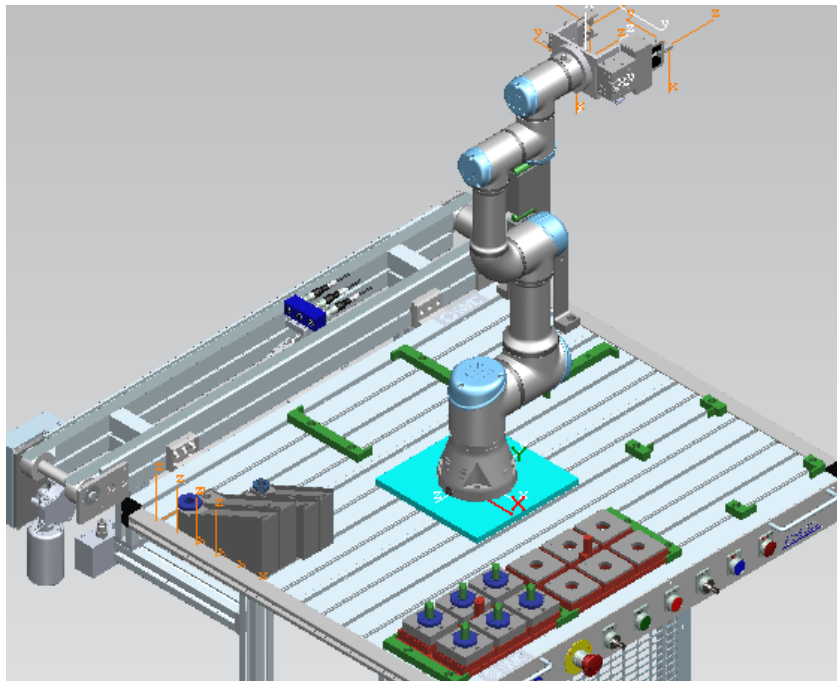


MÁSTER UNIVERSITARIO EN INGENIERÍA DE CONTROL,
AUTOMATIZACIÓN Y ROBÓTICA

TRABAJO FIN DE MASTER

GEMELO DIGITAL Y PUESTA EN MARCHA VIRTUAL DE ESTACIÓN DE MONTAJE ROBOTIZADA.



Estudiante: Lorenzo Ochoa, Bertha

Directora: Burgos Fernandez, Arantzazu

Resumen castellano

Este proyecto se enfoca en la automatización de una estación de montaje robotizada. La estación de montaje consiste en un brazo robótico de Universal Robots, UR3e, capaz de realizar tareas de ensamblaje y manipulación de componentes.

La automatización se lleva a cabo mediante un enfoque dual. En primer lugar, se emplea un gemelo digital utilizando la herramienta Tecnomatix, que permite simular y controlar todas las operaciones de ensamblaje. Esto incluye la creación de secuencias de tareas, interacciones entre componentes y ensamblaje de productos.

El gemelo digital también incluye un sistema de simulación que permite evaluar el comportamiento del proceso y corregir posibles problemas antes de la validación en el entorno real. Esto garantiza una mayor coherencia entre el gemelo digital y la estación de montaje física.

En segundo lugar, se implementa un sistema de control físico basado en un PLC Siemens S7-1500, CPU 1516-3 PN/DP, que se encarga de coordinar y supervisar las tareas en tiempo real. El PLC se comunica con el gemelo digital a través de la red Profinet-IO y gestiona las señales necesarias para el control del robot y la estación de montaje. Además, se utiliza un HMI (WinCC Advance), para la interacción y supervisión del operario.

Resumen euskara

Proiektu hau muntaia-estazio robotizatu baten automatizaziora bideratzen da. Muntaia-estazioa Universal Robotsen (UR3e) beso robotiko bat da, osagaiak mihiztatzeko eta manipulatzeko lanak egiteko gai dena.

Automatizazioa ikuspegi dual baten bidez gauzatzen da. Lehenik eta behin, biki digital bat erabiltzen da Tecnomatix tresna erabiliz, mihizatze-eragiketa guztiak simulatzeko eta kontrolatzeko aukera ematen duena. Horren barruan sartzen da ataza-sekuentziak sortzea, osagaien arteko elkarreraginak eta produktuen mihiztadura.

Biki digitalak simulazio-sistema bat ere badu, eta horrek aukera ematen du prozesuaren portaera ebaluatzeko eta ingurune errealean baliozkotzearen aurretik sor daitezkeen arazoak zuzentzeko. Horrek biki digitalaren eta muntaia fisikoko estazioaren arteko koherentzia handiagoa bermatzen du.

Bigarrenik, PLC Siemens S7-1500, CPU 1516-3 PN/DP batean oinarritutako kontrol fisikoko sistema bat ezartzen da, zereginak denbora errealean koordinatzeaz eta gainbegiratzeaz arduratzen dena. PLCa biki digitalarekin komunikatzen da Profinet-IO sarearen bidez, eta robota eta muntaia-estazioa kontrolatzeko beharrezkoak diren seinaleak kudeatzen ditu. Gainera, HMI (WinCC Advance) bat erabiltzen da langilearen interakziorako eta gainbegiratzeko.

Resumen inglés

This project focuses on the automation of a robotic assembly station. The assembly station consists of a Universal Robots robotic arm, UR3e, capable of performing assembly and component manipulation tasks.

The automation is carried out through a dual approach. Firstly, a digital twin is employed using the Tecnomatix tool, which allows for the simulation and control of all assembly operations. This includes the creation of task sequences, interactions between components, and product assembly.

The digital twin also includes a simulation system that enables the evaluation of process behavior and the correction of potential issues before validation in the real environment. This ensures greater consistency between the digital twin and the physical assembly station.

Secondly, a physical control system is implemented based on a Siemens S7-1500 PLC, CPU 1516-3 PN/DP, responsible for coordinating and supervising tasks in real-time. The PLC communicates with the digital twin via the Profinet-IO network and manages the necessary signals for robot and assembly station control. Additionally, an HMI (WinCC Advance) is used for operator interaction and supervision

1.	MEMORIA.....	10
1.1.	INTRODUCCIÓN.....	10
1.2.	CONTEXTO.....	12
1.3.	OBJETIVOS.....	13
1.4.	ALCANCE DEL TRABAJO.....	14
1.5.	BENEFICIOS QUE APORTA EL TRABAJO.....	15
1.5.1.	Beneficios sociales	15
1.5.2.	Beneficios económicos.....	15
1.5.3.	Beneficios técnicos.....	16
1.6.	ANALISIS DE ALTERNATIVAS	18
1.7.	ANALISIS DE RIESGOS.....	22
1.8.	DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA.....	25
2.	METODOLOGÍA DE DESARROLLO DEL TRABAJO.....	26
2.1.	INTRODUCCIÓN.....	26
2.2.	DESCRIPCIÓN FUNCIONAL DE LA ESTACIÓN.....	27
2.3.	DESCRIPCIÓN DE LOS SERVICIOS	30
2.4.	METODOLOGÍA DEL DISEÑO.....	32
2.4.1.	ELEMENTOS CAD DE LA ESTACIÓN	32
2.4.2.	GEMELO DIGITAL.....	42
2.4.2.1.	Modelado Visual	43
2.4.2.2.	Modelado Funcional	46
2.4.2.2.1.	Propiedades Cinemáticas del Robot UR3e.....	47
2.4.2.2.2.	Parámetros de Movimiento de las Pinzas Neumáticas.....	49
2.4.2.2.3.	Flujo de Material	51
2.4.2.2.4.	Interacciones y Ensamblaje.....	53
2.4.2.3.	Validación del Gemelo Digital.....	60
2.4.2.3.1.	Validación del gemelo digital en la simulación	60
2.4.2.3.2.	Validación del gemelo digital en la realidad	62
2.5.	CONTROL DESDE EL PLC.....	68
2.5.1.	Configuración hardware de los dispositivos en el PLC	69
2.5.2.	Señales de control.....	71

2.5.2.1.	HMI	73
2.5.2.2.	Programa de control	75
2.6.	PUESTA EN MARCHA VIRTUAL (virtual commissioning).....	84
2.7.	PLANIFICACIÓN	86
3.	ASPECTOS ECONÓMICOS	90
4.	CONCLUSIONES	92
4.1.	Desarrollos futuros.....	92
5.	BIBLIOGRAFÍA.....	94
6.	ANEXO II	95

ÍNDICE DE FIGURAS

Figura 1. Conjunto de ensamblaje	27
Figura 2. Estación robotizada.....	28
Figura 3. Robot UR3e	33
Figura 4. Herramienta del Robot	34
Figura 5. Pinza 1	35
Figura 6. Pinza 2	36
Figura 7. Mesa de trabajo	37
Figura 8. Piezas del conjunto	39
Figura 9. Alimentador de Tapas interiores y Rodamientos	40
Figura 10. Alimentador de Bulones.....	40
Figura 11. Palet de entrada.....	41
Figura 12. Alimentador de Tapas exteriores.....	41
Figura 13. Ubicaciones potenciales.....	45
Figura 14. Modelado visual final de la estación de montaje	46
Figura 15. Operaciones del robot	48
Figura 16. Robot Progam Inventory.....	49
Figura 17. Instancia de la operación IN_CASO1_1.....	52
Figura 18. Selección del objeto CAD	52
Figura 19. Material flow simulation	54
Figura 20. Sequence Editor	55
Figura 21. Transiciones de la operación OP13	56
Figura 22. Transiciones asociadas a la operación Init.....	57
Figura 23. Transiciones asociadas a la operación Init_Piezas.....	58
Figura 24. Ejemplo de vinculación de los rodamientos	59
Figura 25. Señales del robot.....	61
Figura 26. Panel de simulación	61
Figura 27. Archivo con formato Script original	63
Figura 28. Archivo con formato Script modificado	64
Figura 29. OLP Commands de las operaciones Pick y Place, respectivamente.	65
Figura 30. Posición de las juntas del robot, en la TeachPendant	66
Figura 31. Posición de las juntas del robot, en Tecnomatix	67
Figura 32. Dispositivos Hardware de la estación	70
Figura 33. Dispositivos hardware vinculados en TIA Portal.....	71
Figura 34. HMI.....	74
Figura 35. Secuencia principal	76
Figura 36. Secuencia de marcha de preparación	78

Figura 37. Secuencia de validación de sensores, parte 1	80
Figura 38. Secuencia de validación de sensores, parte 2	81
Figura 39. Secuencia de emergencia	83
Figura 40. Flujo de comunicación entre los sistemas	84
Figura 41. Diagrama de Gant	89

ÍNDICE DE TABLAS

Tabla 1. Matriz de probabilidad e impacto de los riesgos	24
Tabla 2. Servicios de la estación	31
Tabla 3. Señales de control del PLC	71
Tabla 4. Variables para la puesta en marcha virtual	85
Tabla 5. Presupuesto orientativo de los componentes de la estación	90
Tabla 6. Presupuesto orientativo de los costos de desarrollo y licencias de software	90
Tabla 7. Presupuesto orientativo de los costos del Ingeniero de Desarrollo	91

1. MEMORIA

1.1. INTRODUCCIÓN

En la era de la Industria 4.0, la búsqueda constante de mejoras en los procesos de producción ha llevado al desarrollo de nuevas tecnologías que optimizan la eficiencia, la calidad y la productividad. Entre estas tecnologías se destacan el uso de gemelos digitales y herramientas de simulación, que ofrecen la posibilidad de crear réplicas virtuales precisas de sistemas físicos complejos. En particular, Tecnomatix Process Simulate se ha posicionado como una herramienta líder en la industria para el desarrollo de gemelos digitales y la simulación de procesos de fabricación robotizados.

Tecnomatix Process Simulate es una herramienta de software que permite modelar, simular y validar sistemas de producción en entornos virtuales. A través de su interfaz intuitiva y funcionalidades avanzadas, Process Simulate facilita la creación de gemelos digitales, que son réplicas virtuales de sistemas físicos, como líneas de ensamblaje, estaciones de trabajo o de robots industriales. Estos gemelos digitales permiten realizar pruebas, optimizaciones y validaciones antes de la implementación real, lo que reduce los riesgos, los costos y los tiempos asociados a errores en la etapa de producción.

El presente trabajo de fin de máster se centra en la aplicación de Tecnomatix Process Simulate para el desarrollo de un gemelo digital y la puesta en marcha virtual de una estación de montaje robotizada. El objetivo principal es aprovechar las capacidades de esta herramienta para crear un entorno virtual que reproduzca de manera precisa la estación de montaje y su funcionamiento, utilizando el robot UR3e como elemento central.

El robot UR3e, utilizado en el proyecto, es un brazo robótico compacto y flexible de Universal Robots. Este robot es ampliamente utilizado en diversas aplicaciones industriales debido a su alta precisión, facilidad de uso y versatilidad. Por ello, contar con un gemelo digital de este robot podría tener implicaciones significativas para su rendimiento y eficacia en entornos industriales.

A lo largo del trabajo, se ha seguido una metodología que incluye la adquisición del robot UR3e, el diseño y configuración de la estación de montaje, la programación y validación de la secuencia de montaje en la herramienta de simulación. Además, se llevarán a cabo pruebas comparativas entre la estación de montaje real y su representación virtual para evaluar la precisión y las mejoras obtenidas.

Además de la creación del gemelo digital, un aspecto esencial de este proyecto ha sido la puesta en marcha virtual de la estación de montaje robotizada. Este proceso implica el uso del gemelo digital para simular, analizar y optimizar todas las operaciones de montaje antes de su ejecución en el entorno físico. La puesta en marcha virtual se ha convertido en una herramienta crucial para las

empresas modernas, ya que permite identificar y solucionar potenciales problemas de producción sin la necesidad de detener o interrumpir las operaciones en curso. Esto no solo reduce el tiempo de inactividad y los costos asociados, sino que también mejora la seguridad al permitir que los operarios e ingenieros prueben y modifiquen las operaciones en un entorno digital seguro antes de implementarlas en la estación de montaje real.

1.2. CONTEXTO

La industria manufacturera ha experimentado una evolución constante en los últimos años. La cuarta revolución industrial, también conocida como Industria 4.0, ha transformado la industria y la producción gracias a la integración de tecnologías avanzadas como la Inteligencia Artificial, el Internet de las Cosas, la Robótica y, de manera significativa para este proyecto, los gemelos digitales.

Los gemelos digitales son representaciones digitales precisas de sistemas físicos. Se utilizan para simular, optimizar y analizar sistemas en un entorno seguro y controlado, permitiendo a las empresas predecir posibles problemas y mejorar la eficiencia de sus operaciones sin afectar a la producción real. A medida que la industria continúa evolucionando hacia un entorno más automatizado y digitalizado, la demanda y la relevancia de los gemelos digitales se ha disparado.

En este contexto, Tecnomatix Process Simulate ha surgido como una herramienta líder en el desarrollo de gemelos digitales y la simulación de procesos de fabricación. Process Simulate es una herramienta completa que permite modelar, simular y validar estaciones de montaje en entornos virtuales. Su interfaz intuitiva y sus funcionalidades avanzadas facilitan la creación de gemelos digitales precisos, lo que permite realizar pruebas, optimizaciones y validaciones antes de la implementación real.

La automatización industrial es otro factor clave en este contexto. Los robots industriales, como el UR3e que se utiliza en este proyecto, están cada vez más presentes en las líneas de producción y montaje debido a su capacidad para realizar tareas complejas y repetitivas con alta precisión y eficiencia. La capacidad para integrar estos robots con tecnologías de gemelos digitales es una frontera de desarrollo crucial en la industria moderna.

Además, nos ubicamos en el contexto de una estación de montaje robotizada. Este tipo de estaciones son cada vez más comunes en la industria, especialmente en sectores como el automotriz, el aeroespacial o la electrónica, donde se requiere una alta precisión y eficiencia. Con la creación de un gemelo digital para una estación de montaje robotizada y su puesta en marcha virtual, buscamos explorar las posibilidades y ventajas que estas tecnologías pueden ofrecer en este tipo de entornos.

1.3. OBJETIVOS

Este proyecto se enmarca en el contexto de la Industria 4.0 y tiene como principal objetivo explorar los beneficios de los gemelos digitales y su impacto en las estaciones de montaje robotizadas. Para ello, se han definido los siguientes objetivos específicos:

Objetivo General:

- Diseño e implementación de un gemelo digital de una estación de montaje, con un robot UR3e como elemento principal de la estación, y puesta en marcha virtual de la estación.

Objetivo Específicos:

- Selección de la herramienta adecuada: Se evaluarán diferentes herramientas de simulación y se justificará la elección de Tecnomatix Process Simulate como la plataforma principal para el desarrollo del gemelo digital de la estación de montaje robotizada. Se detallarán las capacidades y funcionalidades de la herramienta y se explicará cómo contribuye al cumplimiento de los objetivos del proyecto.
- Diseño y configuración de la estación de montaje: Se definirán los componentes y dispositivos necesarios para simular la estación de montaje, incluyendo las herramientas de ensamblaje, los sensores y cualquier otro elemento relevante para el proceso de montaje robotizado.
- Desarrollar un gemelo digital detallado y funcional de la estación de montaje utilizando la herramienta Process Simulate de Tecnomatix.
- Simular la estación de montaje robotizada, evaluando su eficiencia en diversas tareas de montaje, teniendo en cuenta los tiempos de ciclo, la precisión en las tareas y la seguridad
- Validar el gemelo digital desarrollado de la estación de montaje real. Identificando así, los desafíos y las limitaciones asociados a la creación y aplicación de gemelos digitales en estaciones de montaje robotizadas.
- Realizar una Puesta en Marcha Virtual de la estación de montaje, utilizando herramientas como Tecnomatix, PLCSIM Advanced y un HMI virtual.

1.4. ALCANCE DEL TRABAJO

Este proyecto se enfoca en la creación de un gemelo digital del robot UR3e y su uso para la puesta en marcha virtual de una estación de montaje robotizada utilizando la herramienta Process Simulate de Tecnomatix. En este sentido, el alcance del trabajo se delimita a las siguientes áreas:

- Análisis y comprensión de los requisitos de la estación de montaje real: Se realizará un estudio detallado de la estación de montaje existente, incluyendo sus componentes, procesos de montaje, restricciones y requisitos específicos.
- Modelado de la estación de montaje en el entorno virtual: Utilizando Tecnomatix Process Simulate, se desarrollará un modelo virtual de la estación de montaje. Esto incluirá la creación de representaciones digitales de los componentes, herramientas y equipos utilizados en el proceso de montaje.
- Programación y validación de la secuencia de montaje: Se desarrollará la lógica de control del robot UR3e en Process Simulate para realizar las tareas de montaje en la estación virtual. Se validarán las secuencias de movimiento y las operaciones de ensamblaje para garantizar el correcto funcionamiento del sistema en el entorno virtual.
- Simulación y optimización de los procesos de montaje: Se realizarán pruebas y simulaciones en el gemelo digital para evaluar y mejorar los procesos de montaje. Esto incluirá la optimización de los tiempos de ciclo, la secuencia de tareas y la precisión en las operaciones de montaje.
- Validación del gemelo digital en la estación de montaje real. Esto permitirá identificar las ventajas y limitaciones del gemelo digital, así como las oportunidades de mejora.
- Virtual Commissioning: Antes de implementar físicamente la estación, se realiza una simulación en un entorno virtual. Esto permite probar y refinar el programa de control, así como identificar posibles problemas de diseño o programación.

1.5. BENEFICIOS QUE APORTA EL TRABAJO

En este apartado se exponen los diferentes beneficios que aporta este proyecto.

1.5.1. Beneficios sociales

- Seguridad laboral: La implementación de la puesta en marcha virtual y la formación en el entorno del gemelo digital brindan a los trabajadores la oportunidad de adquirir habilidades y conocimientos avanzados antes de enfrentarse a situaciones reales. Esto reduce los riesgos de accidentes y lesiones laborales, ya que los operadores estarán más preparados para lidiar con las tareas de montaje.
- Capacitación y desarrollo de habilidades: La implementación del gemelo digital brinda oportunidades de capacitación y desarrollo de habilidades para los trabajadores. Ya que estos pueden adquirir conocimientos en el manejo de tecnologías avanzadas, programación y colaboración con robots. Estas habilidades son altamente valoradas en la actualidad y mejoran la empleabilidad de los trabajadores.
- Contribución a la sostenibilidad: La optimización de los procesos de fabricación mediante el gemelo digital permite reducir el desperdicio de recursos y la emisión de gases contaminantes. Al mejorar la eficiencia en la producción, se contribuye a la sostenibilidad y la preservación del medio ambiente, lo que beneficia a la comunidad en general.
- Fomento de la industria tecnológica local: La adopción de tecnologías avanzadas, como la simulación, estimula el desarrollo y la adopción de soluciones tecnológicas en la industria local. Esto promueve la innovación y el crecimiento económico, lo que impacta positivamente en la creación de empleo y en el progreso de la comunidad.
- Mejora de la competitividad empresarial: Al implementar gemelos digitales y tecnologías de la Industria 4.0, las empresas pueden mejorar su competitividad a nivel global. Al ser más eficientes y avanzadas tecnológicamente, pueden posicionarse mejor en el mercado, lo que a su vez beneficia a la economía y a la comunidad.

1.5.2. Beneficios económicos

- Reducción de costos de producción: La implementación del gemelo digital y la puesta en marcha virtual permiten identificar y corregir posibles problemas en el proceso de fabricación antes de su implementación real. Esto minimiza mejora la eficiencia operativa, lo que se traduce en una reducción significativa de los costos de producción.
- Mejora en la calidad del producto: Al optimizar los procesos de fabricación y reducir los errores mediante la simulación y el gemelo digital, se mejora la calidad del producto final. Una mayor calidad en los productos resulta en una reducción de devoluciones y

reclamaciones, lo que a su vez aumenta la satisfacción del cliente y fortalece la reputación de la empresa en el mercado.

- **Incremento de la productividad:** La simulación y la puesta en marcha virtual permiten una mejor planificación y programación de las tareas de montaje, lo que mejora la productividad en la línea de producción. Los operadores pueden estar mejor preparados y familiarizados con las tareas a realizar, lo que reduce el tiempo de inactividad y aumenta la eficiencia en la producción.
- **Ahorro en inversiones:** La simulación y el uso de gemelos digitales evitan la necesidad de realizar pruebas y validaciones real, que pueden ser costosas y requerir una inversión en recursos materiales y humanos. Al realizar todas estas pruebas en el entorno virtual, se reduce la necesidad de invertir en prototipos y maquinaria adicional.
- **Mayor eficiencia en el diseño de procesos:** La capacidad de realizar ajustes y optimizaciones en el gemelo digital permite diseñar y planificar los procesos de montaje de manera más eficiente. Se pueden identificar las mejores estrategias y secuencias de montaje antes de la implementación, lo que resulta en un diseño de procesos más eficiente y rentable.

1.5.3. Beneficios técnicos

- **Validación y optimización de la secuencia de montaje:** La creación del gemelo digital permite validar y optimizar la secuencia de montaje del robot UR3e en la estación de trabajo. Se pueden realizar pruebas exhaustivas y ajustes en el entorno virtual para encontrar la mejor secuencia y configuración, lo que garantiza un flujo de trabajo óptimo en el mundo real.
- **Identificación de problemas y mejoras de diseño:** La simulación en el gemelo digital ayuda a identificar posibles colisiones, interferencias o problemas de accesibilidad en la estación de montaje robotizada. Esto permite realizar cambios de diseño y optimizaciones antes de la implementación real.
- **Reducción del tiempo de puesta en marcha:** Al realizar la puesta en marcha virtual de la estación de montaje con el gemelo digital, se anticipan y corrigen problemas antes de la fase de implementación real. Esto reduce el tiempo necesario para poner en funcionamiento la estación real y permite una transición más rápida a la producción.
- **Pruebas de escenarios:** La simulación en el gemelo digital facilita la realización de pruebas en escenarios complejos que podrían ser peligrosas o costosas de replicar en el mundo real. Esto permite evaluar el comportamiento del robot UR3e en diferentes situaciones y asegurar su correcto funcionamiento en cualquier circunstancia.
- **Ajuste y validación de parámetros:** La simulación permite ajustar y validar los parámetros del robot, como velocidades de movimiento, aceleraciones y precisión, para lograr el mejor rendimiento en la estación de montaje. Esto garantiza un funcionamiento óptimo del robot en la realidad y asegura la calidad y consistencia del proceso de montaje.

- Mejora de la colaboración humano-robot: La simulación en el gemelo digital también facilita la interacción entre humanos y robots en la estación de montaje. Los operadores pueden familiarizarse con la interacción con el robot y realizar pruebas de seguridad en un entorno virtual, lo que mejora la colaboración y la seguridad en el trabajo conjunto.

1.6. ANALISIS DE ALTERNATIVAS

Se explorarán los fundamentos teóricos y las aplicaciones prácticas de las herramientas software disponibles en el mercado, para la creación de gemelos digitales, así como los beneficios y desafíos asociados a su implementación. A partir de este análisis, se seleccionará Tecnomatix Process Simulate como herramienta principal para el desarrollo del gemelo digital de la estación de montaje robotizada.

ROS (“Robot Operatin System”):

ROS es una plataforma de código abierto ampliamente utilizada en robótica. Proporciona una variedad de bibliotecas, herramientas y algoritmos que facilitan el desarrollo de aplicaciones robóticas, incluyendo la simulación de robots y entornos virtuales.

Aunque es una plataforma muy interesante se descarta la posibilidad de utilizarla porque:

- Aunque ROS cuenta con una comunidad activa y numerosos recursos, al tratarse de código abierto y gratuito, no tiene un soporte técnico específico para dirigirse en caso de problemas.
- Se necesita otro sistema operativo (Ubuntu) y en este proyecto se cuenta con Windows.
- Curva de aprendizaje: ROS tiene una curva de aprendizaje pronunciada, lo que puede requerir un tiempo considerable para familiarizarse con sus conceptos y herramientas.
- Dependencia del hardware: La compatibilidad y el rendimiento de ROS pueden variar según el hardware utilizado. Esto puede limitar las opciones de hardware para el proyecto y requerir un análisis detallado de la compatibilidad.

Siemens PLM Software:

Además de Tecnomatix Process Simulate, Siemens PLM Software ofrece otras herramientas de simulación y modelado, como “Plant Simulation” y “Simcenter 3D”. Estas soluciones proporcionan capacidades avanzadas para la creación de gemelos digitales y la optimización de procesos de producción. Sería beneficioso evaluar estas herramientas en términos de funcionalidad, integración con el robot UR3e y facilidad de uso.

Las principales desventajas por las que se descartan estas plataformas son:

- Plant Simulation: Está plataforma está orientada principalmente a la simulación de procesos de producción a nivel de planta, puede tener una curva de aprendizaje pronunciada si se requiere una simulación detallada a nivel de estación de montaje. Además, puede no tener todas las capacidades específicas para la programación y simulación de robots en comparación con plataformas más especializadas.

- Simcenter 3D: Tiene un enfoque principal en la simulación y análisis estructural, por lo que no puede cubrir completamente las necesidades de simulación de una estación de montaje con un robot. Por otro lado, puede requerir módulos o complementos adicionales para abordar aspectos específicos de la automatización y la programación del robot.

Siemens NX-MCD:

Siemens NX es un software de diseño, simulación y fabricación integrado. Además de Tecnomatix Process Simulate, Siemens NX ofrece herramientas avanzadas de modelado y simulación que pueden ser utilizadas para crear gemelos digitales de sistemas de producción completos, incluyendo el robot UR3e y sus interacciones con otros equipos y componentes.

Las principales desventajas por las que se descartan esta plataforma son:

- Si bien es una plataforma de diseño y fabricación sólida, es una herramienta orientada a máquinas. Por lo que, puede requerir módulos adicionales o personalización para abordar la simulación y programación específica de robots.
- Puede tener una curva de aprendizaje pronunciada y requerir experiencia en el uso de software CAD/CAM.

Visual Components:

Visual Components es una plataforma de simulación y diseño 3D que permite crear gemelos digitales de sistemas de producción completos. Ofrece una amplia biblioteca de modelos de robots industriales, incluyendo el robot UR3e, y permite simular su comportamiento y optimizar los procesos de montaje y producción.

Las principales desventajas por las que se descartan esta plataforma son:

- Complejidad de Uso: Visual Components puede resultar complejo para aquellos que no tienen experiencia previa con la herramienta. La curva de aprendizaje puede ser empinada, lo que podría aumentar los tiempos de desarrollo y capacitación del personal.
- El software Visual Components a menudo requiere una configuración de hardware robusta para un rendimiento óptimo, lo que puede traducirse en costos adicionales para actualizar las computadoras y estaciones de trabajo existentes.
- Se encontraron limitaciones en la capacidad de Visual Components para integrarse de manera fluida con el hardware de automatización específico utilizado en el proyecto, como el PLC Siemens S7-1500

RoboDK:

Es un simulador potente y rentable para robots industriales y programación de robots. La ventaja de utilizar las herramientas de simulación y programación offline de RoboDK es que permite programar robots fuera del entorno de producción y se puede programar robots directamente desde el ordenador, eliminando el tiempo de inactividad de producción causado por la programación en el taller. Además, no se requieren habilidades de programación, ya que posee una interfaz intuitiva y se puede programar casi cualquier robot, gracias a su extensa biblioteca con más de 600 brazos robóticos. Por último, se puede exportar programas al robot UR3e, gracias a que los procesadores de RoboDK son compatibles con muchos controladores de robots, entre ellos Robots universales (URP/ script).

Las principales desventajas del software de simulación RoboDK son:

- Limitaciones del fabricante: Esta plataforma está más orientada a los robots colaborativos de Universal Robots, lo que puede limitar su compatibilidad con otros fabricantes de robots y componentes. Esto podría restringir las opciones de personalización y configuración del proyecto.

Siemens Tecnomatix Process Simulate:

Tecnomatix Process Simulate es un software de programación offline de robots, proporciona herramientas para la creación de gemelos digitales, simulación y optimización de procesos de producción. Con Process Simulate, es posible modelar y simular la estación de montaje con el robot UR3e, optimizar su funcionamiento y evaluar diferentes escenarios.

Las principales desventajas de esta herramienta son:

- Costo: Tecnomatix Process Simulate es una herramienta de software comercial, lo que implica un costo asociado para su adquisición y su licencia. Esto puede suponer un desafío para proyectos con restricciones presupuestarias.
- Curva de aprendizaje: Al ser una plataforma más especializada, puede requerir una curva de aprendizaje inicial para aprovechar al máximo sus capacidades. Esto puede implicar una inversión de tiempo y esfuerzo por parte del equipo de desarrollo.

Las principales ventajas por las cuales se elige esta herramienta son:

- Capacidades avanzadas: Ofrece un conjunto completo de herramientas para la planificación y simulación de procesos de producción. Su enfoque en la industria manufacturera y su integración con los gemelos digitales lo convierten en una opción sólida para proyectos de este tipo.

- Integración con hardware y software: Permite la integración con una amplia gama de componentes de hardware y software, lo que brinda flexibilidad y personalización en el diseño de la estación de montaje.
- Experiencia y soporte: Siemens, el proveedor de Tecnomatix Process Simulate, tiene una amplia experiencia en la industria manufacturera y ofrece soporte técnico y actualizaciones regulares para la plataforma. Esto garantiza la disponibilidad de recursos y ayuda profesional en caso de problemas o consultas.

1.7. ANALÍSIS DE RIESGOS

El desarrollo del proyecto Gemelo Digital y Puesta en Marcha Virtual de Estación de Montaje Robotizada con el robot UR3e y la herramienta Process Simulate de Tecnomatix implica la identificación y evaluación de diversos riesgos. Estos riesgos deben ser tomados en cuenta y gestionados adecuadamente para asegurar la ejecución exitosa del trabajo. A continuación, se presenta un análisis de los principales riesgos asociados al proyecto:

Riesgos técnicos:

El riesgo técnico identificado en el proyecto se refiere a la precisión y fidelidad del gemelo digital creado para la estación de montaje robotizada con el robot UR3e. La probabilidad de este riesgo es alta debido a la complejidad de modelar con precisión todos los componentes y movimientos del robot, así como las interacciones con los elementos de la estación de montaje.

Es importante destacar que, en la estación de montaje, el robot UR3e debe manipular palets y encajarlos en zonas delimitadas por marcos en la mesa de trabajo. La diferencia entre las piezas y las zonas es de pocos milímetros, lo que puede resultar en un ajuste crítico y delicado por parte del robot. Si el gemelo digital no refleja con precisión estas dimensiones y movimientos, existe el riesgo de que el robot no pueda realizar las tareas de montaje correctamente.

Además, si el gemelo digital no ha sido adecuadamente validado y optimizado, pueden ocurrir colisiones con componentes de la estación, herramientas de ensamblaje u otros objetos, lo que podría dañar el robot o los elementos de la estación y afectar la integridad del proceso de montaje.

Para mitigar este riesgo, se debe realizar una validación exhaustiva del gemelo digital, asegurándose de que el modelo virtual del robot UR3e y los elementos de la estación sean precisos y estén correctamente configurados. Además, se deben realizar pruebas de colisión y ajustes en el gemelo digital para garantizar que el robot pueda realizar sus tareas de montaje sin problemas y en condiciones seguras. La verificación cuidadosa de las dimensiones, tolerancias y movimientos en el gemelo digital asegurará que el robot pueda interactuar adecuadamente con la estación de montaje y llevar a cabo sus tareas con precisión y sin riesgos de colisión.

Teniendo en cuenta, la matriz de probabilidad e impacto de riesgos (figura 3), la alta probabilidad con la que pueda ocurrir este riesgo y que el robot cuenta con protecciones integradas que detienen el movimiento de éste en caso de detección de colisión, se puede concluir que se trata de un riesgo moderado. Para minimizar las repercusiones del riesgo en el robot, se realiza la validación del gemelo digital a una velocidad reducida, para prevenir las colisiones. También, se ha optado por realizar las posiciones críticas de forma manual en el robot real. Cada posición crítica define una configuración en los ejes del robot, esta configuración se introduce forma manual posición específica en la simulación.

Riesgos de tiempo y recursos:

El desarrollo y la implementación de un gemelo digital y la puesta en marcha virtual pueden requerir una inversión considerable de tiempo y recursos. Riesgos como retrasos en la adquisición de datos, falta de personal cualificado o problemas técnicos inesperados podrían afectar el cronograma y aumentar los costos del proyecto. El impacto que puede surgir debido a este riesgo es que el avance del proyecto sea muy lento. Según la matriz de probabilidad se trata de un riesgo bajo ya que, la probabilidad del riesgo es moderada y el impacto del riesgo es muy bajo. Por lo que no se realiza ninguna acción de prevención por este riesgo.

Riesgos de seguridad:

La interacción entre el gemelo digital y la célula robotizada real podría presentar riesgos de seguridad si no se implementan las medidas adecuadas para garantizar la protección de los operadores y de la estación de montaje. Aunque el gemelo digital es una réplica virtual de la estación de montaje, las acciones y decisiones que se tomen en este entorno también pueden afectar la operación real del robot UR3e en la estación y se pueden generar riesgos de golpeo, atrapamiento y otros similares.

Este riesgo, se identifica como un riesgo de seguridad inesperado con una probabilidad moderada de ocurrencia, por lo que se considera que el entorno de trabajo puede ser peligroso para el personal involucrado. No obstante, el hecho de contar con un robot colaborativo en la estación de montaje permite mitigar y reducir los efectos de los riesgos en el personal. Gracias a la naturaleza colaborativa del robot, este puede detectar la presencia de los operadores y detener su movimiento de forma rápida y segura en caso de una posible colisión o situación de riesgo. Esta característica hace que se trate de un riesgo moderado, por lo que se deben establecer protocolos de seguridad y procedimientos claros para la interacción con el robot.

Riesgos de implementación:

Durante la implementación del gemelo digital y la puesta en marcha virtual, puede surgir la falta de compatibilidad entre la herramienta Process Simulate y el robot UR3e, lo que podría dificultar la integración y generar retrasos en el proyecto.

Esto supone un riesgo bajo, debido a que la probabilidad del riesgo es moderada y a que el impacto es muy bajo, por lo que no se realiza ninguna acción preventiva.

En la tabla 1, se puede observar la matriz de probabilidad e impacto de los riesgos:

Tabla 1. Matriz de probabilidad e impacto de los riesgos

Probabilidad	Muy Alta	Moderada	Alta	Alta	Alta	Alta
	Alta	Baja	Moderada	Moderada	Alta	Alta
	Moderada	Baja	Moderada	Moderada	Moderada	Alta
	Baja	Baja	Baja	Moderada	Moderada	Alta
	Muy Baja	Baja	Baja	Baja	Baja	Moderada
		Muy Bajo	Bajo	Moderado	Alto	Muy Alto
		Impacto				

1.8. DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA

La solución propuesta para el proyecto de Gemelo Digital y Puesta en Marcha Virtual de Estación de Montaje Robotizada con el robot UR3e y la herramienta Process Simulate de Tecnomatix se fundamenta en el desarrollo de un gemelo digital preciso y funcional que represente fielmente la estación de montaje robotizada. Esta solución tiene como objetivo principal optimizar y validar el proceso de montaje mediante la simulación en un entorno virtual antes de su implementación real, lo que aporta una serie de ventajas técnicas y económicas.

En primer lugar, se procederá a recopilar todos los datos y especificaciones del robot UR3e y de la estación de montaje, asegurándose de obtener información precisa para la creación del gemelo digital. Mediante la herramienta Process Simulate de Tecnomatix, se desarrollará un modelo virtual detallado del robot y de todos los componentes relevantes de la estación, incluyendo herramientas de ensamblaje, sensores y demás elementos involucrados en el proceso de montaje.

Una vez creado el gemelo digital, se realizará una validación exhaustiva de su precisión y funcionamiento. Se llevarán a cabo pruebas de simulación para verificar que todas las interacciones y movimientos del robot en el entorno virtual se ajusten correctamente a la realidad. Se buscará identificar y corregir posibles problemas o colisiones que puedan surgir en el proceso de montaje para garantizar un funcionamiento seguro y eficiente.

Para la puesta en marcha virtual, se simularán y evaluarán diferentes secuencias de montaje, optimizando la programación del robot UR3e y las operaciones de ensamblaje. Estas simulaciones permitirán identificar y resolver posibles cuellos de botella, optimizando el rendimiento y la productividad de la estación de montaje.

En resumen, la solución propuesta para el trabajo de memoria busca aprovechar las capacidades de los gemelos digitales para optimizar el proceso de montaje en una estación robotizada. La creación de un gemelo digital preciso y funcional, junto con la validación exhaustiva y la optimización de la puesta en marcha virtual, permitirá una implementación más eficiente y segura de la estación de montaje, con beneficios significativos en términos de reducción de costos, mejoras en la calidad del producto y aumento de la productividad general.

2. METODOLOGÍA DE DESARROLLO DEL TRABAJO

2.1. INTRODUCCIÓN

En el transcurso del desarrollo de este proyecto, se ha seguido una metodología estructurada para garantizar un enfoque efectivo y ordenado en el diseño, implementación y validación de una estación de montaje robotizado. Esta metodología se basa en las mejores prácticas de la ingeniería de sistemas y la automatización industrial, y se ha adaptado específicamente para cumplir con los objetivos y requisitos de este proyecto.

La metodología se ha dividido en una serie de fases interconectadas, cada una de las cuales desempeña un papel crucial en la creación exitosa de la estación de montaje robotizado. A continuación, se presenta un resumen de estas fases:

- Descripción sistema: En esta fase inicial, se establecieron la descripción general de la estación, definiendo así todos los elementos que tiene la estación real. También se establecieron los productos a ensamblar.
- Diseño: En esta fase, se definió los diseños de los elementos de la estación de montaje robotizada y su implantación en la simulación virtual.
- Gemelo digital: en esta fase se creó de manera efectiva el Gemelo Digital, a través tres pasos abiertos que permiten su desarrollo y aprovechamiento óptimo (Modelado visual, Modelado funcional y validación del gemelo digital).
- Implementación de un sistema de control adicional basado en un PLC (Controlador Lógico Programable), para garantizar un funcionamiento seguro y eficiente.
- La última fase, de puesta en marcha virtual del proyecto desempeña un papel fundamental en la validación de la estación de montaje robotizada.

2.2. DESCRIPCIÓN FUNCIONAL DE LA ESTACIÓN

La estación robotizada tiene como objetivo funcional ensamblar una pieza compuesta por una base, un rodamiento, un bulón, una tapa interior y una tapa exterior (véase figura 1). De acuerdo con las condiciones específicas del palet de bases que se introduzca en la estación y del servicio seleccionado, el robot UR3e realizará diferentes operaciones de ensamblaje para completar el conjunto.

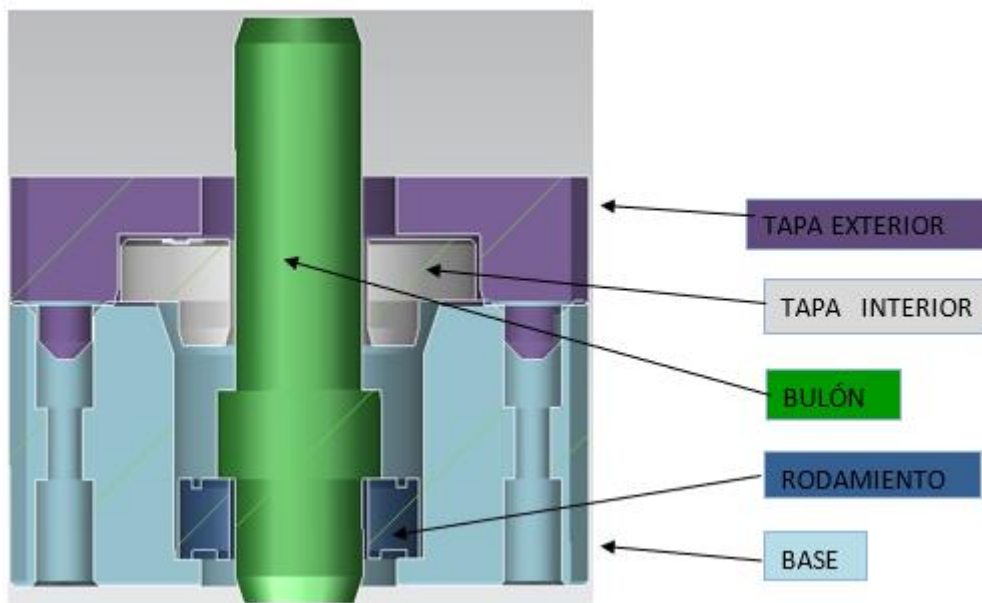


Figura 1. Conjunto de ensamblaje

El proceso comienza con la introducción del palet de bases en la estación. Este palet puede contener entre 1 y 6 bases, y cada base puede llevar otros elementos adicionales, como rodamientos, bulones o tapas interiores.

Según las condiciones y los servicios elegidos (ver apartado 2.3), el robot UR3e realizará diferentes operaciones de ensamblaje. El robot UR3e se encargará de seleccionar y manipular los elementos necesarios para cada operación de ensamblaje. Utilizando su herramienta, el robot agarrará los componentes correspondientes y los ensamblará de manera precisa sobre la base.

El sistema está diseñado para ser altamente flexible y adaptable a diferentes configuraciones de bases y servicios de ensamblaje. La secuencia de operaciones se ajusta dinámicamente según las características del palet de bases y el servicio seleccionado, lo que permite un proceso eficiente y personalizado.

Una vez finalizado el ensamblaje de la pieza, el producto terminado se coloca en la zona de salida, listo para su posterior proceso o distribución.

Para conseguir el objetivo funcional de la estación, el sistema consta de los siguientes elementos, que se pueden observar en la figura 2:

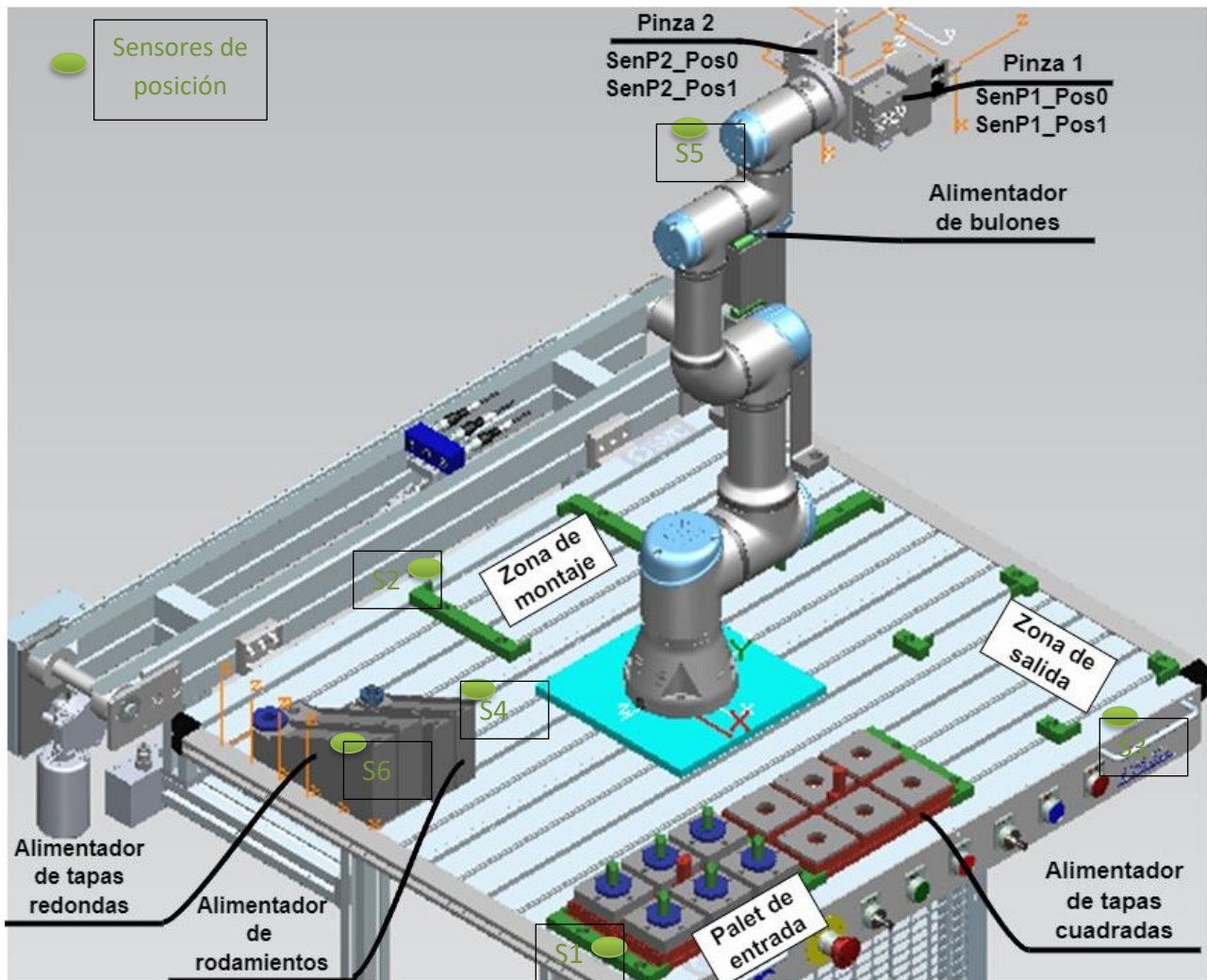


Figura 2. Estación robotizada

- Robot UR3e: Es el componente central del sistema, encargado de realizar las tareas de manipulación y ensamblaje de los productos. Cuenta con una garrucha personalizada que incluye dos pinzas controladas mediante aire comprimido y equipadas con sensores inductivos para el monitoreo del estado de las pinzas (abierta o cerrada).
- Mesa: Proporciona la plataforma de trabajo para el robot UR3e y el montaje de los productos. En la mesa se encuentran diferentes piezas fabricadas con una impresora 3D, entre ellas, destacan los marcos que se utilizan para ubicar los palets.
- Palets: El sistema cuenta con dos palets, uno destinado a contener tapas y otro para las bases de los productos a ensamblar. El palet de tapas tiene una ubicación fija en la mesa, mientras que el palet de bases se puede colocarse en diferentes puntos a lo largo del proceso de montaje.

- Alimentadores por Gravedad: Son tres alimentadores incorporados al sistema para suministrar de manera continua los componentes necesarios para el proceso de ensamblaje. Estos alimentadores garantizan una operación ininterrumpida y una alimentación constante de las piezas requeridas.
- Circuitos Neumáticos: El sistema cuenta con circuitos neumáticos que permiten controlar las pinzas del robot UR3e. El aire comprimido se utiliza para abrir y cerrar las pinzas de forma precisa durante el proceso de manipulación.
- Sensores Inductivos: El sistema dispone de cuatro sensores, que se encuentran en las pinzas del robot y proporcionan información en tiempo real sobre el estado de estas, permitiendo un control preciso y seguro de las acciones de manipulación.
- Sensores de Posición: El sistema dispone de seis sensores de posición.
 - S1: Sensor en la zona de entrada, para verificar que el palet ha entrado en la estación.
 - S2: Sensor en la zona de montaje, para verificar que el palet se encuentra en la zona de montaje.
 - S3 Sensor en la zona de salida, para verificar que el palet se encuentra en la zona de salida.
 - S4: Sensor en el alimentador de rodamientos: Para verificar que existen suficientes rodamientos para la ejecución del ensamblaje. Esta verificación solo se realiza al inicio del ensamblaje.
 - S5: Sensor en el alimentador de bulones: Para verificar que existen suficientes bulones para la ejecución del ensamblaje. Esta verificación solo se realiza al inicio del ensamblaje.
 - S6: Sensor en el alimentador de tapas interiores: Para verificar que existen suficientes tapas interiores para la ejecución del ensamblaje. Esta verificación solo se realiza al inicio del ensamblaje.

En conjunto, estos elementos conforman un sistema altamente automatizado y eficiente para llevar a cabo el montaje de productos. La combinación del robot UR3e con su herramienta, la mesa, los palets, los alimentadores por gravedad y los circuitos neumáticos y sensores inductivos garantiza una operación continua y precisa, optimizando la productividad y asegurando la calidad del proceso de ensamblaje.

2.3. DESCRIPCIÓN DE LOS SERVICIOS

La estación robotizada ofrece una variedad de servicios de ensamblaje, cada uno diseñado para satisfacer diferentes requerimientos de montaje en la pieza principal. Estos servicios permiten una personalización efectiva del proceso de ensamblaje de acuerdo con las características específicas del palet de bases que se introduce en la estación. A continuación, se describen detalladamente cada uno de los 10 servicios disponibles:

1. Servicio: Añadir rodamiento a la base.

Descripción: En este servicio, el robot UR3e selecciona el rodamiento y los coloca con precisión sobre la base correspondiente del palet, completando la primera etapa del ensamblaje.

2. Servicio: Añadir rodamiento y bulón a la base.

Descripción: El robot ejecuta la colocación del rodamiento y el bulón sobre la base correspondiente del palet. Esta operación garantiza un ensamblaje más completo y resistente de la pieza.

3. Servicio: Añadir rodamiento, bulón y tapa interior a la base.

Descripción: El robot lleva a cabo la instalación del rodamiento y el bulón, y posteriormente coloca la tapa interior sobre la base. Este servicio agrega una capa adicional de protección y funcionalidad a la pieza.

4. Servicio: Añadir rodamiento, bulón, tapa interior y tapa exterior a las bases.

Descripción: Este servicio implica el ensamblaje completo de la pieza principal. El robot instala el rodamiento, el bulón, la tapa interior y finalmente coloca la tapa exterior para concluir el montaje.

5. Servicio: Añadir bulón a la base.

Descripción: El robot se enfoca en agregar un bulón a la base del palet, lo que proporciona una sujeción adicional en el ensamblaje de la pieza.

6. Servicio: Añadir bulón y tapa interior a la base.

Descripción: En este servicio, el robot realiza la instalación de un bulón y, posteriormente, colocar la tapa interior sobre la base del palet.

7. Servicio: Añadir bulón, tapa interior y la tapa exterior a las bases.

Descripción: Este servicio incluye la colocación de un bulón, la tapa interior y la tapa exterior en la base del palet, logrando un ensamblaje completo y funcional de la pieza principal.

8. Servicio: Añadir tapa interior a la base.

Descripción: El robot se enfoca exclusivamente en colocar la tapa interior sobre la base del palet, proporcionando una cobertura adecuada a la pieza.

9. Servicio: Añadir tapa interior y la tapa exterior a la base.

Descripción: En este servicio, el robot instala la tapa interior y, posteriormente, coloca la tapa exterior, completando el montaje de la pieza con una combinación de ambas tapas.

10. Servicio: Añadir tapa exterior a la base.

Descripción: El robot realiza la instalación exclusiva de la tapa exterior sobre la base del palet, otorgando un acabado final a la pieza principal.

La versatilidad de los servicios y la precisión del robot UR3e garantizan un ensamblaje eficiente y de alta calidad. En la tabla 2, se puede observar los servicios de ensamblaje:

Tabla 2. Servicios de la estación

SERVICIOS	PALET DE ENTRADA	TAREAS A REALIZAR
1	BASE	Rodamiento
2		Rodamiento Bulón
3		Rodamiento Bulón Tapa interior
4		Rodamiento Bulón Tapa interior Tapa exterior
5	BASE + RODAMIENTO	Bulón
6		Bulón Tapa exterior
7		Bulón Tapa interior Tapa exterior
8	BASE + RODAMIENTO + BULÓN	Tapa interior
9		Tapa interior Tapa cuadrada
10	BASE + RODAMIENTO + BULÓN + TAPA BLANCA	Tapa exterior

2.4. METODOLOGÍA DEL DISEÑO

El presente apartado tiene como objetivo proporcionar una visión integral de la metodología de diseño utilizada en el desarrollo del proyecto.

En la primera parte de este apartado, se detallarán los elementos CAD que conforman la estación robotizada real, incluyendo el diseño mecánico del robot UR3e, la herramienta con las dos pinzas neumáticas y los elementos de la mesa, como los marcos para ubicar los palets y las piezas impresas en 3D. Se destacará la integración de los sensores inductivos que aportan control y retroalimentación al proceso de manipulación y ensamblaje.

En la segunda parte, se describirá el desarrollo del gemelo digital utilizando la herramienta Process Simulate de Tecnomatix. Se explicará el proceso de modelado y simulación, detallando la creación del entorno virtual de la estación, la programación del robot UR3e y la interacción con los elementos del palet de bases. Se mostrarán las simulaciones realizadas para validar y optimizar el proceso de ensamblaje, así como los ajustes realizados en el gemelo digital para reflejar fielmente el comportamiento del sistema real.

2.4.1. ELEMENTOS CAD DE LA ESTACIÓN

En este apartado, se describen en detalle los elementos CAD que conforman la estación robotizada. Estos componentes son fundamentales para la implementación y funcionamiento adecuado de la estación de montaje robotizada.

Para la representación gráfica de algunos elementos, se requiere conocer las medidas precisas, lo que garantiza la adecuada interacción y ensamblaje virtual en el gemelo digital. Elementos como la herramienta del robot con las dos pinzas neumáticas, los marcos para ubicar los palets y las piezas impresas en 3D, demandan dimensiones exactas para lograr una simulación realista y precisa.

Por otro lado, cabe destacar que tanto la mesa de trabajo como el robot UR3e cuentan con modelos de simulación proporcionados por los fabricantes en formato JT CAD. Estos modelos predefinidos simplifican el proceso de introducción en el software de simulación y aseguran la precisión de su representación virtual. Gracias a estos modelos predeterminados, se optimiza el tiempo y los recursos destinados al modelado de la mesa de trabajo y el robot, lo que permite centrarse en el diseño y ajuste de otros elementos críticos del proceso de montaje.

2.4.1.1. Robot UR3e:

El robot UR3e (figura 3), fabricado por Universal Robots, es el componente principal de la estación robotizada. Se trata de un robot colaborativo de 6 ejes, diseñado para interactuar de forma segura con los operadores humanos en un entorno de trabajo compartido. El proveedor proporciona el modelo CAD del robot UR3e. El cual incluye su estructura mecánica, articulaciones, motores y controladores, permitiendo su simulación y programación en el gemelo digital.

Los elementos básicos del brazo robótico son:

- Base: donde se monta el brazo robótico.
- Hombro y Codo: hacen movimientos más amplios.
- Muñeca 1 y Muñeca 2: hacen movimientos más finos.
- Muñeca 3: donde se acopla la herramienta a la Brida de herramienta.

Es el robot más pequeño de la gama UR3. Sin embargo, es un robot pequeño pero potente, el UR3e tiene una carga útil nominal de 3 kg y un radio de alcance de 500 mm; Esto influye en el peso máximo que es capaz de cargar (incluyendo el peso de la garra, que en este caso es una pinza cuyo peso es de 0.75kg). La carga útil nominal del brazo también depende de la compensación del centro de gravedad de la carga (distancia entre el centro de la brida de la herramienta y el centro de gravedad de la carga útil sujeta).



Figura 3. Robot UR3e

2.4.1.2. Herramienta del Robot (figura 4):

La Herramienta es un elemento clave que se acopla al extremo del robot UR3e. Esta herramienta cuenta con dos pinzas neumáticas controladas por aire comprimido y equipadas con sensores inductivos para la detección del estado de las pinzas (abiertas o cerradas). El diseño CAD de la herramienta refleja su geometría, mecanismos de apertura y cierre, así como la ubicación de los sensores inductivos para su posterior simulación y funcionamiento en el gemelo digital.

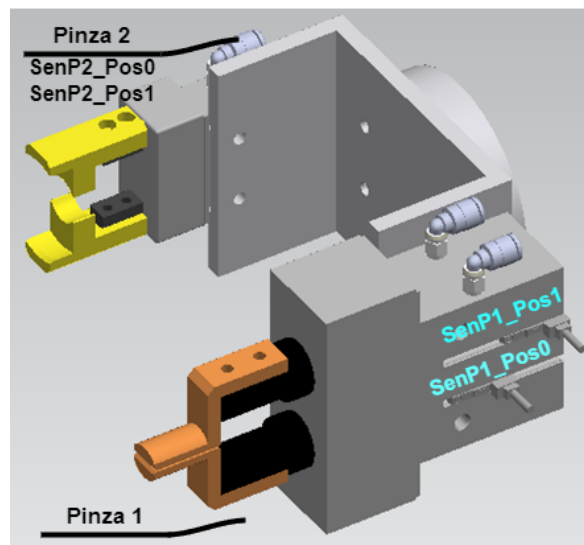


Figura 4. Herramienta del Robot

Las dos pinzas han sido diseñadas para adaptarse a aplicaciones que requieren un alto grado de precisión en una amplia gama de entornos. La precisión del dedo se logra mediante el uso de un mecanismo de leva de cuña, que permite la capacidad de repetición a ± 0.01 mm y produce características de carga axial mucho más altas que permiten su uso en procedimientos de prensa de inserción.

Pinza 1:

La pinza 1, es una pinza paralela de la serie modelo MHKL2-25D (véase la figura 5).

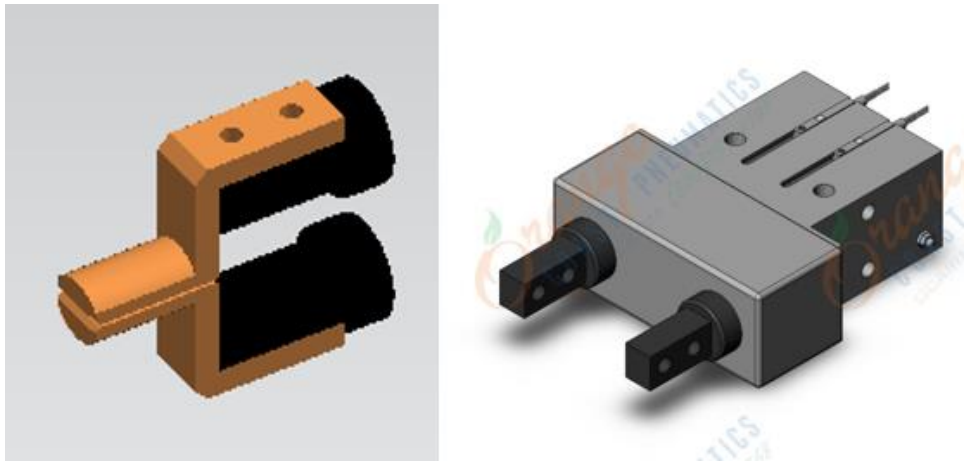


Figura 5. Pinza 1

Esta pinza, desempeña una función esencial en el proceso de ensamblaje en la estación robotizada. Es controlada mediante un sistema neumático, está diseñada para manipular y transportar tres tipos de componentes: los rodamientos, las tapas interiores y las tapas exteriores. Cada uno de estos elementos posee un orificio en su centro, aunque los diámetros de estos orificios son diferentes entre sí para cada tipo de componente. Por otro lado, la pinza tiene dos sensores Inductivos, las cuales proporcionan la siguiente información sobre el estado de las mismas:

- SenP1_Pos0: Pinza 1 abierta.
- SenP1_Pos1: Pinza 1 cerrada.

Cuando la pinza 1 se encuentra en posición cerrada, se introduce de manera precisa en el orificio del componente seleccionado.

Para capturar el componente, la pinza se abre al activar el sistema neumático, asegurando un agarre firme y seguro. Este mecanismo de apertura y cierre permite que la pinza 1 sujete el componente con precisión y suavidad, evitando daños o deformaciones en el material.

Una vez que la pinza ha sujetado el componente de manera adecuada, el robot UR3e, transporta la pieza desde su ubicación inicial hasta el lugar destinado en el proceso de ensamblaje. La precisión y confiabilidad del agarre de la pinza 1 garantizan que los componentes sean colocados en las posiciones requeridas con exactitud milimétrica.

Pinza 2:

La pinza 2, es una pinza paralela de la serie modelo MHKL2-16D (véase la figura 6).

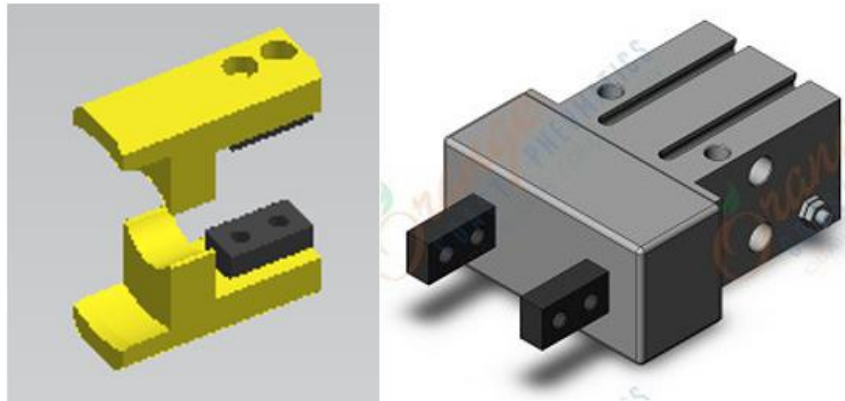


Figura 6. Pinza 2

La pinza 2 de la herramienta desempeña un papel clave en el proceso de montaje y transporte de la estación robotizada. Esta versátil pinza, también controlada mediante un sistema neumático.

De igual forma que la pinza anterior, la pinza tiene dos sensores Inductivos, las cuales proporcionan la siguiente información sobre el estado de estas:

- SenP2_Pos0: Pinza 2 abierta.
- SenP2_Pos1: Pinza 2 cerrada.

Por otro lado, esta pinza tiene una doble funcionalidad, lo que contribuye significativamente a la eficiencia y precisión del ensamblaje.

En su primera función, la pinza es responsable de agarrar el palet de las bases, ubicado en la zona de entradas, y transportarlo con seguridad hacia la zona de montaje y, posteriormente, hacia la zona de salida. Cuando la pinza se cierra, sujeta firmemente el cilindro ubicado en el palet de bases. Gracias a esta capacidad de agarre y liberación controlada, el robot UR3e puede manipular y posicionar el palet con precisión milimétrica, garantizando un transporte fluido y eficiente a lo largo del proceso de montaje.

En su segunda función, la pinza es utilizada para coger los bulones requeridos en el ensamblaje de la pieza principal. Del mismo modo que con el palet de bases, los bulones tienen un cilindro con un diámetro uniforme, lo que permite que la pinza los sujete con seguridad al cerrarse. Este proceso de agarre preciso asegura que los bulones sean manipulados y colocados con la máxima exactitud, garantizando un montaje preciso y seguro.

2.4.1.3. Mesa

La mesa de trabajo (figura 7) del fabricante proporciona la plataforma de trabajo para la estación robotizada. Esta mesa es un elemento esencial que permite la correcta disposición y organización de los componentes durante el proceso de montaje.

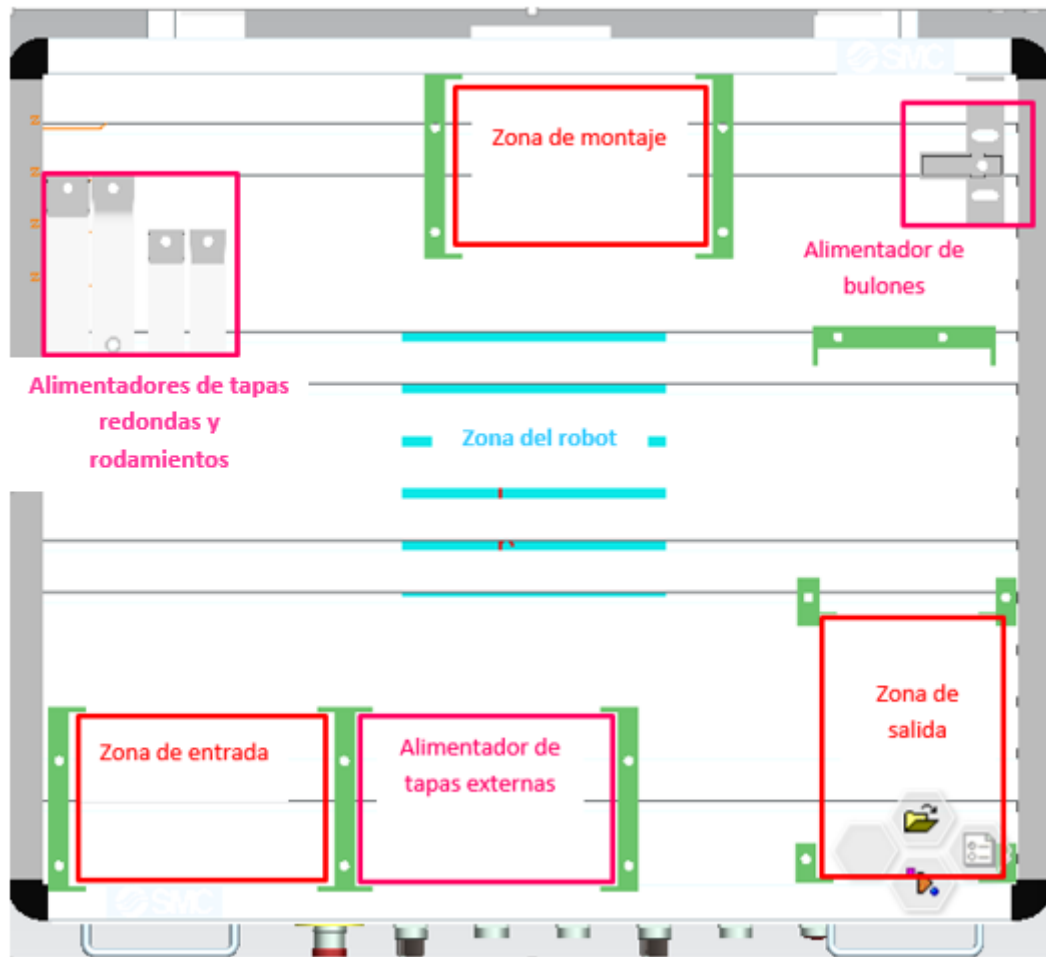


Figura 7. Mesa de trabajo

La mesa, elemento esencial en la estación robotizada, se encuentra organizada de manera eficiente para optimizar el proceso de ensamblaje del conjunto. Esta mesa se compone de diversas zonas delimitadas para garantizar un flujo de trabajo óptimo y sincronizado.

En primer lugar, se encuentran las cuatro zonas delimitadas por marcos 3D, diseñados para alojar los palets y los elementos esenciales del proceso de ensamblaje. La "Zona de Entrada de Palets" es el punto inicial donde se introducen los palets que contienen las bases y otros componentes necesarios para el montaje, según sea el caso. La "Zona de Montaje" es el área principal donde se lleva a cabo el ensamblaje propiamente dicho, y los marcos 3D aseguran la correcta disposición de

los elementos durante este proceso. La "Zona de Salida" es el punto de destino donde se coloca el palet con el conjunto ensamblado una vez completado el montaje. Además, una "Zona para el Palet Alimentador de Tapas exteriores " está destinada específicamente para recibir el palet que contiene las tapas exteriores requeridas para el ensamblaje final.

La mesa también incorpora dos áreas dedicadas a los "Alimentadores de Gravedad". En la primera zona se encuentran los alimentadores de rodamientos y de tapas interiores y en la segunda zona se encuentra el alimentador de bulones. Los alimentadores de gravedad proveen de forma continua y eficiente los elementos necesarios para el proceso de ensamblaje, asegurando un suministro constante durante la operación.

Por último, en el centro de la mesa se ubica el robot, el protagonista del proceso de montaje. Esta "Zona Central" es el punto desde el cual el robot UR3e realiza la manipulación y ensamblaje de los componentes con precisión y destreza. Gracias a esta ubicación estratégica, el robot puede acceder a todas las áreas de la mesa, facilitando una interacción fluida con los palets y los alimentadores de gravedad.

2.4.1.4. Piezas del conjunto

Marcos para Ubicar Palets: Los marcos definen las áreas de ubicación para los palets de bases y del palet de las tapas exteriores.

Piezas de ensamblaje: En la figura 8, se puede ver las piezas del conjunto a ensamblar:

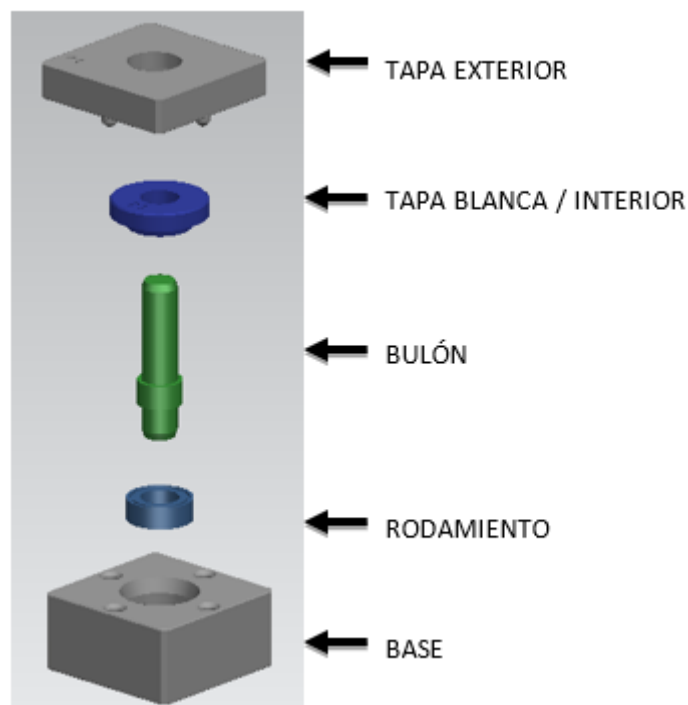


Figura 8. Piezas del conjunto

- **Bases:** Las bases son una parte fundamental del conjunto que se ensambla en la estación robotizada. Las bases actúan como la estructura principal sobre la cual se ensamblan los demás elementos, proporcionando una base sólida y estable para el conjunto final.
- **Rodamientos:** Los rodamientos son componentes esenciales para permitir el movimiento suave y fluido de las partes del conjunto.
- **Bulones:** Los bulones, también conocidos como tornillos o pernos, son elementos de fijación utilizados para unir las diferentes partes del conjunto.
- **Tapas interiores:** Las tapas interiores son componentes que se utilizan para cubrir el orificio en la base.
- **Tapas exteriores:** Las tapas exteriores son similares a las tapas interiores. Esta se usa para cubrir la pieza en conjunto.

Alimentadores por gravedad: Los alimentadores en la estación robotizada se dividen en tres tipos. La fabricación de cada alimentador se ha realizado en función de la pieza del conjunto a suministrar:

- Alimentador de Tapas interiores y Rodamientos (figura 9): Estos alimentadores tienen una forma de rampa y operan mediante la fuerza de la gravedad. En este caso, las tapas interiores y los rodamientos son dispuestos en la parte superior de la rampa, y gracias a la inclinación de esta, las piezas deslizan por la superficie y se desplazan hacia el extremo inferior. La disposición inclinada de la rampa permite un suministro constante.

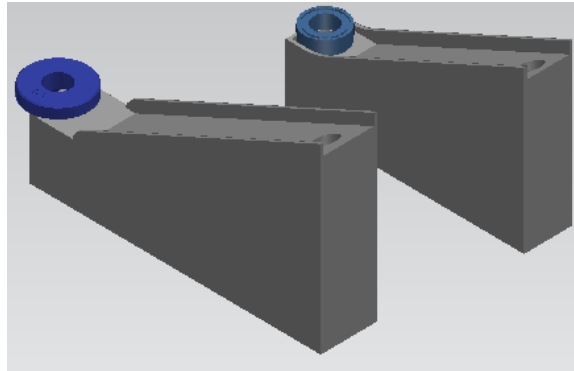


Figura 9. Alimentador de Tapas interiores y Rodamientos

- Alimentador de Bulones (figura 10): El último tipo de alimentador es el de bulones. Este alimentador tiene un formato vertical y también funciona por gravedad. Los bulones son colocados en una zona abierta superior, desde la cual caen debido a la fuerza de la gravedad. La pinza del robot UR3e es capaz de recoger los bulones en la zona abierta inferior para utilizarlos en el proceso de ensamblaje. La disposición vertical de este alimentador permite una fácil recogida de los bulones por parte del robot.

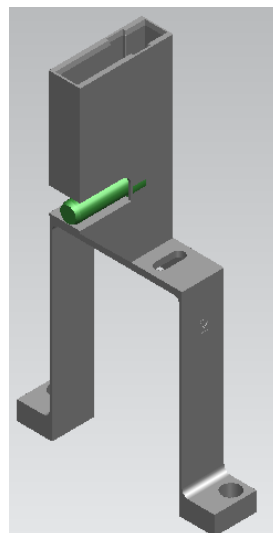


Figura 10. Alimentador de Bulones

Palet de entrada (figura11): El palet es una pieza 3D esencial en la estación robotizada, diseñado para almacenar y organizar hasta seis piezas, que pueden ser tapas exteriores o bases, utilizadas en el proceso de ensamblaje del conjunto. Este palet está estructurado con seis contenedores

individuales, como se puede ver en la figura 11, cada uno de ellos destinado a alojar una pieza específica. La disposición ordenada de las piezas en el palet es crucial para que el robot UR3e realice la manipulación y ensamblaje de manera precisa y eficiente.

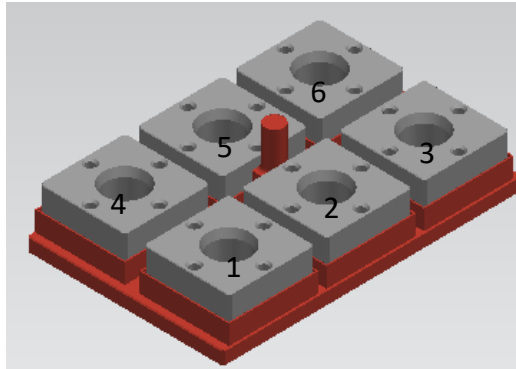


Figura 11. Palet de entrada

Cada pieza en el palet está debidamente enumerada, lo que permite una identificación clara y precisa de las piezas y su respectivo orden en el montaje del conjunto. La numeración proporciona una guía para el robot, permitiéndole coger los elementos en el orden adecuado, lo que resulta en un proceso de ensamblaje sin errores y evita posibles confusiones.

Gracias a la organización y estructura eficiente del palet, se facilita la manipulación y transporte de las piezas, garantizando una interacción fluida con el robot UR3e.

Alimentador de Tapas exteriores: Este alimentador está diseñado para acomodar las tapas exteriores. Las piezas que contiene este alimentador son similares a las bases que se encuentran en el palet de entrada, por lo que se utiliza un palet como alimentador. Como se puede observar en la figura 12, el diseño del alimentador permite una disposición ordenada de las tapas, asegurando que estén listas y posicionadas adecuadamente para el proceso de montaje.

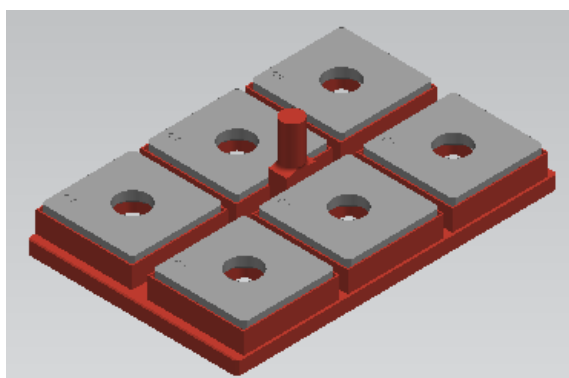


Figura 12. Alimentador de Tapas exteriores

2.4.2. GEMELO DIGITAL

En el contexto de los objetivos del proyecto, se destaca la necesidad de distinguir dos contextos esenciales: el Gemelo Digital en sí mismo y los servicios y consecuencias que se derivan de su implementación. Teniendo en cuenta esto y para lograr una creación efectiva del Gemelo Digital, se establece un método con tres pasos abiertos que permiten su desarrollo y aprovechamiento óptimo:

- Modelado visual.
- Modelado funcional.
- validación del gemelo digital.

Por otro lado, también es relevante tener en cuenta que en Tecnomatix Process Simulate existen los modos "Standalone" y "Line Simulation". Los cuales son dos configuraciones de trabajo que ofrecen diferentes enfoques para el uso de la herramienta, dependiendo de los requerimientos y objetivos del usuario. A continuación, se describen ambos modos:

Modo Standalone:

El modo "Standalone" es una configuración basada en tiempos, en la que Tecnomatix se utiliza como una herramienta independiente y autónoma, sin estar integrada con otros sistemas o software externos. En este modo, el usuario puede realizar simulaciones y análisis de procesos de forma aislada, sin la necesidad de conectar con otros programas de diseño o fabricación.

Esta modalidad es especialmente útil para realizar simulaciones de manera rápida y eficiente, sin requerir el acceso a datos externos o una conexión en tiempo real con el sistema de producción. Permite realizar pruebas y análisis de escenarios, evaluar el rendimiento de una estación de trabajo o proceso de ensamblaje, y realizar optimizaciones sin depender de otros sistemas.

Line Simulation:

El modo "Line Simulation" es una configuración basada en eventos, en la que Tecnomatix se integra con otros sistemas y software en un entorno de línea de producción más complejo. En este modo, Tecnomatix se conecta con otras herramientas de diseño, planificación y automatización para simular y coordinar toda una línea de producción.

El modo "Line Simulation" es ideal para el diseño y análisis de líneas de producción completas, donde múltiples estaciones de trabajo y procesos están interconectadas. Permite simular y optimizar la secuencia de operaciones en toda la línea, asegurando un flujo eficiente de material y una coordinación adecuada entre las diferentes estaciones.

Además, en este modo, Tecnomatix puede conectarse con sistemas de control en tiempo real, lo que permite la realización de simulaciones de producción en un entorno más cercano al de la producción real. Esto facilita el análisis de escenarios complejos y la toma de decisiones informadas para mejorar el rendimiento y la eficiencia de la línea de producción.

A continuación, se describen detalladamente cada uno de los pasos del método:

2.4.2.1. Modelado Visual

En esta etapa inicial, se realiza el modelado visual de la estación de montaje robotizada utilizando la herramienta Process Simulate. Se importan los modelos CAD de todos los elementos que componen la estación, como el robot UR3e, la mesa, las pinzas neumáticas, los palets, los alimentadores por gravedad y las piezas impresas en 3D.

El objetivo es obtener una representación virtual precisa y realista de la estación de montaje, donde cada componente se encuentra ubicado en su posición correspondiente y con sus dimensiones exactas. Este modelado visual permitirá visualizar la estación en su totalidad y preparar el terreno para el siguiente paso del método.

Implementación

El proceso de implementación de los modelos CAD en el proyecto es una etapa esencial para lograr una simulación virtual exitosa de la estación de montaje robotizada. Inicialmente, se procede al diseño y creación meticulosa de los modelos CAD de todos los elementos que conforman la estación de montaje o, en su defecto, se obtienen los modelos proporcionados por el fabricante. Estos modelos deben representar con precisión las dimensiones y características real de cada componente.

Una vez obtenidos los modelos CAD, se realiza la importación de estos a la herramienta Process Simulate. Esta herramienta ofrece soporte para una variedad de formatos de archivos CAD, como JT CAD, STEP o IGES, lo que facilita la integración de los modelos previamente creados.

Durante el proceso de importación, es esencial asegurarse de que los modelos CAD se ubiquen correctamente en la estación de montaje virtual para que reflejen fielmente la disposición real de los elementos en la estación real. Esta representación precisa es fundamental para obtener una réplica fiable de la estación de montaje en la simulación.

En el entorno del proyecto en Process Simulate, al realizar la creación del proyecto, se crea una carpeta por defecto. Dentro de esta carpeta, se añade una subcarpeta denominada "Librería", que servirá para organizar los elementos de la estación. La carpeta "Librería" se divide a su vez en dos subcarpetas: "Parts" y "Resources". En "Parts", se agrupan los elementos móviles que el robot manipulará, como bases, bulones y palets. En "Resources", se alojan el modelo del robot UR3e, la mesa, los alimentadores y otros recursos necesarios para crear la estación de montaje virtual.

Un paso crucial para el correcto funcionamiento del proyecto es vincularlo a la carpeta "Librería". Esto se logra mediante la configuración de la dirección de esta carpeta en la herramienta Process Simulate, asegurándose de que todos los modelos y recursos necesarios estén disponibles para su uso en la simulación. Esto se realiza en la pestaña de "File, Opciones, Disconnected" haciendo clic en los tres puntos y seleccionando la carpeta "librería".

Asimismo, al añadir cada elemento por primera vez en el simulador requiere seguir pasos específicos para garantizar que aseguren una correcta integración y representación del componente en la simulación. Estos pasos pueden variar según el tipo de elemento que se esté añadiendo, ya sea un robot, una herramienta del robot, un palet, un alimentador o cualquier otro componente de la estación de montaje. A continuación, se detallan los pasos generales para añadir elementos por primera vez en el simulador:

- Acceso al Modo "Standalone".
- Modelado del Componente: Antes de añadir el elemento al simulador, se debe asegurar que se cuenta con el modelo CAD del componente en cuestión. Este modelo CAD debe ser preciso y representar adecuadamente las características reales del elemento objeto de representación.
- Si el modelo tiene un archivo "TuneData.xml", es necesario eliminarlo para que la herramienta cree automáticamente el archivo con la ubicación específica del modelo.
- Clasificar el tipo de componente de todos los elementos mediante el comando "Define Component Type". Dependiendo del tipo de elemento que se esté añadiendo, se debe clasificar adecuadamente en el simulador. Por ejemplo, el robot será clasificado como "tipo Robot", la herramienta como "gripper", los palets y las piezas móviles como "part prototype", etc.
- Los componentes pueden añadirse a la herramienta seleccionando "Resources" o "Parts", según sea el caso, en "Object Tree" y utilizando la opción "Insert Component".

Una vez que todos los elementos de la estación de montaje robotizada han sido importados y configurados adecuadamente en la herramienta Process Simulate, se procede al modelado del escenario. El escenario es el entorno virtual en el que se desarrollará la simulación del proceso de ensamblaje de las piezas, y su diseño preciso es esencial para obtener resultados realistas y útiles.

En primer lugar y antes de realizar el montaje físico del robot UR3e sobre la mesa, es importante determinar la ubicación óptima para colocar el robot sobre ella. Este paso se lleva a cabo mediante el uso de la herramienta Robot Smart Place en el software Process Simulate.

El proceso para elegir la ubicación idónea del robot es el siguiente:

Configuración del Escenario: Se crea un escenario virtual en la herramienta Process Simulate que representa el entorno de trabajo de la estación de montaje. En este escenario, se posicionan visualmente la mesa y el robot UR3e.

Activación del Robot Smart Place: Se activa la función Robot Smart Place en la herramienta para comenzar el proceso de búsqueda de ubicaciones óptimas para el robot.

Definición de Puntos Clave: Se identifican los puntos clave dentro de la estación de montaje donde el robot deberá realizar tareas de ensamblaje y manipulación de piezas. Estos puntos pueden incluir la ubicación de los palets con las piezas a ensamblar, las áreas de alimentadores por gravedad y los puntos de ensamblaje específicos.

Generación de Ubicaciones Potenciales: Con los puntos clave definidos, el Robot Smart Place genera una serie de ubicaciones potenciales para el robot dentro del escenario, como se puede observar en la figura 13. Estas ubicaciones se distribuyen estratégicamente alrededor de los puntos clave para permitir un acceso eficiente a las piezas y optimizar el alcance del robot.

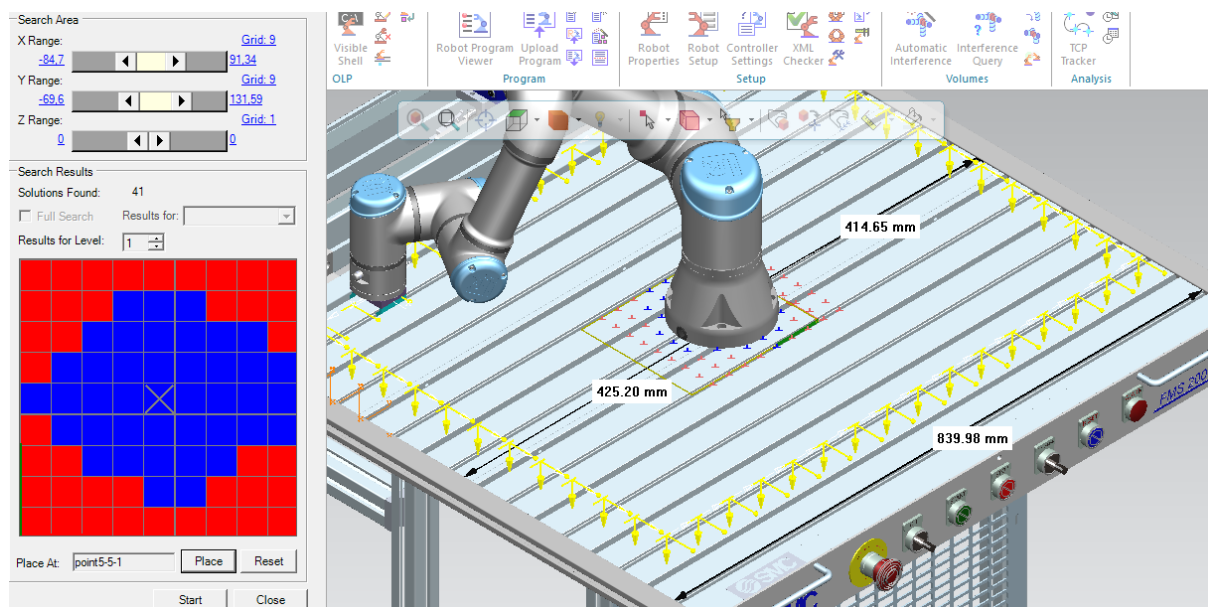


Figura 13. Ubicaciones potenciales

Evaluación de las Ubicaciones: Cada ubicación potencial se evalúa en función de su capacidad para alcanzar los puntos clave y realizar las operaciones de ensamblaje de manera eficiente. Se tienen en cuenta factores como la distancia recorrida por el robot, la minimización de colisiones y la facilidad de acceso a las piezas.

Selección de la Ubicación Óptima: Por último, se selecciona la ubicación que mejor cumple con los criterios de eficiencia y accesibilidad establecidos. Esta ubicación se considera la idónea para colocar el robot en la estación de montaje.

Una vez determinada la ubicación óptima del robot UR3e en el escenario virtual, se inicia la implementación de los demás componentes de la estación de montaje sobre la mesa. Este proceso implica colocar y ensamblar todos los elementos relevantes en sus ubicaciones correspondientes en el entorno virtual y físico de la estación robotizada.

Una vez completado el modelado visual, se obtiene una representación virtual completa de la estación de montaje robotizada, como se puede observar en la figura 14.

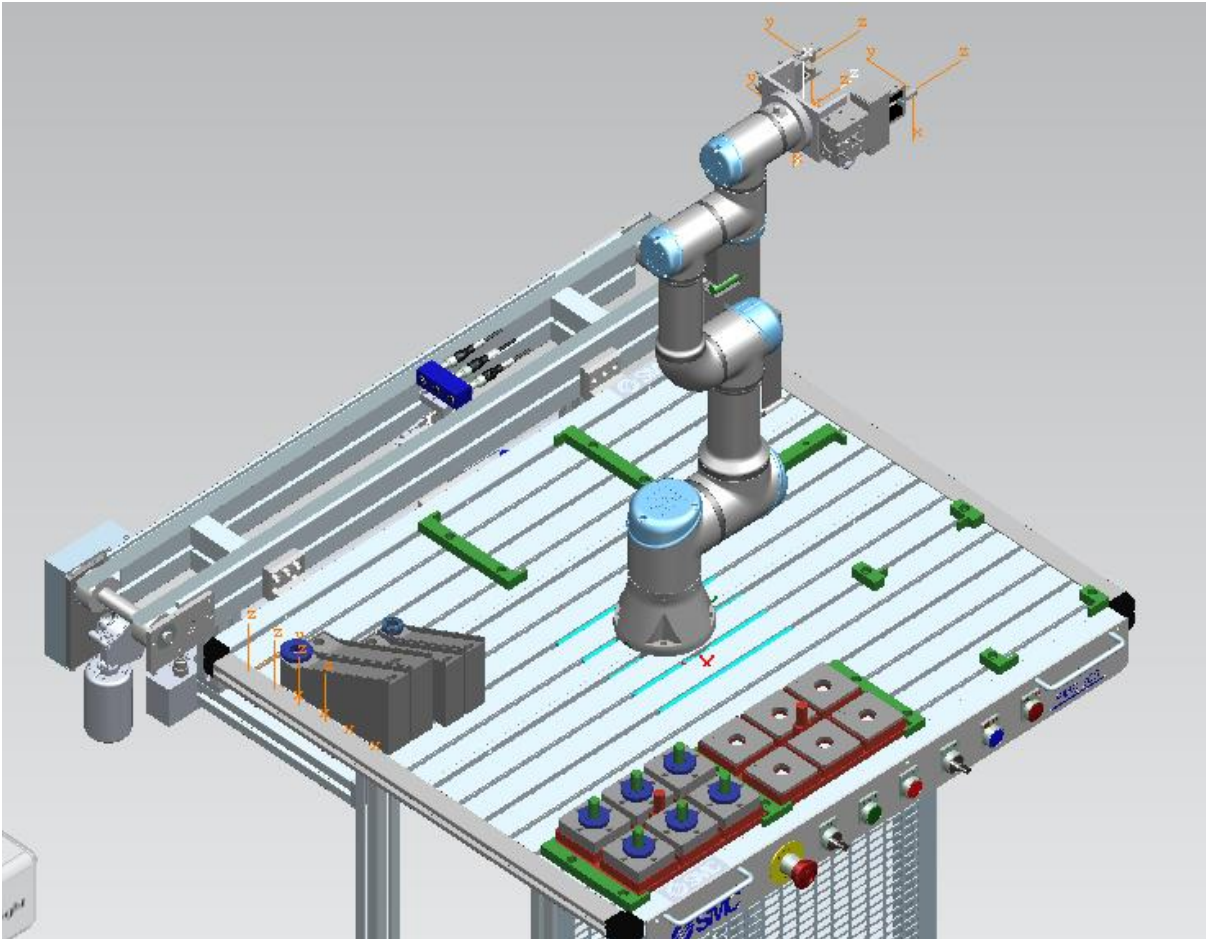


Figura 14. Modelado visual final de la estación de montaje

2.4.2.2. Modelado Funcional

En esta fase, se realiza el modelado funcional de la estación de montaje robotizada en el Gemelo Digital. Es decir, se asignan atributos y comportamientos específicos a cada elemento para simular su funcionamiento en el entorno virtual.

Se configuran las propiedades cinemáticas del robot UR3e, se establecen los parámetros de movimiento de las pinzas neumáticas, se definen las trayectorias de los palets y se especifican los flujos de material en los alimentadores por gravedad. Además, se establecen las interacciones entre los elementos para reflejar fielmente el proceso de ensamblaje de las piezas.

2.4.2.2.1. *Propiedades Cinemáticas del Robot UR3e*

En este apartado se configuran con precisión las propiedades cinemáticas del robot UR3e. Esto abarca aspectos como su alcance, velocidad y aceleración. Asimismo, se definen sus puntos de agarre y puntos de referencia. Además, se detallan las operaciones específicas que el robot debe realizar dentro de la estación de montaje.

Cada operación está diseñada para llevar a cabo una tarea particular en el proceso de ensamblaje, contribuyendo así a la creación del producto final. Teniendo en cuenta que, para lograr esto, se implementa una lógica de programación en el robot que optimiza la secuencia de tareas y asegura que cada palet sea ensamblado de manera completa antes de avanzar al siguiente.

La programación del robot se diseña de manera que se realice un ciclo completo de todas las tareas, si fuese necesario (colocación de rodamientos, bulones, tapas interiores y tapas exteriores) para cada palet individual antes de pasar al siguiente. Esto implica que el robot trabajará en un palet a la vez, asegurando que todas las piezas se ensamblen antes de moverse al siguiente palet.

También, al empezar el proceso de ensamblaje, se realiza una verificación de la cantidad de bases que contiene en palet. Si el palet está incompleto o las bases vienen con elementos añadidos, el robot tiene que ejecutar las tareas correspondientes para completar ese palet, con los elementos asignados antes de avanzar al siguiente.

A continuación, se detallarán las tareas creadas para que el robot realice el proceso de ensamblaje de la estación de montaje robotizada:

- **OP13 - Transporte de Palet de Entrada a la zona de montaje:** El robot ejecuta esta operación para trasladar el palet desde la zona de entrada hasta la zona de montaje. La precisión en el posicionamiento del palet es esencial para garantizar un proceso de ensamblaje eficiente.
- **Tarea 1_Rodamiento:** En esta tarea, se agrupan las operaciones necesarias para colocar el rodamiento en la pieza. El robot recoge el rodamiento del alimentador correspondiente y lo posiciona con precisión sobre la base adecuada en el palet.
- **Tarea 2_Bulón:** El robot lleva a cabo esta tarea, que engloba las operaciones para colocar el bulón en la estructura. El bulón es recogido del alimentador correspondiente y posicionado con exactitud sobre el rodamiento previamente colocado.
- **Tarea 3_TapaInt:** En esta tarea se realizan las operaciones requeridas para la colocación de las tapas interiores. El robot recoge la tapa interior del alimentador correspondiente y la posiciona con cuidado sobre el bulón previamente ensamblado.
- **Tarea 3_TapaExt:** Similar a la tarea anterior, esta operación agrupa las acciones para colocar las tapas exteriores en la estructura. El robot recoge la tapa exterior del alimentador apropiado y la coloca con precisión sobre la base correspondiente.

- **OP34 - Transporte de Palet de Montaje a Salida:** La última operación implica que el robot transporte el palet ensamblado desde la zona de montaje a la zona de salida. La colocación adecuada del palet es esencial para evitar daños a las piezas ensambladas.

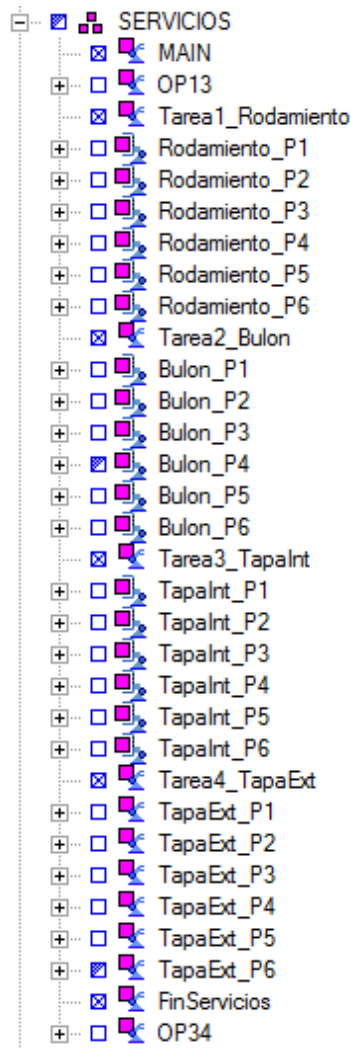




Figura 15. Operaciones del robot

En la figura 15 anterior, se puede observar todas las operaciones que realiza el robot. Estas operaciones se definen y coordinan a través de un programa principal denominado "MAIN". Este programa principal tiene la responsabilidad de gestionar y coordinar las diferentes tareas que el robot debe ejecutar, así como determinar cuántas operaciones se realizarán en cada tarea. El "MAIN " actúa como el controlador central que guía al robot a través del proceso de ensamblaje en la estación de montaje.

Una vez creadas todas las operaciones que realiza el robot, es necesario diseñar el programa de control del robot. En este contexto, emerge como un elemento fundamental la característica conocida como "Robot Program Inventory" dentro de la herramienta Tecnomatix, que permite la creación y organización de programas de control para robots en un entorno de simulación y

desempeña un papel fundamental para garantizar la sincronización y coordinación efectiva de todas las operaciones que el robot UR3e debe realizar en el proceso de ensamblaje.

Para crear el programa de control del robot, es necesario elegir en el robot y elegir "Robotic Program Inventory" (figura 16), presionar el icono  y asignarle un nombre, como por ejemplo "UR3e_Program". A continuación, se le asigna el programa como por defecto al robot con el icono .

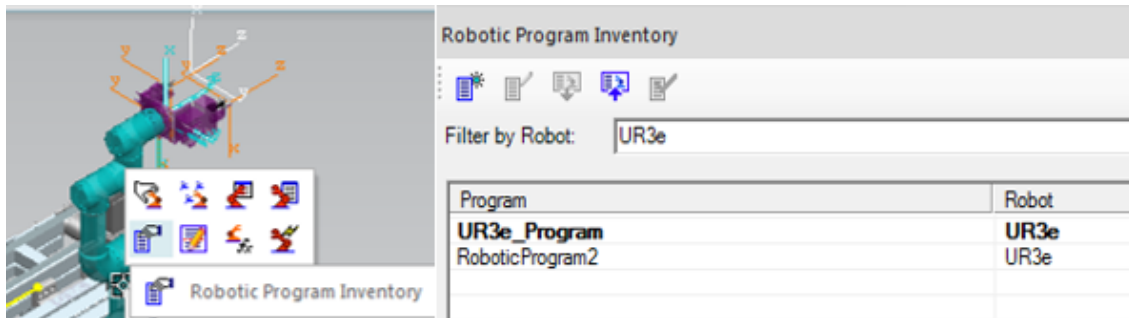




Figura 16. Robot Program Inventory

Por último, a través del icono "open in Program Editor"  se añade el programa principal "MAIN" presionando el icono . A este programa se le vinculan todas las operaciones que el robot puede realizar y se le asigna un valor de "Path" igual a 1. De esta forma, cuando se esté realizando la simulación en Línea (explicación en detalle en el apartado 3.4.2.3) y el robot reciba el valor del Path igual a 1, a través de la variable del robot "programNumber", el robot solo podrá ejecutar los comandos que ordene el "MAIN".

2.4.2.2.2. **Parámetros de Movimiento de las Pinzas Neumáticas**

En esta fase de modelado funcional, se hace hincapié en definir con precisión cómo se mueven las pinzas neumáticas en respuesta a las señales y comandos. Los "Parámetros de Movimiento" incluyen aspectos como la velocidad de apertura y cierre de las pinzas, los límites de su movimiento, y las restricciones basadas en las características mecánicas y físicas de las pinzas reales.

La técnica de "joint dependency" en Tecnomatix se aplica para establecer las conexiones lógicas y funcionales entre las diferentes partes móviles de las pinzas neumáticas. Esto asegura que el movimiento de un dedo de la pinza afecte adecuadamente el movimiento de los otros dedos de manera realista y coherente, simulando así el comportamiento sincronizado de las pinzas en el mundo físico.

De igual forma, el "Pose Editor" es una herramienta clave en este proceso. Ya que permite definir y ajustar las posiciones y orientaciones de los componentes móviles, como los dedos de las pinzas, de manera precisa. Con esta herramienta se definen dos poses en cada pinza:

Pinza 1: Esta pinza es la encargada de manejar los rodamientos, las tapas interiores y las tapas exteriores en el proceso de ensamblaje.

En esta tarea, uno de los aspectos fundamentales es definir los diámetros de los agujeros por donde la pinza 1 agarra los rodamientos y las tapas interiores. Estos diámetros son idénticos, lo que permite que la pinza pueda manipular ambos elementos de manera uniforme. Por otro lado, las tapas exteriores presentan un diámetro diferente, lo que conlleva la necesidad de establecer tres poses distintas para lograr un control preciso y efectivo de su manipulación.

Las poses definidas son las siguientes:

- **HOME:** Esta es la posición inicial, la pinza se encuentra en su estado inicial, con las mandíbulas cerradas y lista para realizar movimientos y tareas.
- **Rodamiento_CLOSED:** Esta pose se configura para la manipulación de los rodamientos y las tapas interiores. Con esta pose la apertura de la pinza se adapta para asegurar un agarre firme y preciso de estos componentes.
- **TapaExt_CLOSED:** Esta pose se especifica para las tapas exteriores, que presentan un diámetro distinto al de los rodamientos y las tapas interiores. La configuración de esta pose permite que la apertura de la pinza pueda ajustarse de manera óptima para agarrar y manipular las tapas exteriores de manera eficiente.

Pinza 2 Esta pinza desempeña un papel crucial al encargarse de tomar los bulones y los palets en el proceso de ensamblaje. Es importante destacar que tanto los bulones como los palets tienen un diámetro de agarre idéntico, lo que simplifica la configuración de movimientos y poses.

Para asegurar un control preciso y efectivo de la pinza 2, se definen dos poses clave:

- **HOME:** En esta posición, la pinza 2 se encuentra en su estado inicial, con las mandíbulas abiertas y lista para realizar movimientos y tareas.
- **Bulon_Closed:** Esta pose se configura para la captura de los bulones y los palets. Esta única pose permite que la pinza se cierre de manera óptima, garantizando un agarre seguro y confiable de los componentes.

2.4.2.2.3. Flujo de Material

En este apartado es importante comprender y detallar el flujo de material que caracteriza la estación de montaje robotizada, considerando tanto los componentes de los Alimentadores por Gravedad como los elementos presentes en los palets de entrada y de tapas exteriores. Para una representación coherente y efectiva, se ha segmentado esta consideración en dos grupos principales: las variantes del palet de entrada y los elementos de ensamblaje.

Variantes del Palet de Entrada:

Este aspecto del modelado funcional se centra en la diversidad de vertientes que puede tener el palet de entrada. Esto implica comprender cómo los elementos en el palet se presentan. Se consideran tanto los diferentes números de bases que el palet puede contener, como también las posibles combinaciones de rodamientos, bulones y tapas interiores que pueden estar presentes en las bases. Estas variantes generan cuatro grupos específicos, denominados "CASO", cada uno compuesto por seis elementos.

Para gestionar eficientemente estos flujos, se debe crear un conjunto de operaciones que abarque cada una de las situaciones presentes en los diferentes casos de entrada. Cada operación individual está vinculada a un objeto específico y único dentro de la estación.

Un ejemplo ilustrativo de esta estrategia es la creación de una operación nominalmente denominada "IN_CASO1_1", que se puede observar en las imágenes 17. A esta operación se le asigna el objeto CAD correspondiente, en este caso denominado "IN_Caso1_1" (figura 18). Dicho objeto representa el palet de entrada con una única base y sin ningún otro elemento adicional.

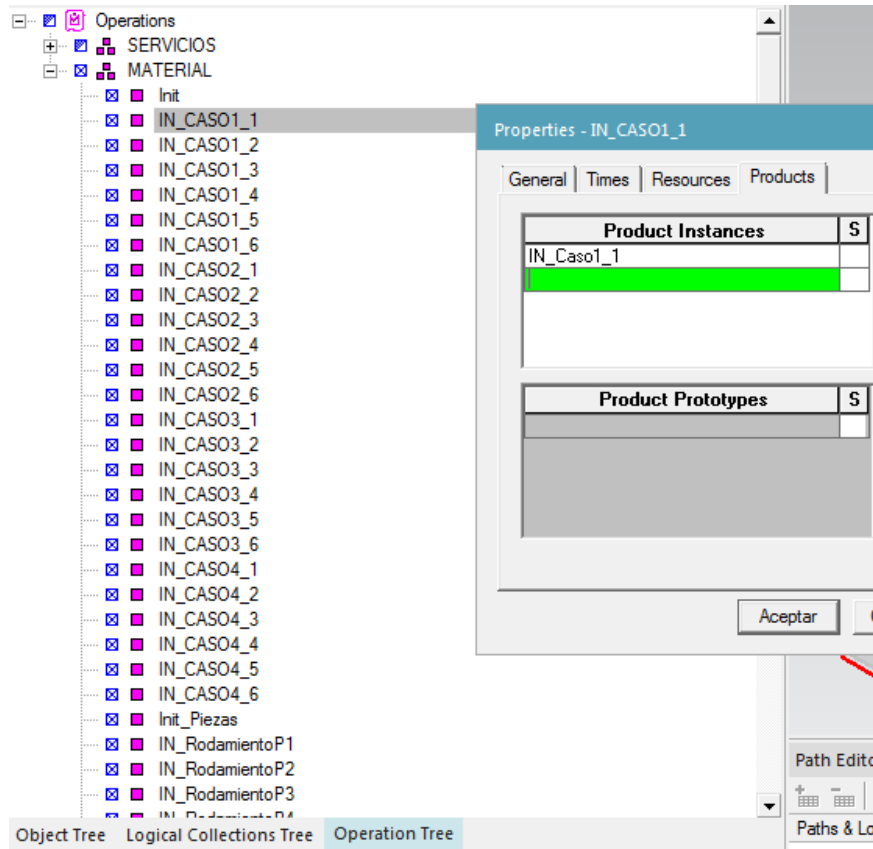


Figura 17. Instancia de la operación IN_CASO1_1

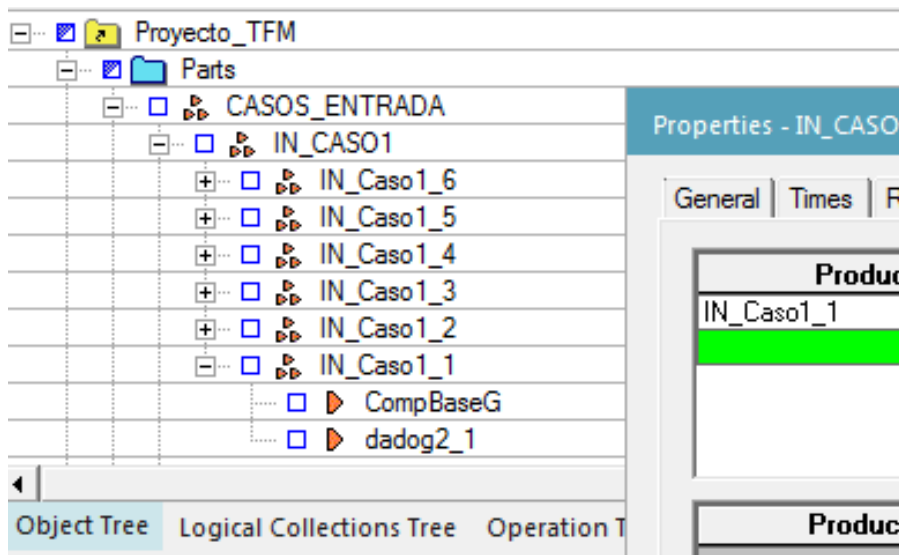


Figura 18. Selección del objeto CAD

Elementos de Ensamblaje:

Este aspecto del modelado funcional se centra en los elementos de ensamblaje propiamente dichos. Se detalla cómo el robot interactúa con los rodamientos, bulones, tapas interiores y tapas exteriores, desde su recogida de los alimentadores por gravedad hasta su colocación en las ubicaciones adecuadas en las bases. Además, se establece cómo el robot realiza las secuencias de ensamblaje, desde su recolección de los alimentadores por gravedad hasta su colocación precisa en las ubicaciones destinadas sobre las bases.

Siguiendo una lógica similar a la aplicada en las variantes del palet de entrada, se crean operaciones específicas para cada elemento. Cada uno de estos componentes únicos del ensamblaje cuenta con su propia operación asignada, en un total de seis elementos exclusivos por cada alimentador.

2.4.2.2.4. Interacciones y Ensamblaje

Se definen las interacciones entre los elementos de la estación, modelando con precisión cómo las piezas son manipuladas, ensambladas y desplazadas. Se programan las secuencias de operación para lograr un ensamblaje fluido y preciso.

Material flow simulation:

En esta etapa se define cómo los diversos elementos interactúan entre sí durante el proceso de ensamblaje. Es fundamental reconocer que, para llevar a cabo esta fase, es necesario cambiar la herramienta al modo "Line simulation".

Una vez que se han establecido todas las operaciones para las diferentes variantes del palet de entrada y los elementos de ensamblaje, el siguiente paso es configurar el flujo de materiales en la sección de "material flow simulation". En este proceso, se realiza la vinculación de cada operación que genera una instancia de un elemento móvil específico con las tareas en las cuales se utilizarán dichos elementos, como se puede observar en la figura 19.

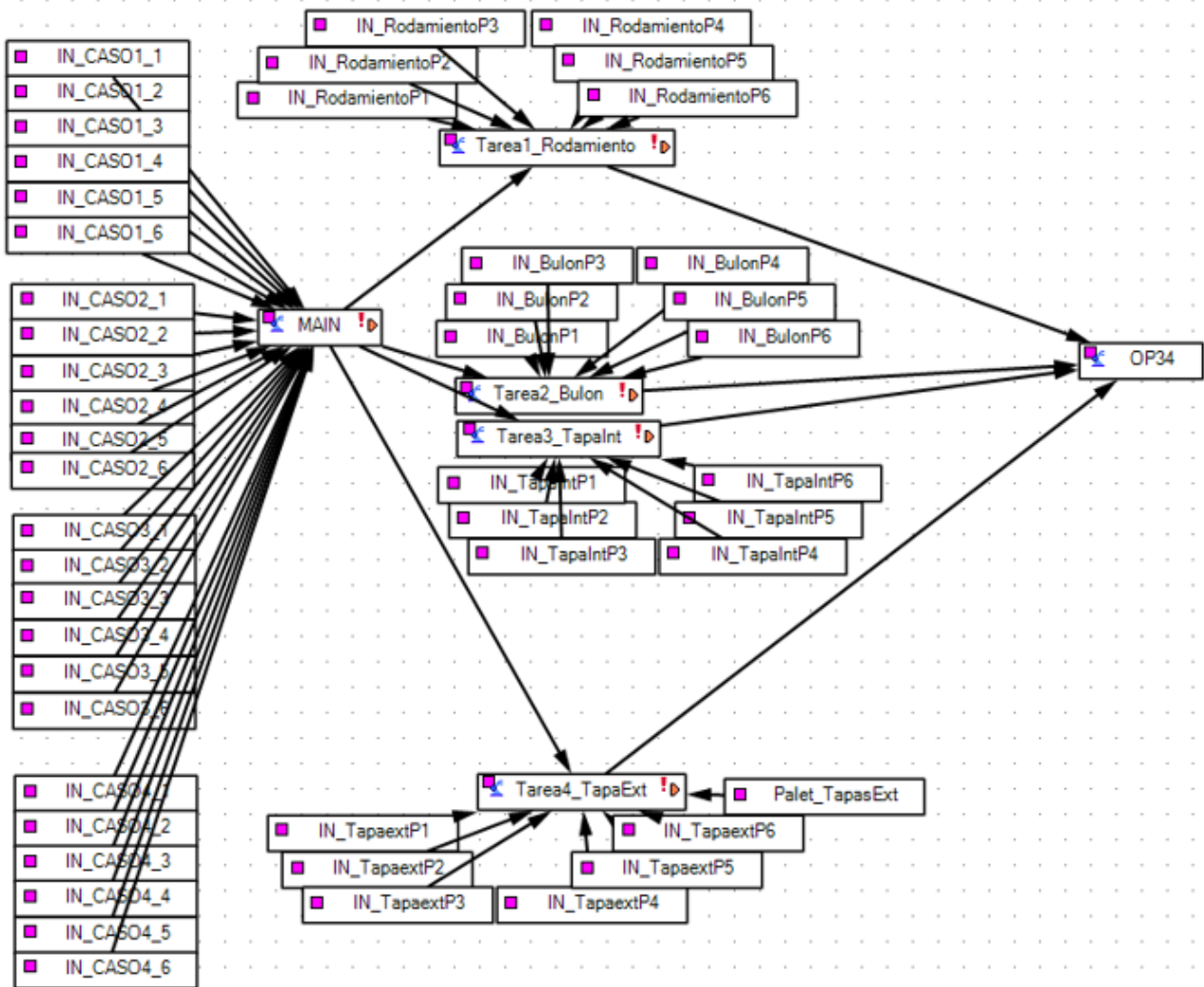


Figura 19. Material flow simulation

Sequence Editor:

En el proceso de modelado funcional, la fase de Interacciones y Ensamblaje se ve enriquecida por la utilización de una herramienta clave: el Sequence Editor (Editor de Secuencias). Esta herramienta desencadena una representación secuencial detallada de cómo los componentes interactúan y se ensamblan en la estación de montaje robotizada.

El Sequence Editor permite establecer y ajustar minuciosamente la secuencia en la que se llevan a cabo las operaciones de ensamblaje. Cada paso es definido con precisión, desde la recogida de los elementos móviles de los alimentadores por gravedad hasta su colocación en las ubicaciones adecuadas en las bases. Además, se pueden programar transiciones fluidas entre las distintas fases del proceso, garantizando que cada acción se realice de manera coordinada y sin conflictos.

Por otro lado, Sequence Editor permite visualizar y validar el flujo de operaciones antes de ejecutar la simulación completa. Esto facilita la identificación de posibles problemas o áreas de mejora en la secuencia de ensamblaje. A medida que se ajustan y perfeccionan las secuencias en esta fase de

modelado, se logra una simulación más precisa y efectiva de cómo los elementos interactúan y se ensamblan en la estación de montaje robotizada

Mediante el Sequence Editor, se traza una representación detallada de las secuencias de acciones que rigen las operaciones de ensamblaje. Su versatilidad radica en la capacidad de crear dos tipos de secuencias cruciales para el funcionamiento de la estación:

Secuencia de Servicios: En esta secuencia, se establece la lógica que guía las tareas y operaciones que el robot debe realizar (figura 20). Cada servicio, como la colocación de rodamientos, bulones o tapas, se programa meticulosamente en función de una secuencia temporal optimizada. Esto implica activar los movimientos del robot en el momento adecuado, coordinar la recogida y colocación de elementos y garantizar que todas las acciones se ejecuten en armonía.

Es importante destacar que la Secuencia de Servicios está directamente influenciada por la operación “MAIN”, la cual tiene vinculada un programa principal. Este programa principal guía las acciones generales del robot, gobierna la secuencia de tareas y operaciones en la Secuencia de Servicios. Esto asegura que todas las acciones realizadas por el robot se coordinen de manera coherente y eficiente por lo que no es necesario definir las transiciones para pasar de una acción a otra, si no se elige una transición simultanea (figura 21).

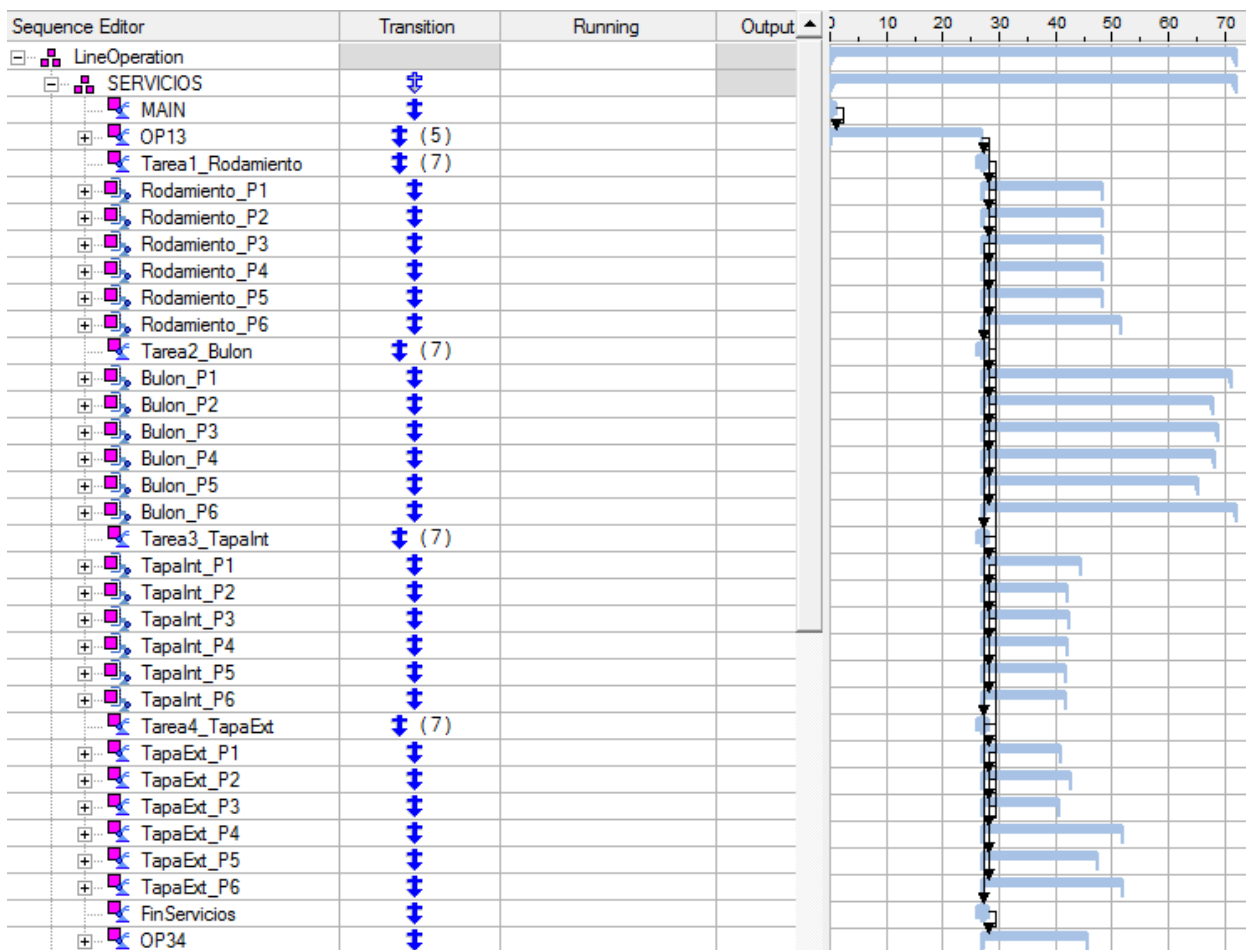
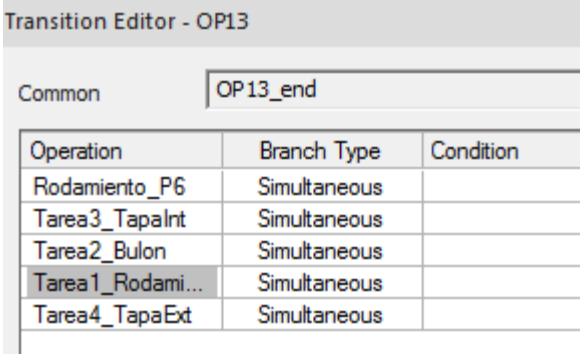


Figura 20. Sequence Editor



Operation	Branch Type	Condition
Rodamiento_P6	Simultaneous	
Tarea3_TapaInt	Simultaneous	
Tarea2_Bulon	Simultaneous	
Tarea1_Rodami...	Simultaneous	
Tarea4_TapaExt	Simultaneous	

Figura 21. Transiciones de la operación OP13

Secuencia de Materiales: Esta secuencia se centra en el flujo de los elementos móviles desde los palets con las bases hasta los elementos móviles de los alimentadores por gravedad. Se define cómo los componentes se desplazan a lo largo del proceso, desde su recogida hasta su ensamblaje en las piezas finales.

Esta segunda secuencia, introduce un nivel adicional de complejidad para garantizar un flujo de componentes fluido y preciso en la estación de montaje robotizada. A diferencia de la secuencia de servicios, en la secuencia de materiales es esencial definir transiciones bien coordinadas para determinar la aparición de elementos específicos en el proceso de ensamblaje.

De igual forma que la secuencia de servicios, la secuencia de materiales tiene dos operaciones “Init” o “Init_piezas” que controlan las demás operaciones. Sin embargo, estas operaciones no tienen vinculadas ningún programa y la forma de controlar el flujo de materiales se realiza con unas condiciones asociadas.

Operación Init: Esta operación marca el inicio de una de las ramificaciones de la secuencia de materiales y controla la secuencia. Su función es crucial para establecer el flujo inicial de los elementos. Las condiciones asociadas a esta operación son responsables de determinar qué elemento será introducido en el proceso de ensamblaje (figura 21). Dependiendo de estas condiciones, se dirigirá la secuencia hacia la aparición de un componente específico, como un rodamiento, un bulón o una tapa además de las bases.

Transition Editor - Init

Common RE(UR3e_startProgram)

Operation	Branch Type	Condition
IN_CASO3_1	Alternative	NItems = 1 AND Casos_Entrada = 3
IN_CASO2_4	Alternative	NItems = 4 AND Casos_Entrada = 2
IN_CASO2_6	Alternative	NItems = 6 AND Casos_Entrada = 2
IN_CASO4_3	Alternative	NItems = 3 AND Casos_Entrada = 4
IN_CASO4_1	Alternative	NItems = 1 AND Casos_Entrada = 4
IN_CASO1_5	Alternative	NItems = 5 AND Casos_Entrada = 1
IN_CASO2_2	Alternative	NItems = 2 AND Casos_Entrada = 2
IN_CASO1_3	Alternative	NItems = 3 AND Casos_Entrada = 1
IN_CASO1_4	Alternative	NItems = 4 AND Casos_Entrada = 1
IN_CASO3_5	Alternative	NItems = 5 AND Casos_Entrada = 3
IN_CASO4_5	Alternative	NItems = 5 AND Casos_Entrada = 4
IN_CASO4_4	Alternative	NItems = 4 AND Casos_Entrada = 4
IN_CASO1_6	Alternative	NItems = 6 AND Casos_Entrada = 1
IN_CASO3_3	Alternative	NItems = 3 AND Casos_Entrada = 3
IN_CASO1_1	Alternative	NItems = 1 AND Casos_Entrada = 1
IN_CASO3_2	Alternative	NItems = 2 AND Casos_Entrada = 3
IN_CASO3_6	Alternative	NItems = 6 AND Casos_Entrada = 3
IN_CASO3_4	Alternative	NItems = 4 AND Casos_Entrada = 3
IN_CASO4_2	Alternative	NItems = 2 AND Casos_Entrada = 4
IN_CASO2_1	Alternative	NItems = 1 AND Casos_Entrada = 2
IN_CASO2_3	Alternative	NItems = 3 AND Casos_Entrada = 2
IN_CASO4_6	Alternative	NItems = 6 AND Casos_Entrada = 4
IN_CASO1_2	Alternative	NItems = 2 AND Casos_Entrada = 1
IN_CASO2_5	Alternative	NItems = 5 AND Casos_Entrada = 2

Figura 22. Transiciones asociadas a la operación Init

Operación Init_piezas: La otra rama de la secuencia de materiales se inicia con esta operación principal. Similar a Init, Init_piezas controla la secuencia y está vinculada a condiciones específicas que determinan qué elemento será introducido en el proceso de ensamblaje. Las condiciones asociadas a esta operación permiten la aparición de otro conjunto de componentes, lo que completa las opciones disponibles en el flujo de materiales. Sin embargo, es importante señalar que las condiciones asociadas a esta operación adquieren una mayor complejidad debido a su dependencia de un conjunto más amplio de variables.

En particular, las condiciones asociadas a esta operación (figura 23) están intrincadamente vinculadas al caso de entrada, el servicio requerido y una variable denominada "contador". Esta variable, cuyo valor evoluciona a lo largo del proceso, es modificada según la tarea que está ejecutando el robot. De esta manera, el valor del contador se convierte en un marcador fundamental que guía la toma de decisiones y define las transiciones de los elementos en la secuencia de materiales.

Transition Editor - Init_Piezas

Common Edit C

Operation	Branch Type	Condition
IN_RodamientoP4	Alternative	RE(Casos_Entrada = 1 AND Contador = 4)
IN_RodamientoP2	Alternative	RE(Casos_Entrada = 1 AND Contador = 2)
IN_RodamientoP6	Alternative	RE(Casos_Entrada = 1 AND Contador = 6)
IN_TapaextP1	Alternative	RE((Service = 4 OR Service = 10 OR Service = 7 OR Service = 9) AND (NItems = 1 OR NItems = 2 OR NItems = 3 OR NItems = 4 OR NItems = 5 OR NItems = 6))
IN_TapaIntP4	Alternative	RE((Service = 3 OR Service = 4 OR Service = 6 OR Service = 7 OR Service = 8 OR Service = 9) AND Contador = 16)
IN_TapaIntP3	Alternative	RE((Service = 3 OR Service = 4 OR Service = 6 OR Service = 7 OR Service = 8 OR Service = 9) AND Contador = 15)
IN_TapaextP4	Alternative	RE((Service = 4 OR Service = 10 OR Service = 7 OR Service = 9) AND (NItems = 4 OR NItems = 5 OR NItems = 6))
IN_TapaIntP5	Alternative	RE((Service = 3 OR Service = 4 OR Service = 6 OR Service = 7 OR Service = 8 OR Service = 9) AND Contador = 17)
IN_BulonP6	Alternative	RE((Service = 2 OR Service = 3 OR Service = 4 OR Service = 5 OR Service = 6 OR Service = 7) AND Contador = 12)
IN_RodamientoP5	Alternative	RE(Casos_Entrada = 1 AND Contador = 5)
IN_RodamientoP3	Alternative	RE(Casos_Entrada = 1 AND Contador = 3)
IN_TapaIntP1	Alternative	RE((Service = 3 OR Service = 4 OR Service = 6 OR Service = 7 OR Service = 8 OR Service = 9) AND Contador = 13)
IN_TapaextP5	Alternative	RE((Service = 4 OR Service = 10 OR Service = 7 OR Service = 9) AND (NItems = 5 OR NItems = 6))
IN_BulonP3	Alternative	RE((Service = 2 OR Service = 3 OR Service = 4 OR Service = 5 OR Service = 6 OR Service = 7) AND Contador = 9)
IN_BulonP1	Alternative	RE((Service = 2 OR Service = 3 OR Service = 4 OR Service = 5 OR Service = 6 OR Service = 7) AND Contador = 7)
IN_TapaextP6	Alternative	RE((Service = 4 OR Service = 10 OR Service = 7 OR Service = 9) AND NItems = 6)
IN_BulonP5	Alternative	RE((Service = 2 OR Service = 3 OR Service = 4 OR Service = 5 OR Service = 6 OR Service = 7) AND Contador = 11)
Palet_TapasExt	Alternative	RE((Service = 4 OR Service = 7 OR Service = 9 OR Service = 10))
IN_BulonP4	Alternative	RE((Service = 2 OR Service = 3 OR Service = 4 OR Service = 5 OR Service = 6 OR Service = 7) AND Contador = 10)
IN_RodamientoP1	Alternative	RE(Casos_Entrada = 1 AND Contador = 1)
IN_TapaextP3	Alternative	RE((Service = 4 OR Service = 10 OR Service = 7 OR Service = 9) AND (NItems = 3 OR NItems = 4 OR NItems = 5 OR NItems = 6))
IN_TapaextP2	Alternative	RE((Service = 4 OR Service = 10 OR Service = 7 OR Service = 9) AND (NItems = 2 OR NItems = 3 OR NItems = 4 OR NItems = 5 OR NItems = 6))
IN_TapaIntP2	Alternative	RE((Service = 3 OR Service = 4 OR Service = 6 OR Service = 7 OR Service = 8 OR Service = 9) AND Contador = 14)
IN_TapaIntP6	Alternative	RE((Service = 3 OR Service = 4 OR Service = 6 OR Service = 7 OR Service = 8 OR Service = 9) AND Contador = 18)
IN_BulonP2	Alternative	RE((Service = 2 OR Service = 3 OR Service = 4 OR Service = 5 OR Service = 6 OR Service = 7) AND Contador = 8)

Figura 23. Transiciones asociadas a la operación Init_Piezas

Ensamblaje:

Una etapa crítica en el proceso de simulación es la operación final denominada OP34, que se dedica a trasladar el palet desde la zona de montaje hasta la zona de salida. La relevancia de esta operación radica en su capacidad para demostrar de manera tangible y efectiva cómo las piezas que han sido añadidas a las bases en la zona de montaje interactúan de manera conjunta durante el desplazamiento del palet de un punto a otro.

En esta operación, se busca visualizar el resultado integral del proceso de ensamblaje, donde cada componente en las bases ha sido cuidadosamente colocado y ensamblado. Al mover el palet, se puede apreciar cómo las piezas previamente ensambladas interactúan, manteniendo su disposición y relación dentro de las bases. Esta simulación proporciona una representación precisa y realista de cómo los elementos individuales se unen para formar un conjunto funcional.

En la figura 24, se puede observar cómo se realiza la vinculación. Antes de ejecutar la operación de recogida (pick) del palet, se lleva a cabo un proceso de vinculación de los elementos seleccionados, una estrategia fundamental para asegurar la correcta interacción y movimiento coordinado durante la manipulación y transporte.

Este proceso de vinculación se lleva a cabo a través de comandos OLP (Offline Programming) dentro de la operación en cuestión. Los comandos OLP permiten establecer conexiones lógicas entre los componentes, creando una relación que asegura que los elementos designados se mantengan unidos y se muevan como una entidad única.

La vinculación se realiza de manera específica y selectiva, identificando los elementos que deben mantenerse juntos durante el traslado del palet. Estos elementos pueden incluir rodamientos, bulones, tapas interiores y tapas exteriores, cuya correcta disposición es crucial para garantizar la funcionalidad y el rendimiento del conjunto final. En la figura 24, se puede observar cómo se vinculan los rodamientos.

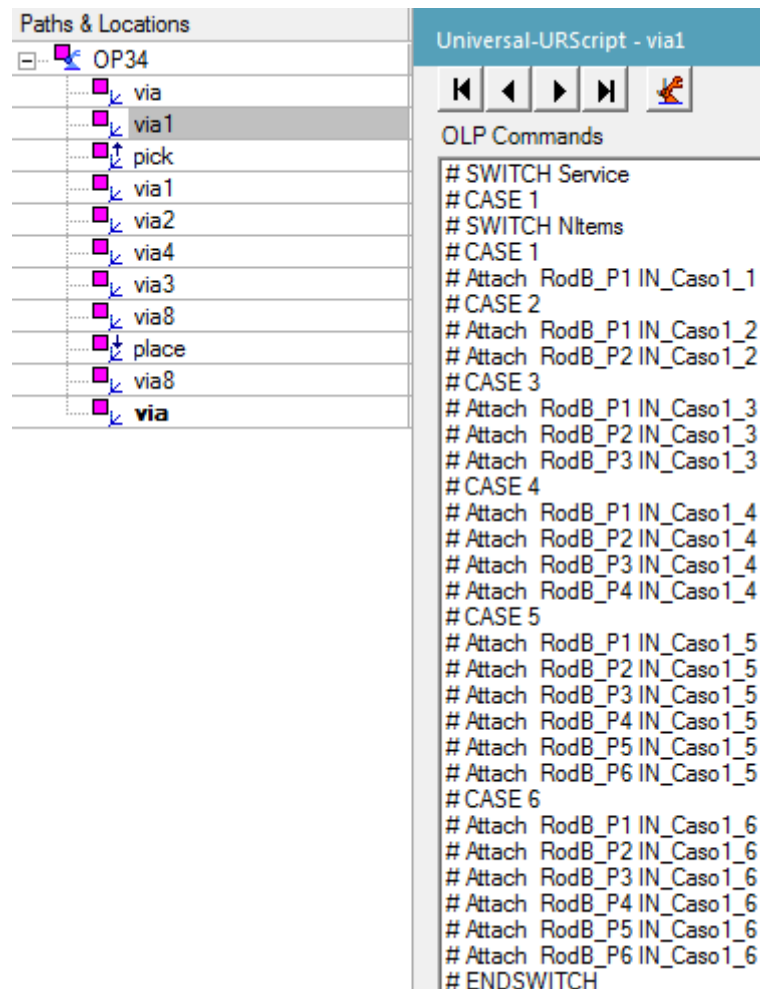


Figura 24. Ejemplo de vinculación de los rodamientos

De igual forma cuando se envía la señal de “Palet_Recogido”, se vacían los elementos que se encuentran en la zona de salida, ya que se entiende que el palet ha salido de la estación. Esta operación se ejecuta con el comando “Blank”, que oculta los elementos deseados.

2.4.2.3. Validación del Gemelo Digital

La fase de validación del gemelo digital es un paso esencial en el proceso de desarrollo y validación de la estación de montaje robotizada. Una vez que se ha completado el modelado funcional, la asignación de tareas, las interacciones y el ensamblaje de los componentes en el entorno virtual, es necesario llevar a cabo una rigurosa validación para garantizar que el gemelo digital sea una representación precisa y fiable de la estación real.

Esta etapa de validación implica una serie de pasos críticos destinados a evaluar la eficacia y la precisión del gemelo digital en simular el proceso de ensamblaje real. Estos pasos incluyen:

2.4.2.3.1. Validación del gemelo digital en la simulación

Se realiza una simulación completa del proceso de ensamblaje utilizando el gemelo digital. Durante esta simulación, se ejecutan las secuencias de tareas programadas, se monitorizan las interacciones entre los componentes y se observa cómo se desarrolla cada fase del proceso. Esto permite identificar posibles problemas o inconsistencias que puedan surgir durante la operación real.

La inclusión de señales adecuadas en la simulación es fundamental para recrear situaciones reales. Estas señales pueden representar entradas y salidas de sensores, estados de componentes, movimientos del robot, tiempos de ciclo, entre otros aspectos relevantes.

La mayor parte de las señales del robot son predefinidas automáticamente y llevan el nombre distintivo que comienza con “UR3e” (figura 25). Son generadas internamente por la herramienta Tecnomatix, en función de los parámetros y propiedades definidos para el robot y otros elementos.

Además de las señales predefinidas, se han creado señales personalizadas para capturar condiciones y eventos específicos dentro del proceso de ensamblaje (figura 25). Estas señales diseñadas a medida se ajustan a las necesidades particulares del proyecto y permiten monitorizar detalles críticos que influyen en el flujo de trabajo. Por ejemplo, se han creado señales para registrar la introducción del palet de bases, la cantidad de bases contenidas en el palet, el servicio a realizar y otras variables clave.

Robot Signals - " UR3e "

PLC Signal Name	Robot Signal Name	I/O	Signal Function	HW T...	Address	External Connection
UR3e_startProgram	startProgram	Q	Starting Program	BOOL	No Address	
UR3e_programNumber	programNumber	Q	Program Number	BYTE	No Address	
UR3e_emergencyStop	emergencyStop	Q	Program Emergency Stop	BOOL	No Address	
UR3e_programPause	programPause	Q	Program Pause	BOOL	No Address	
UR3e_programEnded	programEnded	I	Ending Program	BOOL	No Address	
UR3e_mirorProgramNumber	mirorProgramNumber	I	Mirror Program Number	BYTE	No Address	
UR3e_errorProgramNumber	errorProgramNumber	I	Error Program Number	BOOL	No Address	
UR3e_robotReady	robotReady	I	Robot Ready	BOOL	No Address	
Service	Service	Q		DINT	No Address	
NItems	NItems	Q		DINT	No Address	
Casos_Entrada	Casos_Entrada	Q		DINT	No Address	
Palet_Recogido	Palet_Recogido	Q		BOOL	No Address	
Contador	Contador	I		DINT	No Address	
Servicio_MalElegido	Servicio_MalElegido	I		BOOL	No Address	
UR3e_at_HOME	HOME	I	Pose Signal	BOOL	No Address	
UR3e_at_pos1	pos1	I	Pose Signal	BOOL	No Address	

Figura 25. Señales del robot

Teniendo todas las señales generadas, pasamos a la utilización del "Simulation Panel" (figura 26) de la herramienta Tecnomatix, herramienta es esencial en esta etapa. Este panel proporciona una interfaz interactiva que permite configurar y ejecutar la simulación, así como monitorear y analizar los resultados en tiempo real. Entre las funciones clave del Panel de Simulación se encuentra la capacidad de añadir y gestionar señales, las cuales son esenciales para simular el funcionamiento realista de los componentes y la interacción entre ellos.

Simulation Panel

Simulation	I...	O...	LB	Forced!	F...	A...	Robot Signal Name
[-] Proyecto_TFM							
UR3e_startProgram		●		<input type="checkbox"/>	■	Q	startProgram
UR3e_programNumber		0		<input checked="" type="checkbox"/>	1	Q	programNumber
Service		0		<input checked="" type="checkbox"/>	1	Q	Service
NItems		0		<input checked="" type="checkbox"/>	6	Q	NItems
Casos_Entrada		0		<input checked="" type="checkbox"/>	1	Q	Casos_Entrada
Palet_Recogido		●		<input type="checkbox"/>	■	Q	Palet_Recogido
Servicio_MalElegido	■			<input type="checkbox"/>	■	I	Servicio_MalElegido

Figura 26. Panel de simulación

En la figura 26. Se puede observar todas las señales utilizadas para realizar la simulación de la estación. A continuación, se realizará una breve explicación de estas:

UR3e_startProgram: Esta señal marca el inicio del programa de control del robot. Cuando se activa, inicia la ejecución de las secuencias y tareas programadas en el gemelo digital.

UR3e_programNumber: Esta variable almacena un valor numérico que se utiliza para determinar qué programa específico debe ejecutarse en el robot. En el proyecto, solo el programa principal

"MAIN" tiene asignado un valor de "1". Esto implica que al asignar el valor "1" a esta variable, el programa principal "MAIN" se activa y toma el control del robot para guiar su comportamiento y las secuencias de tareas definidas en ese programa. Al asignar valores numéricos diferentes a otros programas, podrías crear diferentes escenarios o secuencias de tareas para el robot y luego controlar cuál de estos programas se ejecuta, al modificar el valor de la señal UR3e_programNumber. Esta flexibilidad permite simular y evaluar diversas situaciones y comportamientos del robot en el gemelo digital, lo que resulta fundamental para la validación y optimización del proceso de ensamblaje en la estación de montaje robotizada.

Service: Esta señal representa el tipo de servicio deseado. Dependiendo del valor de esta señal, el robot realizará una secuencia particular de tareas para ensamblar las piezas correspondientes.

Nitems: Indica el número total de elementos en un palet. Ayuda a controlar y ajustar la secuencia de ensamblaje según la cantidad de piezas presentes en el proceso.

Casos_Entrada: Esta señal describe el caso de entrada específico. Dependiendo de esta variable, se selecciona la secuencia adecuada de tareas y operaciones para el ensamblaje.

Palet_Recogido: Indica si el palet ha sido recogido de la zona de salida. Esta señal puede ser relevante para garantizar que las operaciones solo se ejecuten cuando la estación esta libre.

Servicio_MalElegido: Esta señal detecta si se ha seleccionado un servicio incorrecto. Puede ser utilizada para activar una alarma o detener la simulación si se elige un servicio inapropiado.

2.4.2.3.2. Validación del gemelo digital en la realidad

Se compara el comportamiento de la simulación con el comportamiento real en la estación de montaje real. Esta comparación incluye factores como tiempos de ciclo, movimientos de los elementos, interacciones y secuencias de ensamblaje.

Cualquier discrepancia identificada se analiza y se aborda para garantizar la coherencia entre el gemelo digital y la realidad. La resolución de las discrepancias puede implicar ajustes en el gemelo digital, como modificar parámetros, mejorar la programación de las tareas o realizar cambios en las interacciones entre los elementos. El objetivo es asegurar una coherencia y concordancia sólidas entre el gemelo digital y la realidad, de modo que la simulación sea una representación confiable y precisa del proceso de ensamblaje.

Para llevar a cabo la validación de las operaciones que el robot realiza, se debe considerar que cada operación comprende varios movimientos y acciones específicas llevadas a cabo por el robot. El proceso para verificar cada operación involucra la generación y descarga de un programa que la herramienta Tecnomatix es capaz de crear.

En primer lugar, se debe configurar el robot en la simulación utilizando el controlador específico del robot de Universal Robots. Esta configuración permite que la herramienta Tecnomatix genere un archivo en formato "scripts" que es comprensible para el robot real.

Una vez configurado el robot en la simulación, se procede a generar el archivo de programa necesario utilizando la opción "download to robot" dentro de la herramienta. Este archivo de programa contiene todas las instrucciones y movimientos necesarios para ejecutar la operación específica en el robot real y se transfiere al robot mediante un pendrive.

Como se puede observar en la figura 27, el archivo generado a través del Process Simulate contiene una cabecera específica del entorno de simulación. Por lo tanto, es necesario realizar una adaptación al script generado para que sea comprensible por el robot real. En este proceso, se extraen únicamente los movimientos y acciones generados en el script original, eliminando la información relacionada con el entorno de simulación (figura 28).

```

# Process Simulate Disconnected 16.0.1 to Universal-URScript
# Controller version      : 3.14
# File                   : Prueba3
# Study                   : Proyecto_TFM
# User                    : User
# Generation Date        : 07/07/2021 at 12:04:50
# Robot                   : UR3e
#-----
def Prueba2():
  set_tcp(p[-0.0000000915,0.0000000740,0.0180002423,0.0000000591,0.0000000584,-1.5707963270])
  #$ "via"
  movej([0.0055371805,-1.5731506454,0.0084337042,-1.5768758634,-0.0055376785,0.0000056220], r=0)
  #$ "vial"
  movej([0.7249839984,-1.1390171617,1.6585434701,-2.0903227862,-1.5707964778,2.2957806720], r=0)
  #$ "pick"
  movej([0.7249799741,-1.1065687438,1.6898548559,-2.1540825899,-1.5707964778,2.2957766477], r=0)
  # Destination sai6199-gv
  # Drive Bulon_CLOSED
  # Destination sai6199-gv
  # WaitDevice Bulon_CLOSED
  # Grip TCP_Home
  #$ "vial"
  movej([0.7249845538,-1.1390136201,1.6585379781,-2.0903208358,-1.5707964778,2.2957812274], r=0)
  #$ "place"
  movej([0.7249812559,-1.1065785563,1.6898706813,-2.1540886028,-1.5707964778,2.2957779295], r=0)
  # Release TCP_Home
  # Destination sai6199-gv
  # Drive HOME
  # Destination sai6199-gv
  # WaitDevice HOME
  #$ "vial"
  movej([0.7249845538,-1.1390136201,1.6585379781,-2.0903208358,-1.5707964778,2.2957812274], r=0)
  #$ "via"
  movej([0.0055371805,-1.5731506454,0.0084337042,-1.5768758634,-0.0055376785,0.0000056220], r=0)
end
  
```

Figura 27. Archivo con formato Script original

```

set_tcp(p[-0.0000000915,0.0000000740,0.0180002423,0.0000000591,0.0000000584,-1.5707963270])
#$ "via"
movej([0.0055371805,-1.5731506454,0.0084337042,-1.5768758634,-0.0055376785,0.0000056220], r=0)
#$ "via1"
movej([0.7249839984,-1.1390171617,1.6585434701,-2.0903227862,-1.5707964778,2.2957806720], r=0)
#$ "pick"
movej([0.7249799741,-1.1065687438,1.6898548559,-2.1540825899,-1.5707964778,2.2957766477], r=0)
# Destination sai6199-gv
# Drive Bulon_CLOSED
# Destination sai6199-gv
# WaitDevice Bulon_CLOSED
# Grip TCP_Home
#$ "via1"
movej([0.7249845538,-1.1390136201,1.6585379781,-2.0903208358,-1.5707964778,2.2957812274], r=0)
#$ "place"
movej([0.7249812559,-1.1065785563,1.6898706813,-2.1540886028,-1.5707964778,2.2957779295], r=0)
# Release TCP_Home
# Destination sai6199-gv
# Drive HOME
# Destination sai6199-gv
# WaitDevice HOME
#$ "via1"
movej([0.7249845538,-1.1390136201,1.6585379781,-2.0903208358,-1.5707964778,2.2957812274], r=0)
#$ "via"
movej([0.0055371805,-1.5731506454,0.0084337042,-1.5768758634,-0.0055376785,0.0000056220], r=0)
  
```

Figura 28. Archivo con formato Script modificado

El proceso de cargar y ejecutar el script adaptado en el robot implica seguir un protocolo específico. A continuación, se detallan los pasos necesarios para reproducir el programa descargado en la consola del robot:

- **Introducción del Pendrive:** Se Inserta el pendrive en la interfaz USB de la consola del robot.
- **Creación del Programa de Robot:** En la consola del robot, se crea un nuevo programa de robot en el que se ejecutará el script. Este programa actuará como contenedor para las operaciones que se llevarán a cabo.
- **Agregar la Operación Script:** Dentro del programa de robot, se utiliza la función proporcionada por la interfaz de la consola para agregar una operación de tipo "script". Esta operación se utiliza para cargar el archivo script.
- **Selección del Archivo:** Cuando se agrega la operación de script, se selecciona el previamente guardado en el pendrive.
- **Iniciar la Ejecución:** Después de configurar la operación de script, se inicia la ejecución del programa de robot en la consola. El robot comenzará a cargar y ejecutar el script adaptado desde el pendrive.

Durante el proceso de comprobación, surgieron dos discrepancias significativas que requirieron atención y ajustes. Estas discrepancias se detallan a continuación:

Acciones de Apertura y Cierre de Pinzas: Se detectó una discrepancia en la ejecución de las acciones de apertura y cierre de las pinzas. Estas operaciones no se estaban realizando de manera

adecuada según lo previamente simulado. Esta discrepancia se originó en la diferencia entre la configuración de las pinzas en el entorno de simulación y las características reales del robot. Para resolver esta discrepancia y asegurar un funcionamiento coherente en ambos entornos, se implementaron medidas correctivas específicas en el programa de control del robot. Se añadieron dos líneas adicionales de comandos en las respectivas operaciones de apertura y cierre de pinzas (figura 29). Estas líneas son específicas y comprendidas solo por el robot real:

- `set_standard_digital_out(0, True)`: este comando tiene dos parámetros.
 - El primer parámetro activa la salida digital “0” que está vinculada a la pinza 1, utilizada para manipular palets y bulones. La pinza 2, utilizada para manipular rodamientos, tapas interiores y tapas exteriores, está vinculada a la salida digital “1”.
 - El segundo parámetro “True” o “False”, se utiliza para definir la posición de la pinza. False, se le indica a la pinza que permanezca en su posición de reposo (pinza 1, abierta; pinza 2; cerrada). True, cambia la pinza a la posición activa de la pinza.

- `sleep(1.0)`: Este comando introduce un breve período de espera, en este caso de 1 segundo, para permitir que el robot real procese y ejecute la instrucción antes de continuar con las siguientes operaciones.

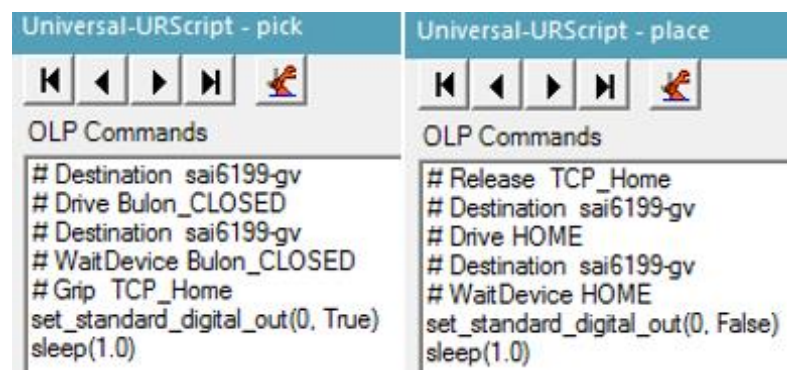


Figura 29. OLP Commands de las operaciones Pick y Place, respectivamente.

Puntos Críticos y Precisión: Se identificó que varios puntos críticos, como la recogida y colocación de palets, así como la disposición de elementos de ensamblaje en las bases, que presentaban una holgura tan pequeña que, necesitaron modificaciones para garantizar la alineación y el posicionamiento preciso. Para garantizar esta alineación se llevó a cabo un proceso de ajuste minucioso de las ubicaciones. Cada punto crítico se revisó y recalibró, asegurando que los elementos se dispusieran de manera precisa y coherente. Este ajuste fue necesario para reflejar con precisión la ejecución real de las tareas y garantizar que las operaciones se llevaran a cabo sin problemas en la estación de montaje real.

Para obtener los puntos exactos de las operaciones. En primer lugar, fue visitar la estación de montaje real y configurar el robot en la posición deseada para la operación en cuestión. Esto permitió obtener una configuración específica y precisa de las articulaciones del robot que se correspondía con el punto objetivo (figura 30).

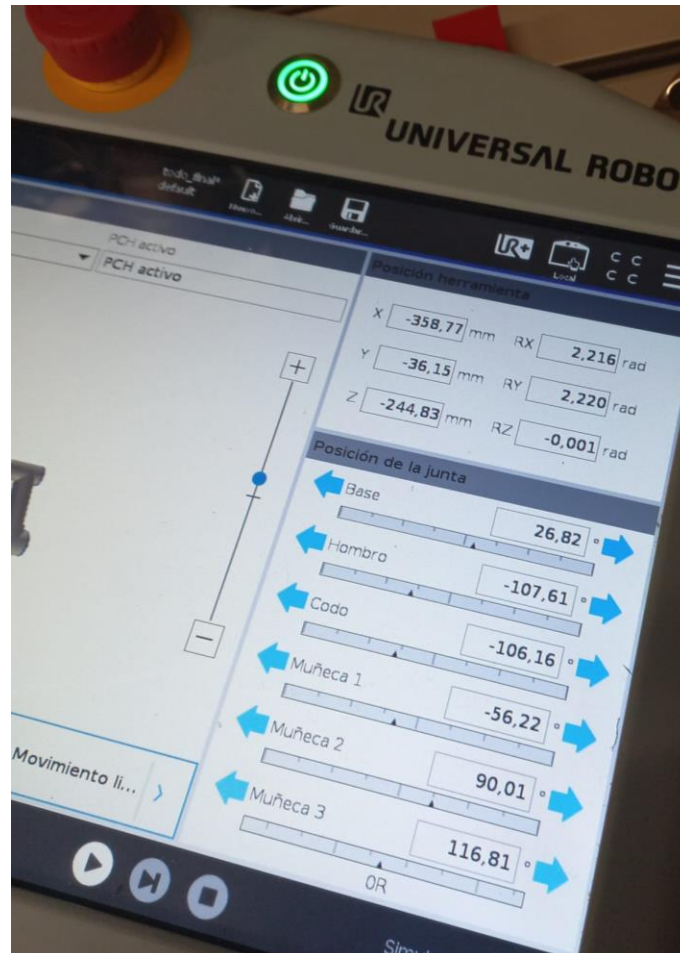


Figura 30. Posición de las juntas del robot, en la TeachPendant

Una vez que se obtuvo la configuración real del robot en esa posición, se llevó esta información a la simulación. En el entorno de simulación, se accedió a la operación específica que se deseaba ajustar y se abrió la pestaña "Robot". Dentro de la pestaña "Robot", se seleccionó la opción "Reach" y luego "Robot Jog". Esta acción abrió una ventana que mostraba los valores de las articulaciones del robot en ese momento (figura 31). Al modificar cuidadosamente estos valores y ajustarlos para que coincidieran con la configuración real obtenida del robot físico, se aseguró que la posición en la simulación fuera idéntica a la posición real.

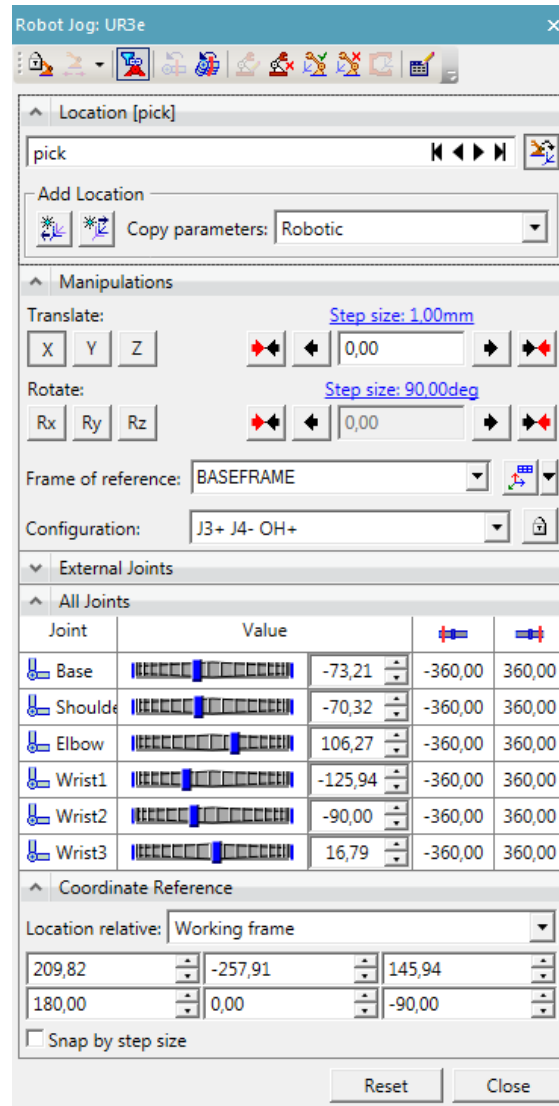


Figura 31. Posición de las juntas del robot, en Tecnomatix

Este proceso de ajuste permitió alcanzar un alto nivel de exactitud en la simulación de los puntos críticos y aseguró que las operaciones en el gemelo digital se ejecutaran de manera coherente con el comportamiento real del robot en la estación de montaje.

2.5. CONTROL DESDE EL PLC

Tal como se ha detallado en secciones anteriores, el control de la estación de montaje robotizada se basa en un programa de control principal (MAIN) vinculado al robot, el cual, en función de diversas variables, dirige las acciones del robot en la simulación, por ejemplo, realizar el movimiento del robot desde su posición de inicio a la posición de coger palet, respectivamente.

Como primera simulación se ha utilizado el panel de simulación de Tecnomatix, con la cual se pueden simular las variables necesarias para que el programa de control principal pueda ejecutarse correctamente. No obstante, para mejorar la coordinación y supervisión de estas variables, se ha implementado un sistema de control adicional basado en un PLC (Controlador Lógico Programable), para garantizar un funcionamiento seguro y eficiente.

El PLC elegido, es un PLC de Siemens S7-1500, reconocido por su robustez y versatilidad y desempeña varias funciones clave:

Supervisión Centralizada: El PLC asume la responsabilidad de supervisar y controlar las variables críticas para el proceso de ensamblaje. Esto abarca aspectos como el estado de los sensores, la elección de los casos de entrada, la cantidad de bases en los palets, y otras variables relevantes.

Forzado de Variables: El PLC tiene la capacidad de forzar variables de control específicas según sea necesario. Por ejemplo, puede iniciar la fase de preparación del sistema para que la estación se encuentre preparada para el ensamblaje, habilitar el pulsador de inicio del robot o detener el proceso en situaciones de emergencia.

Coordinación de Secuencias: El PLC coordina las secuencias de inicio y parada, asegurándose de que todos los componentes de la estación estén en las condiciones adecuadas antes de iniciar el proceso y que se lleve a cabo una parada segura al concluir la operación.

Seguridad: El PLC monitorea los sistemas de seguridad, incluyendo botones de emergencia y toma medidas en caso de una emergencia para garantizar la seguridad en la operación.

Interacción con Tecnomatix: El PLC establece comunicación bidireccional con Tecnomatix para recibir información sobre las condiciones de la estación de montaje y transmitir señales de control. Esto garantiza que el PLC esté en sincronía con el estado simulado de la estación.

Interacción con HMI: El PLC interactúa con la interfaz hombre-máquina (HMI) virtual para mostrar información relevante y permitir al usuario tomar decisiones basadas en la simulación.

Realizar el control desde el PLC conlleva una serie de ventajas:

- **Mayor Coordinación:** La inclusión de un PLC permite una coordinación más efectiva de las variables de control, garantizando que todas las condiciones estén en su lugar antes de iniciar el proceso.
- **Mejor Supervisión:** El PLC proporciona una plataforma centralizada para supervisar y controlar el proceso, lo que facilita la detección temprana de problemas y la toma de decisiones informadas.
- **Mayor Seguridad:** El PLC se encarga de las funciones de seguridad críticas, como la gestión de situaciones de emergencia, lo que mejora la seguridad general del sistema.
- **Facilita la Puesta en Marcha Real:** La implementación de un PLC en la simulación ayuda a replicar con mayor precisión el control que se utilizará en la puesta en marcha real, simplificando la transición entre el entorno virtual y el físico.

A lo largo de este punto, se explicará cómo se establecen las conexiones entre el PLC y el gemelo digital, se definen las señales y variables necesarias, y se muestra el programa lógico de control para simular las operaciones de la estación de montaje.

2.5.1. Configuración hardware de los dispositivos en el PLC

En primer lugar, se puede observar en la figura 32, los dispositivos necesarios para que el dispositivo de control (PLC) controle el sistema y estos se describen a continuación.

- **PLC Siemens S7-1500, CPU 1516-3 PN/DP:** Es el núcleo del Sistema de control y gracias a sus capacidades de comunicación Profinet (PN) y profibus (DP), facilita la conectividad y el intercambio de datos con otros dispositivos y sistemas.
- **HMI WIN CC Advanced:** La comunicación efectiva entre los operadores y el sistema es posible gracias a un HMI. Esta interfaz intuitiva permite la visualización de datos en tiempo real, así como la supervisión y el control de las operaciones de la estación de montaje robotizada.
- **Archivo GSDML:** El archivo GSDML (Device Description File) es esencial para la configuración de los dispositivos Profinet. Contiene información detallada sobre las características y capacidades de los dispositivos, lo que permite una integración más precisa y eficiente en el sistema de control. Para la realización de este proyecto, se utiliza el archivo GSDML proporcionado por el fabricante universal robots, para el control del robot utilizado.

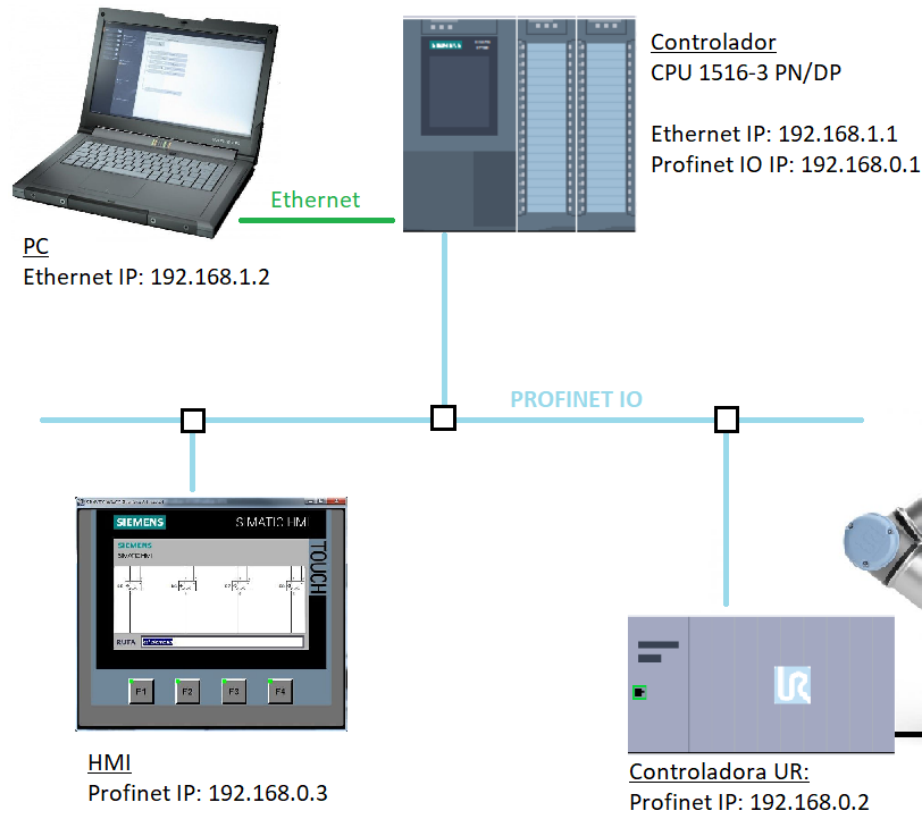


Figura 32. Dispositivos Hardware de la estación

A continuación, utilizando la herramienta de Siemens TIA Portal para su configuración y programación, se introducen todos los dispositivos hardware (PLC, HMI y archivo GSDML) en el proyecto se conectan a una red PROFINET-IO creada, donde se les asigna una dirección IP fija (figura 33).

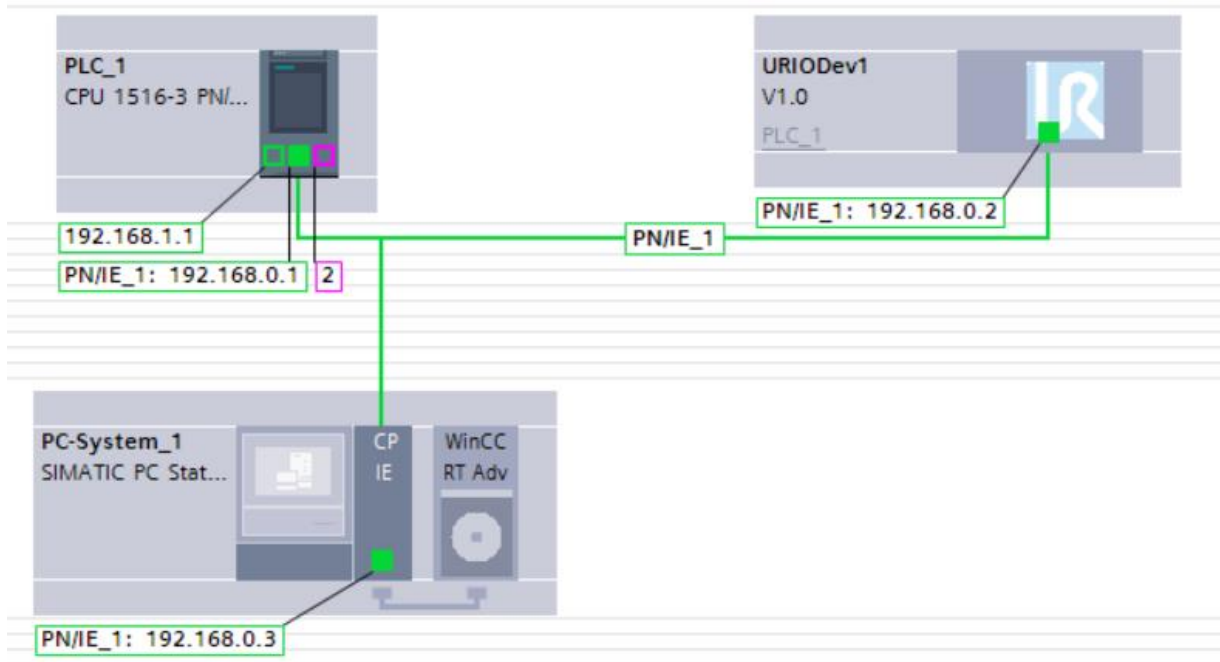


Figura 33. Dispositivos hardware vinculados en TIA Portal

Utilizando Profinet-IO como la red de comunicación, se asegura que los componentes de hardware se comuniquen de manera efectiva y en tiempo real, lo que a su vez permite la sincronización precisa y la ejecución exitosa de las operaciones en la estación de montaje.

2.5.2. Señales de control

Antes de pasar al programa de control es necesario definir las señales necesarias para el mismo (tabla 3). Teniendo en cuenta cuales señales están vinculadas al gemelo digital y cuales están vinculadas al HMI.

Tabla 3. Señales de control del PLC

Variable	Tipo	Dirección	Conexión	Funcionamiento
Pulsador_Marcha_Prep	BOOL	%M0.0	HMI	Activación secuencia de preparación de la estación
Pulsador_Marcha	BOOL	%M0.1	HMI	Activación de la estación
Pulsador_Emergencia	BOOL	%M0.2	HMI	Activación del estado de emergencia de la estación
Pulsador_Rearme	BOOL	%M0.3	HMI	Desactivación del estado de emergencia
Pulsador_ParoFC	BOOL	%M0.4	HMI	Paro Fin de ciclo
Sol_Marcha_Prep	BOOL	%M10.0	Variable de proceso	Solicitud Marcha Preparación
Fin_Marcha_Prep	BOOL	%M10.1	Variable de proceso	Finalización Marcha Preparación
Sol_Validar_Sensores	BOOL	%M10.2	Variable de proceso	Solicitud comprobación sensores de la estación

Variable	Tipo	Dirección	Conexión	Funcionamiento
Fin_Validar_Sensores	BOOL	%M10.3	Variable de proceso	Finalización comprobación sensores
HabilitarMarcha	BOOL	%M10.4	HMI	Habilitación del pulsador de marcha
Sensor_ZonaEntrada	BOOL	%I0.0	Estación Real	Presencia del palet en la zona de entrada
Sensor_ZonaMontaje	BOOL	%I0.1	Estación Real	Presencia del palet en la zona de montaje
Sensor_ZonaSalida	BOOL	%I0.2	Estación Real	Presencia del palet en la zona de salida
Sensor_Rodam	BOOL	%I0.3	Estación Real	Presencia de 6 rodamientos en el alimentador específico
Sensor_Bulones	BOOL	%I0.4	Estación Real	Presencia de 6 bulones en el alimentador específico
Sensor_TapasInt	BOOL	%I0.5	Estación Real	Presencia de 6 tapas interiores en el alimentador específico
SenP1_Pos0	BOOL	%I0.6	Estación Real	Pinza 1 (Rodamientos) abierta
SenP2_Pos1	BOOL	%I0.7	Estación Real	Pinza 2 (Bulones) cerrada
PilotoVerde	BOOL	%M10.5	HMI	Indicador de estación en servicio
PilotoRojo	BOOL	%M10.6	HMI	Indicador de estación en estado de emergencia
Mensaje1	BOOL	%M11.0	HMI	Indicador de falta de solicitud de servicio
Mensaje2	BOOL	%M11.1	HMI	Indicador de habilitación de pulsador de marcha
Mensaje3	BOOL	%M11.2	HMI	Indicador de palet mal puesto, en zona de montaje
Mensaje4	BOOL	%M11.3	HMI	Indicador de palet mal puesto, en zona de salida
Mensaje5	BOOL	%M11.4	HMI	Indicador de falta de material, rodamientos.
Mensaje6	BOOL	%M11.5	HMI	Indicador de falta de material, bulones.
Mensaje7	BOOL	%M11.6	HMI	Indicador de falta de material, tapas interiores.
Mensaje8	BOOL	%M11.7	HMI	Indicador de falta de palet en la zona de entrada.
ProgramNum	BYTE	%QW26	Tecnomatix	Variable del robot para elegir un programa de control
Robot_HOME	BOOL	%I1.0	Tecnomatix	Variable del robot en posición de inicio.
UR3e_emergencyStop	BOOL	%I1.1	Tecnomatix	Variable del robot de parada de emergencia.
StartProgram	BOOL	%Q1.0	Tecnomatix	Variable del robot para empezar programa

CasoEntrada_HMI	BYTE	%MQ20	HMI-Teconmatix	Caso de entrada del palet
Servicio	BYTE	%Q22	HMI-Teconmatix	Servicio elegido
NumeroBases	BYTE	%QW24	HMI-Teconmatix	Número de bases que contiene el palet

2.5.2.1. HMI

El HMI (Interfaz Hombre-Máquina) desempeña un papel importante en el proyecto de montaje robotizado, ya que proporciona una interfaz gráfica para que los operadores interactúen con la estación de ensamblaje y supervisen su funcionamiento. Aquí, se describen algunas de las características y funciones clave del HMI (figura 34) en este proyecto:

Seleccionar Configuraciones: Los operadores pueden utilizar el HMI para seleccionar configuraciones específicas, como el caso de entrada del palet, el número de bases en el palet y el servicio deseado. Estos ajustes afectan directamente el funcionamiento del robot y la estación

- Tiene un apartado donde se describe los números de casos de entrada que se pueden elegir en la zona de características del palet de entrada. En esta zona se puede elegir el número de caso de entrada, estando limitada la variable de 1 a 4. De igual forma se puede elegir la cantidad de bases con las que el palet es introducido, esta variable también está limitada de 1 a 6.
- También tiene un apartado donde se describe los servicios que se pueden elegir en la zona de solicitud de servicio. En esta zona se puede elegir el servicio deseado, estando limitada la variable de 1 a 10 servicios.

Control: El HMI permite a los operadores iniciar, detener el proceso de ensamblaje. Esto se logra mediante la incorporación de pulsadores. Estos se tratan de:

- Pulsador de Marcha de preparación: Este pulsador se utiliza para preparar la estación para que pueda realizar los montajes de manera segura y eficaz.
- Pulsador de Marcha: este pulsador se utiliza para iniciar el proceso de ensamblaje. Actúa como un disparador que pone en marcha todas las operaciones y secuencias necesarias para llevar a cabo el ensamblaje de las piezas.

Seguridad: Proporciona controles de seguridad, como botones de paro de emergencia y rearme, para garantizar la seguridad de los operadores y del equipo.

Indicadores de Estado: Incorpora indicadores visuales, como luces LED virtuales, que muestran el estado de la estación, como "En Servicio" o "Emergencia". Estos indicadores proporcionan retroalimentación rápida sobre el estado actual de la estación.

Avisos y Mensajes: Puede generar alarmas y mensajes en pantalla para informar a los operadores sobre eventos importantes o situaciones de emergencia. Por ejemplo, puede mostrar un de que falta un elemento en los alimentadores o si se activa un botón de emergencia.










<p style="text-align: center;">Nº Caso entrada</p> <p>Caso 1: Base Caso 2: Base + Rodamiento Caso 3: Base + Rodamiento + Bulón Caso 4: Base + Rodamiento + Bulón + Tapa blanca</p> <hr/> <p style="text-align: center;">Características del palet de entrada</p> <p>Nº caso entrada..... <input type="text" value="▼"/> Nº elementos en el palet..... <input type="text" value="▼"/></p> <hr/> <p style="text-align: center;">Solicitud de Servicio</p> <p>Nº de servicio deseado..... <input type="text" value="▼"/></p>	<p style="text-align: center;">Servicios</p> <p>Servicio 1: Añadir rodamientos Servicio 2: Añadir rodamientos y bulones Servicio 3: Añadir rodamientos, bulones y tapas interiores Servicio 4: Añadir rodamientos, bulones, tapas interiores y tapas exteriores Servicio 5: Añadir bulones Servicio 6: Añadir bulones y tapas interiores Servicio 7: Añadir bulones, tapas interiores y tapas exteriores Servicio 8: Añadir tapas interiores Servicio 9: Añadir tapas interiores y tapas exteriores Servicio 10: Añadir tapas exteriores</p>														
<table style="width: 100%; text-align: center;"> <tr> <td>Marcha Preparación</td> <td>Marcha</td> <td>Emergencia</td> <td>Rearme</td> <td>Paro Fin de ciclo</td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </table>	Marcha Preparación	Marcha	Emergencia	Rearme	Paro Fin de ciclo						<p style="text-align: center;">Estado de la estación</p> <table style="width: 100%; text-align: center;"> <tr> <td></td> <td></td> </tr> <tr> <td>En servicio</td> <td>En emergencia</td> </tr> </table>			En servicio	En emergencia
Marcha Preparación	Marcha	Emergencia	Rearme	Paro Fin de ciclo											
															
															
En servicio	En emergencia														
<p style="text-align: center;">AVISOS</p> <p>Mensaje 1: Solicitud de servicio Mensaje 2: Habilitación Pulsador de Marcha Mensaje 3: Quitar palet de la zona de montaje Mensaje 4: Quitar palet de la zona de salida Mensaje 5: Añadir rodamientos a su alimentador Mensaje 6: Añadir bulones a su alimentador Mensaje 7: Añadir tapas interiores a su alimentador Mensaje 8: No hay palet en la zona de entrada Mensaje 9: Estado de emergencia. Pulse Rearme para liberar las pinzas</p>															

Figura 34. HMI

2.5.2.2. Programa de control

En esta sección, vamos a presentar el programa de control de nuestra estación de montaje robotizada utilizando el lenguaje GRAFCET. Explicaremos los estados, las transiciones y las condiciones que definen el comportamiento de la estación. Este enfoque nos permitirá ilustrar de manera efectiva cómo se coordina y controla cada aspecto de la operación, desde la inicialización hasta la ejecución de las tareas de ensamblaje y la gestión de situaciones de emergencia.

Secuencia Principal (figura 35): La secuencia principal se ha diseñado para garantizar una operación segura y eficiente. A continuación, se detalla esta secuencia paso a paso:

- Inicio de la secuencia (etapa 1): La secuencia de control comienza cuando se pulsa el pulsador de "Marcha de Preparación" en el HMI. El sistema verifica que la estación esté preparada para comenzar el proceso, estando las zonas de montaje y salida libres de palets.
- Verificación de sensores (etapa 2): El PLC realiza una verificación de los sensores en la estación, para garantizar que todas las condiciones sean adecuadas para el inicio.
 - Se comprueba que los alimentadores estén llenos y listos para suministrar los componentes necesarios.
 - Se verifica que no haya palets presentes en la zona de montaje ni en la zona de salida.
 - Se confirma que el palet necesario para el ensamblaje se encuentra en la zona de entrada
- Selección del Servicio Deseado (etapa 3): Una vez que se han verificado los sensores, se activa una señal para indicar al operador que seleccione el servicio deseado, el caso de entrada y la cantidad adecuada de bases que contiene el palet, a través del HMI.
- Habilitación del Pulsador de Marcha (etapa 4): Cuando se ha seleccionado correctamente el servicio, el caso de entrada y la cantidad adecuada de bases que contiene el palet, se habilita el pulsador de "Marcha" en el HMI. Al mismo tiempo, se actualiza el valor de la variable "Program Number", que asigna al robot el programa principal "MAIN", para que ordene realizar al robot una operación u otra.
- Inicio del Programa del Robot (etapa 5): En este punto, se espera a que el operador pulse el pulsador de "Marcha" en el HMI. Cuando se pulsa, se activa la señal "StartProgram" del robot, lo que permite al robot ejecutar la secuencia que le ordene el programa "MAIN". A su vez, el piloto verde en el HMI se enciende, indicando que la "estación está en servicio".
- Finalización de la Operación (etapa 6): Cuando el programa del robot ha finalizado y el palet se encuentra en la zona de salida, el programa de control diverge en dos opciones.
 - En la primera opción, si no se pulsa el pulsador de "Paro de Fin de Ciclo" y se saca el palet de la zona de salida, el programa de control regresa al estado de verificación de sensores y se pone a la espera de un nuevo palet.
 - En la segunda opción, si se pulsa el pulsador de "Paro de Fin de Ciclo" y se saca el palet de la estación, el programa de control finaliza, entendiendo que no se desea realizar más ensamblajes.

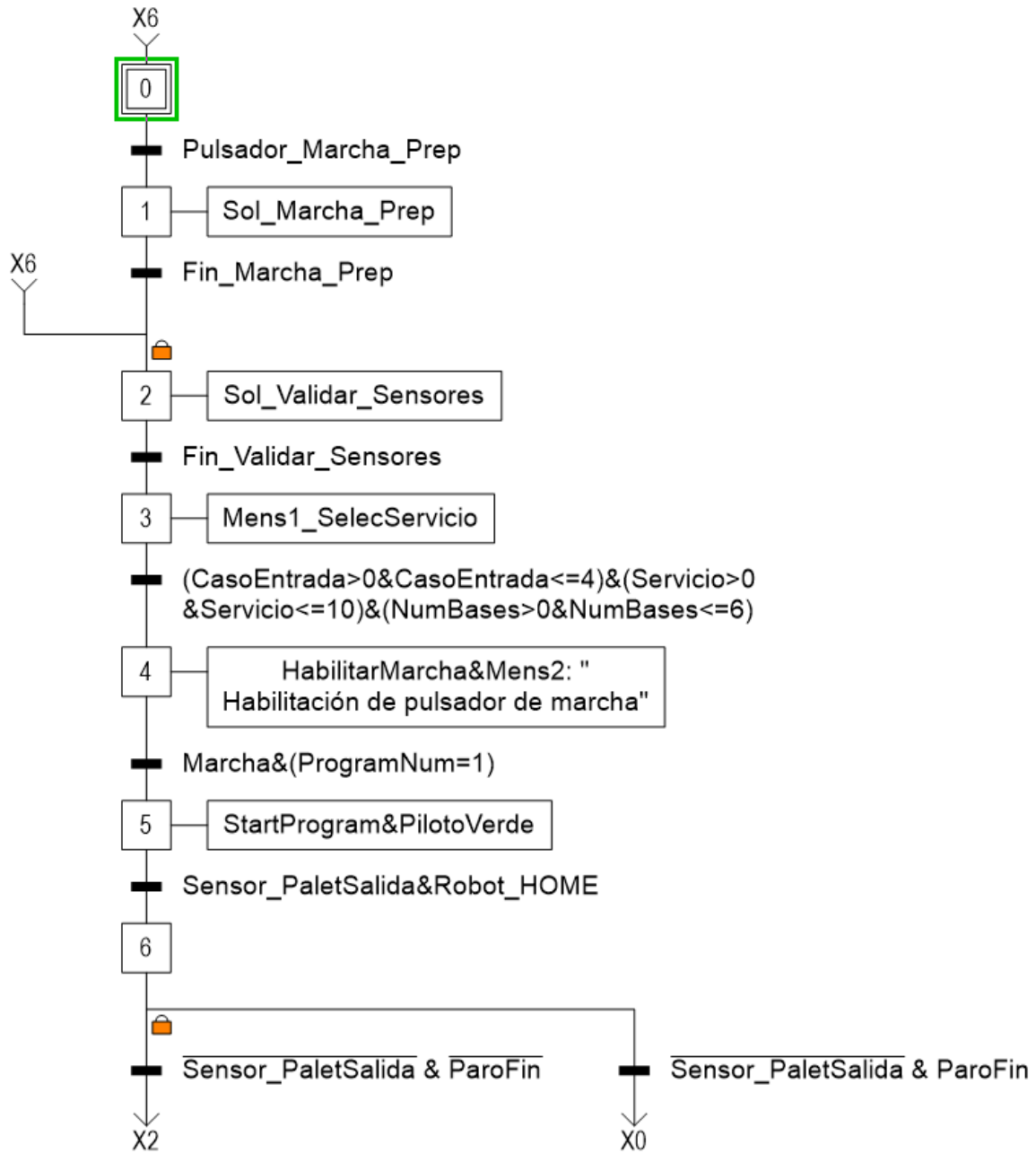


Figura 35. Secuencia principal

Secuencia de marcha de preparación (figura 36): Esta secuencia es esencial para prevenir posibles problemas o conflictos en la estación de montaje, asegurando un funcionamiento seguro y eficiente. A continuación, se detalla esta secuencia paso a paso:

- Inicio de la secuencia (etapa 11): la secuencia se inicia cuando se activa la variable Sol_Marcha_Prep.
- Verificación de la Zona de Montaje:
 - Comprueba si hay un palet en la zona de montaje.
 - Si se detecta un palet, muestra un mensaje para que el operario retire el palet.
- Verificación de la Zona de Salida:
 - Si la zona de montaje está vacía, se pasa a verificar la zona de salida.
 - Comprueba si hay un palet en la zona de salida.
 - Si se detecta un palet, muestra un mensaje para que el operario retire el palet de la zona de salida.
- Finalización de la Secuencia:
 - Después de verificar ambas zonas y asegurarse de que estén vacías, la secuencia de marcha de preparación se considera completa.
 - La estación está ahora lista y vacía para iniciar la siguiente fase de operación.

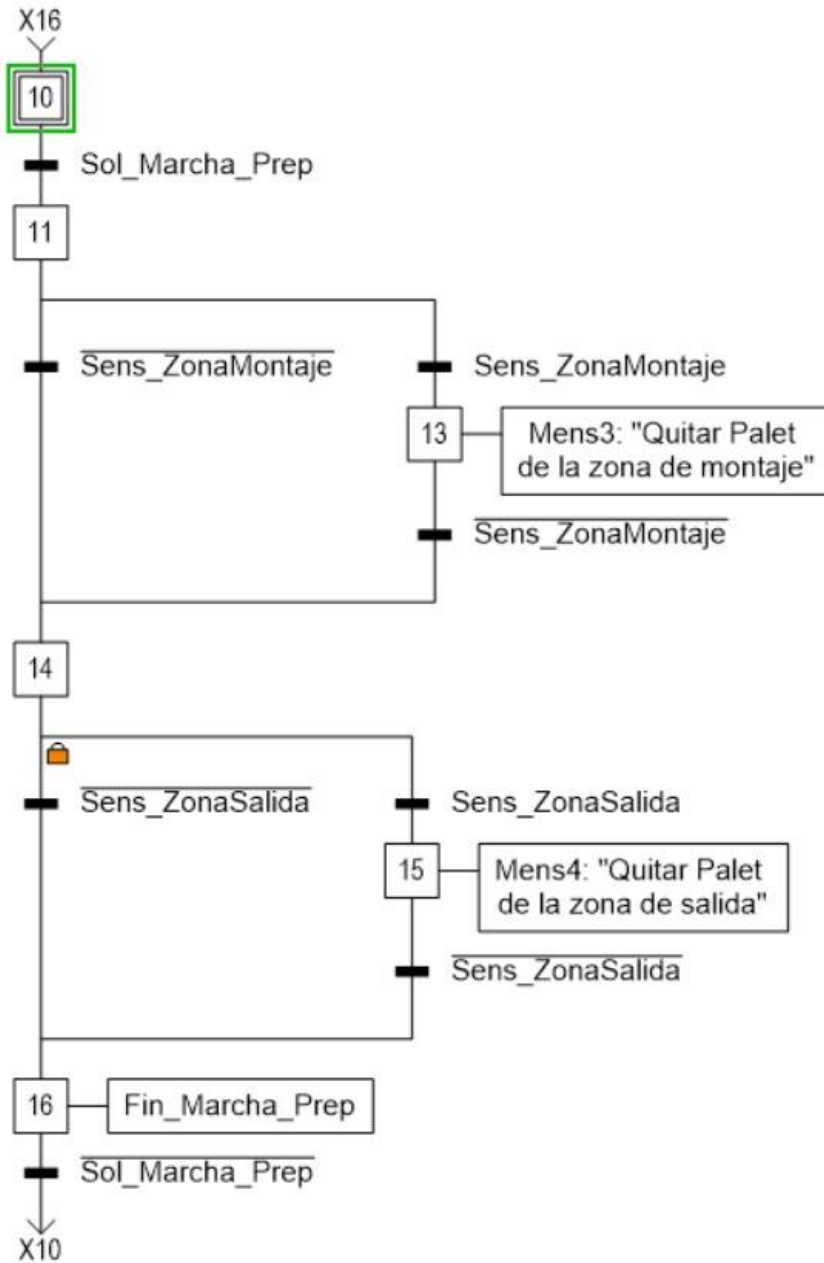


Figura 36. Secuencia de marcha de preparación

Secuencia de validación de sensores (figuras 37 y38): Esta secuencia garantiza que todos los elementos necesarios estén en su lugar antes de iniciar cualquier proceso de ensamblaje. A continuación, se detalla esta secuencia paso a paso:

- Inicio de la secuencia (etapa 21): la secuencia se inicia cuando se activa la variable Sol_validar_Sensores.
- Verificación de la Zona de Montaje:
 - Comprueba si hay un palet en la zona de montaje.
 - Si se detecta un palet, muestra un mensaje para que el operario retire el palet.
- Verificación de Rodamientos:
 - Comprueba si hay suficientes rodamientos disponibles.
 - Si no hay suficientes rodamientos, muestra un mensaje para que el operario llene el alimentador de rodamientos.
- Verificación de Bulones:
 - Verifica si hay suficientes bulones disponibles.
 - Si no hay suficientes bulones, muestra un mensaje para que el operario añada bulones.
- Verificación de Tapas Interiores:
 - Verifica si hay suficientes tapas interiores disponibles.
 - Si no hay suficientes tapas interiores, muestra un mensaje para que el operario añada tapas interiores.
- Verificación de Palet en la Zona de Entrada:
 - Comprueba si hay un palet en la zona de entrada.
 - Si no se detecta ningún palet, muestra un aviso indicando que falta el palet en la zona de entrada.
- Verificación de Palet en la Zona de Montaje:
 - Asegura que no haya ningún palet en la zona de montaje.
 - Si se detecta un palet, muestra un mensaje para que el operario retire el palet.
- Finalización de la Secuencia:
 - Después de realizar todas estas verificaciones, la secuencia de validación de sensores se da por finalizada.

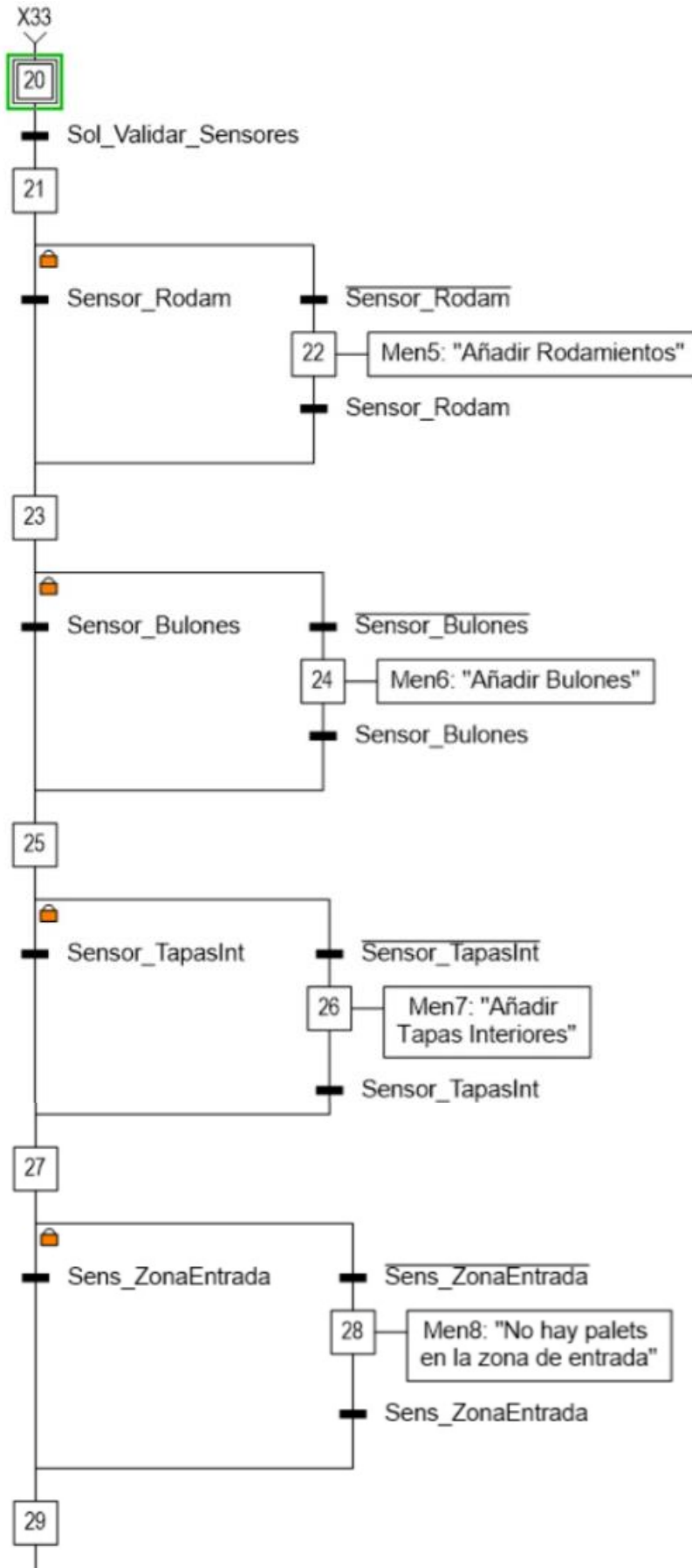


Figura 37. Secuencia de validación de sensores, parte 1

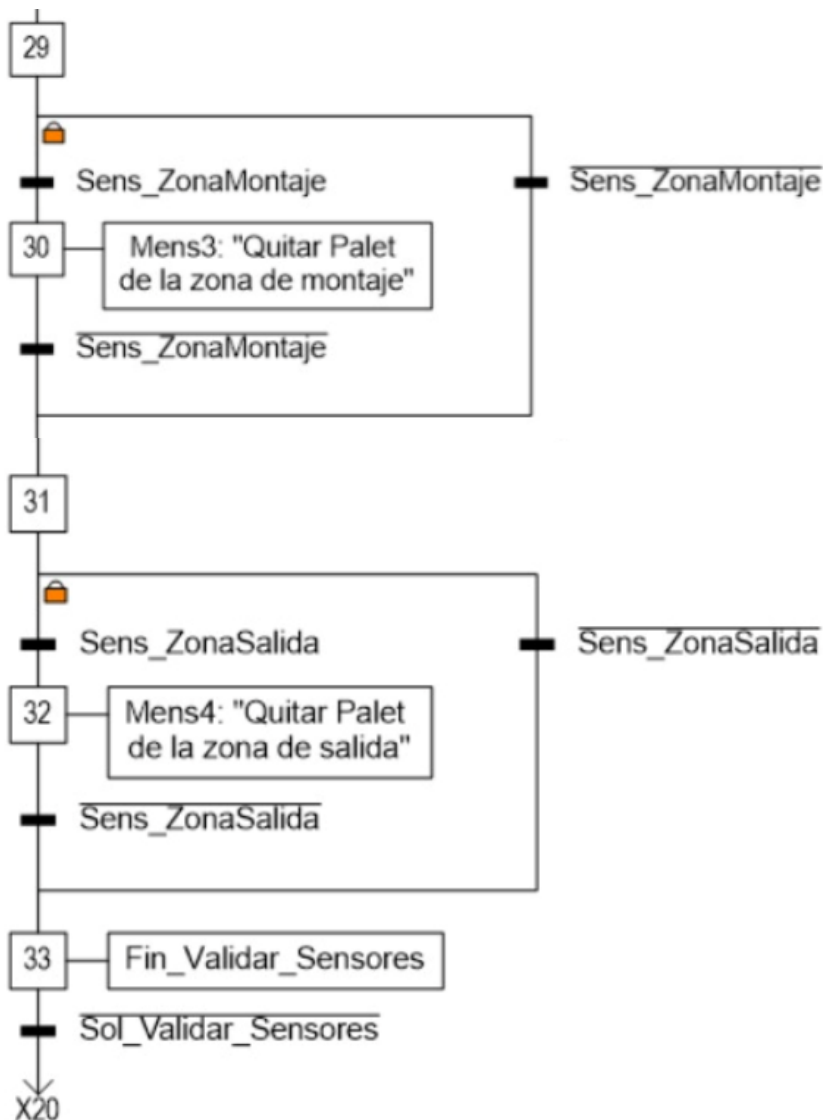


Figura 38. Secuencia de validación de sensores, parte 2

Secuencia de emergencia (figura 39): La secuencia de emergencia garantiza que, en caso de una situación de emergencia, el robot se detenga de inmediato, las pinzas se liberen si es necesario y el sistema se recupere de manera segura antes de reanudar las operaciones normales. Además, proporciona una clara indicación en el HMI sobre el estado de emergencia para el operador. A continuación, se detalla esta secuencia paso a paso:

- Inicio de la secuencia (etapa 100): La secuencia de control comienza cuando se pulsa el pulsador de "Emergencia" en el HMI. En este caso se desencadenan una serie de acciones específicas, y que se deben cumplir obligatoriamente, para garantizar la seguridad y la recuperación del sistema.

- UR3e emergencyStop (etapa 101): Se envía una señal de emergencia al robot para que detenga inmediatamente todas las operaciones.
- Verificación de pinzas (etapa 103): Se verifica la posición de las pinzas del robot para determinar si están en la posición de coger material (abiertas para rodamientos o cerradas para bulones). Si alguna de las pinzas está en la posición de coger material, se muestra un mensaje en el HMI que indica "Pulsar rearme para liberar pinzas" y se espera la acción del operador para pulsar el botón de rearme en el HMI.
- Posición de reposo de las pinzas (etapa 104): Una vez pulsado el botón de rearme, Ambas pinzas se ponen en su posición de reposo, independientemente de su estado anterior (abiertas o cerradas).
- Reset Grafkets (etapa 106): al pulsar el botón de emergencia, se resetean los grafket.
- Piloto estado de emergencia (etapa 107): El piloto rojo del HMI se enciende, indicando que la estación se encuentra en estado de emergencia.
- Grafket a posición inicial (108): Una vez que los tres caminos anteriores han completado su secuencia. Se espera a que el botón de emergencia ya no está pulsado y se pulse el botón de rearme. Es aquí, cuando todos los Grafkets vuelven a su posición de inicio, preparados para continuar con las operaciones normales de la estación.

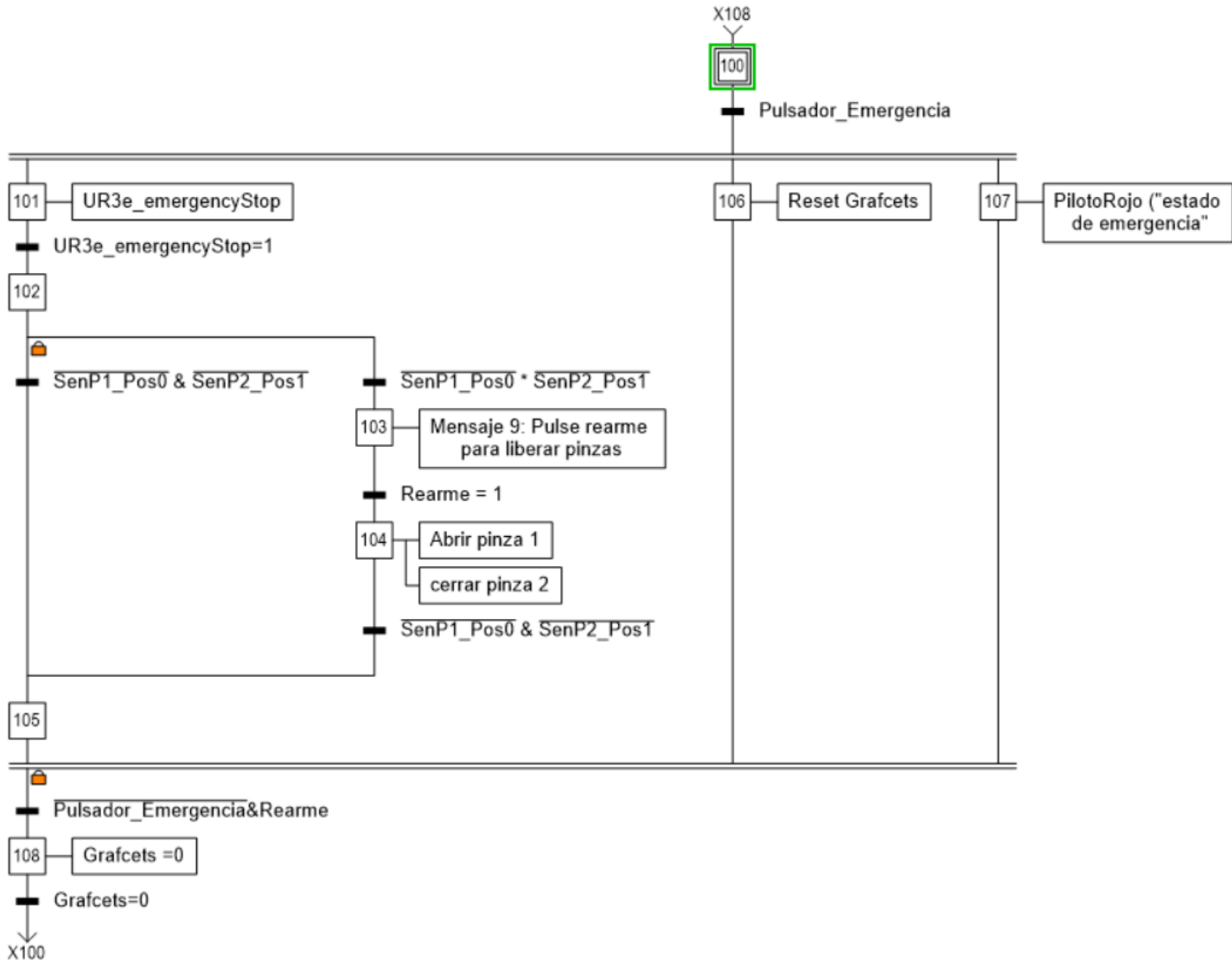


Figura 39. Secuencia de emergencia

2.6. PUESTA EN MARCHA VIRTUAL (virtual commissioning)

La fase de puesta en marcha virtual del proyecto desempeña un papel fundamental en la validación de la estación de montaje robotizada antes de su implementación en el entorno real. En este proyecto, se ha empleado SIMATIC PLCSIM Advanced, una herramienta de simulación avanzada de Siemens, para las soluciones software-in-the-loop (SiL) en la puesta en marcha virtual.

La puesta en marcha virtual distribuida se ha llevado a cabo mediante una integración cuidadosa de tres componentes clave (figura 40): Tecnomatix Process Simulate (PS), una instancia del PLC virtual a través de PLCSIM Advanced y una simulación virtual del HMI. Esta integración se ha realizado en un único PC.

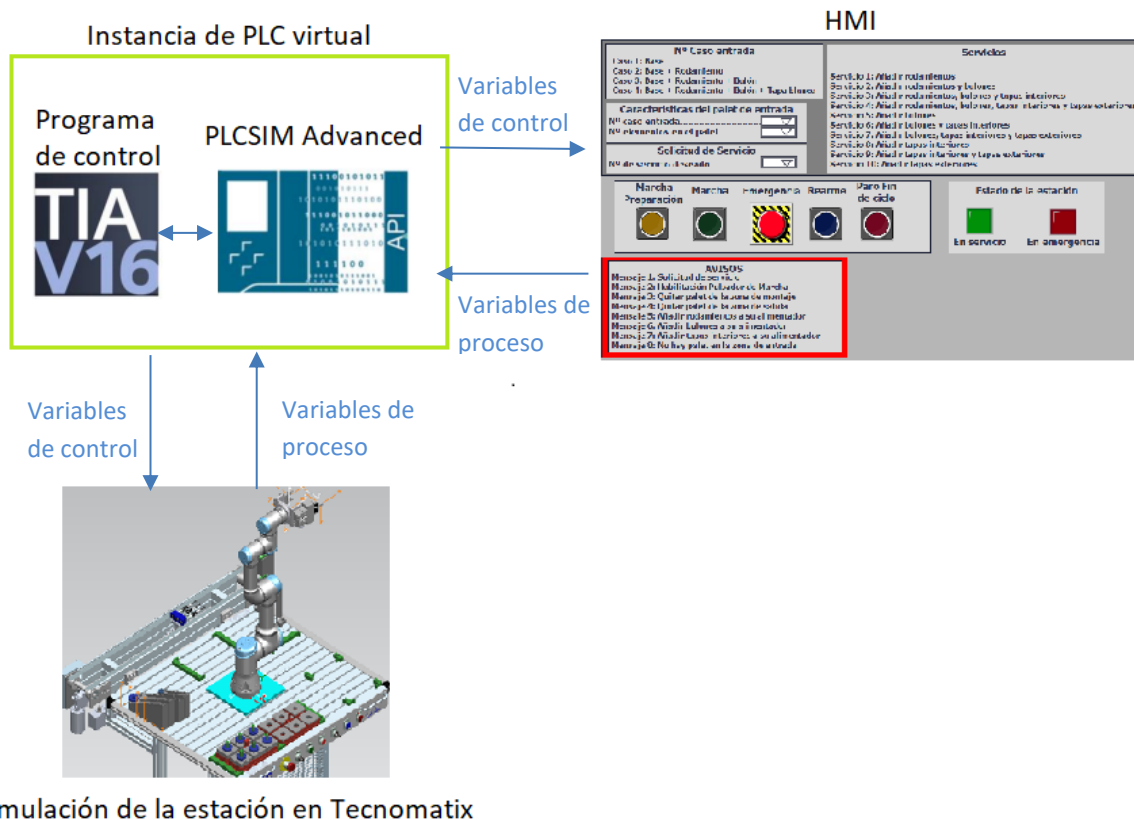


Figura 40. Flujo de comunicación entre los sistemas

Por un lado, Tecnomatix PS ha servido como el entorno principal para la creación y modelado del gemelo digital del sistema de ensamblaje. Se ha utilizado para definir virtualmente todos los aspectos de la estación de montaje robotizada, incluyendo sus operaciones, flujos de material y lógica de control.

Por otro lado, se ha utilizado WinCC Runtime Advanced para crear la interfaz de usuario humano (HMI). La simulación del HMI proporciona una interfaz gráfica que permite la supervisión y el control del proceso de ensamblaje, mediante la interacción con el PLC virtual.

Por último, se ha empleado una instancia del PLC virtual, en particular PLCSIM Advanced, para emular y ejecutar la lógica de control desarrollada para la estación de montaje. Esta instancia virtual del PLC se ha configurado para comunicarse bidireccionalmente con Tecnomatix PS y el simulador del HMI, recibiendo el estado de las variables necesarias para ejecutar correctamente el control del proceso.

Las variables utilizadas en la puesta en marcha virtual (tabla 4), tanto las variables de control como las de proceso, se deben configurar de manera precisa. Esto significa que, si en el PLC se trata de una señal de entrada, en la simulación del gemelo digital, la misma variable debe ser definida como una señal de salida con la correspondiente dirección. Este proceso simplifica enormemente la implementación real de la estación, ya que, al pasar al PLC físico, se usaría el mismo programa que se utilizó durante las pruebas de comprobación en el PLC virtual. Esto garantiza una transición fluida y eficiente de la simulación a la implementación real.

A continuación, se detallan las Entradas/Salidas utilizadas para la puesta en marcha virtual.

Tabla 4. Variables para la puesta en marcha virtual

Variable vinculada a tecnomatix	Tipo	Dirección en PLCSIM Advanced
UR3e_startProgram	BOOL	%Q1.0
UR3e_programNumber	BYTE	%QW26
UR3e_emergencyStop	BOOL	%Q1.1
Service	BYTE	%QW22
NItems	BYTE	%QW24
Cases_Entrada	BYTE	%QW20
UR3e_at_HOME	BOOL	%IW1.0

2.7. PLANIFICACIÓN

Este documento de planificación del proyecto proporciona una visión general de todos los aspectos clave del proyecto, desde su alcance y objetivos hasta los recursos necesarios. Además, presenta una estructura detallada en fases que servirá como guía para la ejecución exitosa del proyecto.

Duración del proyecto: 201 días. La jornada de trabajo media por día estimada es de 3 horas al día. Siendo un total de horas: 603 h.

Fase 1, Definición y planificación inicial

Duración: 18 días.

Tareas:

- Identificación de los objetivos del proyecto.
- Formación.
- Definición de los Requisitos y Alcance del Proyecto.

Recursos utilizados:

- Tecnomatix process simulate
- Documentación facilitada por Siemens con ejemplos para desarrollar.

Fase 2, Diseño y Configuración del Gemelo Digital

Duración: 77 días.

Tareas:

- Selección de la Herramienta de Simulación.
- Modelado de la Estación de Montaje Robotizada en Tecnomatix.
 - Creación de Variantes de Palets.
 - Configuración de Operaciones y Secuencias en el Gemelo Digital.
 - Definición de Variables de Control y Señales.
 - Desarrollo del Programa de Control Principal (MAIN).

Recursos utilizados:

- Tecnomatix process simulate

Fase 3: Validación del gemelo digital

Duración: 43 días.

Tareas:

- Ejecución de Pruebas de Simulación.
- Ejecución de pruebas en el robot real.
- Identificación y Resolución de Discrepancias.
- Validación del Comportamiento del Sistema real.

Recursos utilizados:

- Tecnomatix process simulate
- Estación robotizada real con robot UR3e.

Fase 4: Programación del Control en el PLC Virtual

Duración: 13 días.

Tareas:

- Configuración del PLC Virtual (PLCSIM Advanced).
- Desarrollo de la secuencia de Control Principal.
- Integración de Variables de Control y Proceso.

Recursos utilizados:

- TIA Portal V16

Fase 5: Diseño y Desarrollo del HMI

Duración: 15 días

Tareas:

- Diseño de la Interfaz de Usuario (HMI).
- Creación de Elementos de Control en el HMI (Botones, Indicadores, etc.).
- Integración del HMI con el PLC Virtual.

Recursos utilizados:

- TIA Portal V16
- WinCC Runtime

Fase 7: Documentación y Entrega

Duración: 35 días

Tareas:

- Elaboración de Documentación Técnica.

Recursos utilizados:

- Microsoft Word
- Diferentes fuentes de información

En la figura 41, se puede observar el diagrama de Gantt generado con las fases del proyecto.

3. ASPECTOS ECONÓMICOS

En el desarrollo de este proyecto, se consideraron una serie de aspectos económicos clave que desempeñaron un papel fundamental en su ejecución. A continuación, se detallan los aspectos económicos específicos que comprenden los costos y recursos involucrados:

1. Componentes de la Estación:

El proyecto contó con una inversión inicial para adquirir todos los componentes necesarios para la estación de montaje robotizada. Esto incluye el robot UR3e y los demás componentes de la estación de montaje, el PLC de Siemens y panel para el HMI. En la tabla 5 se puede observar un presupuesto orientativo de los componentes de la estación:

Tabla 5. Presupuesto orientativo de los componentes de la estación

Concepto	Coste (€)
Robot UR3e	18.000
Herramienta del Robot	1.200
Mesa de trabajo SMC	800
Elementos realizados en impresora 3D	200
PLC Siemens S7-1500, CPU 1516-3 PN/DP	1500
Panel para HMI	300
TOTAL	22.000

2. Costos de Desarrollo y Licencias de Software:

Para el desarrollo del proyecto ha sido necesario un ordenador y la adquisición de licencias de software, como Tecnomatix, TiaPortal V16 y WinCC Advanced, constituyó una inversión significativa en el proyecto. Estas licencias permitieron el uso de herramientas de simulación y control esenciales para la estación de montaje robotizada. En la tabla 6 se puede observar un presupuesto orientativo de los costos de desarrollo y licencias de software:

Tabla 6. Presupuesto orientativo de los costos de desarrollo y licencias de software

Concepto	Coste (€)
Ordenador para desarrollo y programación	1.500
Licencia Tecnomatix	8.000
TIA Portal V16	1.500
WinCC Advanced Runtime	500
TOTAL	11.500

3. Costo del Ingeniero de Desarrollo:

Se asignó un ingeniero para el desarrollo del proyecto. Este ingeniero desempeñó un papel fundamental en la planificación, diseño, implementación y supervisión de todo el proyecto. El

costo asociado con las horas trabajadas por el ingeniero se considera un componente importante en el presupuesto total del proyecto. En total, se dedicaron 603 horas de trabajo a lo largo del proyecto. En la tabla 7 se puede observar un presupuesto orientativo de los costos del Ingeniero de Desarrollo:

Tabla 7. Presupuesto orientativo de los costos del Ingeniero de Desarrollo

Concepto	Precio por hora	Horas	Coste (€)
Ingeniero de Desarrollo	30	603	18.090

4. Costos Operativos:

A lo largo del proyecto, se incurrieron en costos operativos que incluyen electricidad, espacio de trabajo y otros gastos menores relacionados con la ejecución del proyecto que no se tienen en cuenta.

Presupuesto Total del Proyecto: 51,590 euros.

4. CONCLUSIONES

En resumen, este proyecto destaca cómo las tecnologías de simulación, gemelos digitales y virtual commissioning pueden revolucionar la forma en que diseñamos y ponemos en marcha sistemas industriales complejos.

A lo largo de este proyecto, se ha llegado a las siguientes conclusiones:

Diseño Eficiente con Tecnomatix: Utilizar Tecnomatix como herramienta principal de diseño y simulación ha demostrado ser una elección acertada. Esta plataforma ha permitido una representación precisa y completa de nuestra estación de montaje robotizada, lo que ha facilitado la optimización de procesos antes de la implementación real.

Virtual Commissioning: La puesta en marcha virtual ha sido un paso fundamental en nuestro proyecto. El uso de PLCSIM Advanced y su conexión con Tecnomatix ha permitido probar y validar nuestro sistema de control antes de comprometernos con hardware físico. Esto no solo ha ahorrado tiempo y costos, sino que también ha mejorado significativamente la confiabilidad y eficiencia de nuestra estación.

Lenguaje GRAFCET: La descripción del programa de control mediante el lenguaje GRAFCET ha sido una elección efectiva para representar la lógica de funcionamiento de nuestra estación de montaje. Ha facilitado la comprensión de la secuencia de control, lo que ha contribuido a un funcionamiento más confiable.

Beneficios del Gemelo Digital: Hemos observado una serie de beneficios tangibles al utilizar un gemelo digital en este proyecto. Esto incluye una reducción significativa en los tiempos de desarrollo y un aumento en la eficiencia del proceso de diseño. Además, el gemelo digital ha permitido identificar y abordar problemas potenciales de manera proactiva, antes de la implementación real.

4.1. Desarrollos futuros

Si bien el proyecto actual ha demostrado con éxito la funcionalidad de una estación de montaje robotizada a través de una puesta en marcha virtual utilizando el PLC virtual de PLCSIM Advanced, sería interesante para futuros desarrollos, realizar la implementación de un PLC físico en lugar del PLC virtual. Esta transición permitirá que el control de la estación se lleve a cabo en el entorno real de producción.

Una vez que se haya implementado el PLC real, se llevaría a cabo una fase de validación en el mundo real. Esto implica realizar pruebas exhaustivas para garantizar que el sistema funcione según lo previsto en un entorno físico. Las pruebas pueden incluir la interacción entre el PLC, el robot, los sensores y otros componentes.

Por otro lado el PLC de la estación de montaje, en un futuro, podría integrarse con otros sistemas de la planta y sistemas de gestión de la producción. Por ejemplo, podría conectarse Sistemas MES (Manufacturing Execution Systems). Los sistemas MES desempeñan un papel esencial en la planificación, seguimiento y control de la producción. La integración permitiría una comunicación bidireccional entre la estación de montaje y el sistema MES, lo que facilitaría la gestión de pedidos, la programación de la producción en tiempo real y la recopilación de datos de producción para análisis y mejora continua.

5. BIBLIOGRAFÍA

- [1] Universal Robots, Programme su robot para una aplicación fiable y robusta, disponible en: <https://ur-training.videomarketingplatform.co/programme-su-robot-para-una-1>
- [2] Siemens, Tecnomatix digital manufacturing software, disponible en: <https://plm.sw.siemens.com/es-ES/tecnomatix/>
- [3] Universal Robots, Robot UR3e – JT file, disponible en: <https://www.universal-robots.com/download/mechanical-e-series/ur3e/robot-ur3e-jt-file-e-series/>
- [4] Siemens, Tecnomatix eMServer Platform 14.0 Help, disponible en: https://docs.plm.automation.siemens.com/tdoc/tecnomatix/14.0/tecnomatix_eMS#uid:indx_xid1015765
- [6] Miguel Redondo Alonso, Advanced automation concepts, disponible en: <https://uvadoc.uva.es/bitstream/handle/10324/37729/TFG-I-1211.pdf?sequence=1&isAllowed=y>
- [7] Jose Alberto Valdez Becerril and Harshith Kumar Prasanna, Migration of 3D Simulation software a Real Digital Twin framework, disponible en: file:///C:/Users/adrmi/Downloads/KumarPrasannaValdezBecerril_2021pdf.pdf
- [8] Avantek, Tecnomatix Process Simulate, disponible en: <https://avantek.es/productos/tecnomatix/process-simulate/>
- [9] Irati Aresti Muñoz, Diseño e implementación del gemelo digital de célula robotizada para entorno inteligente y validación con software in the loop (sil) distribuido, disponible en: <http://hdl.handle.net/10810/53964>
- [10] 3R industria 4.0, gemelos digitales en la industria. Disponible en: <https://industria40.me/blog/gemelos-digitales-en-la-industria/>

6. ANEXO II

A continuación se describe el programa principal vinculado al robot:

MAIN

```
# WaitSignal startProgram 1
# If ((Casos_Entrada ==1 AND (Service == 1 OR Service ==2 OR Service ==3 OR Service==4))
OR(Casos_Entrada ==2 AND (Service == 5 OR Service ==6 OR Service ==7)) OR(Casos_Entrada ==3
AND (Service == 8 OR Service ==9)) OR(Casos_Entrada ==4 AND Service == 10)) Then
# CallPath OP13
# SWITCH Service
# CASE 1
# If (Casos_Entrada ==1) Then
# CallPath Tarea1_Rodamiento
# EndIf
# CASE 2
# If (Casos_Entrada ==1) Then
# CallPath Tarea1_Rodamiento
# CallPath Tarea2_Bulon
# EndIf
# CASE 3
# If (Casos_Entrada ==1) Then
# CallPath Tarea1_Rodamiento
# CallPath Tarea2_Bulon
# CallPath Tarea3_TapaInt
# EndIf
# CASE 4
# If (Casos_Entrada ==1) Then
# CallPath Tarea1_Rodamiento
# CallPath Tarea2_Bulon
# CallPath Tarea3_TapaInt
# CallPath Tarea4_TapaExt
# EndIf
# CASE 5
# If (Casos_Entrada ==2) Then
# CallPath Tarea2_Bulon
# EndIf
# CASE 6
# If (Casos_Entrada ==2) Then
# CallPath Tarea2_Bulon
# CallPath Tarea3_TapaInt
```

```
# EndIf
# CASE 7
# If (Casos_Entrada ==2) Then
# CallPath Tarea2_Bulon
# CallPath Tarea3_TapaInt
# CallPath Tarea4_TapaExt
# EndIf
# CASE 8
# If (Casos_Entrada ==3) Then
# CallPath Tarea3_TapaInt
# EndIf
# CASE 9
# If (Casos_Entrada ==3) Then
# CallPath Tarea3_TapaInt
# CallPath Tarea4_TapaExt
# EndIf
# CASE 10
# If (Casos_Entrada ==4) Then
# CallPath Tarea4_TapaExt
# EndIf
# ENDSWITCH
# CallPath FinServicios
# Else
# SetSignal Servicio_MalElegido = 1
# EndIf
```

Tarea1_Rodamiento

```
# SWITCH NItems
# CASE 1
# SetSignal Contador = 1
# CallPath Rodamiento_P1
# CASE 2
# SetSignal Contador = 1
# CallPath Rodamiento_P1
# SetSignal Contador = 2
# CallPath Rodamiento_P2
# CASE 3
# SetSignal Contador = 1
# CallPath Rodamiento_P1
```



```
# SetSignal Contador = 2
# CallPath Rodamiento_P2
# SetSignal Contador = 3
# CallPath Rodamiento_P3
# CASE 4
# SetSignal Contador = 1
# CallPath Rodamiento_P1
# SetSignal Contador = 2
# CallPath Rodamiento_P2
# SetSignal Contador = 3
# CallPath Rodamiento_P3
# SetSignal Contador = 4
# CallPath Rodamiento_P4
# CASE 5
# SetSignal Contador = 1
# CallPath Rodamiento_P1
# SetSignal Contador = 2
# CallPath Rodamiento_P2
# SetSignal Contador = 3
# CallPath Rodamiento_P3
# SetSignal Contador = 4
# CallPath Rodamiento_P4
# SetSignal Contador = 5
# CallPath Rodamiento_P5
# CASE 6
# SetSignal Contador = 1
# CallPath Rodamiento_P1
# SetSignal Contador = 2
# CallPath Rodamiento_P2
# SetSignal Contador = 3
# CallPath Rodamiento_P3
# SetSignal Contador = 4
# CallPath Rodamiento_P4
# SetSignal Contador = 5
# CallPath Rodamiento_P5
# SetSignal Contador = 6
# CallPath Rodamiento_P6
# ENDSWITCH
```

Tarea2_Bulon

```
# SWITCH NItems
# CASE 1
# SetSignal Contador = 7
# CallPath Bulon_P1
# CASE 2
# SetSignal Contador = 7
# CallPath Bulon_P1
# SetSignal Contador = 8
# CallPath Bulon_P2
# CASE 3
# SetSignal Contador = 7
# CallPath Bulon_P1
# SetSignal Contador = 8
# CallPath Bulon_P2
# SetSignal Contador = 9
# CallPath Bulon_P3
# CASE 4
# SetSignal Contador = 7
# CallPath Bulon_P1
# SetSignal Contador = 8
# CallPath Bulon_P2
# SetSignal Contador = 9
# CallPath Bulon_P3
# SetSignal Contador = 10
# CallPath Bulon_P4
# CASE 5
# SetSignal Contador = 7
# CallPath Bulon_P1
# SetSignal Contador = 8
# CallPath Bulon_P2
# SetSignal Contador = 9
# CallPath Bulon_P3
# SetSignal Contador = 10
# CallPath Bulon_P4
# SetSignal Contador = 11
# CallPath Bulon_P5
# CASE 6
# SetSignal Contador = 7
# CallPath Bulon_P1
# SetSignal Contador = 8
# CallPath Bulon_P2
```

```
# SetSignal Contador = 9
# CallPath Bulon_P3
# SetSignal Contador = 10
# CallPath Bulon_P4
# SetSignal Contador = 11
# CallPath Bulon_P5
# SetSignal Contador = 12
# CallPath Bulon_P6
# ENDSWITCH
```

Tarea3_TapaInt

```
# SWITCH NItems
# CASE 1
# SetSignal Contador = 13
# CallPath TapaInt_P1
# CASE 2
# SetSignal Contador = 13
# CallPath TapaInt_P1
# SetSignal Contador = 14
# CallPath TapaInt_P2
# CASE 3
# SetSignal Contador = 13
# CallPath TapaInt_P1
# SetSignal Contador = 14
# CallPath TapaInt_P2
# SetSignal Contador = 15
# CallPath TapaInt_P3
# CASE 4
# SetSignal Contador = 13
# CallPath TapaInt_P1
# SetSignal Contador = 14
# CallPath TapaInt_P2
# SetSignal Contador = 15
# CallPath TapaInt_P3
# SetSignal Contador = 16
# CallPath TapaInt_P4
# CASE 5
# SetSignal Contador = 13
# CallPath TapaInt_P1
# SetSignal Contador = 14
# CallPath TapaInt_P2
```

```
# SetSignal Contador = 15
# CallPath TapaInt_P3
# SetSignal Contador = 16
# CallPath TapaInt_P4
# SetSignal Contador = 17
# CallPath TapaInt_P5
# CASE 6
# SetSignal Contador = 13
# CallPath TapaInt_P1
# SetSignal Contador = 14
# CallPath TapaInt_P2
# SetSignal Contador = 15
# CallPath TapaInt_P3
# SetSignal Contador = 16
# CallPath TapaInt_P4
# SetSignal Contador = 17
# CallPath TapaInt_P5
# SetSignal Contador = 18
# CallPath TapaInt_P6
# ENDSWITCH
```

```
Tarea4_TapaExt
# SWITCH NItems
# CASE 1
# SetSignal Contador = 19
# CallPath TapaExt_P1
# CASE 2
# SetSignal Contador = 19
# CallPath TapaExt_P1
# SetSignal Contador = 20
# CallPath TapaExt_P2
# CASE 3
# SetSignal Contador = 19
# CallPath TapaExt_P1
# SetSignal Contador = 20
# CallPath TapaExt_P2
# SetSignal Contador = 21
# CallPath TapaExt_P3
```

```
# CASE 4
# SetSignal Contador = 19
# CallPath TapaExt_P1
# SetSignal Contador = 20
# CallPath TapaExt_P2
# SetSignal Contador = 21
# CallPath TapaExt_P3
# SetSignal Contador = 22
# CallPath TapaExt_P4
# CASE 5
# SetSignal Contador = 19
# CallPath TapaExt_P1
# SetSignal Contador = 20
# CallPath TapaExt_P2
# SetSignal Contador = 21
# CallPath TapaExt_P3
# SetSignal Contador = 22
# CallPath TapaExt_P4
# SetSignal Contador = 23
# CallPath TapaExt_P5
# CASE 6
# SetSignal Contador = 19
# CallPath TapaExt_P1
# SetSignal Contador = 20
# CallPath TapaExt_P2
# SetSignal Contador = 21
# CallPath TapaExt_P3
# SetSignal Contador = 22
# CallPath TapaExt_P4
# SetSignal Contador = 23
# CallPath TapaExt_P5
# SetSignal Contador = 24
# CallPath TapaExt_P6
# ENDSWITCH

FinServicios
# CallPath OP34
```

Programación para vincular los elementos ensamblados:

OP34, Universal-URScript – via1 (punto de localización del robot ejecutado antes de realizar el pick de la operación)

```
# SWITCH Service
# CASE 1
# SWITCH NItems
# CASE 1
# Attach RodB_P1 IN_Caso1_1
# CASE 2
# Attach RodB_P1 IN_Caso1_2
# Attach RodB_P2 IN_Caso1_2
# CASE 3
# Attach RodB_P1 IN_Caso1_3
# Attach RodB_P2 IN_Caso1_3
# Attach RodB_P3 IN_Caso1_3
# CASE 4
# Attach RodB_P1 IN_Caso1_4
# Attach RodB_P2 IN_Caso1_4
# Attach RodB_P3 IN_Caso1_4
# Attach RodB_P4 IN_Caso1_4
# CASE 5
# Attach RodB_P1 IN_Caso1_5
# Attach RodB_P2 IN_Caso1_5
# Attach RodB_P3 IN_Caso1_5
# Attach RodB_P4 IN_Caso1_5
# Attach RodB_P5 IN_Caso1_5
# Attach RodB_P6 IN_Caso1_5
# CASE 6
# Attach RodB_P1 IN_Caso1_6
# Attach RodB_P2 IN_Caso1_6
# Attach RodB_P3 IN_Caso1_6
# Attach RodB_P4 IN_Caso1_6
# Attach RodB_P5 IN_Caso1_6
# Attach RodB_P6 IN_Caso1_6
# ENDSWITCH
# CASE 2
# SWITCH NItems
# CASE 1
# Attach RodB_P1 IN_Caso1_1
# Attach BulonLargoP_P1 IN_Caso1_1
# CASE 2
# Attach RodB_P1 IN_Caso1_2
```

Attach RodB_P2 IN_Caso1_2
Attach BulonLargoP_P1 IN_Caso1_2
Attach BulonLargoP_P2 IN_Caso1_2
CASE 3
Attach RodB_P1 IN_Caso1_3
Attach RodB_P2 IN_Caso1_3
Attach RodB_P3 IN_Caso1_3
Attach BulonLargoP_P1 IN_Caso1_3
Attach BulonLargoP_P2 IN_Caso1_3
Attach BulonLargoP_P3 IN_Caso1_3
CASE 4
Attach RodB_P1 IN_Caso1_4
Attach RodB_P2 IN_Caso1_4
Attach RodB_P3 IN_Caso1_4
Attach RodB_P4 IN_Caso1_4
Attach BulonLargoP_P1 IN_Caso1_4
Attach BulonLargoP_P2 IN_Caso1_4
Attach BulonLargoP_P3 IN_Caso1_4
Attach BulonLargoP_P4 IN_Caso1_4
CASE 5
Attach RodB_P1 IN_Caso1_5
Attach RodB_P2 IN_Caso1_5
Attach RodB_P3 IN_Caso1_5
Attach RodB_P4 IN_Caso1_5
Attach RodB_P5 IN_Caso1_5
Attach BulonLargoP_P1 IN_Caso1_5
Attach BulonLargoP_P2 IN_Caso1_5
Attach BulonLargoP_P3 IN_Caso1_5
Attach BulonLargoP_P4 IN_Caso1_5
Attach BulonLargoP_P5 IN_Caso1_5
CASE 6
Attach RodB_P1 IN_Caso1_6
Attach RodB_P2 IN_Caso1_6
Attach RodB_P3 IN_Caso1_6
Attach RodB_P4 IN_Caso1_6
Attach RodB_P5 IN_Caso1_6
Attach RodB_P6 IN_Caso1_6
Attach BulonLargoP_P1 IN_Caso1_6
Attach BulonLargoP_P2 IN_Caso1_6
Attach BulonLargoP_P3 IN_Caso1_6
Attach BulonLargoP_P4 IN_Caso1_6

```
# Attach BulonLargoP_P5 IN_Caso1_6
# Attach BulonLargoP_P6 IN_Caso1_6
# ENDSWITCH
# CASE 3
# SWITCH NItems
# CASE 1
# Attach RodB_P1 IN_Caso1_1
# Attach BulonLargoP_P1 IN_Caso1_1
# Attach TapaInt2_P1 IN_Caso1_1
# CASE 2
# Attach RodB_P1 IN_Caso1_2
# Attach RodB_P2 IN_Caso1_2
# Attach BulonLargoP_P1 IN_Caso1_2
# Attach BulonLargoP_P2 IN_Caso1_2
# Attach TapaInt2_P1 IN_Caso1_2
# Attach TapaInt2_P2 IN_Caso1_2
# CASE 3
# Attach RodB_P1 IN_Caso1_3
# Attach RodB_P2 IN_Caso1_3
# Attach RodB_P3 IN_Caso1_3
# Attach BulonLargoP_P1 IN_Caso1_3
# Attach BulonLargoP_P2 IN_Caso1_3
# Attach BulonLargoP_P3 IN_Caso1_3
# Attach TapaInt2_P1 IN_Caso1_3
# Attach TapaInt2_P2 IN_Caso1_3
# Attach TapaInt2_P3 IN_Caso1_3
# CASE 4
# Attach RodB_P1 IN_Caso1_4
# Attach RodB_P2 IN_Caso1_4
# Attach RodB_P3 IN_Caso1_4
# Attach RodB_P4 IN_Caso1_4
# Attach BulonLargoP_P1 IN_Caso1_4
# Attach BulonLargoP_P2 IN_Caso1_4
# Attach BulonLargoP_P3 IN_Caso1_4
# Attach BulonLargoP_P4 IN_Caso1_4
# Attach TapaInt2_P1 IN_Caso1_4
# Attach TapaInt2_P2 IN_Caso1_4
# Attach TapaInt2_P3 IN_Caso1_4
# Attach TapaInt2_P4 IN_Caso1_4
# CASE 5
# Attach RodB_P1 IN_Caso1_5
```



```
# Attach RodB_P2 IN_Caso1_5
# Attach RodB_P3 IN_Caso1_5
# Attach RodB_P4 IN_Caso1_5
# Attach RodB_P5 IN_Caso1_5
# Attach BulonLargoP_P1 IN_Caso1_5
# Attach BulonLargoP_P2 IN_Caso1_5
# Attach BulonLargoP_P3 IN_Caso1_5
# Attach BulonLargoP_P4 IN_Caso1_5
# Attach BulonLargoP_P5 IN_Caso1_5
# Attach TapaInt2_P1 IN_Caso1_5
# Attach TapaInt2_P2 IN_Caso1_5
# Attach TapaInt2_P3 IN_Caso1_5
# Attach TapaInt2_P4 IN_Caso1_5
# Attach TapaInt2_P5 IN_Caso1_5
# CASE 6
# Attach RodB_P1 IN_Caso1_6
# Attach RodB_P2 IN_Caso1_6
# Attach RodB_P3 IN_Caso1_6
# Attach RodB_P4 IN_Caso1_6
# Attach RodB_P5 IN_Caso1_6
# Attach RodB_P6 IN_Caso1_6
# Attach BulonLargoP_P1 IN_Caso1_6
# Attach BulonLargoP_P2 IN_Caso1_6
# Attach BulonLargoP_P3 IN_Caso1_6
# Attach BulonLargoP_P4 IN_Caso1_6
# Attach BulonLargoP_P5 IN_Caso1_6
# Attach BulonLargoP_P6 IN_Caso1_6
# Attach TapaInt2_P1 IN_Caso1_6
# Attach TapaInt2_P2 IN_Caso1_6
# Attach TapaInt2_P3 IN_Caso1_6
# Attach TapaInt2_P4 IN_Caso1_6
# Attach TapaInt2_P5 IN_Caso1_6
# Attach TapaInt2_P6 IN_Caso1_6
# ENDSWITCH
# CASE 4
# SWITCH NItems
# CASE 1
# Attach RodB_P1 IN_Caso1_1
# Attach BulonLargoP_P1 IN_Caso1_1
# Attach TapaInt2_P1 IN_Caso1_1
# Attach TapaExt2_1 IN_Caso1_1
```

CASE 2

Attach RodB_P1 IN_Caso1_2
Attach RodB_P2 IN_Caso1_2
Attach BulonLargoP_P1 IN_Caso1_2
Attach BulonLargoP_P2 IN_Caso1_2
Attach TapaInt2_P1 IN_Caso1_2
Attach TapaInt2_P2 IN_Caso1_2
Attach TapaExt2_1 IN_Caso1_2
Attach TapaExt2_2 IN_Caso1_2

CASE 3

Attach RodB_P1 IN_Caso1_3
Attach RodB_P2 IN_Caso1_3
Attach RodB_P3 IN_Caso1_3
Attach BulonLargoP_P1 IN_Caso1_3
Attach BulonLargoP_P2 IN_Caso1_3
Attach BulonLargoP_P3 IN_Caso1_3
Attach TapaInt2_P1 IN_Caso1_3
Attach TapaInt2_P2 IN_Caso1_3
Attach TapaInt2_P3 IN_Caso1_3
Attach TapaExt2_1 IN_Caso1_3
Attach TapaExt2_2 IN_Caso1_3
Attach TapaExt2_3 IN_Caso1_3

CASE 4

Attach RodB_P1 IN_Caso1_4
Attach RodB_P2 IN_Caso1_4
Attach RodB_P3 IN_Caso1_4
Attach RodB_P4 IN_Caso1_4
Attach BulonLargoP_P1 IN_Caso1_4
Attach BulonLargoP_P2 IN_Caso1_4
Attach BulonLargoP_P3 IN_Caso1_4
Attach BulonLargoP_P4 IN_Caso1_4
Attach TapaInt2_P1 IN_Caso1_4
Attach TapaInt2_P2 IN_Caso1_4
Attach TapaInt2_P3 IN_Caso1_4
Attach TapaInt2_P4 IN_Caso1_4
Attach TapaExt2_1 IN_Caso1_4
Attach TapaExt2_2 IN_Caso1_4
Attach TapaExt2_3 IN_Caso1_4
Attach TapaExt2_4 IN_Caso1_4

CASE 5

Attach RodB_P1 IN_Caso1_5

Attach RodB_P2 IN_Caso1_5
Attach RodB_P3 IN_Caso1_5
Attach RodB_P4 IN_Caso1_5
Attach RodB_P5 IN_Caso1_5
Attach BulonLargoP_P1 IN_Caso1_5
Attach BulonLargoP_P2 IN_Caso1_5
Attach BulonLargoP_P3 IN_Caso1_5
Attach BulonLargoP_P4 IN_Caso1_5
Attach BulonLargoP_P5 IN_Caso1_5
Attach TapaInt2_P1 IN_Caso1_5
Attach TapaInt2_P2 IN_Caso1_5
Attach TapaInt2_P3 IN_Caso1_5
Attach TapaInt2_P4 IN_Caso1_5
Attach TapaInt2_P5 IN_Caso1_5
Attach TapaExt2_1 IN_Caso1_5
Attach TapaExt2_2 IN_Caso1_5
Attach TapaExt2_3 IN_Caso1_5
Attach TapaExt2_4 IN_Caso1_5
Attach TapaExt2_5 IN_Caso1_5
CASE 6
Attach RodB_P1 IN_Caso1_6
Attach RodB_P2 IN_Caso1_6
Attach RodB_P3 IN_Caso1_6
Attach RodB_P4 IN_Caso1_6
Attach RodB_P5 IN_Caso1_6
Attach RodB_P6 IN_Caso1_6
Attach BulonLargoP_P1 IN_Caso1_6
Attach BulonLargoP_P2 IN_Caso1_6
Attach BulonLargoP_P3 IN_Caso1_6
Attach BulonLargoP_P4 IN_Caso1_6
Attach BulonLargoP_P5 IN_Caso1_6
Attach BulonLargoP_P6 IN_Caso1_6
Attach TapaInt2_P1 IN_Caso1_6
Attach TapaInt2_P2 IN_Caso1_6
Attach TapaInt2_P3 IN_Caso1_6
Attach TapaInt2_P4 IN_Caso1_6
Attach TapaInt2_P5 IN_Caso1_6
Attach TapaInt2_P6 IN_Caso1_6
Attach TapaExt2_1 IN_Caso1_6
Attach TapaExt2_2 IN_Caso1_6
Attach TapaExt2_3 IN_Caso1_6

Attach TapaExt2_4 IN_Caso1_6
Attach TapaExt2_5 IN_Caso1_6
Attach TapaExt2_6 IN_Caso1_6
ENDSWITCH
CASE 5
SWITCH NItems
CASE 1
Attach BulonLargoP_P1 IN_Caso2_1
CASE 2
Attach BulonLargoP_P1 IN_Caso2_2
Attach BulonLargoP_P2 IN_Caso2_2
CASE 3
Attach BulonLargoP_P1 IN_Caso2_3
Attach BulonLargoP_P2 IN_Caso2_3
Attach BulonLargoP_P3 IN_Caso2_3
CASE 4
Attach BulonLargoP_P1 IN_Caso2_4
Attach BulonLargoP_P2 IN_Caso2_4
Attach BulonLargoP_P3 IN_Caso2_4
Attach BulonLargoP_P4 IN_Caso2_4
CASE 5
Attach BulonLargoP_P1 IN_Caso2_5
Attach BulonLargoP_P2 IN_Caso2_5
Attach BulonLargoP_P3 IN_Caso2_5
Attach BulonLargoP_P4 IN_Caso2_5
Attach BulonLargoP_P5 IN_Caso2_5
CASE 6
Attach BulonLargoP_P1 IN_Caso2_6
Attach BulonLargoP_P2 IN_Caso2_6
Attach BulonLargoP_P3 IN_Caso2_6
Attach BulonLargoP_P4 IN_Caso2_6
Attach BulonLargoP_P5 IN_Caso2_6
Attach BulonLargoP_P6 IN_Caso2_6
ENDSWITCH
CASE 6
SWITCH NItems
CASE 1
Attach BulonLargoP_P1 IN_Caso2_1
Attach TapaInt2_P1 IN_Caso2_1
CASE 2
Attach BulonLargoP_P1 IN_Caso1_2

Attach BulonLargoP_P2 IN_Caso1_2
Attach TapaInt2_P1 IN_Caso2_2
Attach TapaInt2_P2 IN_Caso2_2
CASE 3
Attach BulonLargoP_P1 IN_Caso2_3
Attach BulonLargoP_P2 IN_Caso2_3
Attach BulonLargoP_P3 IN_Caso2_3
Attach TapaInt2_P1 IN_Caso2_3
Attach TapaInt2_P2 IN_Caso2_3
Attach TapaInt2_P3 IN_Caso2_3
CASE 4
Attach BulonLargoP_P1 IN_Caso2_4
Attach BulonLargoP_P2 IN_Caso2_4
Attach BulonLargoP_P3 IN_Caso2_4
Attach BulonLargoP_P4 IN_Caso2_4
Attach TapaInt2_P1 IN_Caso2_4
Attach TapaInt2_P2 IN_Caso2_4
Attach TapaInt2_P3 IN_Caso2_4
Attach TapaInt2_P4 IN_Caso2_4
CASE 5
Attach BulonLargoP_P1 IN_Caso2_5
Attach BulonLargoP_P2 IN_Caso2_5
Attach BulonLargoP_P3 IN_Caso2_5
Attach BulonLargoP_P4 IN_Caso2_5
Attach BulonLargoP_P5 IN_Caso2_5
Attach TapaInt2_P1 IN_Caso2_5
Attach TapaInt2_P2 IN_Caso2_5
Attach TapaInt2_P3 IN_Caso2_5
Attach TapaInt2_P4 IN_Caso2_5
Attach TapaInt2_P5 IN_Caso2_5
CASE 6
Attach BulonLargoP_P1 IN_Caso2_6
Attach BulonLargoP_P2 IN_Caso2_6
Attach BulonLargoP_P3 IN_Caso2_6
Attach BulonLargoP_P4 IN_Caso2_6
Attach BulonLargoP_P5 IN_Caso2_6
Attach BulonLargoP_P6 IN_Caso2_6
Attach TapaInt2_P1 IN_Caso2_6
Attach TapaInt2_P2 IN_Caso2_6
Attach TapaInt2_P3 IN_Caso2_6
Attach TapaInt2_P4 IN_Caso2_6

```
# Attach TapaInt2_P5 IN_Caso2_6
# Attach TapaInt2_P6 IN_Caso2_6
# ENDSWITCH
# CASE 7
# SWITCH NItems
# CASE 1
# Attach BulonLargoP_P1 IN_Caso2_1
# Attach TapaInt2_P1 IN_Caso2_1
# Attach TapaExt2_1 IN_Caso2_1
# CASE 2
# Attach BulonLargoP_P1 IN_Caso2_2
# Attach BulonLargoP_P2 IN_Caso2_2
# Attach TapaInt2_P1 IN_Caso2_2
# Attach TapaInt2_P2 IN_Caso2_2
# Attach TapaExt2_1 IN_Caso2_2
# Attach TapaExt2_2 IN_Caso2_2
# CASE 3
# Attach BulonLargoP_P1 IN_Caso2_3
# Attach BulonLargoP_P2 IN_Caso2_3
# Attach BulonLargoP_P3 IN_Caso2_3
# Attach TapaInt2_P1 IN_Caso2_3
# Attach TapaInt2_P2 IN_Caso2_3
# Attach TapaInt2_P3 IN_Caso2_3
# Attach TapaExt2_1 IN_Caso2_3
# Attach TapaExt2_2 IN_Caso2_3
# Attach TapaExt2_3 IN_Caso2_3
# CASE 4
# Attach BulonLargoP_P1 IN_Caso2_4
# Attach BulonLargoP_P2 IN_Caso2_4
# Attach BulonLargoP_P3 IN_Caso2_4
# Attach BulonLargoP_P4 IN_Caso2_4
# Attach TapaInt2_P1 IN_Caso2_4
# Attach TapaInt2_P2 IN_Caso2_4
# Attach TapaInt2_P3 IN_Caso2_4
# Attach TapaInt2_P4 IN_Caso2_4
# Attach TapaExt2_1 IN_Caso2_4
# Attach TapaExt2_2 IN_Caso2_4
# Attach TapaExt2_3 IN_Caso2_4
# Attach TapaExt2_4 IN_Caso2_4
# CASE 5
# Attach BulonLargoP_P1 IN_Caso2_5
```

Attach BulonLargoP_P2 IN_Caso2_5
Attach BulonLargoP_P3 IN_Caso2_5
Attach BulonLargoP_P4 IN_Caso2_5
Attach BulonLargoP_P5 IN_Caso2_5
Attach TapaInt2_P1 IN_Caso2_5
Attach TapaInt2_P2 IN_Caso2_5
Attach TapaInt2_P3 IN_Caso2_5
Attach TapaInt2_P4 IN_Caso2_5
Attach TapaInt2_P5 IN_Caso2_5
Attach TapaExt2_1 IN_Caso2_5
Attach TapaExt2_2 IN_Caso2_5
Attach TapaExt2_3 IN_Caso2_5
Attach TapaExt2_4 IN_Caso2_5
Attach TapaExt2_5 IN_Caso2_5
CASE 6
Attach BulonLargoP_P1 IN_Caso2_6
Attach BulonLargoP_P2 IN_Caso2_6
Attach BulonLargoP_P3 IN_Caso2_6
Attach BulonLargoP_P4 IN_Caso2_6
Attach BulonLargoP_P5 IN_Caso2_6
Attach BulonLargoP_P6 IN_Caso2_6
Attach TapaInt2_P1 IN_Caso2_6
Attach TapaInt2_P2 IN_Caso2_6
Attach TapaInt2_P3 IN_Caso2_6
Attach TapaInt2_P4 IN_Caso2_6
Attach TapaInt2_P5 IN_Caso2_6
Attach TapaInt2_P6 IN_Caso2_6
Attach TapaExt2_1 IN_Caso2_6
Attach TapaExt2_2 IN_Caso2_6
Attach TapaExt2_3 IN_Caso2_6
Attach TapaExt2_4 IN_Caso2_6
Attach TapaExt2_5 IN_Caso2_6
Attach TapaExt2_6 IN_Caso2_6
ENDSWITCH
CASE 8
SWITCH NItems
CASE 1
Attach TapaInt2_P1 IN_Caso3_1
CASE 2
Attach TapaInt2_P1 IN_Caso3_2
Attach TapaInt2_P2 IN_Caso3_2

```
# CASE 3
# Attach TapaInt2_P1 IN_Caso3_3
# Attach TapaInt2_P2 IN_Caso3_3
# Attach TapaInt2_P3 IN_Caso3_3
# CASE 4
# Attach TapaInt2_P1 IN_Caso3_4
# Attach TapaInt2_P2 IN_Caso3_4
# Attach TapaInt2_P3 IN_Caso3_4
# Attach TapaInt2_P4 IN_Caso3_4
# CASE 5
# Attach TapaInt2_P1 IN_Caso3_5
# Attach TapaInt2_P1 IN_Caso3_5
# Attach TapaInt2_P3 IN_Caso3_5
# Attach TapaInt2_P4 IN_Caso3_5
# Attach TapaInt2_P5 IN_Caso3_5
# CASE 6
# Attach TapaInt2_P1 IN_Caso3_6
# Attach TapaInt2_P2 IN_Caso3_6
# Attach TapaInt2_P3 IN_Caso3_6
# Attach TapaInt2_P4 IN_Caso3_6
# Attach TapaInt2_P5 IN_Caso3_6
# Attach TapaInt2_P6 IN_Caso3_6
# ENDSWITCH
# CASE 9
# SWITCH NItems
# CASE 1
# Attach TapaInt2_P1 IN_Caso3_1
# Attach TapaExt2_1 IN_Caso3_1
# CASE 2
# Attach TapaInt2_P1 IN_Caso3_2
# Attach TapaInt2_P2 IN_Caso3_2
# Attach TapaExt2_1 IN_Caso3_2
# Attach TapaExt2_2 IN_Caso3_2
# CASE 3
# Attach TapaInt2_P1 IN_Caso3_3
# Attach TapaInt2_P2 IN_Caso3_3
# Attach TapaInt2_P3 IN_Caso3_3
# Attach TapaExt2_1 IN_Caso3_3
# Attach TapaExt2_2 IN_Caso3_3
# Attach TapaExt2_3 IN_Caso3_3
# CASE 4
```



```
# Attach TapaInt2_P1 IN_Caso3_4
# Attach TapaInt2_P2 IN_Caso3_4
# Attach TapaInt2_P3 IN_Caso3_4
# Attach TapaInt2_P4 IN_Caso3_4
# Attach TapaExt2_1 IN_Caso3_4
# Attach TapaExt2_2 IN_Caso3_4
# Attach TapaExt2_3 IN_Caso3_4
# Attach TapaExt2_4 IN_Caso3_4
# CASE 5
# Attach TapaInt2_P1 IN_Caso3_5
# Attach TapaInt2_P2 IN_Caso3_5
# Attach TapaInt2_P3 IN_Caso3_5
# Attach TapaInt2_P4 IN_Caso3_5
# Attach TapaInt2_P5 IN_Caso3_5
# Attach TapaExt2_1 IN_Caso3_5
# Attach TapaExt2_2 IN_Caso3_5
# Attach TapaExt2_3 IN_Caso3_5
# Attach TapaExt2_4 IN_Caso3_5
# Attach TapaExt2_5 IN_Caso3_5
# CASE 6
# Attach TapaInt2_P1 IN_Caso3_6
# Attach TapaInt2_P2 IN_Caso3_6
# Attach TapaInt2_P3 IN_Caso3_6
# Attach TapaInt2_P4 IN_Caso3_6
# Attach TapaInt2_P5 IN_Caso3_6
# Attach TapaInt2_P6 IN_Caso3_6
# Attach TapaExt2_1 IN_Caso3_6
# Attach TapaExt2_2 IN_Caso3_6
# Attach TapaExt2_3 IN_Caso3_6
# Attach TapaExt2_4 IN_Caso3_6
# Attach TapaExt2_5 IN_Caso3_6
# Attach TapaExt2_6 IN_Caso3_6
# ENDSWITCH
# CASE 10
# SWITCH NItems
# CASE 1
# Attach TapaExt2_1 IN_Caso4_1
# CASE 2
# Attach TapaExt2_1 IN_Caso4_2
# Attach TapaExt2_2 IN_Caso4_2
# CASE 3
```

```
# Attach TapaExt2_1 IN_Caso4_3
# Attach TapaExt2_2 IN_Caso4_3
# Attach TapaExt2_3 IN_Caso4_3
# CASE 4
# Attach TapaExt2_1 IN_Caso4_4
# Attach TapaExt2_2 IN_Caso4_4
# Attach TapaExt2_3 IN_Caso4_4
# Attach TapaExt2_4 IN_Caso4_4
# CASE 5
# Attach TapaExt2_1 IN_Caso4_5
# Attach TapaExt2_2 IN_Caso1_5
# Attach TapaExt2_3 IN_Caso4_5
# Attach TapaExt2_4 IN_Caso4_5
# Attach TapaExt2_5 IN_Caso4_5
# CASE 6
# Attach TapaExt2_1 IN_Caso4_6
# Attach TapaExt2_2 IN_Caso4_6
# Attach TapaExt2_3 IN_Caso4_6
# Attach TapaExt2_4 IN_Caso4_6
# Attach TapaExt2_5 IN_Caso4_6
# Attach TapaExt2_6 IN_Caso4_6
# ENDSWITCH
# ENDSWITCH
```

Programación para que desaparezca el palet de la zona de salida:

```
OP34, Universal-URScript – via
# WaitSignal Palet_Recogido 1
# Blank IN_Caso1_1
# Blank IN_Caso1_2
# Blank IN_Caso1_3
# Blank IN_Caso1_4
# Blank IN_Caso1_5
# Blank IN_Caso1_6
# Blank IN_Caso2_2
# Blank IN_Caso2_1
# Blank IN_Caso2_3
# Blank IN_Caso2_4
# Blank IN_Caso2_5
# Blank IN_Caso2_6
# Blank IN_Caso3_1
```

```
# Blank IN_Caso3_2
# Blank IN_Caso3_3
# Blank IN_Caso3_4
# Blank IN_Caso3_5
# Blank IN_Caso3_6
# Blank IN_Caso4_1
# Blank IN_Caso4_2
# Blank IN_Caso4_3
# Blank IN_Caso4_4
# Blank IN_Caso4_5
# Blank IN_Caso4_6
# Blank RodB_P1
# Blank RodB_P2
# Blank RodB_P3
# Blank RodB_P4
# Blank RodB_P5
# Blank RodB_P6
# Blank BulonLargoP_P2
# Blank BulonLargoP_P3
# Blank BulonLargoP_P4
# Blank BulonLargoP_P5
# Blank BulonLargoP_P6
# Blank TapaInt2_P1
# Blank TapaInt2_P2
# Blank TapaInt2_P3
# Blank TapaInt2_P4
# Blank TapaInt2_P5
# Blank TapaInt2_P6
# Blank TapaExt2_1
# Blank TapaExt2_2
# Blank TapaExt2_3
# Blank TapaExt2_4
# Blank TapaExt2_5
# Blank TapaExt2_6
# Send robotReady 0
```