

MÁSTER UNIVERSITARIO EN INGENIERÍA DE CONTROL,
AUTOMATIZACIÓN Y ROBÓTICA

TRABAJO FIN DE MÁSTER

MODELADO Y PUESTA EN MARCHA VIRTUAL DEL GEMELO DIGITAL DE UNA MÁQUINA FLOWPACK CON LA HERRAMIENTA MELSOFT GEMINI



Estudiante: Romano Añez, Jesica

Director/Directora: Casquero Oyarzabal, Oskar

Codirector/Codirectora: Armentia Díaz de Tuesta, Aintzane

Curso: 2022-2023

Fecha: Bilbao, 17, julio, 2023

RESUMEN

Un gemelo digital es una representación virtual de un dispositivo o proceso de una planta. Las tecnologías para el modelado de gemelos digitales permiten realizar la puesta en marcha virtual o *virtual commissioning* de un proceso, que consiste en la verificación offline del sistema de control del proceso mediante la simulación del funcionamiento conjunto de gemelo digital del proceso y el código del controlador. La puesta en marcha virtual se puede realizar mediante simulación *Software in the Loop* (SiL) o *Hardware in the Loop* (HiL). La simulación SiL implica la virtualización tanto del controlador como la del proceso, mientras que la simulación HiL únicamente implica la virtualización del proceso. En este contexto, en este proyecto se realiza la puesta en marcha virtual mediante simulación SiL de una máquina Flowpack destinada a una línea de producción de la industria alimentaria. Para ello, se utilizan las herramientas software del porfolio PLM de la empresa Mitsubishi Electric, a saber: la nueva herramienta de simulación de Gemelos Digitales (*Gemini*), la herramienta de programación de controladores (*GX Works3*) y la herramienta de programación de HMI (*GT Designer3*).

Palabras clave: Gemelo Digital, puesta en marcha virtual, simulación, MELSOFT Gemini.

LABURPENA

Biki-digital bat gailu jakin baten edota prozesu oso baten irudikapen birtuala da. Prozesua martxan ipini daiteke era birtualean (*virtual commissioning*) biki-digitalak eratzeko dauden teknologiei esker; hau da, sistemaren biki-digitala eta hura menderatzen duen kontrolagailuaren kodea egiaztatu daitezke simulazioen bidez offline. Prozesua bi era ezberdinetan jar daiteke martxan, *Software in the Loop* (SiL) edo *Hardware in the Loop* (HiL) bitartez. SiL simulazioetan prozesua eta kontrolagailua irudikatzen dira era birtualean; HiL-etan aldiz, bakarrik prozesua. Testuinguru honetan, proiektu honetan elikadura-industriarako Flowpack makina bat martxan ipini nahi da era birtualena SiL motako simulazio bitartez. Horretarako, Mitsubishi Electric enpresaren PLM portfolioaren software erramintak erabiliko dira, Biki-Digitalen simulazioetarako auzeratu duten tresna berria (*Gemini*), kontrolagailuak programatzeko erraminta (*GX Works3*) eta HMI pantailak programatzeko erraminta (*GT Designer3*) izango dira aipagarrienak.

Gako Hitzak: Biki Digital, abiarazpen birtuala, simulazioa, MELSOFT Gemini.

ABSTRACT

A digital twin is a virtual representation of a device or process in a plant. Technologies used in modeling of digital twins allow for the virtual commissioning of a process, that consists of the offline verification of its control system through the running simulation of the digital twin along the controller code. The virtual commissioning can be performed using *Software in the Loop* (SiL) simulation or *Hardware in the Loop* (HiL). SiL simulation implies the virtualization not only of the controller but the of process as well, whereas HiL simulation only implies the virtualization of the process. In this context, this project details the virtual commissioning, using SiL simulation, of a Flowpack machine destined to a food industry production line. For this, the following software tools belonging to Mitsubishi Electric's PLM portfolio are used: the new Digital Twin simulation tool (*Gemini*), the controller programming tool (*GX Works3*) and the HMI programming tool (*GT Designer3*).

Key words: Digital Twin, Virtual Commissioning, simulation, MELSOFT Gemini.

ÍNDICE

1. INTRODUCCIÓN.....	1
2. CONTEXTO	3
3. OBJETIVOS	4
4. BENEFICIOS	5
4.1. Beneficios económicos.....	5
4.2. Beneficios técnicos.....	5
4.3. Beneficios sociales.....	6
5. REQUERIMIENTOS.....	6
6. DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA.....	7
7. DISEÑO	8
7.1. Servicios.....	8
7.2. Gemelo Digital.....	8
7.2.1. MELSOFT Gemini.....	9
7.2.2. Modelo CAD	19
7.2.3. Modelo cinemático	22
7.2.4. Validación del Gemelo Digital	27
7.2.5. Señales E/S	28
7.3. Programa de control	28
7.3.1. Diseño hardware del sistema	29
7.3.2. Configuración del módulo de Simple Motion	31
7.3.3. Programación del autómata programable (PLC).....	33
7.3.4. Interfaz máquina-humano (HMI)	35
8. PLAN DE TRABAJO	38
8.1. PT1: Formación.....	38
8.1.1. PT1.T1: MELSOFT Gemini	38
8.1.2. PT1.T2: Programación módulo Simple Motion	38
8.1.3. PT1.T3: GT Designer3	39
8.2. PT2: Desarrollo	39
8.2.1. PT2.T1: Modelado CAD de la máquina Flowpack.....	39
8.2.2. PT2.T2: Modelado del Gemelo Digital.....	40
8.2.3. PT2.T3: Programa de control.....	40
8.2.4. PT2.T4: Conexión Gemelo-PLC.....	40
8.2.5. PT2.T5: Panel de operador	41
8.3. PT3: Pruebas.....	41

8.3.1.	PT3.T1: Importación modelo CAD a MELSOFT Gemini	41
8.3.2.	PT3.T2: Validación del Gemelo Digital en Gemini	41
8.3.3.	PT3.T3: Validación del conexionado PLC-Gemini	42
8.3.4.	PT3.T4: Validación del programa de control	42
8.3.5.	PT3.T5: Simulación completa	43
8.4.	PT4: Administrativo	43
8.4.1.	PT4.T1: Documentación del proyecto	43
9.	PRESUPUESTO	45
10.	RESULTADOS	47
11.	LIMITACIONES	47
12.	CONCLUSIONES Y TRABAJO FUTURO	49
	REFERENCIAS	50

ÍNDICE DE FIGURAS

Figura 1: Revoluciones industriales	1
Figura 2: Tecnologías base de la Industria 4.0.....	2
Figura 3: Representación gráfica del sistema SiL.....	4
Figura 4: Ejemplo de máquina empaquetadora Flowpack.....	7
Figura 5: Diagrama de conjuntos.....	8
Figura 6: 3D Simulator MELSOFT Gemini	9
Figura 7: Librería "eCatalogue"	10
Figura 8: Asistente de componentes	11
Figura 9: Comportamientos referentes a un Transportador	11
Figura 10: Propiedades de un "path" o ruta	12
Figura 11: Path con orden original	12
Figura 12: Path con orden modificado	13
Figura 13: Coordenada "Pathframe"	13
Figura 14: Comportamientos disponibles	14
Figura 15: Ejemplo de "Script de Python"	15
Figura 16: Propiedades disponibles.....	15
Figura 17: Sensor dentro de la librería "Conveyor Utilities"	16
Figura 18: Propiedades del sensor	16
Figura 19: Comportamientos del sensor	17
Figura 20: Tipos de articulación	17
Figura 21: Propiedades de una articulación rotativa.....	18
Figura 22: Feeder y barra de caramelo dentro de "eCatalogue".....	19
Figura 23: Esquema de una máquina Flowpack.....	19
Figura 24: Máquina Flowpack (diseño simplificado).....	20
Figura 25: Máquina Flowpack (diseño completo)	21
Figura 26: Componentes de la máquina empaquetadora.....	21
Figura 27: Parametrizado del "Feeder"	22
Figura 28: Cinta de entrada, características de Transportador	23
Figura 29: Script de Python, asignación de velocidad del Transportador	24
Figura 30: Detalle de los sentidos de giro de los rodillos de sellado.....	25
Figura 31: Path de un Transportador.....	26
Figura 32: Path de un "Feeder"	26
Figura 33: Superficie de transporte transparente	27

Figura 34: Señales de E/S del servidor	28
Figura 35: Configuración hardware (controlador).....	29
Figura 36: Configuración hardware (red de servoamplificadores)	30
Figura 37: Diagrama hardware completo.....	30
Figura 38: Configuración software de la mecánica asociada al servomotor	31
Figura 39: Configuración software del sincronismo (eje 2)	32
Figura 40: Configuración software del sincronismo (ejes 3 y 4).....	32
Figura 41: Variables de una leva electrónica	33
Figura 42: Interfaz máquina-humano, pantalla de Inicio	35
Figura 43: Interfaz máquina-humano, pantalla de Estado	36
Figura 44: Interfaz máquina-humano, pantalla de Automático	37
Figura 45: Interfaz máquina-humano, pantalla de Datos.....	37
Figura 46: Ventana de salida en Gemini.....	48

ÍNDICE DE TABLAS

Tabla 1: Plan de trabajo	44
Tabla 2: Bienes materiales.....	45
Tabla 3: Presupuesto de amortización	45
Tabla 4: Presupuesto del grupo de trabajo	46
Tabla 5: Presupuesto total.....	46

1. INTRODUCCIÓN

Las revoluciones industriales constituyen grandes transformaciones sociales derivadas de avances tecnológicos que han marcado claros hitos en el desarrollo científico de la humanidad. En los últimos 250 años, el mundo ha experimentado tres revoluciones industriales (Figura 1). La primera revolución industrial tuvo lugar entre los siglos XVIII y XIX, y supuso la mecanización de las tareas, principalmente mediante el uso de maquinaria propulsada por vapor, lo que dio lugar al auge del sistema fabril. A finales del siglo XIX y principios del XX tuvo lugar la segunda revolución industrial, en la que los nuevos avances relacionados principalmente con la electricidad y los combustibles fósiles, y los métodos de fabricación de piezas estandarizadas e intercambiables, propiciaron la aparición de fábricas capaces de producir en masa. En la segunda mitad del siglo XX, la llegada de la electrónica y la informática hizo posible la automatización de las líneas de producción, dando lugar a lo que se conoce como la tercera revolución industrial. Esta revolución dio lugar a la era de la automatización de alto nivel en la producción gracias a dos grandes inventos: Los autómatas programables (PLC) y los robots.

Hoy en día, toda la comunidad científica y la sociedad en general aceptan y reconocen que estas tres revoluciones industriales han tenido lugar hasta la fecha. Sin embargo, las personas que vivieron estos momentos de transformación económica y social no los identificaron como tales. Definir un conjunto de acontecimientos como revolución industrial es relativamente fácil con la perspectiva que da la historia: una vez que la transformación de la economía y la sociedad se manifiesta y consolida, es fácil recorrer el camino inverso para encontrar las causas profundas. Sin embargo, es mucho más difícil identificar una revolución desde la perspectiva del presente, ya que es necesario predecir el futuro, imaginando el nivel de ruptura necesario en las estructuras sociales y económicas para hacerla realidad. En este sentido, la industria está experimentando una profunda y vertiginosa transformación que ha dado lugar a un torrente de iniciativas y avances tecnológicos que muchas instituciones han convenido en identificar como la cuarta revolución industrial. Este paradigma radica principalmente en la distribución de la inteligencia en los sistemas, entendida como la capacidad de las entidades que los componen para interactuar interpretando información y tomando decisiones orientadas a la consecución de sus objetivos. La transversalidad de este paradigma permite su aplicación en múltiples ámbitos, por ejemplo, ciudades inteligentes, redes o fábricas inteligentes.

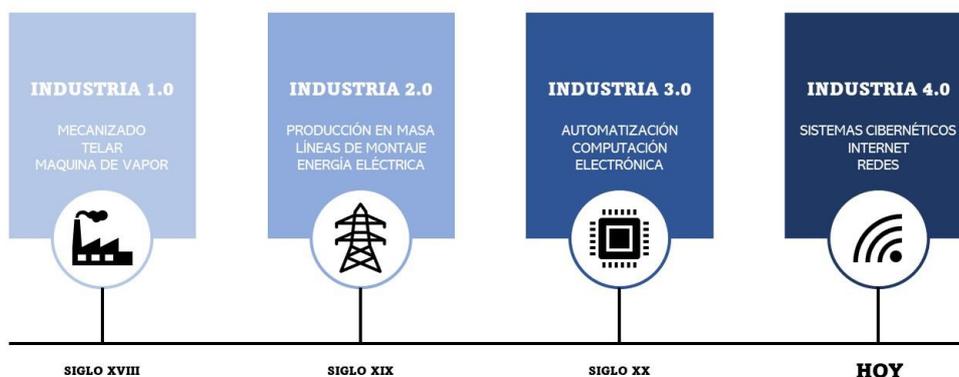


Figura 1: Revoluciones industriales

El concepto de fábrica inteligente o *Smart Factory* se articula en torno a diferentes tecnologías: sistemas ciber-físicos, robótica, inteligencia artificial, la computación en la nube y el Internet of Things, entre otras. El presente Trabajo Fin de Máster (TFM) se enmarca en el área de la simulación, dentro de la cual trata expresamente los aspectos de modelado de gemelos digitales y el de puesta en marcha virtual o *virtual commissioning*.



Figura 2: Tecnologías base de la Industria 4.0

El gemelo digital es una de las tecnologías más disruptivas que han surgido en torno a la digitalización de los procesos productivos [1]. Esta digitalización es la que requiere de una integración total de los procesos mediante la recogida de los datos producidos en las distintas fases del proceso [2]. En este sentido, los sistemas de fabricación actuales son un claro objetivo para la creación de Gemelos Digitales, ya que los distintos problemas que surgen en diseño, desarrollo y prueba de un producto realzan el principal atractivo de esta nueva tecnología.

Un Gemelo Digital puede definirse como una representación virtual de los elementos físicos (incluyendo su cinemática y, opcionalmente, dinámica) y lógicos que constituyen un sistema, dispositivo o servicio a lo largo de su vida útil. Dicha representación puede dividirse en cuatro dimensiones de modelado: geométrica, física, comportamiento y control [3]. Los modelos geométrico y físico en unión con el modelo de comportamiento y control permiten hacer converger el mundo físico con el virtual mediante diversas tecnologías [4]. El Gemelo Digital tiene diversos usos: puede utilizarse como herramienta de recolección, verificación y mantenimiento de datos; como herramienta de análisis del proceso para llevar a cabo mejoras en éste; como herramienta de monitorización del producto durante su vida útil y de soporte a su mantenimiento predictivo [5]; y como herramienta de apoyo al desarrollo de gemelo real durante las fases de diseño, construcción y operación.

Para la simulación de todas las fases en las que puede intervenir un Gemelo Digital, es necesario hablar de la puesta en marcha virtual o *Virtual Commissioning*. La puesta en marcha consiste en el proceso de arranque, regulación y equilibrado de los equipos y sistemas de forma planificada. Se trata un tipo de simulación que ayuda a depurar las fases de diseño y montaje. La puesta en marcha virtual es fundamentalmente el mapeado entre el modelo y el controlador que ejecuta el programa de control. Al realizar la puesta en marcha virtualmente, se puede simular el comportamiento de la estación completa o de partes individuales del proceso mediante la conexión del hardware de control y el gemelo digital. El objetivo final de esta es reducir los errores que puedan darse en la puesta en marcha real de la estación.

La puesta en marcha virtual se puede realizar mediante simulación *Software in the Loop* (SiL) o *Hardware in the Loop* (HiL). La simulación SiL implica la virtualización tanto del controlador lógico programable (PLC) como la del proceso mediante el modelado de un gemelo digital, mientras que la simulación HiL únicamente implica la virtualización del proceso.

Este proyecto trata de la puesta en marcha virtual mediante simulación SiL de una máquina Flowpack destinada a una línea de producción de la industria alimentaria. Para ello, se utilizan las herramientas software del porfolio PLM de la empresa Mitsubishi Electric, a saber: la herramienta de simulación de Gemelos Digitales (*Gemini*), la herramienta de programación de PLC (*GX Works3*) y la herramienta de programación de HMI (*GT Designer3*).

2. CONTEXTO

La influencia del Gemelo Digital afecta a todos los aspectos del desarrollo de su gemelo físico, como son las fases de diseño, de construcción y de operación [6]. En lo que respecta a la fase de diseño, el Gemelo Digital aporta la capacidad de gestionar y perfeccionar el proceso o producto de forma online, de manera que se evitan los altos costes que vienen de la fabricación de prototipos físicos. A su vez, se genera una interacción con el proceso que permite la detección preventiva de errores. Por otro lado, en la fase de construcción, la implantación de un Gemelo Digital ayuda en la optimización de los procesos constructivos, siendo parte del proceso de planificación de un proyecto. Y, por último, en la fase de operación, la obtención de datos en tiempo real a través de los sensores permite realizar una continua recalibración del gemelo digital junto al gemelo físico a lo largo de su vida útil. El procesamiento de estos datos introduce el uso de las tecnologías de Big Data y Cloud Computing lo que permite el funcionamiento fluido del Gemelo Digital [7].

Es por esto por lo que la aparición de los gemelos digitales ha significado una transformación en los procesos industriales, al ser, en el fondo, la digitalización de un sistema, producto o servicio. Es esta digitalización la que permite crear productos inteligentes y tomar decisiones de manera más rápida [8]. En el contexto de la empresa Mitsubishi Electric, se ha visto la posibilidad de mercado de los Gemelos Digitales y es por esto por lo que se ha invertido en el desarrollo de una herramienta software que permita realizar el modelado de estos. Esta herramienta debe ser compatible y completamente funcional dentro del paquete de software ya existente en la empresa.

En este proyecto se lleva a cabo el primer desarrollo de un Gemelo Digital haciendo uso de la nueva herramienta, MELSOFT Gemini, en España. La realización de este proyecto surge al lanzarse dicha herramienta al mercado, viéndose la necesidad de comprender el funcionamiento y las funcionalidades de esta.

Esta herramienta impulsa la introducción de la empresa en la industria 4.0 para hacer frente a los continuos cambios en el mercado, la variación de los productos demandados y la diversificación de estos. Para llegar a cumplir estos objetivos, es imprescindible gestionar la digitalización de la empresa integrando nuevas tecnologías para mejorar así los niveles de automatización y eficiencia.

3. OBJETIVOS

Este proyecto tiene como objetivo principal la puesta en marcha virtual de una máquina Flowpack destinada a una línea de producción de la industria alimentaria. Para alcanzar ese objetivo principal, se plantean los siguientes objetivos secundarios:

- **O1. Desarrollo del Gemelo Digital de la máquina Flowpack.** Se trata de modelar el proceso en una plataforma CAD externa. Se desea obtener una representación sencilla, pero fidedigna desde un punto de vista funcional, con los elementos mínimos necesarios que permitan simular el comportamiento mecatrónico de la máquina. Sin embargo, el nivel de detalle del diseño CAD deberá ser tan preciso como lo requiera la simulación.

Por último, se requiere de la validación del funcionamiento del proceso completo, dentro del entorno de simulación, previo a la implementación del Gemelo Digital. Es recomendable realizar la validación de manera independiente al control que se realiza posteriormente.

- **O2. Diseño del programa de control.** Se trata de desarrollar el programa de PLC necesario para automatizar la máquina de manera que funcione de la manera indicada por el cliente. Esto incluye la configuración hardware de los diferentes dispositivos que intervienen en la automatización del proceso.
- **O3. Simulación de proceso.** Se trata de probar el funcionamiento del programa de control dentro de una simulación que contiene tanto el gemelo digital del sistema físico real, como la instancia de PLC virtual que se utiliza para controlarlo (es decir, una simulación *SiL*).



Figura 3: Representación gráfica del sistema *SiL*

4. BENEFICIOS

La tecnología del Gemelo Digital incrementa la productividad y la rentabilidad de los procesos industriales, ya que al simular el funcionamiento del programa de control se puede analizar y verificar el comportamiento de controlador, reduciendo así el tiempo de la puesta en marcha [9]. Además, el disponer de un Gemelo Digital de todo un proceso industrial permite formar al personal sin la necesidad de estar presentes en la planta o pararla, haciendo que el tiempo de adaptación de los trabajadores sea menor. También permite realizar la optimización del proceso tanto de manera online como off-line, planificar la producción, adquirir datos en tiempo real, etcétera [9, 10]. Así mismo, el desarrollo de un Gemelo Digital trae consigo una reducción de costes y riesgos, así como un incremento de fiabilidad [11].

4.1. Beneficios económicos

Uno de los principales beneficios del Gemelo Digital es la optimización de los procesos que se simulan. El Gemelo Digital permite simular y optimizar el rendimiento de un sistema en tiempo real. Como consecuencia, la eficiencia operativa del sistema será mayor y se verá una disminución en los costes y un ahorro de recursos.

Por otro lado, la implementación de un Gemelo Digital trae consigo la toma de decisiones informada. Es decir, al contar con un modelo digital preciso de un proceso, es posible basar las decisiones en datos confiables y, además, en tiempo real. Esto implica que los riesgos se minimizan y se maximizan los beneficios económicos.

Por último, cabe destacar que al implementar el Gemelo Digital los tiempos de desarrollo se ven reducidos. Esto se debe a que, al poder realizar pruebas y simulaciones virtuales, se acelera el proceso de diseño y desarrollo de productos o procesos. Esto se puede traducir en una reducción en los tiempos de comercialización y, por tanto, se puede considerar una ventaja competitiva.

4.2. Beneficios técnicos

El Gemelo Digital permite probar y refinar diseños en las fases previas a la producción física, por lo que se genera un diseño y producción avanzado. Esto hace que sea posible identificar y corregir errores y problemas con antelación que a su vez lleva a un mayor rendimiento y una optimización del diseño.

De la mano del anterior viene el mantenimiento predictivo. Al realizar una monitorización constante del proceso mediante su Gemelo Digital, es posible realizar una detección temprana de fallos y así poder programar el mantenimiento del proceso físico de manera predictiva. De esta manera se evitan tiempos de inactividad costosos y prolongados.

Además, el gemelo digital de un proceso contribuye a optimizar el proceso de retrofitting de una máquina al permitir simular su comportamiento cuando se modifican sus componentes o se le añaden nuevas funcionalidades.

Por último, un gran beneficio técnico que aporta la implementación del Gemelo Digital es la integración de sistemas. El Gemelo Digital facilita la interoperabilidad entre diferentes sistemas y procesos al proporcionar una plataforma común para compartir datos. De esta manera se mejora la colaboración y la eficiencia en los entornos industriales.

4.3. Beneficios sociales

Los Gemelos Digitales aportan un factor de seguridad para el personal, ya que permiten realizar simulaciones y evaluaciones de situaciones de riesgo sin la necesidad de someter al personal a entornos peligrosos. Esto ayuda a mejorar la seguridad del personal y a reducir accidentes y lesiones.

Por otro lado, un uso poco destacado, pero muy útil de los Gemelos Digitales, es el de entornos de aprendizaje y entrenamiento. Esto se puede llevar a diversos campos; incluso dentro de la ingeniería. Al disponer de un modelo preciso de un proceso físico real, es posible formar a los empleados que deberán trabajar en este de manera simulada, ofreciendo así un entorno de aprendizaje seguro y controlado.

Por último, se debe considerar el aspecto de la sostenibilidad. Al implementar un Gemelo Digital es posible reducir el consumo de recursos, optimizar la eficiencia energética y minimizar el impacto medioambiental, contribuyendo a la sostenibilidad.

5. REQUERIMIENTOS

En lo que respecta a los requerimientos del proyecto, existen dos tipos a tener en cuenta: los requerimientos funcionales, es decir, como tiene que funcionar el proceso; y los requerimientos no funcionales, es decir, los que no tienen que ver con el funcionamiento del proceso.

REQUERIMIENTOS FUNCIONALES

- **Sincronismo de ejes.** Se trata de reflejar el movimiento de tres ejes sincronizados. El funcionamiento completo de la máquina se basa en el sincronismo de los tres ejes principales que permiten realizar paquetes de producto iguales siguiendo un proceso sencillo.
- **Flujo continuo de producto.** Para el correcto funcionamiento del proceso es necesario que el producto a empaquetar fluya por la máquina de manera continua. La distancia entre producto debe ser siempre la misma, de manera que los paquetes sean siempre iguales y contengan el producto en la posición correcta.
- **Simulación del proyecto.** Todo el proyecto será realizado mediante simulación del hardware a la vez que se realiza la simulación del Gemelo Digital como tal. Es obvia la necesidad del funcionamiento fluido entre todos los elementos simulados simultáneamente.

REQUERIMIENTOS NO FUNCIONALES

- **Modelo CAD.** Es necesario disponer de un modelo CAD lo suficientemente detallado para poder realizar un modelado adecuado a partir de él. Este será la base fundamental del proyecto y determinará que tan detallado es posible realizar el Gemelo Digital.
- **Panel de operador.** El proyecto debe constar de un panel de operador constituido por tantas pantallas como sean necesarias para que este permita realizar tanto el control como la monitorización del proceso. También se considera óptimo tener una pantalla de inicio que presente la máquina y la opción sencilla para el paso de una pantalla a otra.

6. DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA

El proyecto trata de la creación del Gemelo Digital de una máquina Flowpack haciendo uso del software de simulación 3D de la empresa Mitsubishi Electric. La máquina Flowpack es, básicamente, una máquina empaquetadora de alimentos. El empaquetado se realiza haciendo uso de bobinas de papel plástico y un cuchillo rotatorio. Existe una entrada de alimentos a través de una cinta transportadora y el papel plástico se va desenrollando de manera que va cubriendo los alimentos. El cuchillo rotatorio suele ser de sellado térmico que permite ir separando los alimentos en paquetes. La salida de los alimentos se realiza mediante otra cinta transportadora.



Figura 4: Ejemplo de máquina empaquetadora Flowpack

La solución propuesta se articula en torno a tres ejes: el Gemelo digital, la programación del autómatas y la programación del panel de operador.

La realización del Gemelo Digital requiere de un diseño previo de la máquina a simular en un entorno CAD. El diseño CAD completo se importa en el entorno de simulación 3D "MELSOFT Gemini", esto habrá que hacerlo teniendo ciertas ideas en cuenta para poder trabajar con los componentes de manera correcta dentro de dicho entorno. Una vez introducido el modelo completo, éste se modela para que tenga la cinemática requerida en simulación. Como parte fundamental del modelado, se identificarán las señales de entrada y salida necesarias para el control de la máquina.

Por otro lado, teniendo en cuenta las señales de entrada y salida comentadas y pensando en el funcionamiento de la máquina, se pasa al diseño del programa de control del sistema. En este caso, será necesario estudiar las opciones de conectividad que ofrecen tanto *Gemini* como el simulador del sistema de control a utilizar, ya que la puesta en marcha se realiza con *SiL*. Como ya se ha indicado anteriormente, la simulación *SiL* implica que tanto el autómatas elegido como la tarjeta *Motion* necesaria para el control sincronizado de los ejes de la máquina serán instancias virtuales. El primer paso para el diseño de programa de control consiste en la configuración del sistema en el entorno de programación "GX Works3", introduciendo el PLC y la tarjeta *Motion* correspondiente y parametrizando los tres ejes necesarios para el control de máquina. El primer eje está relacionado con la primera cinta transportadora y será el eje maestro de la flowpack. El segundo eje es el que controla el movimiento del cuchillo rotatorio. El tercer eje es el que controla el movimiento de la bobina de papel. Estos dos últimos son esclavos del primero y su

movimiento va sincronizado al de este. Una vez configurado el sistema de control, se pasa a su programación con el fin de definir la secuencia diseñada con anterioridad.

Por último, el diseño y la programación de la interfaz máquina-humano o HMI se realiza pensando en ésta como el panel de operador que permite comandar y monitorizar la máquina. Se escoge un modelo de pantalla HMI y se realiza el diseño pensando en las operaciones e información necesarias por parte del personal de planta en lo que a control de la máquina se refiere. Una vez se establece el funcionamiento necesario, se pasa a realizar el mapeo entre las señales de entrada y salida del HMI y las del autómatas programable.

7. DISEÑO

En este apartado se trata el diseño del proyecto en profundidad y se comentan todos los pasos a seguir y todos los aspectos importantes a tener en cuenta en caso de que se desee reproducir o profundizar el trabajo realizado. El diseño del proyecto se realiza pensando en los tres ejes mencionados anteriormente: el Gemelo digital, la programación del autómatas y la programación del panel de operador. Se tiene en cuenta cada eje de manera individual y la conexión entre ellos.

7.1. Servicios

La máquina Flowpack ofrece un único servicio de empaquetado de alimentos en papel plástico.

7.2. Gemelo Digital

Como se observa en la figura 5 es posible separar el Gemelo Digital en dos conjuntos a modelar: el escenario y las máquinas. El conjunto "Escenario" está compuesto por todos los elementos del entorno que no requieren ninguna propiedad cinemática o dinámica. En este caso, se considera escenario a las vallas de seguridad que delimitan el perímetro de la estación.

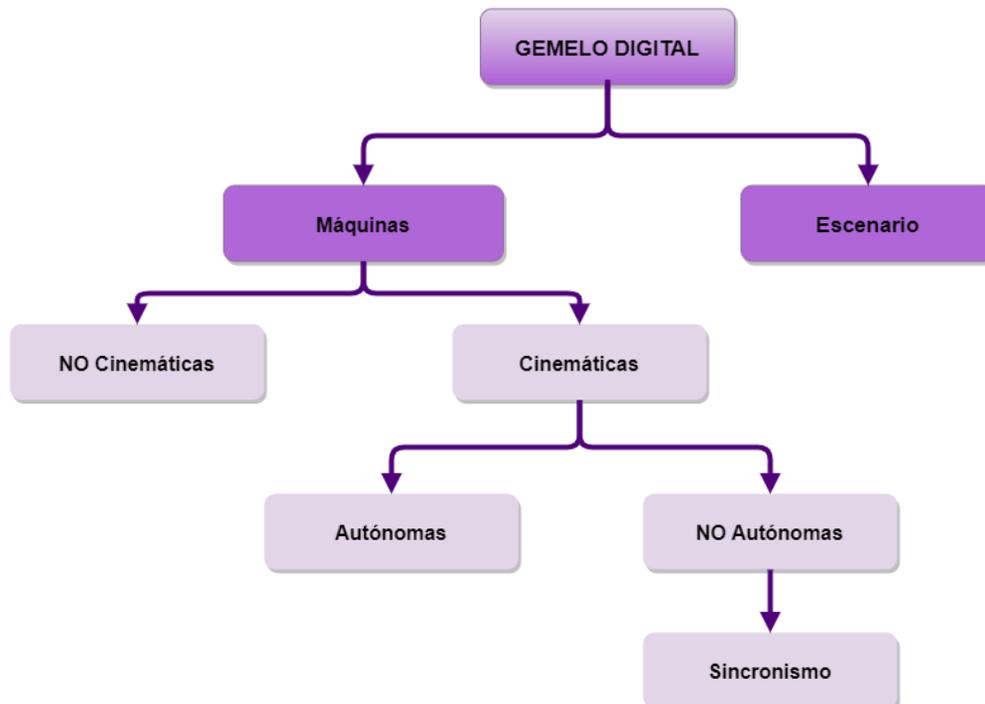


Figura 5: Diagrama de conjuntos

Por otro lado, el conjunto “Máquinas” a su vez está dividido en dos subconjuntos, donde la parte de “No cinemáticas” está compuesta por toda la estructura exterior de la máquina que no realiza ninguna operación. El subconjunto “cinemáticas” está compuesto por todos los elementos de la estación que realizan una operación, sin embargo, estos pueden subdividirse a su vez en “Autónomos” y “No autónomos”. Los elementos “No autónomos” son aquellos que dependen del movimiento de otros para poder realizar su movimiento; es decir, en este caso, los ejes esclavos que siguen el movimiento del eje maestro. Los elementos “Autónomos” son aquellos que realizan sus movimientos de manera individual.

El modelado del Gemelo Digital de la estación se realiza con la herramienta de simulación 3D de Mitsubishi Electric “MELSOFT Gemini” y permite la conexión con el resto de las herramientas software del paquete MELSOFT.

En primer lugar, se comenta el funcionamiento del entorno *Gemini*. A continuación, se pasa a hablar del modelo CAD a diseñar. El siguiente punto es el modelado cinemático que se realiza sobre este modelo CAD. Por otro lado, se trata la validación del Gemelo Digital creado. Y, por último, se establecen las señales de entrada y salida de este gemelo.

7.2.1. MELSOFT Gemini

En este apartado se describen los conceptos básicos necesarios para comprender el modelado del Gemelo Digital en el entorno de “MELSOFT Gemini” (Figura 6).



Figura 6: 3D Simulator MELSOFT Gemini

Dentro de la estación existen, básicamente, dos tipos de movimientos a modelar: movimientos lineales y movimientos giratorios. Los movimientos lineales se encargan de transportar las barras de caramelo del punto de entrada hasta el punto de salida. Por su parte, los movimientos giratorios están relacionados con los ejes repartidos a lo largo de la estación. El análisis de la estación en torno a estos dos tipos de movimientos permite separar las funciones y elementos del entorno en distintos apartados. A continuación, se comentan los elementos necesarios para el modelado de la estación de este proyecto.

7.2.1.1. Librería de elementos

Un aspecto bastante útil y cómodo del entorno de simulación “Gemini” es la librería de elementos que pone a disposición del usuario. Esta librería de elementos se conoce con el nombre de “eCatalogue” (Figura 7). En esta se pueden encontrar diversos elementos que pueden introducirse en el espacio de trabajo para así poder realizar ciertas funcionalidades sin necesidad de realizar ningún modelado extra. Más adelante se comentarán ciertos elementos que se introducen al espacio de trabajo, por ejemplo, el generador de producto que se encuentra en la entrada de la estación. Esta colección de elementos accesibles hace que el modelado de cualquier estación sea más rápido y sencillo que si fuese necesario tener que importar todos los elementos de una herramienta CAD externa y modelarlos en *Gemini*.

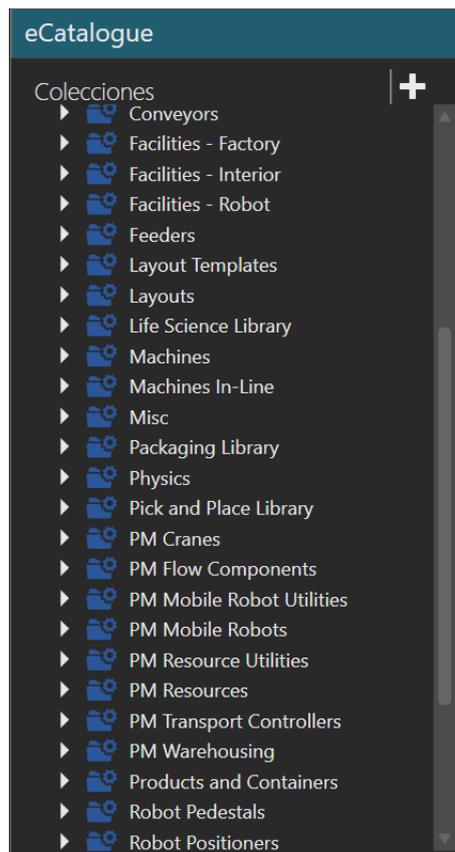


Figura 7: Librería "eCatalogue"

7.2.1.2. Asistente

Para modelar los movimientos lineales es necesario entender el funcionamiento del asistente de modelado que ofrece el entorno. Este asistente permite generar diversos componentes, por ejemplo, un efector terminal, un posicionador o incluso un transportador (Figura 8). El asistente se encarga de crear las propiedades y comportamientos necesarios para el modelado y control del componente que se escoja.

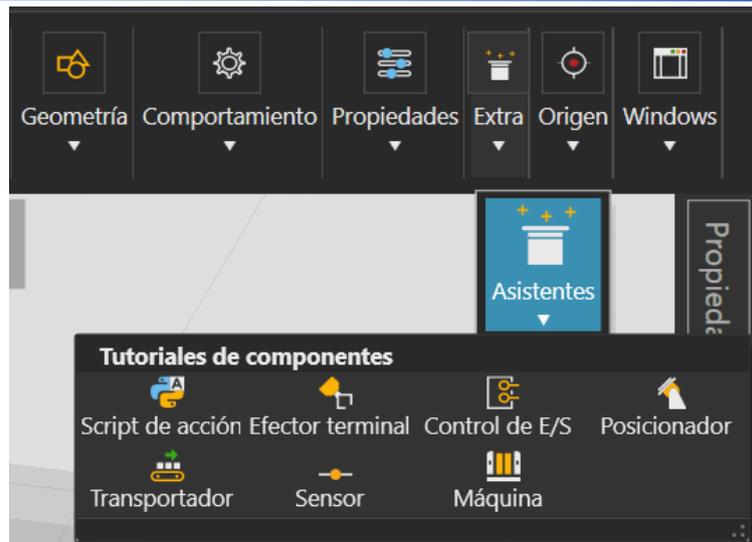


Figura 8: Asistente de componentes

7.2.1.3. Transportador

Para los movimientos lineales necesarios en la estación, se escogerá el componente “Transportador” y se trabajará con sus propiedades y comportamientos. En el caso del transportador únicamente se generan comportamientos, ya que sólo éstos son necesarios para poder transportar elementos. Los comportamientos básicos necesarios aparecen en la figura 9:

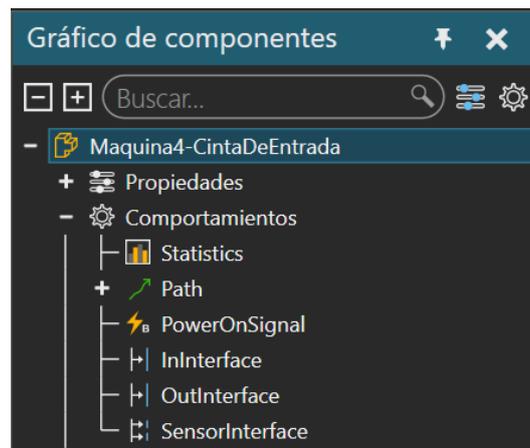


Figura 9: Comportamientos referentes a un Transportador

Entre los diferentes comportamientos generados, cabe destacar la importancia del *path* o ruta. En el caso del “Transportador” se genera un “One Way Path”, es decir, una ruta unidireccional. Este tipo de ruta permite contener y mover elementos en un sentido a lo largo de una ruta definida por puntos establecidos mediante ejes de coordenadas auxiliares (Figura 10).

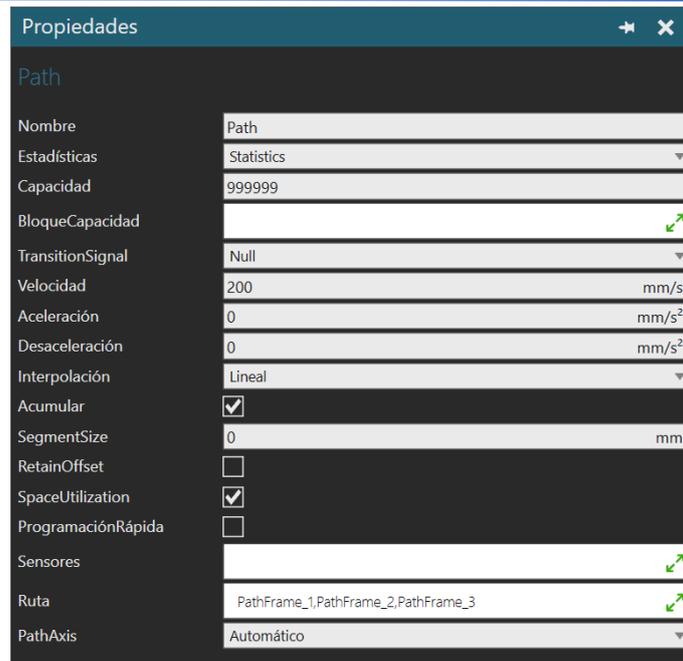


Figura 10: Propiedades de un "path" o ruta

Dentro de las propiedades del comportamiento ruta (Figura 10), es importante establecer la capacidad de la ruta, ya que de ésta depende cuántos elementos puede haber a la vez en la cinta. Otra propiedad clave es la velocidad a la que se transportan los elementos a lo largo de la ruta. En la propiedad "Ruta" es donde se establecen los puntos que generan la trayectoria a seguir por los elementos transportados. Su posición predeterminada genera una trayectoria lineal establecida por tres puntos posicionados en la superficie superior del transportador. Dentro de esta propiedad "Ruta" es importante el orden en el que se introducen dichos puntos, ya que éste establece la trayectoria total de los elementos transportados. Por ejemplo, si se mantiene el orden predeterminado, el elemento se moverá del punto 1 al 2 y del punto 2 al 3 (Figura 11). Sin embargo, si se alterase el orden de los puntos, de manera que en la lista aparezca "PathFrame_3" antes que "PathFrame_2" el elemento tendría otra trayectoria completamente distinta (Figura 12).

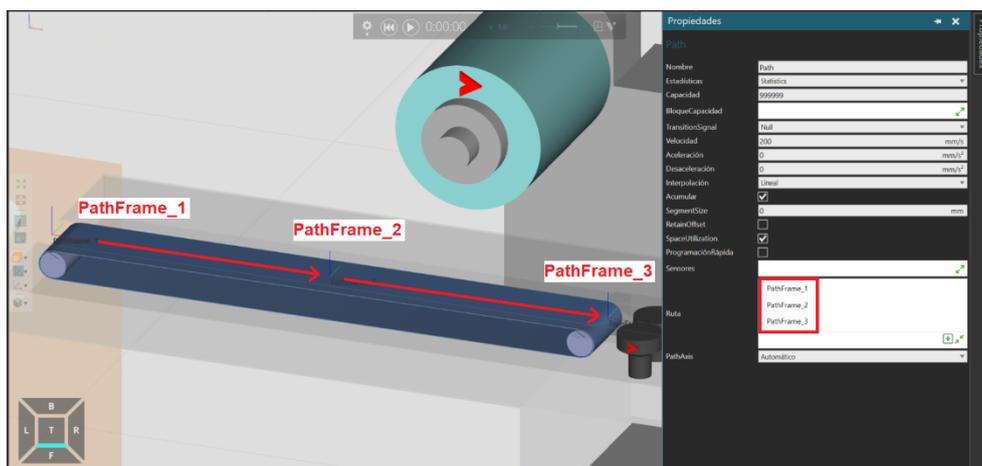


Figura 11: Path con orden original

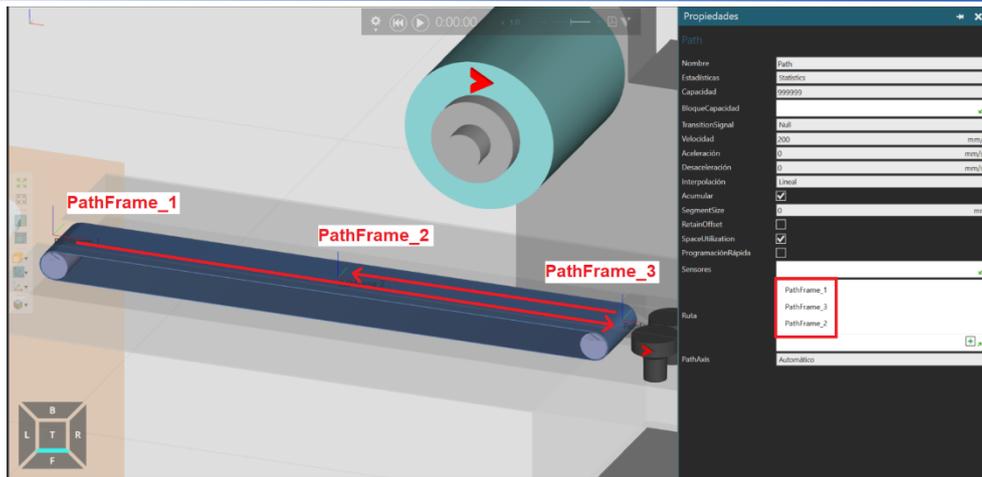


Figura 12: Path con orden modificado

La posición de estos puntos en el espacio es modificable mediante la posición de sus ejes de coordenadas respecto del sistema de coordenadas “Mundo”, “Padre” y “Objeto” (Figura 13), lo cual permite definir perfectamente la trayectoria de los elementos según las necesidades de cada componente transportador.

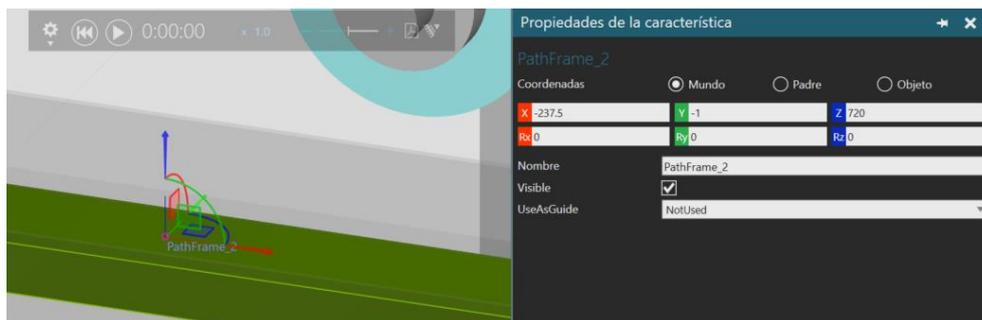


Figura 13: Coordenada “Pathframe”

Un transportador puede tener comportamientos adicionales a los básicos generados mediante el asistente. Concretamente, tal y como se muestra en la Figura 14, a las propiedades y los comportamientos que existen en el momento de generación del componente se le pueden sumar todos los que sean necesarios para tener el control deseado del componente.

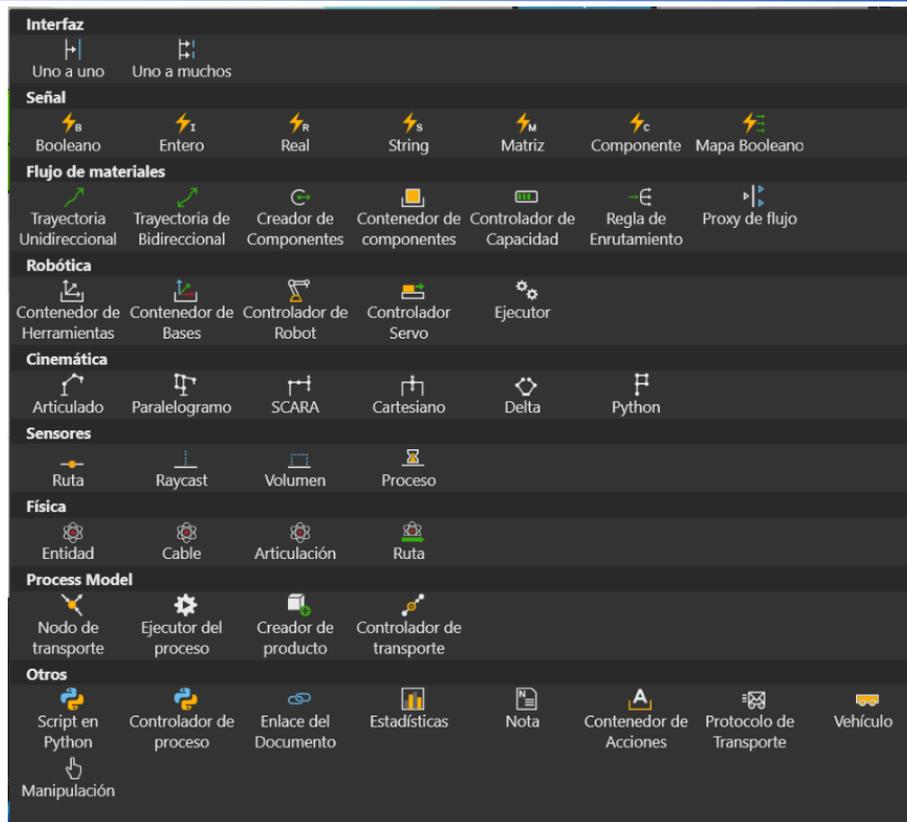


Figura 14:Comportamientos disponibles

7.2.1.4. Script de Python

El “Script de Python” es un tipo de comportamiento que puede resultar muy útil a la hora de gestionar las características de un componente y modelar su comportamiento. Se trata de un intérprete de código Python 2.7 en el cual se puede ejecutar secuencias de código Python en el entorno de simulación (Figura 15). El aspecto más interesante dentro del diseño de esta estación es el hecho de que se puede acceder y modificar tanto las propiedades como los comportamientos del componente en el que se genere el script. Por ejemplo, es posible activar y desactivar la ruta asociada al componente de manera que cumpliéndose las condiciones que se establezcan se pueda dejar de transportar elementos. Otra opción muy útil es poder acceder al valor de las señales que se pueden introducir, siendo estas señales de valor booleano, entero, real, etcétera, como puede verse en la Figura 14. Y, por último, en lo que respecta a esta estación es necesario comentar que es posible acceder y modificar los valores de las distintas propiedades que se introducen en el componente.

```

Conveyor #1::ConfigScript
1 from vcScript import *
2 import vcMatrix
3
4 comp = getComponent()
5 app = getApplication()
6
7 finalized = False
8 userJogging = False # indicates if user is currently hovering over the comp
9
10 def OnFinalize():
11     global finalized
12     finalized = True #flag to differentiate connection events from PnP with events ri
13
14
15 # Try to take the props from the preceding conveyor
16 def unifyProps(otheriface, offset):
17     '''
18     This function is called when component is connected to another component
19     '''
20     if comp in app.SelectionManager.getSelection(Vc_Selection_Component):
21         proplist = [
22             "Presets",
23             "ConveyorHeight",
24             "ConveyorWidth",
25             "ConveyorSpeed",
26             "Advanced::RollerRadius",
27             "Advanced::RollerSpacing",
28             "Advanced::BeltRollers",
29             "Advanced::GuideRailHeight",
30             "Advanced::GuideRailHeightOffset",

```

Figura 15: Ejemplo de "Script de Python"

7.2.1.5. Propiedad (valor real)

El tipo de propiedad más significativo para el modelado de esta estación es la propiedad real (Figura 16). Esto se debe a que, hablando en términos de conectividad, desde el simulador es posible conectar las propiedades de un componente a una señal externa. Esto es muy útil si se tiene en cuenta que es posible modificar el valor de las propiedades de un componente desde un script en Python. Además, dentro de dicho script es posible asociar el valor de una propiedad introducida por el usuario con un comportamiento intrínseco de un componente que de otra manera no sería modificable. Este es un punto clave en el modelado de la estación.

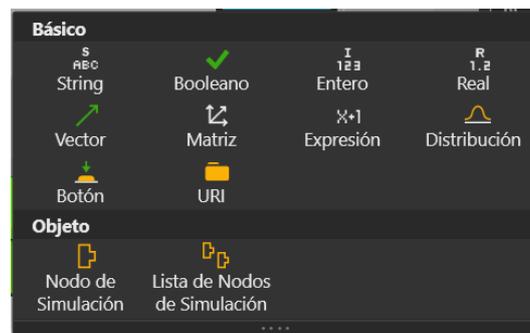


Figura 16: Propiedades disponibles

7.2.1.6. Conveyor sensor

Este es un elemento asociado a un elemento transportador (Figura 17), es decir, primeramente, es necesario ya sea modelar un transportador siguiendo la información anterior o introducir una cinta conveyor de las disponibles en el "eCatalogue".

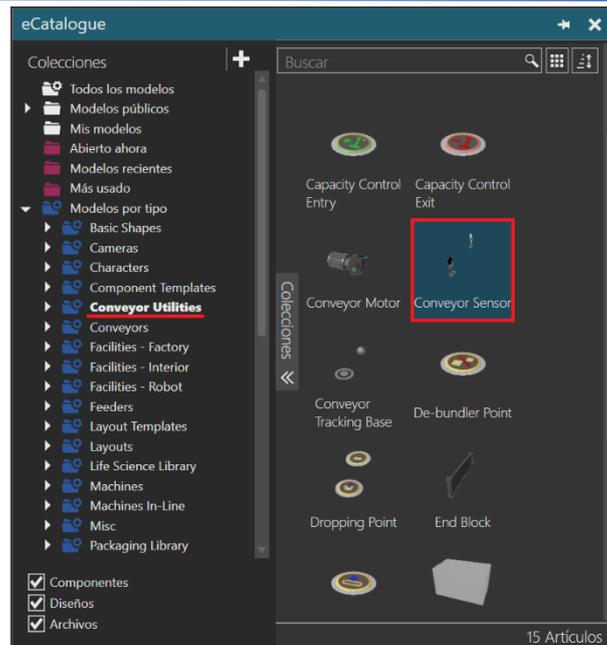


Figura 17: Sensor dentro de la librería "Conveyor Utilities"

Una vez acoplado a un transportador, el sensor detecta el paso de un elemento, moviéndose por dicho transportador, al pasar entre sus dos componentes. Dentro de las propiedades y comportamientos que se generan al introducir este tipo de sensor, para este proyecto solo habrá que centrarse en las más básicas, ya que como puede verse en las Figuras 18 y 19, las funcionalidades del elemento son extensas.

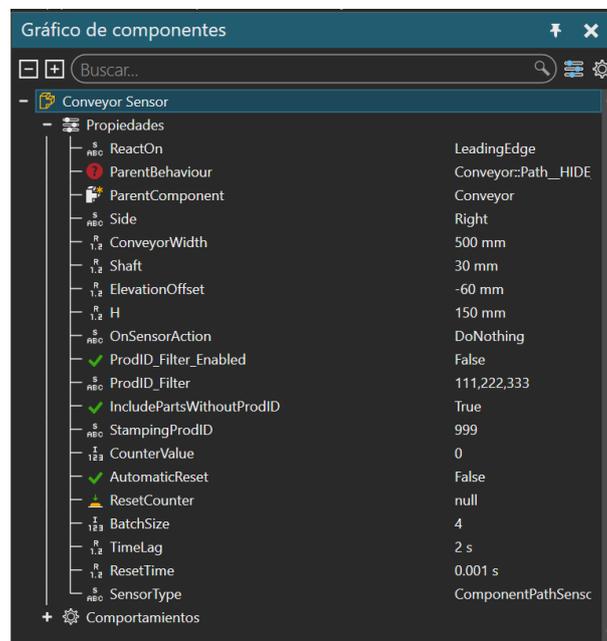


Figura 18: Propiedades del sensor

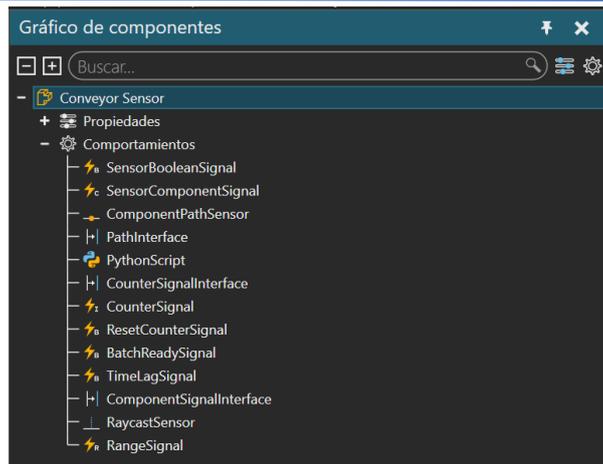


Figura 19: Comportamientos del sensor

Las propiedades por destacar son las correspondientes a las dimensiones del sensor, en concreto, su ancho. Esta propiedad permite ajustar el sensor al ancho del transportador en el que se acople. Este ajuste se realizará automáticamente al acoplarlo a una cinta de las disponibles en la librería. No obstante, si se desea acoplar a un transportador modelado mediante el “Asistente”, en tal caso es necesario realizar este ajuste manualmente.

Por otro lado, el comportamiento a destacar en este proyecto es la señal booleana de “SensorBooleanSignal” que es la que se activa y desactiva cuando un elemento pasa entre los dos componentes del sensor. El valor de la señal es TRUE cuando el elemento se encuentra entre ambos componentes y pasa a FALSE cuando el elemento sigue por el transportador.

7.2.1.7. Articulación

No existe una opción que genere los comportamientos asociados a los movimientos angulares directamente. Es necesario introducir una “Articulación” lo cual genera una nueva articulación secundaria para el nodo seleccionado. Estas articulaciones pueden ser de diversos tipos: fija, rotativa, traslacional, seguidor rotacional, seguidor traslacional o personalizado (Figura 20).

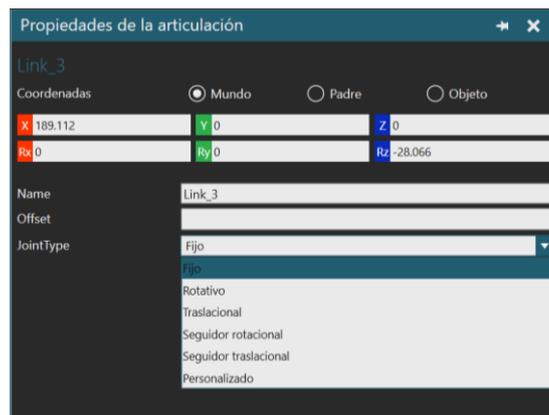


Figura 20: Tipos de articulación

En el caso de movimiento angular es necesario introducir una articulación rotativa. Una vez introducida es importante establecer su posición en el espacio, además de establecer el eje de rotación. Por otro lado, para que más adelante sea posible controlar el movimiento de esta articulación, es necesario asignarle un controlador. También es importante destacar las propiedades de “valor inicial” y “nombre” de la articulación, en el ejemplo “J3” (Figura 21). El

nombre de la articulación permite realizar un control por posición de cada una de las articulaciones ya que, es posible realizar una conexión directa con una variable externa.

Propiedades de la articulación		
Link_3		
Coordenadas <input checked="" type="radio"/> Mundo <input type="radio"/> Padre <input type="radio"/> Objeto		
X	189.112	Y 0
Z	0	
Rx	0	Ry 0
Rz	-28.066	
Name	Link_3	
Offset		
JointType	Rotativo	
Eje	+Z	
Propiedades de articulaciones		
Name	J3	
Controller	ServoController	
Valor inicial	0	
J3	0	
Expresión de valor	VALUE	
Límite Min.	-180,0	
Límite Máx.	180,0	
Velocidad Máx.	100	/s
Aceleración Máx.	500	/s ²
Desaceleración Máx.	500	/s ²
Tiempo de Latencia	0	s
Tiempo de Asentamiento	0	s

Figura 21: Propiedades de una articulación rotativa

7.2.1.8. Feeder

Un elemento clave en la estación es aquel que genera los elementos que se desean procesar o empaquetar. Como se desea poder generar un elemento específico y no una forma geométrica básica, se introduce un “Feeder” parametrizable en este aspecto (Figura 21). Primero es necesario introducir el elemento que se desea generar en el entorno, lo más sencillo es utilizar algún elemento disponible en la librería del entorno, aunque se podría generar un elemento importado desde un modelo CAD. En este caso, existe una barra de caramelo que puede servir de producto para empaquetar (Figura 22).

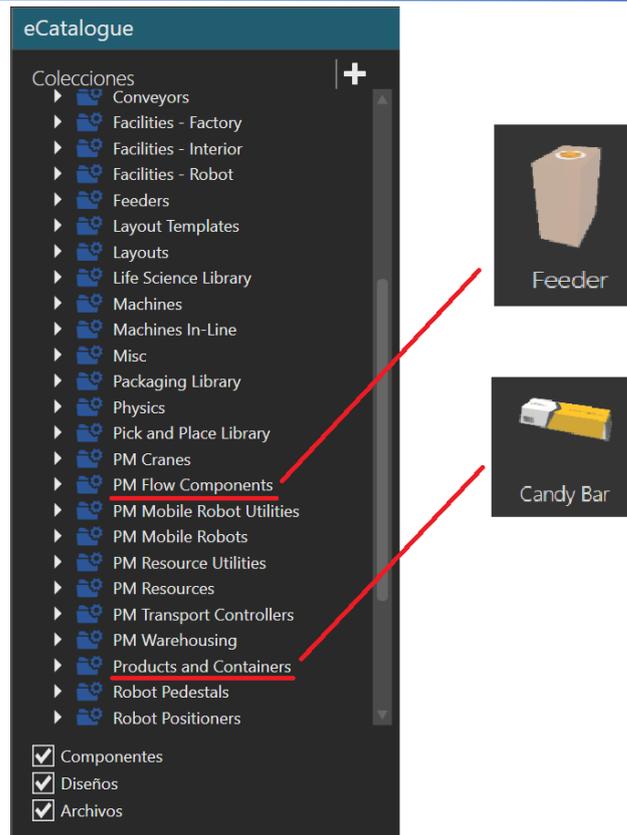


Figura 22: Feeder y barra de caramelo dentro de "eCatalogue"

7.2.2. Modelo CAD

En este apartado se describe el diseño del Gemelo Digital de la máquina Flowpack empaquetadora de alimentos. El primer paso es crear el modelo CAD de la máquina y será necesario realizarlo en un entorno externo, ya que *Gemini* no posee ninguna funcionalidad que permita crear modelos 3D de cero. Para comenzar, se parte de la base de que para ver el funcionamiento de la máquina y saber que se trata de una máquina Flowpack es suficiente con disponer de un modelo 3D simplificado de esta como el dispuesto en la Figura 23.

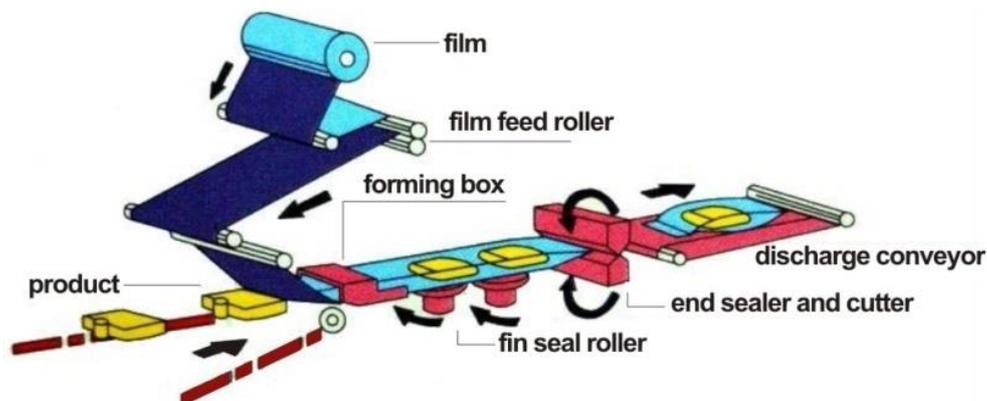


Figura 23: Esquema de una máquina Flowpack

Teniendo en cuenta que el funcionamiento de la máquina se puede desglosar en tres ejes, este será el punto de partida para el diseño 3D. El primer eje es el que controla la cinta transportadora que lleva los alimentos y los cilindros que sellan el paquete en la parte inferior. El segundo eje controla el movimiento rotatorio del cuchillo. Y, por último, el tercero controla el desbobinado del papel plástico que envuelve el producto.

Se realiza un diseño preliminar (Figura 24) en el que sólo se introduce una cinta transportadora simple, los cilindros de sellado, la bobina de papel y el cuchillo rotatorio. Con este ejemplo como base se diseñan las dimensiones de la máquina de manera que quede a escala 1:1; más adelante, esto será útil para realizar la programación del sincronismo.

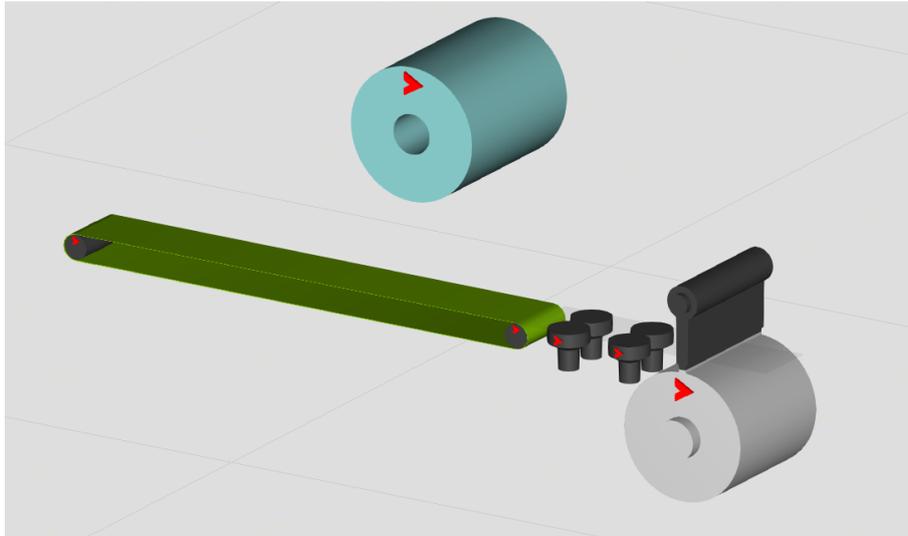


Figura 24: Máquina Flowpack (diseño simplificado)

Una vez claras las dimensiones base, se pasa a crear un escenario algo más completo que permita seguir el movimiento de los ejes. Para no perder la visión del objetivo principal de la máquina, que es el movimiento sincronizado de los ejes, se decide crear el cuerpo de la máquina con un material traslúcido. Por otro lado, se introduce una cinta transportadora de salida que será la que lleve el producto empaquetado fuera de la máquina. La versión final del proceso se muestra en la Figura 25.

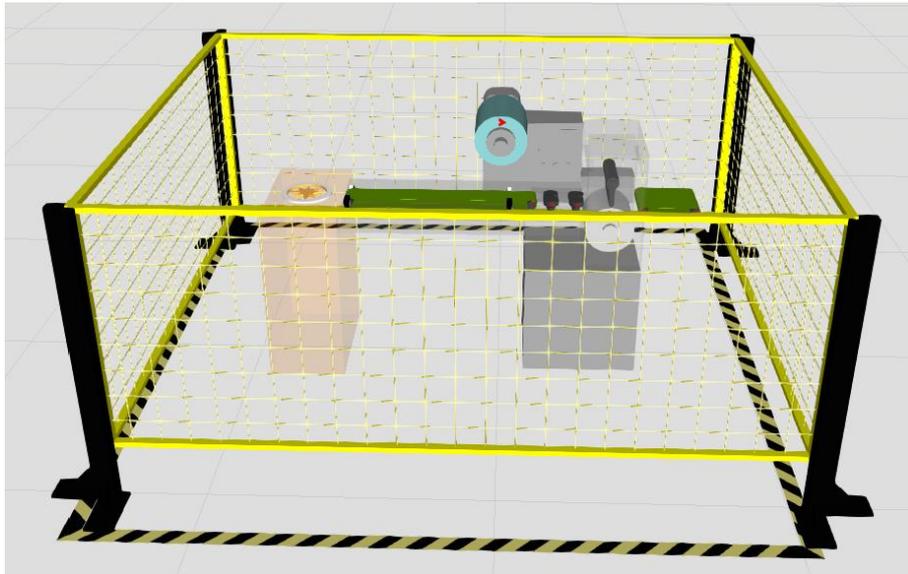


Figura 25: Máquina Flowpack (diseño completo)

Los términos que se utilizaran a partir de este punto para dirigirse a los diversos componentes de la máquina empaquetadora son los establecidos en la Figura 26.

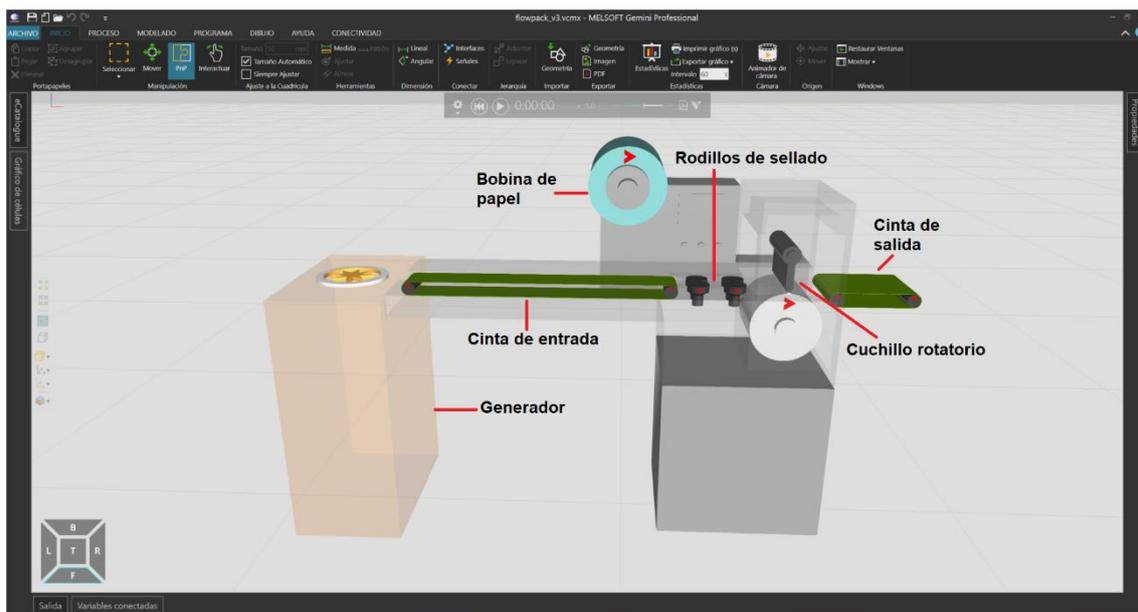


Figura 26: Componentes de la máquina empaquetadora

Como se comentaba anteriormente, es necesario tener en cuenta un par de aspectos al importar el CAD creado en el entorno de *Gemini*:

- Primero, es necesario exportar los archivos de manera individual para poder trabajar sobre ellos más adelante. Esto hace que también haya que tener en cuenta las cotas relativas entre elementos en la máquina final. Esto se debe a que al importar los archivos de manera individual cada uno de ellos aparecerá en el origen de la cota mundo del entorno, por lo tanto, será necesario posicionar cada uno de los elementos. Se ha considerado que lo más sencillo es escoger un elemento como principal y colocar el resto de los elementos según sus cotas relativas a este.

- Segundo, el único elemento necesario que se introduce directamente como nuevo en el entorno Gemini es el generador de elementos que se posiciona en la entrada de la cinta inicial. En este caso, se desea generar algún tipo de material alimenticio con el fin de que la simulación sea lo más parecido a la realidad posible. Para ello, una vez introducido el generador en el entorno y posicionado en el lugar deseado, hay que introducir en el entorno el elemento deseado, a saber: una barra de golosina de la que se dispone en la librería de *Gemini*.

Tanto en el caso del generador como en el de la barra de golosina es necesario realizar un redimensionamiento con el fin de que ambos se ciñan a la escala en la que se está trabajando. El generador al introducirlo genera un elemento genérico, por lo que es necesario parametrizarlo de manera que genere el elemento seleccionado (Figura 27).

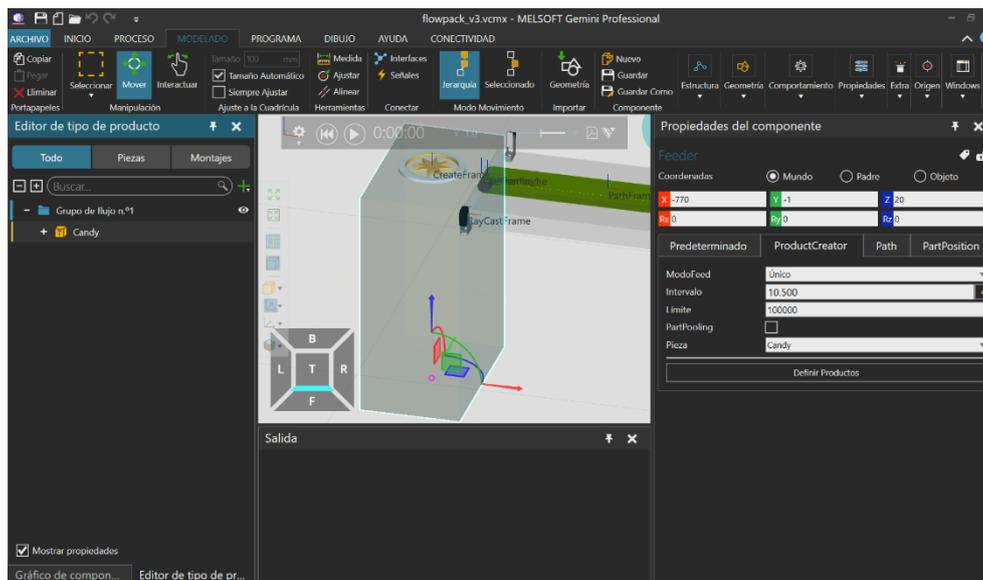


Figura 27: Parametrizado del "Feeder"

Con esto, se tiene el modelo CAD de la máquina listo para ser modelado más allá de la parte mecánica.

7.2.3. Modelo cinemático

Con el modelo listo se pasa a realizar el modelado cinemático, es decir, darle a cada elemento la capacidad de movimiento que necesita. Con el fin de que sea visualmente claro el movimiento sincronizado entre los tres ejes, se desea hacer girar los rodillos de ambas cintas con la velocidad correspondiente.

7.2.3.1. Cinta de entrada

Primero, es necesario dotar a la cinta de las características necesarias que permitan que otros elementos se puedan transportar sobre esta. Esto se consigue mediante el asistente del que dispone *Gemini*. Sin embargo, habrá que tener en cuenta un detalle al crear una superficie de transporte haciendo uso del asistente: si se selecciona toda la cinta, incluyendo los rodillos, posteriormente no será posible dar a cada rodillo el movimiento alrededor de su propio eje. Por lo tanto, el primer paso será separar la cinta en distintos componentes, de manera que se tengan tres elementos independientes: la superficie de transporte y cada rodillo por separado. Una vez

separados, se indica con el asistente que la superficie será un transportador y, por tanto, aparecerán las características necesarias para su control (Figura 28).

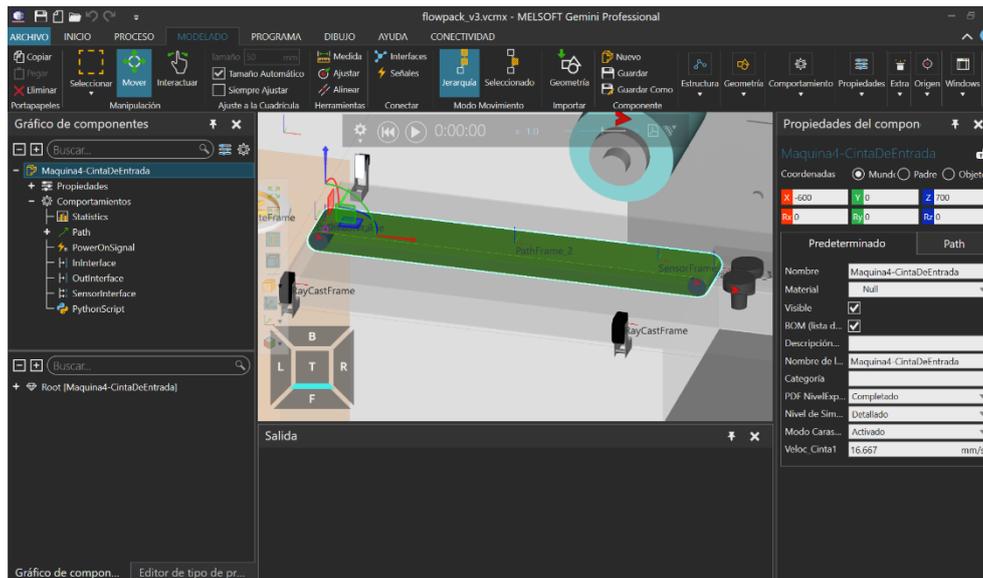
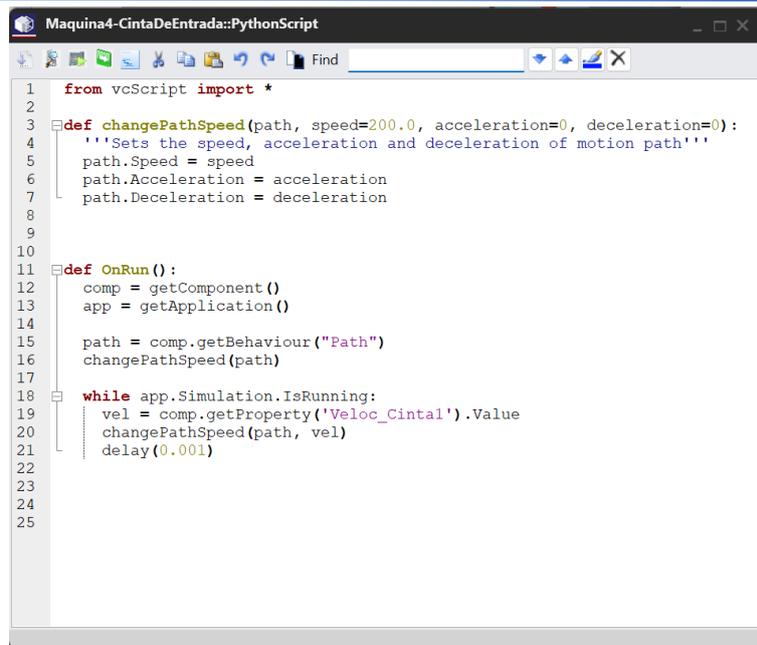


Figura 28: Cinta de entrada, características de Transportador

La variable de la cinta a controlar es la velocidad, lo cual es posible realizar dentro del entorno de Gemini manualmente a través de la función “Velocidad” del *path* que se genera con el asistente. Sin embargo, en el momento de realizar el mapeado de las señales entre simulador y PLC, se puede ver que no es posible mapear esta característica con las variables del autómat. Esto crea la necesidad de generar una variable externa que apunte a la característica intrínseca de la superficie de transporte. Para ello, se crea una variable con el nombre “Veloc_Cinta1” y mediante un script de Python (Figura 29) se relaciona dicha variable con la característica “Speed” del *path* correspondiente a esta cinta. De esta manera, el valor que se le asigne a la variable creada pasa a ser la velocidad que llevan los productos cuando se transportan a lo largo de la cinta de entrada.



```

1  from vcScript import *
2
3  def changePathSpeed(path, speed=200.0, acceleration=0, deceleration=0):
4      '''Sets the speed, acceleration and deceleration of motion path'''
5      path.Speed = speed
6      path.Acceleration = acceleration
7      path.Deceleration = deceleration
8
9
10
11 def OnRun():
12     comp = GetComponent()
13     app = getApplication()
14
15     path = comp.getBehaviour("Path")
16     changePathSpeed(path)
17
18     while app.Simulation.IsRunning:
19         vel = comp.getProperty('Veloc_Cintal').Value
20         changePathSpeed(path, vel)
21         delay(0.001)
22
23
24
25

```

Figura 29: Script de Python, asignación de velocidad del Transportador

Por otro lado, para cada uno de los rodillos de la cinta hay que generar una articulación que permita la rotación. En este caso se crea una articulación rotativa alrededor del eje Y positivo, de manera que los ejes giren en sentido horario. Para esto, se posicionan los ejes de coordenadas de la articulación en el origen de la base del cilindro. Al crear la articulación, se generan características asociadas, como su valor. En este caso, al tratarse de una articulación rotativa, el valor de esta se da en grados angulares. Esta característica es la que se mapea con una variable del PLC y es la base del control del rodillo.

7.2.3.2. Rodillos de sellado

Los rodillos de sellado tienen un modelado similar al descrito anteriormente con la diferencia de que para estos la rotación es alrededor del eje Z y los ejes de coordenadas de las articulaciones se posicionan sobre la tapa superior de cada uno de ellos.

Cabe destacar que para los dos rodillos más próximos al frente de la máquina se establece el eje Z positivo como eje de giro (sentido antihorario) mientras que para los dos rodillos restantes se establece el sentido de giro contrario. De esta manera se consigue el efecto de sellado del paquete por la parte inferior de este.

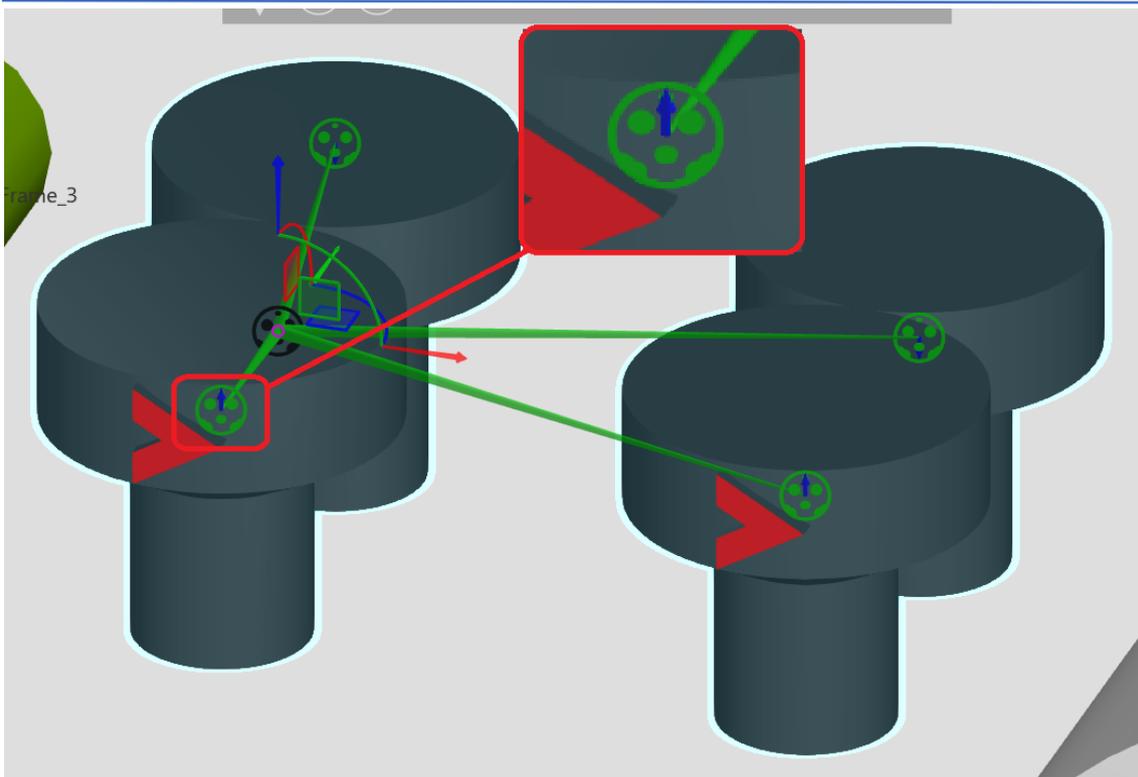


Figura 30: Detalle de los sentidos de giro de los rodillos de sellado

7.2.3.3. Cuchillo rotatorio

El cuchillo lleva a cabo la separación del producto en paquetes. Este cuchillo básicamente se comporta como 2 ejes, por lo que se generan un par de articulaciones rotativas para estos. El cilindro de la parte superior se modela de manera que la rotación sea alrededor del eje Y negativo (sentido antihorario) mientras que el cilindro inferior tendrá una rotación en sentido horario.

7.2.3.4. Bobina de film

Para la bobina se realiza un modelado similar, se establece una articulación con eje de giro en sentido horario.

7.2.3.5. Cinta de salida

Una vez sellado el paquete se encuentra la cinta de salida que lleva los productos fuera de la máquina. Al igual que para la cinta de entrada, esta se separa en tres elementos distintos, la superficie de transporte y los dos rodillos independientes. A la superficie de transporte se le aplican las características necesarias haciendo uso del asistente. Además, es necesario crear una variable de velocidad y un script de Python para poder controlar su velocidad. Se sigue el mismo código que en la cinta de entrada, teniendo en cuenta el nombre que se le asigna esta nueva variable "Veloc_Cinta2".

7.2.3.6. Flujo del producto

Al crear las cintas de transporte haciendo uso del asistente, este crea un *path* o camino que seguirá todo elemento que entre en contacto con la superficie superior de transporte. Por defecto este crea un camino que sigue una línea recta, marcando el recorrido mediante tres puntos en los que se generan ejes de coordenadas auxiliares (Figura 31). En cuanto un elemento se mueve por la cinta y llega al punto final de su *path*, este elemento desaparece. Esto se debe a que, si no hay otro elemento concatenado al final de la cinta, el software asume que es el final del recorrido para el elemento.

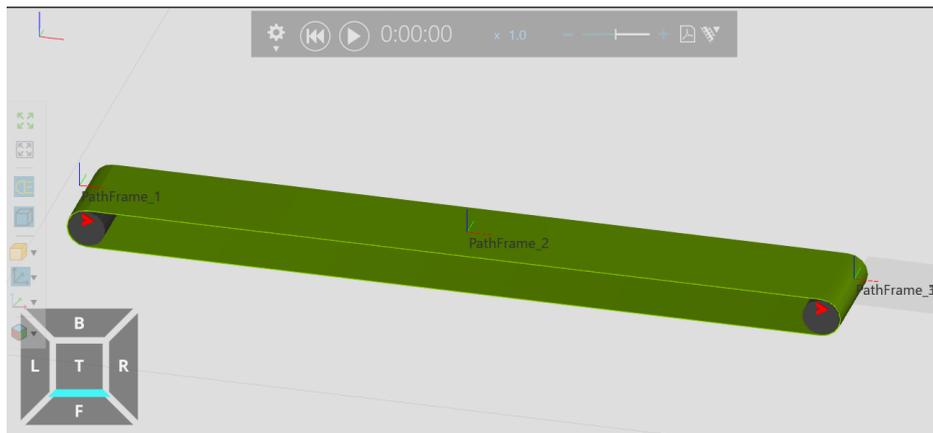


Figura 31: Path de un Transportador

En el generador ocurre lo mismo, este tiene un *path* intrínseco que lleva los elementos que genera desde el punto de aparición hasta un extremo del generador. Este *path* al ser más corto está marcando por dos puntos en los que aparecen ejes de coordenadas auxiliares (Figura 32).

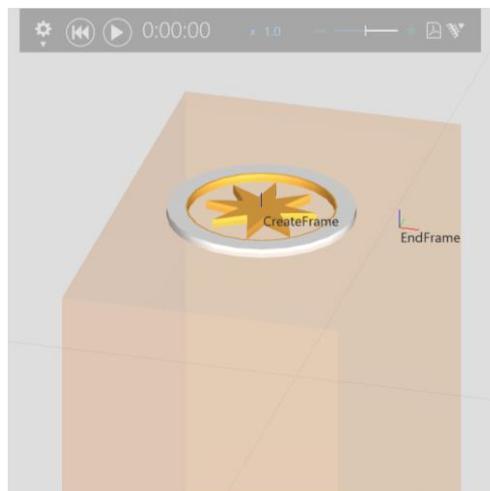


Figura 32: Path de un "Feeder"

En la pestaña de proceso se establece la barra de caramelo como el producto a ser generado por el elemento "Feeder". Por lo que habrá que sustituir el cilindro básico que genera el elemento de manera predeterminada por la barra que se ha introducido en el escenario. Por último, se oculta la barra de caramelo que se utiliza como base para que no sea visible en todo momento durante la simulación.

Al introducir el generador de elementos en el escenario es necesario que no esté conectado al *path* de ningún otro elemento. Una vez asignadas las características de transportador a la cinta de entrada, es posible conectar el generador de elementos con la entrada de la cinta. Básicamente, se concatenan recorridos de manera que cuando la barra de caramelo llegue al final de un recorrido pase al inicio del siguiente sin desaparecer.

7.2.3.7. Superficie de transporte transparente

Teniendo en cuenta la forma de funcionamiento de los recorridos o *paths*, se puede ver la problemática que surge al querer llevar la barra de caramelo de la salida de la cinta de entrada hasta la entrada de la cinta de salida al no haber una superficie que una ambas zonas (Figura 33). Por esta razón, se introduce una superficie transparente entre ambas cintas. Como es de esperar, a esta superficie transparente se le asigna el comportamiento de transportador con el fin de poder concatenar los recorridos de la cinta de entrada, la superficie transparente y la cinta de salida.

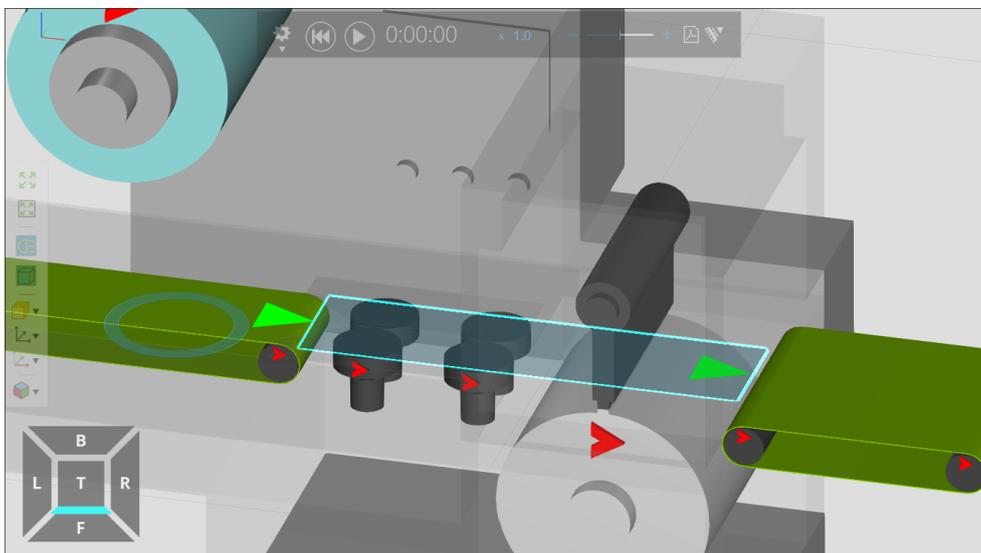


Figura 33: Superficie de transporte transparente

7.2.4. Validación del Gemelo Digital

Una vez terminado el modelo de la estación es recomendable realizar una validación previa al mapeado de señales y la realización del programa de control. En este caso “Gemini” no ofrece un apartado específico para la validación del Gemelo Digital, pero si es posible comprobar que todo funciona de la manera deseada.

Al iniciar la simulación pulsando el “play” en el panel de control de esta, se comprueba que el generador de elementos funciona correctamente, ya que aparece la primera barra de caramelo. Como se ha comentado anteriormente, es posible asignar un valor de velocidad al *path* asociado a la cinta de entrada, por lo que, si se le asigna un valor previamente, una vez iniciada la simulación se podrá ver que la cinta se mueve. Cuando la primera barra de caramelo hace contacto con el primer punto del *path* asociado a la cinta, la barra comienza a moverse a lo largo de la cinta. Esto sucederá con los tres transportadores que se han modelado.

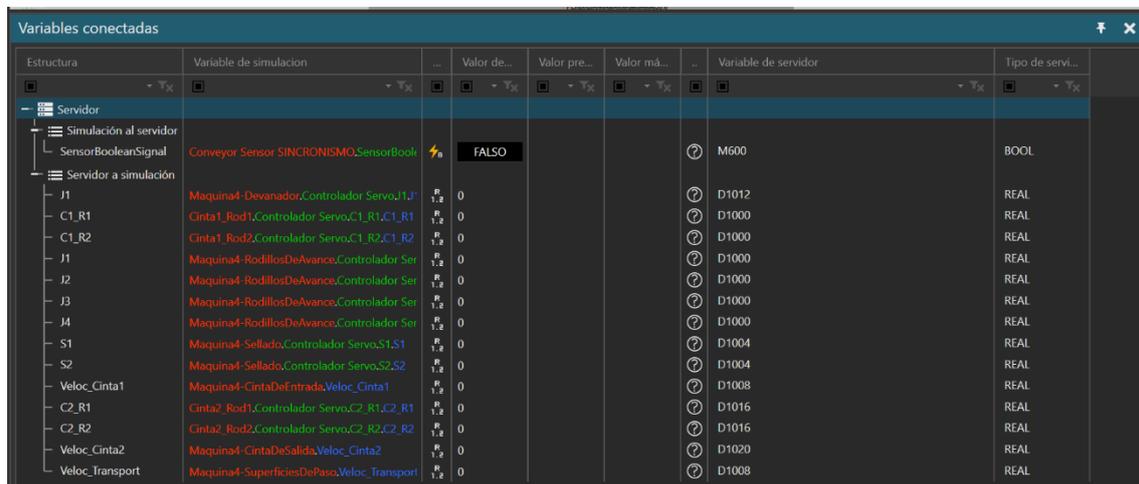
Además, es posible comprobar que los ejes rotativos están modelados correctamente cambiando el valor de cada una de las articulaciones manualmente. Al cambiar el valor de estas con la simulación activa se comprueba que los ejes giran en el sentido deseado.

Por último, se puede comprobar que los sensores acoplados a la cinta de entrada se activan cuando pasa por ellos una barra de caramelo.

7.2.5. Señales E/S

Una vez validado el Gemelo Digital, se realiza el diseño de señales de entrada y salida necesarias para el control de la máquina. Hay que tener en cuenta que las señales que se consideran entrada en el simulador son las señales de salida del controlador y viceversa. En este caso, se desea controlar la velocidad de la cinta de entrada, la posición de los rodillos asociados a dicha cinta, la posición de los rodillos de avance, la posición del cuchillo rotatorio, la posición de la bobina de papel, la velocidad de la cinta de salida y la posición de los rodillos asociados a esta. Además, son necesarios sendos sensores para la activación diversas etapas de la secuencia de trabajo.

Todas las variables relacionadas con la velocidad y la posición de los distintos elementos son consideradas variables de entrada para el simulador y se recogen en la carpeta de “Servidor a simulación” del apartado de Conectividad de *Gemini* (Figura 34). Por otro lado, los sensores de los que se provee la cinta de entrada son señales de salida del simulador y de entrada para el controlador, por lo que se recogen en la carpeta de “Simulación al servidor”.



Estructura	Variable de simulación	Valor de...	Valor pre...	Valor má...	Variable de servidor	Tipo de servi...
Servidor						
Simulación al servidor						
SensorBooleanSignal	Conveyor Sensor SINCRONISMO_SensorBoo...	FALSO			M600	BOOL
Servidor a simulación						
J1	Maquina4-Devanador_Controlador Servo,J1	0			D1012	REAL
C1_R1	Cinta1_Rod1_Controlador Servo,C1_R1,C1_R1	0			D1000	REAL
C1_R2	Cinta1_Rod2_Controlador Servo,C1_R2,C1_R2	0			D1000	REAL
J1	Maquina4-RodillosDeAvance_Controlador Ser...	0			D1000	REAL
J2	Maquina4-RodillosDeAvance_Controlador Ser...	0			D1000	REAL
J3	Maquina4-RodillosDeAvance_Controlador Ser...	0			D1000	REAL
J4	Maquina4-RodillosDeAvance_Controlador Ser...	0			D1000	REAL
S1	Maquina4-Sellado_Controlador Servo,S1,S1	0			D1004	REAL
S2	Maquina4-Sellado_Controlador Servo,S2,S2	0			D1004	REAL
Veloc_Cinta1	Maquina4-CintaDeEntrada_Veloc_Cinta1	0			D1008	REAL
C2_R1	Cinta2_Rod1_Controlador Servo,C2_R1,C2_R1	0			D1016	REAL
C2_R2	Cinta2_Rod2_Controlador Servo,C2_R2,C2_R2	0			D1016	REAL
Veloc_Cinta2	Maquina4-CintaDeSalida_Veloc_Cinta2	0			D1020	REAL
Veloc_Transport	Maquina4-SuperficiesDePaso_Veloc_Transport	0			D1008	REAL

Figura 34: Señales de E/S del servidor

7.3. Programa de control

El siguiente paso una vez realizado y verificado el modelado del Gemelo Digital es el diseño del programa de control. En este apartado se realiza una descripción de su desarrollo haciendo uso de la herramienta de programación de PLC del paquete MELSOFT llamada “GX Works3” y de la herramienta de programación de HMI del paquete llamada “GT Designer3”.

Se comienza con la configuración hardware del sistema. Posteriormente, se pasa a comentar los apartados del programa de control. A continuación, se diseña el mapa de señales entre autómatas y Gemini. Por último, se describe el panel de operador diseñado.

7.3.1. Diseño hardware del sistema

La configuración hardware de la máquina Flowpack está compuesta por un PLC al que se le acopla un módulo de control de movimiento que permite el control por velocidad y posición que es necesario para los ejes de la estación (Figura 35). La CPU escogida es de la serie iQ-F de Mitsubishi Electric, en concreto un FX5-32MR/ES. Por otro lado, el módulo de control de movimiento compatible escogido es el FX5-40SSC-S que permite controlar hasta 4 ejes de forma síncrona.

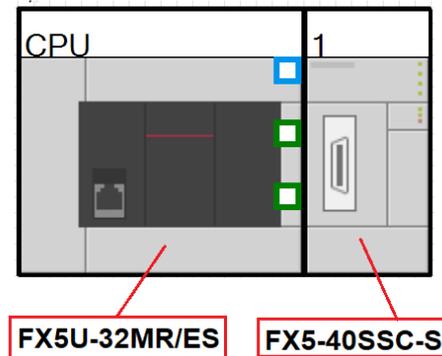


Figura 35: Configuración hardware (controlador)

Entre las funciones de este módulo se encuentra la posibilidad del control síncrono avanzado, concretamente el control de leva. Este control mediante leva es necesario para el control del cuchillo rotatorio que realiza el corte de los paquetes. Estas levas electrónicas se pueden generar de forma automática ingresando ciertos datos en los registros del controlador. Tanto la parametrización y configuración de los ejes como el procesamiento de la leva se realizan desde la misma herramienta de programación “GX Works 3”. Este controlador permite obtener la posición real en cada instante de tiempo de cada uno de los servomotores que dependen de él.

Una vez escogido el controlador se ha de pasar a escoger los servoamplificadores necesarios para realizar la aplicación de máquina empaquetadora. En este caso, se escogen 4 amplificadores iguales de la familia MR-JE, concretamente la serie B (Figura 36). Esta serie es la que permite realizar funciones de sincronismo entre ejes y es compatible con el controlador seleccionado previamente. Al realizarse la puesta en marcha mediante SiL no es necesario tener en cuenta la resolución del encoder ni el tipo de servomotor a seleccionar, ya que serán simulados.

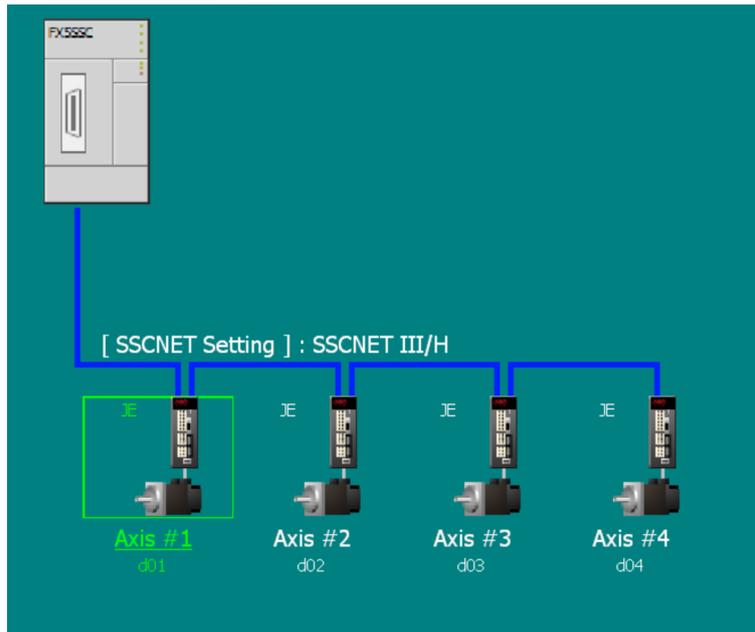


Figura 36: Configuración hardware (red de servoamplificadores)

Por último, se introduce un interfaz máquina-humano (HMI) que se utilizara como panel de operador para la estación. En este caso, se escoge un modelo de la serie “GOT Simple”, concretamente una GS21, ya que es suficiente para la funcionalidad establecida. De esta manera, la configuración hardware completa del proceso es la expuesta en la Figura 37.

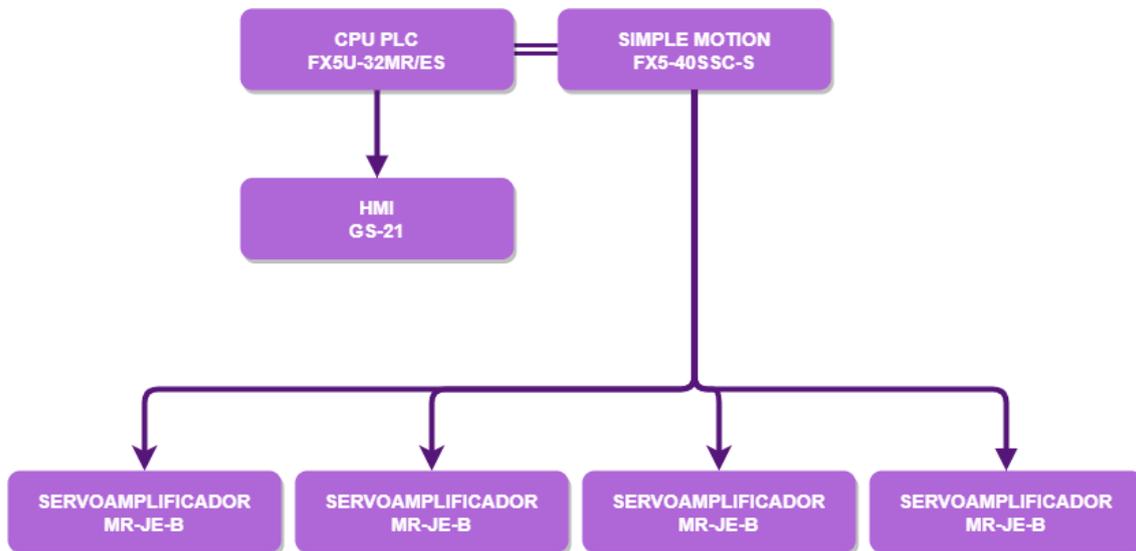


Figura 37: Diagrama hardware completo

7.3.2. Configuración del módulo de Simple Motion

La configuración del módulo se realiza desde un software integrado en el software “GX Works3” llamado “MELSOFT Simple Motion Module Setting”. Este software es necesario para realizar el sincronismo entre dos servomotores. Si tal sincronismo no fuese necesario, sería posible parametrizar los amplificadores desde su display o desde el software específico “MELSOFT MR Configurator2”. La configuración que se aprecia en la Figura 36 se realiza desde este el “MELSOFT Simple Motion Module Setting” donde se parametriza la comunicación SSCNET-III que es la red de control de movimiento por conexión de fibra óptica de Mitsubishi Electric.

Por otro lado, es necesario configurar la mecánica de cada servomotor, esto se lleva a cabo en la pestaña “Compute Basic Parameters” de cada uno de los ejes (Figura 38). En esta pestaña se introduce el tipo de configuración de máquina, el diámetro externo de los rodillos, la relación de reducción en caso de haberla y la resolución del encoder de cada servomotor en unidades de Pulsos/Revolución. Una vez introducidos estos datos, el software calcula automáticamente la relación entre avance por cada revolución y pulsos por revolución y, por tanto, el avance por cada pulso. De esta forma no será necesario introducir el número de pulsos a la hora de realizar un avance, sino que introduciendo el avance deseado en cada eje, el programa automáticamente calcula el número de pulsos que ha de enviar en segundo plano.

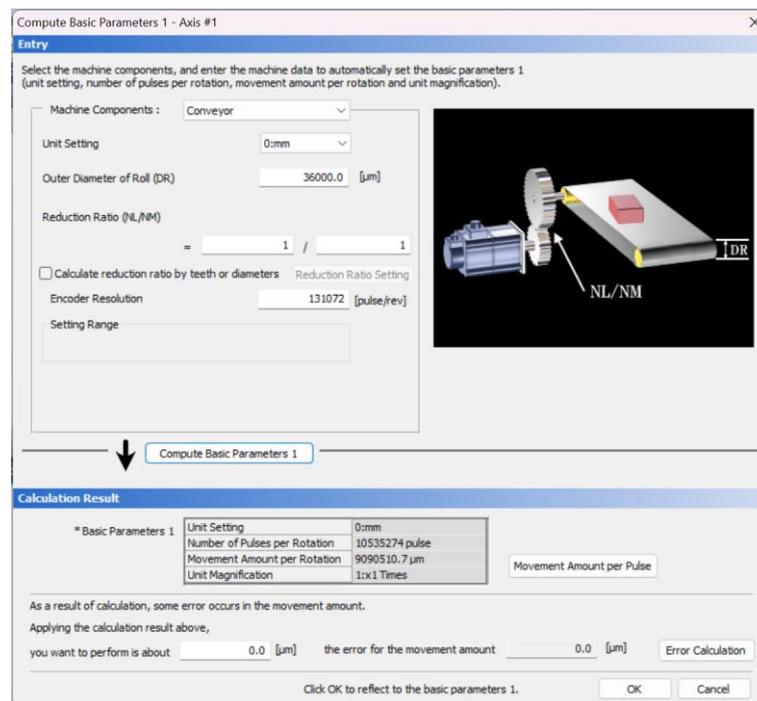


Figura 38: Configuración software de la mecánica asociada al servomotor

Por último, se ha de configurar el sincronismo de los ejes entre sí, comenzando por establecer el eje maestro. En este caso se establece el eje 1 como eje maestro de los ejes restantes, siendo éste el que se asigna a la cinta de entrada. Puede verse en la Figura 39 donde se introduce el valor necesario en el Pr.400 “1: Servo Input Axis”, lo cual se realiza para los tres ejes sincronizados. Esto quiere decir que la cinta de entrada guiará el movimiento del cuchillo rotatorio, del desbobinador y de la cinta de salida. Por tanto, la velocidad de alimentación de las barras de caramelo será la que establezca la funcionalidad del resto de ejes.

Es necesario establecer que el eje 2, es decir, el cuchillo rotatorio, sigue un perfil de leva electrónica. Se le indica al eje que número de leva debe seguir en el parámetro Pr. 440 (Figura 39).

Item	Axis #1	Axis #2
Pr.431:Slippage at auxiliary shaft clutch OFF	0 pulse	0 pulse
Speed change gear		
Pr.434:Speed change gear arrangement	3:After Auxiliary Shaft Composite Gear	3:After Auxiliary Shaft Composite Gear
Pr.435:Speed change gear smoothing time constant	0 ms	0 ms
Speed change ratio		
Pr.436:Numerator	1	1
Pr.437:Denominator	1	1
Output axis		
Cam axis cycle unit		
Pr.438:Unit setting selection	0:Use Unit of Main Input Axis	0:Use Unit of Main Input Axis
Pr.438:Unit	0:mm	0:mm
Pr.438:Number of decimal places	0	0
Pr.439:Cam axis length per cycle	36000000 pulse	110.0000 mm
Pr.441:Cam stroke amount	3600000.0 μm	360.00000 degree
Pr.440:Cam No.	0	1
Pr.444:Cam axis phase compensation advance time	0 μs	0 μs
Pr.445:Cam axis phase compensation time constant	10 ms	10 ms
Pr.446:Synchronous control deceleration time	0 ms	0 ms
Pr.447:Output axis smoothing time constant	0 ms	0 ms
Synchronous control initial position parameter	Set the parameter for the initial alignment when starting the synchronous	

Figura 39: Configuración software del sincronismo (eje 2)

El eje 3 y el eje 4 no siguen ningún perfil de leva, sino que van sincronizados al eje 1 mediante relaciones de engranajes, yendo el eje 3 a la mitad de velocidad que el eje 1, y el eje 4 al doble de velocidad (Figura 40).

Item	Axis #1	Axis #3	Axis #4
Synchronous control module setting	Set each module parameter.		
Main shaft			
Main input axis			
Pr.400:Type	0:Invalid	1:Servo Input Axis	1:Servo Input Axis
Pr.400:Axis No.	0	2	3
Sub input axis			
Pr.401:Type	0:Invalid	0:Invalid	0:Invalid
Pr.401:Axis No.	0	0	0
Main shaft composite gear			
Pr.402:Main	1:Input+	1:Input+	1:Input+
Pr.402:Sub	0:No Input	0:No Input	0:No Input
Main shaft gear			
Pr.403:Numerator	1	1	2
Pr.404:Denominator	1	2	1
Main shaft clutch			
Main shaft clutch control setting			
Pr.405:ON control mode	0:No Clutch (Direct Coupled Operation)	0:No Clutch (Direct Coupled Operation)	0:No Clutch (Direct Coupled Operation)
Pr.405:OFF control mode	0:OFF Control Invalid	0:OFF Control Invalid	0:OFF Control Invalid
Pr.405:High-speed input request signal	0	0	0
Pr.406:Main shaft clutch reference address setting	0:Current Value after Main Shaft Composite Gear	0:Current Value after Main Shaft Composite Gear	0:Current Value after Main Shaft Composite Gear
Pr.407:Main shaft clutch ON address	0 pulse	0.00000 degree	0.00000 degree
Pr.408:Movement amount before main shaft clutch ON	0 pulse	0.00000 degree	0.00000 degree
Pr.409:Main shaft clutch OFF address	0 pulse	0.00000 degree	0.00000 degree
Pr.410:Movement amount before main shaft clutch OFF	0 pulse	0.00000 degree	0.00000 degree
Pr.411:Main shaft clutch smoothing system	0:Direct	0:Direct	0:Direct

Figura 40: Configuración software del sincronismo (ejes 3 y 4)

7.3.3. Programación del autómatas programable (PLC)

El programa está dividido en dos bloques: bloque de inicialización y bloque de ejecución cíclica. En el primero, se cargan los valores iniciales a ciertas variables de comando de los servoamplificadores. Estos valores podrán actualizarse por programa en el bloque de ejecución cíclica. El bloque de ejecución cíclica está dividido a su vez en seis bloques de programación para favorecer su modularidad, facilitar su comprensión y evitar errores.

7.3.3.1. *AxisEnable*

En este bloque se gestiona la activación de los ejes que se hayan configurado en el módulo de control de movimiento.

7.3.3.2. *AxisControl*

En este bloque se instancia el FB de control del eje tantas veces como ejes se hayan configurado en el módulo de control de movimiento. En todos ellos la activación de la FB se realiza cuando el eje está presente, ya sea el eje real o uno de los auxiliares. Mediante esta FB se controla las opciones de comando del eje y se monitoriza el estado de este.

7.3.3.3. *CrearLevas*

Para lograr el sincronismo del eje 2 con el eje 1, es decir, del cuchillo rotatorio con la cinta de entrada, es necesario configurar la leva electrónica que deberá seguir el eje 2. La creación de esta leva electrónica se realiza mediante la FB asociada al módulo de control de movimiento o Simple Motion "M+FX5SSC_MakeRotaryCutterCam_00A". La creación de la leva electrónica depende del valor de las siguientes variables (Figura 41):

- "Cam resolution": Resolución de la leva.
- "Sheet length": Longitud del ciclo del maestro, es decir, la longitud del paquete.
- "Sheet synchronization width": Ancho de sincronización, es decir, distancia en la que los ejes van sincronizados.
- "Synchronous axis length": Longitud del eje síncrono.

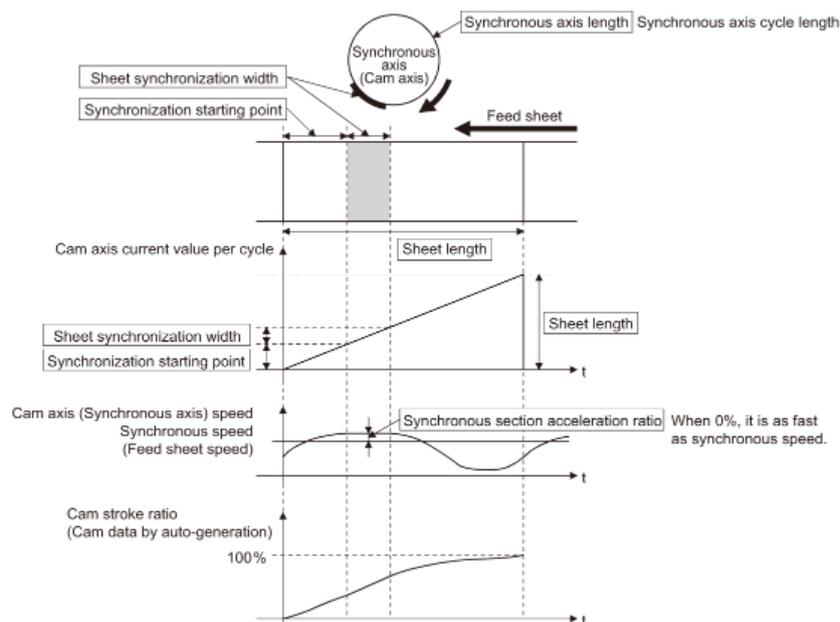


Figura 41: Variables de una leva electrónica

En este bloque se instancia esta FB y se establecen las variables que recogen los valores de las variables necesarias para la creación de la leva desde el HMI. Cabe destacar que es necesario dar el valor de la variable “sheet length” al parámetro de longitud del ciclo de la configuración del módulo. Además, es necesario asignar un número de leva cada vez que se genere una leva electrónica nueva.

7.3.3.4. EstadoEjes

En este apartado se gestiona el estado de los ejes. Por una parte, se gestiona el *Servo ON* de todos los ejes, el cual viene dado al activar el comando de servo presente y el comando de *Enable*. Por otro lado, se gestiona el movimiento *Home* de cada uno de los ejes, el cual permite llevar estos a la posición que se considera el cero del eje. Por último, se gestiona el *Reset* de los ejes y el estado de error que pueda darse en la simulación.

En este bloque se establecen las señales que permiten visualizar el estado de los ejes en el HMI y las señales que permiten activar los comandos de *Servo ON*, *Home* y *Reset*.

7.3.3.5. MapeadoGemini

Este bloque es el encargado de gestionar la conexión del programa de control con el simulador del Gemelo Digital, *Gemini*. En este se controla el movimiento del eje 1, la cinta de entrada, en función de la señal que activara el *JOG* desde el HMI y de la señal de uno de los sensores dispuestos en la cinta de entrada.

Por otro lado, se debe gestionar el cambio de unidades necesario de la variable de valor actual de cada uno de los ejes. El eje 1 está configurado como cinta transportadora, por lo que su valor actual está dado en unidades de longitud, concretamente micrómetros. Sin embargo, la posición de los rodillos en la articulación de *Gemini* debe introducirse en grados angulares. Por lo tanto, se debe realizar el cambio de unidades de micrómetros a grados teniendo en cuenta la longitud del perímetro del rodillo, es decir, su radio. Para ello, primero se debe realizar un cambio de tipo variable de *Integer* a *Real* y después, realizar el cambio de unidades descrito. Para el resto de los ejes, que están configurados directamente en grados en el módulo, simplemente una vez modificado el tipo de variable se debe tener en cuenta que ésta dará el valor en grados angulares con cinco decimales.

Por último, como se ha comentado anteriormente para controlar las cintas transportadoras dentro del entorno de *Gemini* se debe controlar su velocidad. Para ello, al igual que para la posición, se debe cambiar el tipo de variable de *Integer* a *Real* y, además, se debe tener en cuenta que la velocidad en el módulo de Simple Motion está dada en mm/min. La velocidad de las cintas transportadoras en *Gemini* han de darse en mm/seg.

7.3.3.6. Sincronismo

En este último bloque se realiza la gestión del sincronismo de los ejes y las señales que van al panel de operador con respecto al sincronismo. En este caso, el sincronismo se activa teniendo en cuenta la señal que viene del HMI y la señal que viene del segundo sensor dispuesto en la cinta de entrada. Además, se gestiona la señal que muestra en el HMI si el sincronismo está activado. Por último, se asigna la leva electrónica a seguir en función del número de leva que se le asigna a cada una de ellas durante su creación.

7.3.4. Interfaz máquina-humano (HMI)

En este apartado se describe el panel de operador diseñado para el control y la supervisión de la estación completa.

7.3.4.1. Pantalla de inicio

En primer lugar, se crea una pantalla de inicio que presenta la máquina a controlar y desde la cual se puede acceder a las tres pantallas de control y supervisión diseñadas mediante los botones dispuestos en la parte inferior (Figura 42). Se marca en azul la pantalla en la que se encuentra el operador actualmente.

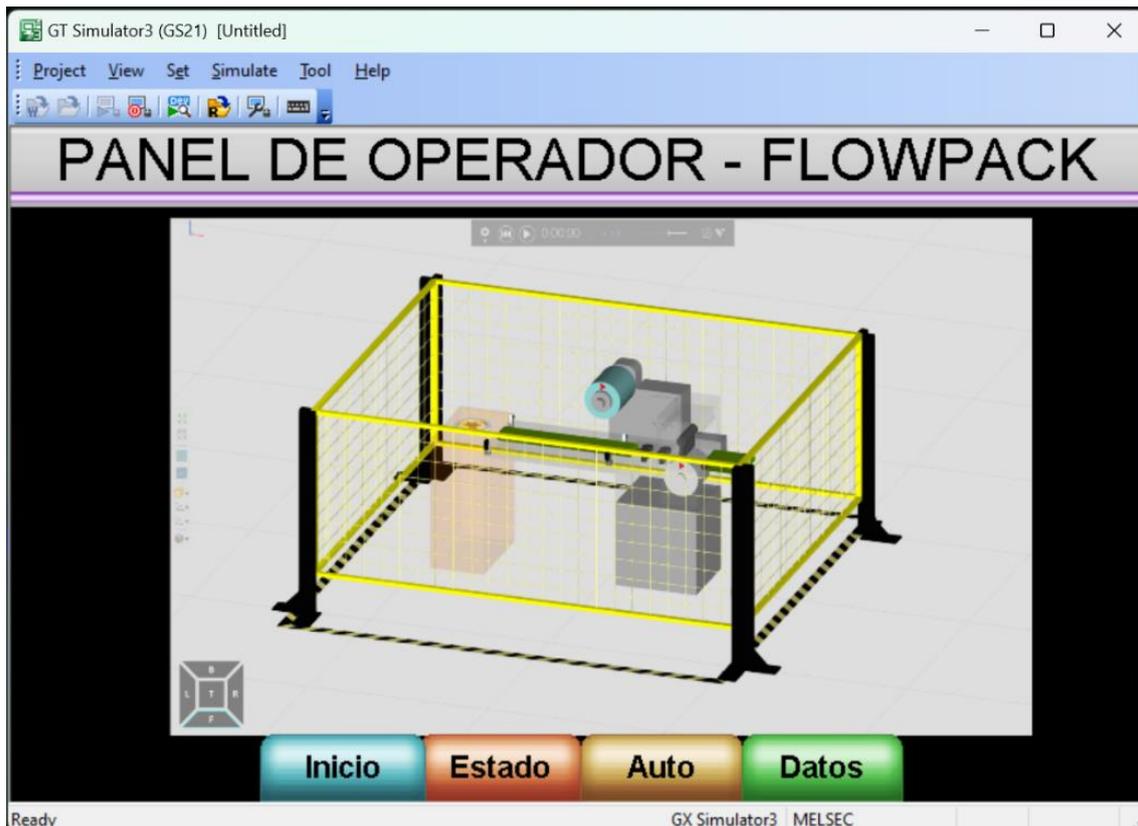


Figura 42: Interfaz máquina-humano, pantalla de Inicio

7.3.4.2. Pantalla de estado

La segunda pantalla diseñada se corresponde con el bloque de control “EstadoEjes” (Figura 43), en esta se puede controlar la activación de los ejes y comprobar que todos ellos han realizado la función correctamente. Al pulsar el botón de “Servo ON” se activan los cuatro ejes a la vez y se puede comprobar que la operación se ha realizado correctamente mediante las lámparas que muestran el estado de los cuatro ejes por separado.

Además, se puede realizar el posicionado a *Home* desde esta pantalla y, a su vez, supervisar que todos los ejes se encuentran en la posición deseada. Al igual que para la función anterior, al pulsar el botón “Home” se realiza el posicionado de los cuatro ejes simultáneamente y se puede comprobar que este se ha realizado correctamente mediante las lámparas de estado de cada uno de los ejes.

Por último, en esta pantalla es posible realizar el reseteo de los cuatro ejes simultáneamente, utilizando el botón de “Reset”. Además, se comprueba si alguno de ellos presenta un error mediante la lámpara de “Error” que se enciende en tal caso.



Figura 43: Interfaz máquina-humano, pantalla de Estado

7.3.4.3. Pantalla de movimiento automático

En esta pantalla (Figura 44) se controla el movimiento *Jog* del eje maestro, así como la activación y la desactivación del sincronismo de los ejes 2, 3 y 4. Al pulsar el botón de *Jog*, se activa la señal necesaria para éste. Sin embargo, el eje no comenzará a moverse hasta que no se active la señal del sensor correspondiente. El sincronismo se activa pulsando el botón de “Activar sincronismo” de esta pantalla, una vez el sensor correspondiente se activa en la cinta de entrada y se ve el estado a través de la lámpara “No Sincronismo/Sincronismo”. Cuando se pulsa el botón de “Desactivar sincronismo” se realizará el reseteo de la señal de control síncrono.

Además, se supervisa la correcta activación del sincronismo mediante la lámpara que se enciende cuando el control síncrono de los tres ejes está activado. Por último, se supervisa la posición de cada uno de los cuatro ejes mediante un display numérico para cada eje, teniendo en cuenta las unidades de la variable para cada uno de ellos.



Figura 44: Interfaz máquina-humano, pantalla de Automático

7.3.4.4. Pantalla de creación de leva

La última pantalla diseñada cumple la función de creación de las levas electrónicas (Figura 45). En esta pantalla se introducen los valores de las variables necesarias para la creación de las levas mediante la FB comentada anteriormente. Es necesario activar dicha FB, lo cual se realiza pulsando el botón de “Generar leva” y se comprueba que se ha realizado esta función correctamente mediante la lámpara de estado “Leva no generada/Leva generada”.



Figura 45: Interfaz máquina-humano, pantalla de Datos

8. PLAN DE TRABAJO

El plan de trabajo se distribuye en cuatro grandes paquetes de trabajo que contienen a su vez todas las tareas a realizar en el proyecto. Estos paquetes de trabajo son: formación, desarrollo, pruebas y administrativo.

8.1. PT1: Formación

Este paquete de trabajo refleja todas las labores de formación realizadas que han permitido adquirir los conocimientos necesarios para la realización del proyecto. Al tratarse de un proyecto en el que se utilizan diversos softwares, es de esperar la necesidad de un periodo de formación en estos antes de poder iniciar el desarrollo de dicho proyecto.

8.1.1. PT1.T1: MELSOFT Gemini

El objetivo será hacer uso de todos los recursos disponibles sobre el software para familiarizarse con este. Con este objetivo se modelizan sistemas sencillos para así conocer las funciones básicas necesarias para el funcionamiento de la máquina empaquetadora.

Responsable: Jesica Romano

Participantes: Jesica Romano

Resultados: varios Gemelos Digitales sencillos que desarrollen movimientos y funcionamientos que son necesarios en la máquina empaquetadora. Al menos un modelo que funcione mediante la conexión de este al PLC y comparta valores de diversas variables entre ambos.

Recursos materiales: ordenador portátil, software MELSOFT Gemini, tutoriales disponibles y el manual del software.

Duración: 3 semanas (06/02/23 - 24/02/23, 40h)

8.1.2. PT1.T2: Programación módulo Simple Motion

Por un lado, en lo que respecta a la programación del autómatas no es necesaria una formación exhaustiva. Sin embargo, es necesario conocer el aspecto de programación y configuración del módulo de control de movimiento o Simple Motion.

Responsable: Jesica Romano

Participantes: Jesica Romano

Resultados: Conocimientos básicos para programar y configurar el módulo de control de movimiento de manera que se sincronicen tres ejes. La configuración y generación de una leva electrónica automática de manera que sea posible customizar el funcionamiento.

Recursos materiales: ordenador portátil, software MELSOFT GX Works3 y el manual de programación del módulo FX5.40SSC-S.

Duración: 1 semana (27/02/23 - 03/03/23, 30h)

8.1.3. PT1.T3: GT Designer3

Será necesario conocer las funciones básicas que permitan controlar y monitorizar el proceso desde el HMI que funciona como panel de operador. Se realizan pequeños programas con distintas funciones que puedan ser de valor en el panel de operador final.

Responsable: Jesica Romano

Participantes: Jesica Romano

Resultados: Un programa de HMI con varias pantallas y la posibilidad de pasar de una a otra mediante un botón. Además, desde este programa se muestra el valor de variables de PLC en pantalla y se puede introducir el valor de una variable de PLC desde la pantalla.

Recursos materiales: ordenador portátil, software MELSOFT GT Designer3 y el manual de programación de la serie GS-21.

Duración: 1 semana (06/03/23 - 10/03/23, 12h)

8.2. PT2: Desarrollo

8.2.1. PT2.T1: Modelado CAD de la máquina Flowpack

El primer paso para el desarrollo del Gemelo Digital es el diseño y modelado 3D de la máquina en un entorno CAD. Se realiza un análisis de dimensiones y se realiza un primer diseño básico a partir del cual se irá aumentando hasta tener un modelo lo suficientemente detallado. Una vez creado el diseño mecánico de la máquina se diseña la máquina con diferentes materiales para que sean distinguibles una vez introducido en la herramienta de modelado de Gemelo Digital.

Responsable: Jesica Romano

Participantes: Jesica Romano

Resultados: un modelo CAD detallado que permita realizar un Gemelo Digital de una máquina Flowpack funcional.

Recursos materiales: ordenador portátil, recursos online y entorno CAD.

Duración: 1 semana (13/03/23 - 17/03/23, 20h)

8.2.2. PT2.T2: Modelado del Gemelo Digital

El objetivo principal de esta tarea es introducir el modelo realizado en la tarea anterior (PT2.T1) y aplicando los conocimientos adquiridos en la tarea de formación PT1.T3, obtener un Gemelo Digital funcional de la máquina empaquetadora diseñada.

Responsable: Jesica Romano

Participantes: Jesica Romano

Resultados: un modelo con todos los movimientos necesarios modelizados. Además de una lista con todas las variables de control necesarias tanto de entrada como de salida del modelo en el entorno MELSOFT Gemini.

Recursos materiales: ordenador portátil, recursos online y software MELSOFT Gemini.

Duración: 4 semanas (20/03/23 - 24/03/23 ; 10/04/23 - 28/04/23, 80h)

8.2.3. PT2.T3: Programa de control

Esta tarea envuelve tanto el diseño hardware del sistema de control como el diseño y desarrollo del programa de control necesario para controlar el modelo final adquirido en la tarea PT2.T3. Durante esta tarea se realiza cualquier cambio o modificación necesaria en el modelo que se haya pasado por alto en el modelado y sea necesario para el programa de control.

Responsable: Jesica Romano

Participantes: Jesica Romano

Resultados: un programa de control en el entorno de GX Works3 que permita la conexión entre el PLC y modulo Simple Motion simulados y el entorno Gemini.

Recursos materiales: ordenador portátil, software MELSOFT GX Works3 y software MELSOFT Gemini.

Duración: 2,5 semanas (01/05/23 - 17/05/23, 50h)

8.2.4. PT2.T4: Conexión Gemelo-PLC

Con el modelo adquirido en la tarea PT2.T2 y el programa de control adquirido en la tarea PT2.T3 se pasa a realizar el conexionado de las señales de entrada y salida de la máquina con las del programa de control.

Responsable: Jesica Romano

Participantes: Jesica Romano

Resultados: el mapeado de las señales de control del sistema.

Recursos materiales: ordenador portátil, software MELSOFT GX Works3 y software MELSOFT Gemini.

Duración: 2 horas (17/05/2023)

8.2.5. PT2.T5: Panel de operador

El objetivo de esta tarea es el diseño y desarrollo del panel de operador que permita el control y monitorización de la máquina en el entorno GT Designer3. Este panel deberá comunicarse con el PLC simulado y poder acceder a las distintas variables de este para realizar funciones de escritura y lectura.

Responsable: Jesica Romano

Participantes: Jesica Romano

Resultados: una pantalla de inicio que presente la máquina, una pantalla que permita controlar y ver el estado de los servoamplificadores, una pantalla que permita activar el funcionamiento automático de la máquina y una pantalla que permita crear levas electrónicas automáticas.

Recursos materiales: ordenador portátil, software MELSOFT GT Designer3 y el manual de programación de la serie GS-21.

Duración: 1 semana (22/05/23 - 26/05/23, 10h)

8.3. PT3: Pruebas

8.3.1. PT3.T1: Importación modelo CAD a MELSOFT Gemini

Se trata de realizar la importación del modelo CAD diseñado en la tarea PT2.T1 al entorno de Gemini de la manera óptima. Se desea que el modelo en Gemini tenga el mismo orden y jerarquía que en el entorno CAD.

Responsable: Jesica Romano

Participantes: Jesica Romano

Resultados: un modelo en Gemini con el que sea posible trabajar a nivel de componente y que permita el modelado de estos de la manera necesaria para el correcto funcionamiento de la máquina.

Recursos materiales: ordenador portátil y software MELSOFT Gemini.

Duración: 30 minutos (18/05/2023)

8.3.2. PT3.T2: Validación del Gemelo Digital en Gemini

Esta tarea trata de realizar todas las pruebas necesarias manualmente dentro del entorno de Gemini que permitan saber si todos los movimientos modelados en la tarea PT2.T2 se han realizado correctamente.

Responsable: Jesica Romano

Participantes: Jesica Romano

Resultados: el modelo de la máquina final donde se ha comprobado el correcto funcionamiento de todos los elementos.

Recursos materiales: ordenador portátil y software MELSOFT Gemini.

Duración: 2 días (18/05/23 - 19/05/23, 8h)

8.3.3. PT3.T3: Validación del conexionado PLC-Gemini

El objetivo de esta tarea es con el modelo validado en la tarea PT3.T2, se pasará a probar el programa de control haciendo una simulación del PLC y módulo de control de movimiento. Sin embargo, se realizará una comprobación manual del funcionamiento del mapeado de las señales, es decir, no se implementará la leva y el sincronismo necesarios.

Responsable: Jesica Romano

Participantes: Jesica Romano

Resultados: el mapeado de señales final que permite controlar la máquina de manera correcta.

Recursos materiales: ordenador portátil, software MELSOFT GX Works3 y software MELSOFT Gemini.

Duración: 1 día (29/05/2023, 4h)

8.3.4. PT3.T4: Validación del programa de control

Partiendo del programa de control generado en la tarea PT2.T3 y teniendo en cuenta la validación del mapeado realizada en la tarea PT3.T3, se validará el programa de control completo. Se generará la leva electrónica automática necesaria y se configurará el sincronismo de los ejes de manera que se pueda validar el funcionamiento de la máquina.

Responsable: Jesica Romano

Participantes: Jesica Romano

Resultados: un programa de control que permite controlar la máquina empaquetadora en el entorno Gemini.

Recursos materiales: ordenador portátil, software MELSOFT GX Works3 y software MELSOFT Gemini.

Duración: 2 días (30/05/23 - 31/05/23, 8h)

8.3.5. PT3.T5: Simulación completa

En esta tarea se simula el sistema PLC-Simple Motion, la pantalla HMI y el modelo, y se pone a funcionar todo simultáneamente. Se comprueba que el funcionamiento es el obtenido en las pruebas manuales previas.

Responsable: Jesica Romano

Participantes: Jesica Romano

Resultados: el Gemelo Digital final de la máquina empaquetadora de alimentos.

Recursos materiales: ordenador portátil, software MELSOFT GX Works3, software MELSOFT GT Designer3 y software MELSOFT Gemini.

Duración: 1 día (01/06/2023, 4h)

8.4. PT4: Administrativo

8.4.1. PT4.T1: Documentación del proyecto

En esta tarea se redacta una documentación formal que recoge todo el trabajo realizado a lo largo del proyecto, así como toda la información que se considera de importancia para comprender y analizar la solución final propuesta.

Responsable: Jesica Romano

Participantes: Jesica Romano

Resultados: la memoria final del trabajo de fin de máster.

Recursos materiales: ordenador portátil y Microsoft Office Word.

Duración: 4 semanas (05/06/23 - 30/06/23, 40h)

9. PRESUPUESTO

En este apartado se presenta el presupuesto del proyecto desglosado, dividiendo este en bienes materiales y recurso humanos. En el caso de este proyecto, dentro de los bienes materiales se trata únicamente con bienes inventariables, es decir, que se utilizaran en varios proyectos para llegar a amortizarlos. Los materiales utilizados en el proyecto son: un ordenador portátil y un monitor externo, así como las licencias de todos los softwares necesarios para el modelado, desarrollo y simulación del Gemelo Digital.

Tipo de herramienta	Descripción	Cantidad	Precio
Lenovo Thinkpad X13 Yoga	Ordenador portátil	1	601,40 €
Lenovo D24-20 24"	Monitor de 24"	1	100,00 €
MELSOFT GX Works3	Licencia del software de programación del PLC (v 1.090U)	1	283,39 €
MELSOFT GT Designer3	Licencia del software de programación del HMI (v 1.290C)	1	186,82 €
MELSOFT Gemini	Licencia del software de simulación (v 1.45)	1	13.000,00 €
FreeCAD	Licencia del software de diseño CAD	1	0,00 €
TOTAL			14.171,61 €

Tabla 2: Bienes materiales

Como se ha comentado anteriormente, estos bienes son amortizables, por lo que se ha de calcular la amortización de estos. Para ello, se tiene en cuenta el tiempo de amortización de cada uno de ellos, siendo el del ordenador y el monitor de cuatro años, mientras que las licencias son amortizables a un año. Como el tiempo de desarrollo del proyecto es menor a ambos tiempos, se realiza el cálculo para el periodo de cinco meses.

Tipo de herramienta	Cuota de adquisición	Tiempo de amortización (años)	Tiempo de uso (meses)	Amortización
Lenovo Thinkpad X13 Yoga	601,40 €	4	5	62,65 €
Lenovo D24-20 24"	100,00 €	4	5	10,42 €
MELSOFT GX Works3	283,39 €	1	5	118,08 €
MELSOFT GT Designer3	186,82 €	1	5	77,84 €
MELSOFT Gemini	13.000,00 €	1	5	5.416,67 €
FreeCAD	0,00 €	1	5	0,00 €
TOTAL				5.685,65 €

Tabla 3: Presupuesto de amortización

Por otro lado, en lo que respecta a los recursos humanos, se unifican las horas totales correspondientes a cada uno de los paquetes de trabajo descritos en el plan de trabajo y se suman las horas de reunión que requeridas con el director del proyecto y el supervisor asignado en la empresa.

Tarea	Operario	Duración (h)	Tarifa (€/h)	Precio (€)
PT1: Formación	Ingeniera junior	82	30	2.460,00 €
Apoyo en la formación	Ingeniero senior (supervisor)	8	60	480,00 €
PT2: Desarrollo	Ingeniera junior	162	30	4.860,00 €
Supervisión del desarrollo	Ingeniero senior (supervisor) e Ingeniero senior (director)	4	60	240,00 €
Apoyo en la configuración de Simple Motion	Ingeniero senior (supervisor)	3	60	180,00 €
Supervisión del programa de control	Ingeniero senior (supervisor) e Ingeniero senior (director)	4	60	240,00 €
PT3: Pruebas	Ingeniera junior	24,5	30	735,00 €
Supervisión de la simulación final	ingeniero senior (director)	2	60	120,00 €
PT4: Administración	Ingeniera junior	40	30	1.200,00 €
TOTAL				10.515,00 €

Tabla 4: Presupuesto del grupo de trabajo

Por último, se obtiene el presupuesto total calculando los costes indirectos y asociado a posibles imprevistos que puedan surgir a lo largo de la realización del proyecto.

Concepto	Importe
Presupuesto de inventariables	5.685,65 €
Presupuesto de recursos humanos	10.515,00 €
SUBTOTAL	16.200,65 €
Costes indirectos (10%)	1.620,07 €
Imprevistos (5%)	810,03 €
TOTAL	34.831,40 €

Tabla 5: Presupuesto total

El presupuesto total del desarrollo del proyecto es de **TREINTA Y CUATRO MIL OCHOCIENTOS TREINTA Y UNO CON CUARENTA.**

10. RESULTADOS

En lo que respecta al Gemelo Digital, hay que distinguir dos partes: el desarrollo y el modelado. En lo que al desarrollo del Gemelo Digital respecta, se han cumplido todos los objetivos establecidos inicialmente. Es decir, se ha creado un modelo CAD de la máquina empaquetadora a simular y se ha importado el diseño CAD al entorno de simulación Gemini, siendo posible trabajar sobre los componentes por separado. No ha sido necesario manipular el diseño realizado previamente para poder pasar a realizar el modelado del Gemelo Digital. Esto supone que el diseño de la máquina se ha realizado de manera óptima teniendo en cuenta todos sus aspectos críticos. Por otro lado, en lo que el modelado del Gemelo Digital respeta, se considera que se ha realizado un modelado lo suficientemente detallado para poder realizar la puesta en marcha virtual. En este sentido, se han introducido todos los elementos, comportamientos y propiedades necesarios para pasar de un modelo CAD estático a una máquina empaquetadora con movimiento. Además, se ha conseguido que los tres ejes de la máquina realicen sus movimientos de manera sincronizada, lo cual es uno de los objetivos más importantes de esta parte del proyecto. En resumen, se ha creado un gemelo digital completamente funcional en el que todos los movimientos y elementos han sido validados manualmente. Por último, cabe destacar que ha sido posible conectar el software de simulación con el de programación de PLC.

En lo que respecta al proyecto de automatización para el control y la monitorización de la máquina empaquetadora, cabe destacar que tanto su configuración hardware como software se han realizado satisfactoriamente. Se ha alcanzado tanto el objetivo de controlar la máquina, como su monitorización. Desde el programa del PLC se gestionan los comandos de control de cada uno de los servoamplificadores que se han introducido en la configuración hardware. A través de estos comandos es posible realizar diversos modos de operación controlando la velocidad o la posición del servo en función del modo de operación seleccionado. Para el caso de la máquina empaquetadora se realiza un movimiento “Jog” en el que se controla la velocidad de cada uno de los ejes. Mediante estos comandos es posible activar los ejes, realizar el movimiento a “Home”, realizar el “Reset” de los ejes, etc. La monitorización permite conocer el estado de cada uno de los ejes, es decir, para saber si los ejes se han activado correctamente, si han realizado los movimientos deseados, si ha ocurrido un error, etc. Para realizar la monitorización y el control del proceso se ha realizado la programación del panel de operador que consta de tres pantallas.

Por último, se realiza la puesta en marcha virtual del proceso a través del simulador “Gemini”, que comunica el PLC virtual, el HMI virtual y el Gemelo digital en una simulación SiL.

11. LIMITACIONES

A la hora de plantear el proyecto, se debe discernir sobre la posibilidad de realizar la puesta en marcha virtual mediante Hardware in the Loop (HiL) además de mediante Software in the Loop (SiL). Como se ha comentado en apartados anteriores, realizar la puesta en marcha mediante SiL significa que el controlador debe simularse junto con el gemelo digital de forma simultánea. Sin embargo, la puesta en marcha virtual mediante HiL implica que para simular el funcionamiento de la máquina se ha de conectar el gemelo digital a la configuración hardware real. Esto supone una limitación en el marco del proyecto, ya que no se dispone de todo el hardware real. Concretamente, se dispone del PLC y la pantalla HMI, pero no del módulo de control de movimiento o Simple Motion. A su vez, es necesario conectar los cuatros servoamplificadores a

dicho módulo Simple Motion. Esto último su pone un problema adicional, ya que no se dispone de tal número de servoamplificadores.

Por otro lado, a la hora de generar el gemelo digital, el primer paso es generar el modelo CAD de la máquina. En este caso, el software Gemini no permite crear este modelo desde cero. Esto podría considerarse una limitación, ya que es necesario disponer de una licencia adicional de un software de CAD para poder llevar a cabo el proyecto. Dado que el diseño de la máquina Flowpack se ha planteado en términos funcionales más que de representatividad de la realidad, ha sido posible diseñar la máquina haciendo uso del software libre FreeCAD.

Por último, la principal limitación del proyecto se encuentra en la gran capacidad de procesamiento requerida para la puesta en marcha virtual mediante SiL. Concretamente, el ordenador portátil utilizado durante el proyecto no era capaz de realizar la simulación SiL de la puesta en marcha virtual de forma fluida. No sólo es necesario simular el gemelo digital, sino hay que también hay que simular el PLC y la pantalla HMI. Uno de los puntos más exigentes de la puesta en marcha virtual es la simulación del PLC, el módulo de control de movimiento y el movimiento de los cuatro servoamplificadores, ya que no sólo se simula el funcionamiento de estos elementos, sino también entre éstos y con el Gemelo Digital. La simulación de la pantalla HMI también agrega una exigencia adicional a la puesta en marcha virtual.

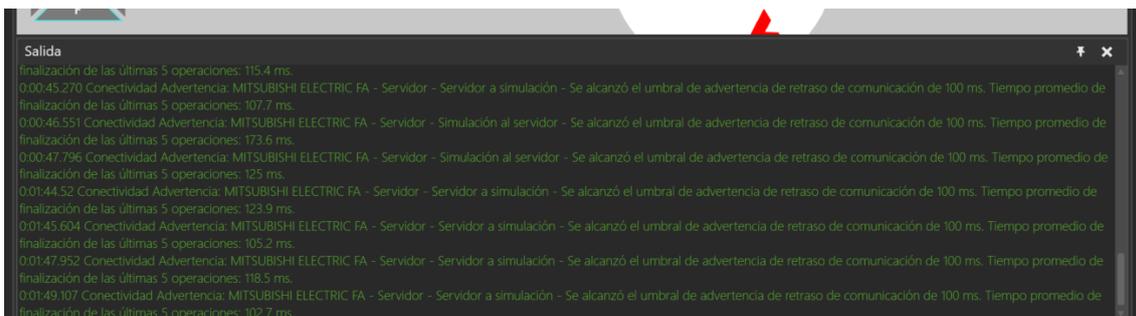


Figura 46: Ventana de salida en Gemini

Se ha comprobado que esta limitación en la capacidad de procesamiento necesaria para realizar la simulación SiL de la puesta en marcha virtual mediante la integración de todas las herramientas del entorno Mitsubishi, implica un desfase en la comunicación entre el controlador y el simulador, lo cual se traduce en un desfase entre los movimientos que dependen de la comunicación y los movimientos que no dependen de esta comunicación. Esto provoca que la simulación del corte de paquetes no sea fluida. Al monitorizar tanto la simulación como los ejes, se puede comprobar que el módulo de control de movimiento sigue la leva electrónica establecida correctamente. Además, los dos ejes restantes realizan el movimiento siguiendo la configuración realizada en dicho módulo. Por otro lado, se percibió un empeoramiento de la simulación en el momento en el que se introdujo la simulación de la pantalla HMI. Por todo esto, se concluye que los problemas de visualización que se dan en la simulación se derivan de la alta capacidad de procesamiento requerida por el conjunto de herramientas que intervienen en la puesta en marcha virtual.

12. CONCLUSIONES Y TRABAJO FUTURO

Este trabajo surge dentro de Mitsubishi Electric debido a la necesidad de incorporar las soluciones basadas en Gemelo Digital al porfolio de ofertas ya existentes en la empresa. Para ello, se estableció como requisito el análisis de la herramienta MELSOFT Gemini. Así, este trabajo surge como una primera aproximación al desarrollo de Gemelos Digitales con el objetivo de ir adentrándose en las diversas funcionalidades que ofrece la integración de las distintas herramientas del paquete MELSOFT.

En este trabajo se ha presentado una propuesta para la creación del gemelo digital de una máquina Flowpack haciendo uso de la herramienta MELSOFT Gemini. Además, se ha hecho uso de distintos programas software del paquete ofrecido por Mitsubishi Electric para poder realizar su puesta en marcha virtual. La mayor parte del peso del proyecto ha recaído en el modelado del gemelo digital en Gemini, aunque cabe destacar que la configuración del módulo de control de movimiento o Simple Motion también ha supuesto una carga considerable.

La solución final propuesta resulta en tres bloques de trabajo diferenciados, a saber: el modelado del gemelo digital, la programación del controlador (configuración hardware y software) y la programación de la pantalla HMI. Al tratarse de un proyecto sin precedentes en el uso del paquete MELSOFT, se consideró necesaria una fase de formación en torno al simulador 3D “Gemini” y a la configuración del módulo de control de movimiento dentro de GX Works3.

Se han alcanzado todos los objetivos inicialmente planteados, obteniendo un gemelo digital funcional de una máquina Flowpack en el entorno Gemini con un controlador de Mitsubishi Electric. En este sentido, se espera que el presente documento tenga un papel instructivo para esos futuros proyectos, en lo que a la metodología de creación de un gemelo digital respecta.

Por último, se espera de este proyecto que sirva de base para posibles trabajos futuros. En este sentido, se observan cuatro posibles líneas de trabajo a futuro para continuar con este proyecto.

- Realizar la puesta en marcha virtual mediante *SiL* de manera distribuida. Es decir, simular por un lado, el gemelo digital y, por otro lado, el controlador y la pantalla HMI, en sendos ordenadores. Esta idea surge al observar que no existen problemas de simulación cuando se simulan sólo el PLC con el módulo Simple Motion y la pantalla HMI para realizar el control y la monitorización. Además, también se observa que el simulador del gemelo digital no presenta ningún problema de procesamiento de forma aislada.
- Realizar la puesta en marcha virtual mediante *HiL*. Es otra manera de intentar contrarrestar el problema de capacidad de procesamiento del ordenador. Al implementar el sistema hardware completo físicamente, solo sería necesario realizar la simulación del gemelo digital. Esta opción exigiría menos al procesador y posiblemente solucionase los problemas de visualización descritos anteriormente.
- Mejorar la pantalla HMI incluyendo algunas funcionalidades en el panel de operador que permitan la completa automatización del sistema. Por ejemplo, sería interesante introducir una entrada numérica en la pantalla “Movimiento automático” que permita escoger el número de leva que se desea seguir con el eje del cuchillo rotatorio.
- La adición de más elementos para completar la línea de producción iniciada en este proyecto. Un elemento interesante podría ser un robot al final de la cinta que realizase un movimiento de “Pick and place” con el fin de colocar los caramelos empaquetados en un pallet final o en algún tipo de almacenamiento.

REFERENCIAS

- [1] D. Y. Jeong, M. S. Baek, T. B. Lim, Y. W. Kim, S. H. Kim, Y. T. Lee, W. S. Jung, and I. B. Lee, “Digital twin: Technology evolution stages and implementation layers with technology elements,” *IEEE Access*, vol. 10, pp. 52609–52620, 2022.
- [2] R. Stark, C. Fresemann, and K. Lindow, “Development and operation of digital twins for technical systems and services,” *CIRP Annals*, vol. 68, pp. 129–132, 1 2019.
- [3] F. Tao, B. Xiao, Q. Qi, J. Cheng, and P. Ji, “Digital twin modeling,” *Journal of Manufacturing Systems*, vol. 64, pp. 372–389, 7 2022.
- [4] Q. Qi, F. Tao, T. Hu, N. Anwer, A. Liu, Y. Wei, L. Wang, and A. Y. Nee, “Enabling technologies and tools for digital twin,” *Journal of Manufacturing Systems*, vol. 58, pp. 3– 21, 1 2021.
- [5] H. van der Valk, H. Haße, F. M“oller, and B. Otto, “Archetypes of digital twins,” *Business and Information Systems Engineering*, vol. 64, pp. 375–391, 6 2022.
- [6] O’CONNOR, Chris. Digital Twin: Simple, but detailed. EEUU, IBM, junio 2017. Disponible en: <https://www.youtube.com/watch?v=RaOejcczPas>
- [7] BOUZÓN, Irene. Gemelos digitales. En: *CTIC News* [en línea]. 29 de abril de 2019 [consulta: 25 mayo 2021]. Disponible en: <https://www.fundacionctic.org/es/actualidad/gemelos-digitales>
- [8] HEMMELGARN, Tony. The Digital Twin: Realizing Transformation. Berlín, Siemens Software, mayo 2017. Disponible en: <https://youtu.be/oWeAYJ8CN70>
- [9] A. Koulouris, N. Misailidis, and D. Petrides, “Applications of process and digital twin models for production simulation and scheduling in the manufacturing of food ingredients and products,” *Food and Bioproducts Processing*, vol. 126, pp. 317–333, 3 2021.
- [10] A. Hyre, G. Harris, J. Osho, M. Pantelidakis, K. Mykoniatis, and J. Liu, “Digital twins: Representation, replication, reality, and relational (4rs),” *Manufacturing Letters*, vol. 31, pp. 20–23, 1 2022.
- [11] E. VanDerHorn and S. Mahadevan, “Digital twin: Generalization, characterization and implementation,” *Decision Support Systems*, vol. 145, p. 113524, 6 2021.