

Jornadas de Automática

Arquitectura modular Unity-PLC para gemelos digitales industriales

Emiliano Pérez^{a,*}, María La Calle^a, Pilar Merchán^a, Santiago Salamanca^a

^aEscuela de Ingenierías Industriales, Universidad de Extremadura, Avda. de Elvas s/n, 06006 Badajoz, España.

To cite this article: Pérez, E., La Calle, M., Merchán, P., Salamanca, S. 2026. A modular Unity-PLC architecture for industrial digital twins. Jornadas de Automática, 47. <https://doi.org/10.17979/ja-cea.2026.47.13874>

Resumen

Este trabajo presenta una arquitectura modular para el desarrollo de gemelos digitales industriales en Unity conectados a autómatas Siemens mediante S7.NetPlus. La propuesta se comprueba funcionalmente sobre una maqueta Fischertechnik formada por una cinta transportadora y una troqueladora, controlada por un PLC Siemens S7-1200. El sistema organiza la comunicación en capas: interfaces HMI, sensores virtuales, actuadores, un bloque común de entradas/salidas y un módulo central de intercambio con el PLC. Esta separación permite añadir sensores, motores, finales de carrera o elementos de interfaz sin reescribir la lógica de comunicación. El gemelo digital permite tres modos de operación: simulación, control sincronizado de planta y supervisión directa de actuadores físicos. Además de su valor como banco de pruebas, la aplicación se ha empleado para diseñar prácticas docentes de automatización con dificultad progresiva, integrando programación en TIA Portal, validación virtual y ejecución sobre maqueta real.

Palabras clave: Gemelo digital, Automatización industrial, PLC, Unity, S7.NetPlus, Simulación

A modular Unity-PLC architecture for industrial digital twins

Abstract

This paper presents a modular architecture for developing industrial digital twins in Unity connected to Siemens PLCs through S7.NetPlus. The proposal is functionally checked on a Fischertechnik training plant composed of a conveyor belt and a die cutter, controlled by a Siemens S7-1200 PLC. The system structures communication into layers: HMI interfaces, virtual sensors, actuators, a shared input/output block and a central PLC exchange module. This separation makes it possible to add sensors, motors, limit switches or interface elements without rewriting the communication logic. The digital twin supports three operating modes: full simulation, synchronized plant control and direct supervision of physical actuators. Besides its value as a test bench, the application has been used to design automation laboratory exercises with increasing difficulty, combining TIA Portal programming, virtual validation and execution on the physical training plant.

Keywords: Digital twin, Industrial automation, PLC, Unity, S7.NetPlus, Simulation

1. Introducción

Los gemelos digitales se han consolidado como una herramienta clave en la Industria 4.0 al permitir representar, simular y supervisar sistemas físicos mediante modelos virtuales conectados a datos de operación (Kritzinger et al., 2018). En automatización industrial, esta tecnología facilita la validación temprana de programas de control, la reducción de errores du-

rante la puesta en marcha y la creación de entornos de formación más próximos a una planta real.

En el ámbito docente, el uso de maquetas físicas controladas por PLC sigue siendo fundamental para aprender sensores, actuadores, Grafset y programación en entornos como TIA Portal. Sin embargo, el acceso al laboratorio suele estar limitado por disponibilidad de equipos, horarios y riesgo de

*Autor para correspondencia: emilianoph@unex.es

manipulación incorrecta. Un gemelo digital interactivo permite preparar y validar programas antes de operar sobre la instalación real, manteniendo la lógica de control industrial y reduciendo la dependencia inicial del hardware.

Este planteamiento se relaciona con el concepto de fábricas de aprendizaje, en las que se reproducen entornos productivos realistas para desarrollar competencias de ingeniería mediante actividades prácticas. Una aplicación de este concepto, enriquecida con más características es el presentado en (Abele et al., 2017; Masaev et al., 2024).

En esta línea, se han desarrollado casos docentes de gemelos digitales de sistemas discretos empleando Unity, TIA Portal, y maquetes industriales, apoyándose en el framework Game4Automation para resolver la conexión entre el modelo virtual y la planta física (Balla et al., 2023). También se han propuesto plataformas Unity-PLC para fabricación inteligente, donde Unity actúa como interfaz de monitorización y control de un sistema físico gobernado por PLC Siemens mediante comunicación TCP/IP y S7Net (Wang et al., 2024).

Este trabajo describe el diseño de un gemelo digital desarrollado en Unity de una maqueta controlada por un PLC. La contribución principal no se limita al modelo 3D, sino a la arquitectura modular implementada en Unity para organizar entradas, salidas, sensores, actuadores e interfaces HMI de forma escalable. La comunicación con el PLC se realiza mediante S7.NetPlus (S7NetPlus, 2026), permitiendo tanto simulación como sincronización con planta física.

El resto del artículo se organiza como sigue. La Sección 2 describe la maqueta física empleada como banco de pruebas. La Sección 3 presenta la aplicación desarrollada, incluyendo el modelo en Unity, la arquitectura de comunicación y el ciclo de intercambio con el PLC. La Sección 4 recoge las prácticas docentes planteadas sobre el gemelo digital. Finalmente, las Secciones 5 y 6 discuten el alcance de la propuesta y resumen las conclusiones del trabajo.

2. Sistema físico

El sistema físico utilizado es una maqueta didáctica Fischertechnik (Fischertechnik, 2026) de 24 V compuesta por una cinta transportadora y una troqueladora. La cinta incluye sensores ópticos en ambos extremos y dos señales de actuación para controlar el sentido de movimiento. La troqueladora incorpora un mecanismo vertical con dos finales de carrera, superior e inferior, y dos señales de actuación para los movimientos de subida y bajada. La Figura 1 muestra la maqueta utilizada como referencia física.

El control se realiza con un PLC Siemens S7-1200 CPU 1211C AC/DC/RLY, conectado a la maqueta mediante entradas y salidas digitales. En la configuración empleada, las entradas físicas I0.0 e I0.1 recogen los sensores de la cinta, mientras que I0.2 e I0.3 corresponden a los finales de carrera de la troqueladora. Las salidas Q0.0 y Q0.1 gobiernan el movimiento de la cinta, y Q0.2 y Q0.3 controlan los movimientos de la troqueladora. La Tabla 1 resume el mapeo utilizado entre señales lógicas, direcciones virtuales y direcciones físicas.

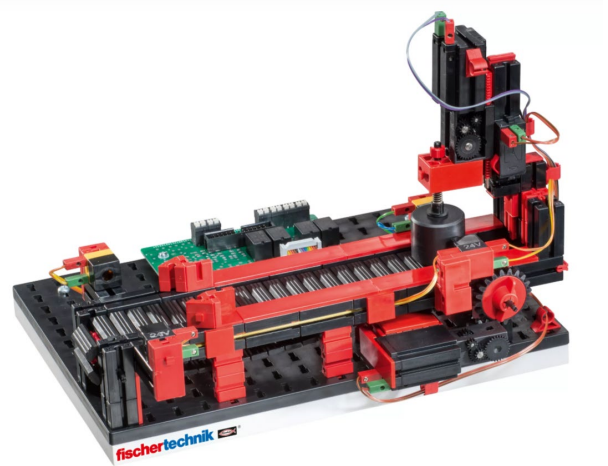


Figura 1: Maqueta Fischertechnik de troqueladora con cinta transportadora.

Tabla 1: Mapa de señales principales.

Señal	Tipo	Virt./fís.	Función
SensorR	E	I3.0/I0.0	Sensor derecho de la cinta
SensorL	E	I3.1/I0.1	Sensor izquierdo de la cinta
FinalDown	E	I3.2/I0.2	Final inferior de troqueladora
FinalUp	E	I3.3/I0.3	Final superior de troqueladora
MotorR	S	Q2.0/Q0.0	Cinta hacia derecha
MotorL	S	Q2.1/Q0.1	Cinta hacia izquierda
TroquelUp	S	Q2.2/Q0.2	Subida de troqueladora
TroquelDown	S	Q2.3/Q0.3	Bajada de troqueladora

3. Aplicación desarrollada

El gemelo digital se ha construido en Unity combinando modelos 3D, físicas de objetos y scripts de control. La cinta transportadora se representa mediante una estructura fija y una banda móvil. El movimiento se resuelve con dos mecanismos complementarios: desplazamiento visual de la textura de la banda y arrastre físico de las cajas mediante el motor de físicas. Esta solución permite que el usuario perciba el movimiento continuo de la cinta y que los objetos transportados interactúen con sensores y zonas de detección.



Figura 2: Modelo 3D de la cinta transportadora con troqueladora integrado en Unity.

La aplicación incorpora una HMI principal con botones de arranque, parada, emergencia, generación de cajas e indicadores de estado. También se incluye una HMI de supervisor desde la que se pueden accionar directamente motores físicos de la cinta y de la troqueladora. La arquitectura permite tres modos de funcionamiento. En modo simulación, Unity envía al PLC las entradas virtuales generadas por el modelo y aplica al gemelo las salidas calculadas por el programa del autómat

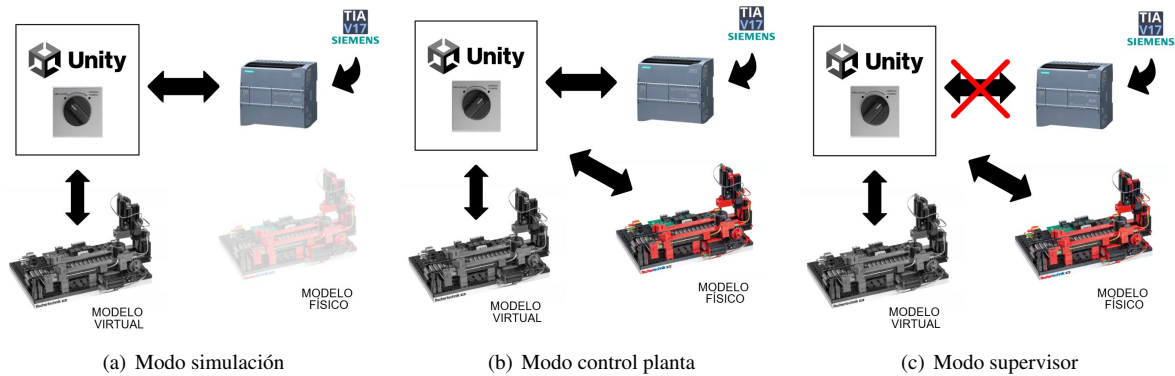


Figura 3: Modos de funcionamiento del gemelo digital.

(Figura 3a). En modo control planta, Unity sincroniza sensores y actuadores reales con el modelo virtual (Figura 3b). En modo supervisor, Unity fuerza directamente las salidas físicas del PLC para tareas de comprobación y mantenimiento (Figura 3c).

3.1. Arquitectura de comunicación en Unity

La arquitectura de comunicación se ha diseñado para evitar una dependencia rígida entre cada objeto de la escena y una dirección concreta del PLC. Para ello, los elementos funcionales se conectan a una interfaz común de entradas y salidas, y un módulo central se encarga de traducir esas señales a direcciones del autómatas. Esta organización por capas separa la detección y actuación en Unity, el estado lógico común de entradas/salidas y el intercambio de variables con el PLC.

El bloque *InterfazPLC* define listas de puntos lógicos de entrada y salida. Cada punto contiene un nombre, un tipo y un valor booleano. Sobre esta estructura se conectan los componentes de Unity: *EntradaUI* convierte botones y conmutadores de la HMI en señales digitales; *SalidaUI* actualiza LEDs, textos o imágenes a partir de salidas del PLC; *Sensor* y *BarrierSensor* transforman detecciones físicas del entorno 3D en entradas lógicas; y *Actuador* y *Motor* aplican las salidas recibidas a movimientos o cambios de estado en el modelo virtual.

3.1.1. Sensores y actuadores

Los sensores virtuales se han dividido en dos responsabilidades. El componente *BarrierSensor* se coloca sobre una zona de detección de Unity configurada como *trigger*; cuando una caja entra o sale de esa zona, el script notifica al componente *Sensor*. Este segundo componente aplica la lógica normalmente abierta o normalmente cerrada y actualiza el punto de entrada asociado en *InterfazPLC*. La separación permite reutilizar el mismo sensor lógico con distintas geometrías de detección: barreras ópticas de la cinta, finales de carrera de la troqueladora u otros dispositivos discretos.

La salida sigue un criterio análogo. *Actuador* lee el valor de un punto de salida y encapsula el cambio de estado. *Motor* se vincula al actuador y transforma esa orden en movimiento o en el comportamiento gráfico correspondiente. Así, la lógica de comunicación no depende de si la salida mueve una cinta, sube la troqueladora, cambia una textura o activa una señal

visual. La Figura 4 resume la secuencia de scripts que intervienen desde los elementos de la escena y la HMI hasta el módulo de intercambio con el PLC.

3.1.2. Interfaz común y módulo de intercambio

El papel de *InterfazPLC* es actuar como conexión entre la escena de Unity y la comunicación industrial. Cada elemento de la maqueta virtual se enlaza a puntos con nombre, no directamente a direcciones de memoria. Esto hace que la escena sea más legible y permite que el mismo objeto se use en modo simulación, planta o supervisor cambiando únicamente el mapeo de direcciones.

El módulo *PLCExchange* centraliza la conexión con el PLC. Este componente mantiene una lista de interfaces, asigna a cada punto una dirección virtual y, cuando procede, una dirección física. Durante la ejecución lee y escribe de forma cíclica mediante la librería S7.NetPlus, que permite acceder desde C# a las variables del PLC Siemens. En modo simulación, las entradas del modelo se escriben como entradas virtuales y las salidas virtuales gobiernan el gemelo. En modo planta, las entradas físicas se copian a las virtuales y las salidas virtuales se reflejan tanto en Unity como en las salidas físicas. En modo supervisor, la HMI de Unity gobierna directamente las salidas físicas y lee las entradas reales.

3.1.3. Ciclo de lectura y escritura

El ciclo de intercambio se ejecuta en cada actualización de Unity siempre que exista conexión con el PLC. Primero se recorren las interfaces registradas en *PLCExchange*. Para cada punto lógico se obtiene su tipo, su dirección virtual y, si está definida, su dirección física. La Figura 5 resume la relación entre el ciclo de actualización de Unity y el ciclo de scan del PLC. Después, el sentido del intercambio depende del modo activo:

1. En modo simulación, los sensores y entradas HMI de Unity actualizan los puntos de entrada de *InterfazPLC*; *PLCExchange* escribe esos valores en las entradas virtuales del PLC y lee las salidas virtuales calculadas por el programa de control para actualizar actuadores, motores e indicadores.
2. En modo control de planta, *PLCExchange* lee las entradas físicas del PLC y las copia a sus entradas virtuales equivalentes. El programa del PLC trabaja sobre esas

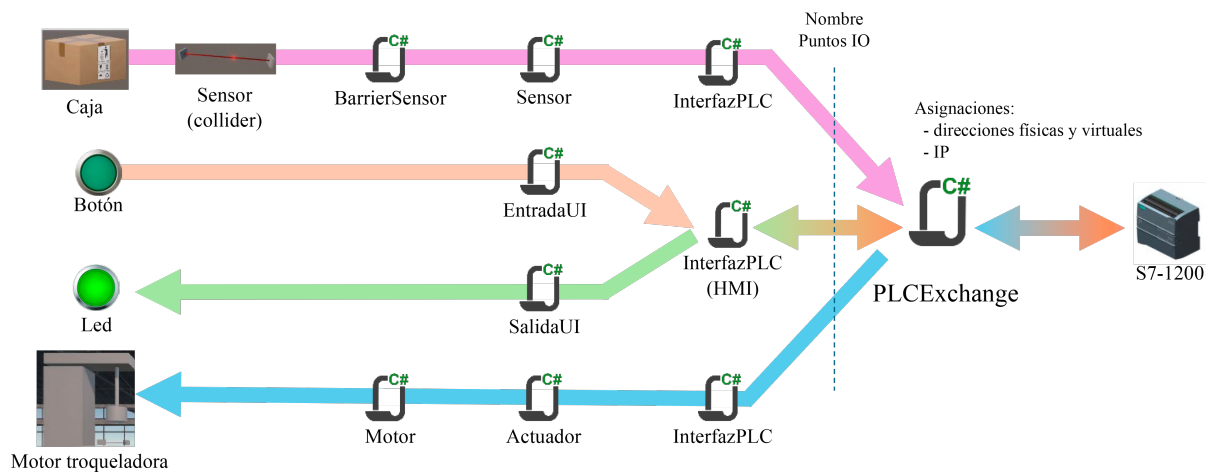


Figura 4: Secuencia de scripts que conectan sensores, entradas HMI, salidas visuales y actuadores con la interfaz lógica y el módulo de intercambio Unity-PLC.

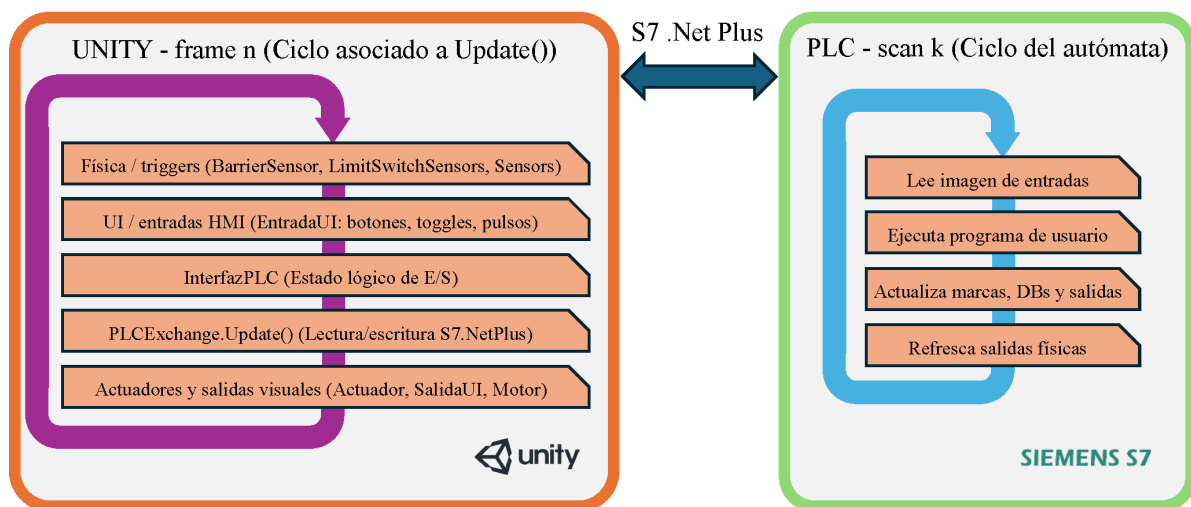


Figura 5: Relación entre el ciclo de actualización de Unity y el ciclo de scan del PLC. Ambos ciclos se ejecutan de forma independiente y se acoplan mediante la lectura y escritura periódica de variables a través de S7.NetPlus.

entradas virtuales y sus salidas virtuales se aplican simultáneamente al gemelo digital y, cuando existe mapeo físico, a las salidas reales de la maqueta.

- En modo supervisor, solo las interfaces HMI marcadas como supervisor actúan sobre direcciones físicas. Las entradas de esta HMI escriben salidas físicas y sus salidas visuales leen entradas reales, dejando al margen la lógica del programa de control para tareas puntuales de comprobación.

Para evitar escrituras redundantes, el módulo mantiene una caché del último valor enviado a cada dirección y solo ejecuta una escritura cuando el valor cambia. Esta decisión reduce tráfico de comunicación y simplifica el seguimiento de eventos discretos, aunque en este trabajo no se ha realizado una caracterización temporal intensiva de latencias o tasas de fallo.

3.1.4. Gestión de modos y seguridad

Los tres modos de funcionamiento se consideran mutuamente excluyentes desde el punto de vista del mando. El modo supervisor tiene prioridad sobre el modo planta y debe emplearse únicamente como herramienta de prueba sobre la ma-

queta didáctica. Para evitar órdenes incompatibles, la arquitectura separa la HMI de supervisor de la HMI normal y permite bloquear el cambio de modo planta cuando el supervisor está activo. Además, la lógica de aplicación debe incluir enclavamientos básicos, como impedir la activación simultánea de motores opuestos, respetar finales de carrera de la troqueladora y mantener operativa la señal de parada o emergencia. Estas medidas son suficientes para el banco docente desarrollado, pero no sustituyen los requisitos de seguridad industrial que serían necesarios en una planta real.

La escalabilidad se refuerza mediante scripts personalizados del editor de Unity. Estos editores filtran los puntos disponibles según el tipo de componente, generan mapeos desde cada *InterfazPLC* y limitan la selección de byte y bit. De este modo, la incorporación de un nuevo sensor, actuador o elemento HMI se realiza añadiendo un punto a la interfaz y seleccionándolo en el inspector, sin modificar el núcleo de comunicación.

4. Ejemplos de aplicación

El gemelo digital se ha utilizado para preparar prácticas de la asignatura de Automatización. El flujo de trabajo propuesto

comienza con el diseño de una propuesta de Grafset que satisfaga la especificación y su implementación en TIA Portal. A continuación, el estudiante valida el programa en modo simulación, usando las cajas y sensores virtuales de Unity. Una vez comprobado el comportamiento, se pasa al modo control planta para ejecutar la misma lógica sobre la maqueta real. La Tabla 2 resume los enunciados de las prácticas propuestas.

La Práctica 1 plantea el arranque y parada de la cinta mediante los botones de la HMI. La validación en Unity permite observar inmediatamente si los botones *Start* y *Stop*, los LEDs y el movimiento de la cinta responden de forma coherente con la propuesta programada.

En la Práctica 2 se solicita contar las cajas que atraviesan la cinta, seleccionando un contador ascendente y definiendo correctamente el flanco de detección para evitar doble conteo. La simulación permite generar cajas desde ambos extremos, repetir el ensayo rápidamente y comprobar la correspondencia entre la detección virtual, la entrada enviada al PLC y la actualización de indicadores.

La Práctica 3 tiene como objetivo medir el tiempo que tarda una caja en recorrer la distancia entre dos sensores de la cinta. Para ello se emplea un temporizador con retención y se condiciona su activación al estado de los sensores.

La Práctica 4 amplía la lógica de arranque y parada mediante una subrutina de mando e incorpora una secuencia de emergencia. El estudiante debe distinguir entre parada normal, parada por emergencia y rearme del sistema, trabajando con flancos y estados persistentes.

La Práctica 5 incorpora la troqueladora y exige coordinar el avance de cajas con el movimiento vertical del útil. El objetivo es evitar atascos mediante la intervención del operador y la lectura de finales de carrera. Esta práctica explota la arquitectura modular completa: sensores de barrera, finales de carrera virtuales, etc.

En conjunto, las prácticas cubren una progresión de dificultad que permite que el estudiante trabaje con los mismos conceptos de automatización que en una instalación física, pero con una etapa previa de depuración visual e interactiva. La Figura 6 recoge propuestas de Grafset empleadas como referencia en las prácticas.

5. Discusión

El trabajo presentado debe interpretarse como una propuesta arquitectónica y una comprobación funcional de prototipo. La comunicación bidireccional entre Unity y el PLC se ha probado durante el desarrollo sobre la maqueta real, verificando el intercambio de señales, la coherencia de los tres modos de operación y la posibilidad de añadir nuevos elementos mediante puntos lógicos de entrada/salida. Sin embargo, no se ha realizado una caracterización temporal intensiva; por tanto, no se aportan métricas de latencia, tasa de fallo o desviación entre eventos físicos y virtuales.

La comunicación directa mediante S7.NetPlus resulta adecuada porque permite una integración ligera desde C#. No obstante, otras alternativas ofrecen ventajas distintas según el objetivo de uso. La Tabla 3 resume la posición de la propuesta frente a herramientas habituales en automatización docente e

industrial. La comparación no pretende establecer una superioridad general, sino delimitar el alcance del prototipo desarrollado.

Tabla 3: Comparación cualitativa con alternativas.

Alternativa	Ventaja	Limitación o papel
S7.NetPlus	Acceso directo desde C# a PLC Siemens	Ligera, pero ligada a Siemens
OPC UA PLCSIM Adv.	Interoperabilidad estándar Simulación avanzada del PLC	Requiere más infraestructura No modela la planta 3D
Factory I/O Game4Automation	Escenas docentes listas Integración Unity-industria	Menor control interno Framework específico

Desde el punto de vista docente, el sistema se ha empleado como demostrador y se ha probado de forma preliminar con un grupo muy reducido de estudiantes para detectar problemas de uso y valorar su aplicabilidad en prácticas. Estas pruebas no constituyen una evaluación docente formal. Como siguiente paso, sería necesario realizar una evaluación sistemática con grupos completos, comparando tiempos de depuración, errores típicos en la programación del PLC, autonomía del alumno y transferencia al laboratorio físico.

6. Conclusiones

Se ha implementado un gemelo digital industrial en Unity conectado a un PLC Siemens S7-1200 mediante S7.NetPlus. El resultado principal es una arquitectura modular de comunicación que separa la definición de señales, los elementos funcionales del gemelo y el intercambio con el PLC. Esta estructura permite añadir sensores, actuadores e interfaces HMI mediante puntos lógicos de entrada/salida y facilita el cambio entre simulación, control sincronizado de planta y supervisión directa.

La propuesta se ha comprobado funcionalmente durante el desarrollo sobre una maqueta Fischertechnik de cinta transportadora con troqueladora. Estas pruebas permitieron verificar la comunicación bidireccional, el mapeo entre señales virtuales y físicas y el comportamiento básico de los tres modos de operación. No obstante, no se ha realizado todavía una campaña experimental intensiva que cuantifique latencias, tasas de fallo o desviaciones temporales entre planta y simulación.

Desde el punto de vista docente, el entorno permite probar programas de TIA Portal antes de actuar sobre la maqueta real y ofrece una transición gradual entre simulación y laboratorio. Como trabajos futuros se plantea medir cuantitativamente la sincronización entre maqueta y gemelo digital, estudiar la interoperabilidad mediante OPC UA, MQTT u otros protocolos estándar, ampliar la biblioteca de componentes reutilizables y evaluar el impacto docente del sistema con grupos completos de estudiantes.

Agradecimientos

Este trabajo ha sido financiado por la Consejería de Educación, Ciencia y Formación Profesional de la Junta de Extremadura a través de la Ayuda a Grupos de Investigación GR24059, cofinanciada por Fondos FEDER.

Tabla 2: Enunciados de las prácticas propuestas para el gemelo digital de la cinta transportadora.

Práctica 1 Al pulsar START, el sistema se activa y se señala mediante el LED ON. Si se detecta un objeto en uno de los sensores de la cinta, esta se pone en marcha en el sentido adecuado para trasladarlo hasta el extremo opuesto. El sentido de movimiento no puede modificarse hasta que el objeto abandone la cinta. Una vez completado el transporte, el sistema queda preparado para detectar un nuevo objeto desde cualquiera de los dos extremos. Si se pulsa STOP, el sistema se detiene y será necesario pulsar START para reiniciarlo.	Práctica 2 Al pulsar START, el sistema se activa y la cinta comienza a desplazarse en sentido forward, señalizándose el funcionamiento mediante el LED ON. Un contador registra las cajas que llegan al extremo opuesto y detiene la cinta al alcanzar 10 unidades, momento en el que el operario realiza el empaquetado. Durante esta intervención, el LED OP parpadea con un periodo de 1 s encendido y 1 s apagado. Una vez finalizado el empaquetado, el operario pulsa OP para reiniciar el contador y reanudar la marcha de la cinta. Si se pulsa STOP, el sistema se detiene; al pulsar de nuevo START, la cinta reanuda el funcionamiento conservando el valor del contador.	Práctica 3 Se desea medir el tiempo que tarda una pieza en recorrer la cinta en ambos sentidos. En la etapa inicial se ponen a cero los temporizadores. Al pulsar START, la cinta arranca en sentido backward. El primer temporizador comienza al detectar la pieza en el Sensor 1 y se detiene al llegar al Sensor 2 y se inicia un segundo temporizador. Cuando la pieza vuelve a pasar por el Sensor 2, se detiene el segundo temporizador, la cinta se para y ambos tiempos quedan almacenados hasta que se pulsa STOP y se regresa a la etapa inicial.	Práctica 4 Al pulsar START, el sistema se activa y se indica mediante el LED ON. Cuando se detecta un objeto en cualquiera de los extremos de la cinta, esta se mueve en el sentido necesario para trasladarlo hasta el sensor del extremo opuesto, deteniéndose al alcanzarlo. Si se pulsa STOP mientras la cinta está en movimiento, el transporte continúa hasta que el objeto llega al final y, a continuación, el sistema se detiene; si la cinta estaba parada, el sistema se apaga inmediatamente. Además, se incorpora una seta de emergencia: si se activa durante el funcionamiento, la cinta se detiene de forma inmediata, se enciende el LED EM y el LED OP parpadea para indicar que se requiere intervención. Tras resolver la emergencia, el operador desactiva la seta y pulsa OP para reanudar el funcionamiento normal.	Práctica 5 Al pulsar START, el sistema se activa y, como medida de seguridad, la máquina asciende hasta su posición superior. Una vez alcanzada dicha posición, si se detecta una caja en el sensor derecho (SR), se habilita el movimiento de la cinta para trasladarla hacia el extremo izquierdo. Cuando la caja llega al sensor izquierdo (SL), la máquina descende hasta su final de carrera inferior y vuelve a subir. A continuación, la cinta se detiene y el LED OP parpadea para indicar que el operador debe retirar la caja. Tras la retirada, el operador pulsa OP y el sistema vuelve al estado inicial de espera, preparado para detectar una nueva caja en SR y repetir el ciclo.
--	---	--	---	--

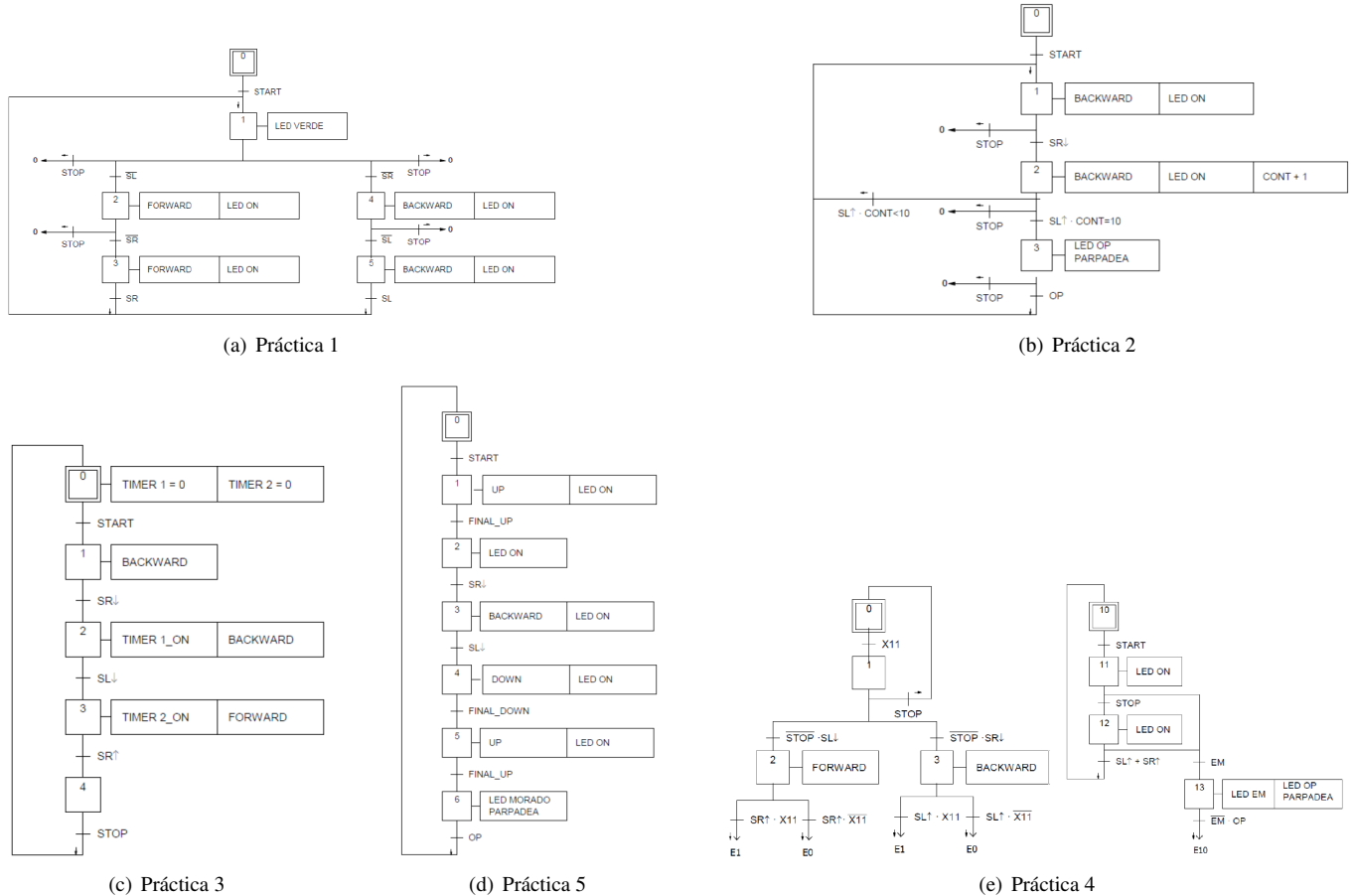


Figura 6: Propuestas de Grafcet empleadas como referencia en las prácticas docentes.

Referencias

- Abele, E., Chrysosolouris, G., Sihni, W., Metternich, J., El Maraghy, H., Seliger, G., Sivard, G., El Maraghy, W., Hummel, V., Tisch, M., Seifermann, S., 2017. Learning factories for future oriented research and education in manufacturing. *CIRP Annals* 66 (2), 803–826.
DOI: 10.1016/j.cirp.2017.05.005
- Balla, M., Haffner, O., Kučera, E., Cigánek, J., 2023. Educational case studies: Creating a digital twin of the production line in tia portal, unity, and game4automation framework. *Sensors* 23 (10), 4977.
DOI: 10.3390/s23104977
- Fischertechnik, 2026. Training models for automation. <https://www.fischertechnik.de/>, accedido: 2026-06-06.

- Kritzinger, W., Karner, M., Traar, G., Henjes, J., Sihni, W., 2018. Digital twin in manufacturing: A categorical literature review and classification. *IFAC-PapersOnLine* 51 (11), 1016–1022.
DOI: 10.1016/j.ifacol.2018.08.474
- Masaev, S., Minkin, A., Troyak, E., Khrulkevich, A., 2024. Digital twin in advanced training of engineering specialists. *arXiv:2412.13429*.
URL: <https://arxiv.org/abs/2412.13429>
- S7NetPlus, 2026. S7.net plus. <https://github.com/S7NetPlus/s7netplus>, accedido: 2026-06-06.
- Wang, H., Yang, Z., Zhang, Q., Sun, Q., Lim, E., 2024. A digital twin platform integrating process parameter simulation solution for intelligent manufacturing. *Electronics* 13 (4), 802.
DOI: 10.3390/electronics13040802