

# Práctica

## Programación PLC 1

Introducción a la aplicación de simulación de planta Factory IO



### Recursos utilizados

Aplicación Factory IO

TIA-Portal

PLCSim

### Automatización y Control

Ángel Gaspar González Rodríguez



UNIVERSIDAD DE JAÉN

## 1. Objetivos y conceptos fundamentales

Factory IO es una aplicación para la simulación de procesos industriales en 3D, diseñado para interactuar con las entradas y salidas de un PLC. De esta manera, permite visualizar las acciones a que daría lugar la secuencia de instrucciones de un código en el proceso (virtual en este caso) que queremos controlar y, al mismo tiempo, recibir las entradas esperadas provenientes de los sensores del proceso que se verían afectados.

Por ejemplo, podemos simular el funcionamiento de una cinta transportadora que dispone de sensores de fin de carrera. En este caso, una salida proveniente de un PLC real, o bien de un simulador de PLCs, se enlazaría con Factory IO y la activación de dicha salida provocaría el movimiento de los rodillos de la cinta, que sería visualizado en nuestra pantalla mediante el desplazamiento de la caja. Unos instantes más tarde se activaría un sensor de fin de carrera en el proceso simulado y esto provocaría la activación de una determinada entrada en nuestro PLC.



En general, este tipo de aplicaciones, encuadradas en la Industria 4.0 bajo el nombre de Gemelos digitales, permiten la comunicación con PLCs reales y simulados de diferentes firmas: Siemens, Allen-Bradely, CodeSys... En realidad, un gemelo digital debería completarse con la realimentación de información procedente del sistema real de manera que el gemelo digital se actualice con dicha información.

En esta práctica se persigue aprender a diseñar una estación sencilla y enlazar sus entradas y salidas con las de un PLC simulado mediante PLCSim. Simultáneamente se crearía un programa de control sencillo residente en el PLC que controle el funcionamiento de dicha estación.

El control de una estación simulada en Factory IO involucra los siguientes pasos:

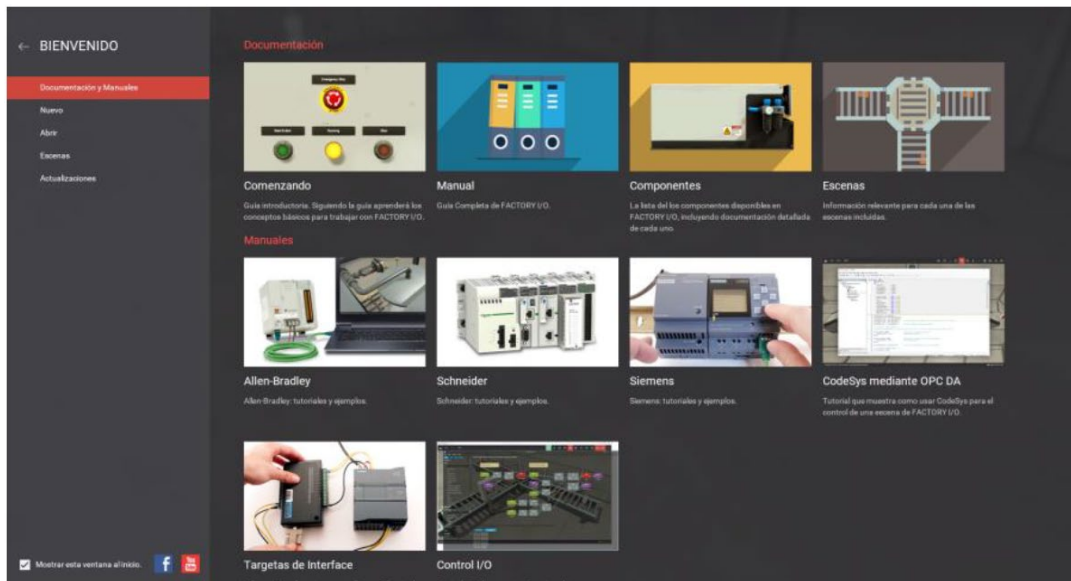
1. Diseño de la estación virtual que simula la estación real a controlar
2. Codificación del programa de control en el entorno de programación (en nuestro caso TIA Portal) y carga en el dispositivo de control (PLC real o simulado)
3. Enlazado de las entradas de Factory IO (por ejemplo actuación sobre cintas transportadoras) con las variables de salida definidas para nuestro PLC. Análogamente, enlazado de las salidas de Factory IO (por ejemplo, sensores de fin de carrera) con las variables de entrada definidas para nuestro PLC.

# 1 Abriendo una escena de ejemplo

## 1.1 Selección de escena

Posiblemente la manera más rápida de comenzar a explorar las posibilidades de Factory IO sea abriendo alguno de los proyectos (también llamadas escenas) de ejemplo. A partir de ahí, el usuario puede realizar modificaciones para comprobar las distintas funcionalidades de cada elemento soporte, de cada entrada o de cada salida.

La parte superior izquierda de la pantalla de inicio contiene un menú con las principales opciones de partida: manuales de ayuda, creación de nuevo proyecto, apertura de uno existente, o apertura de una escena prediseñada a modo de plantilla.



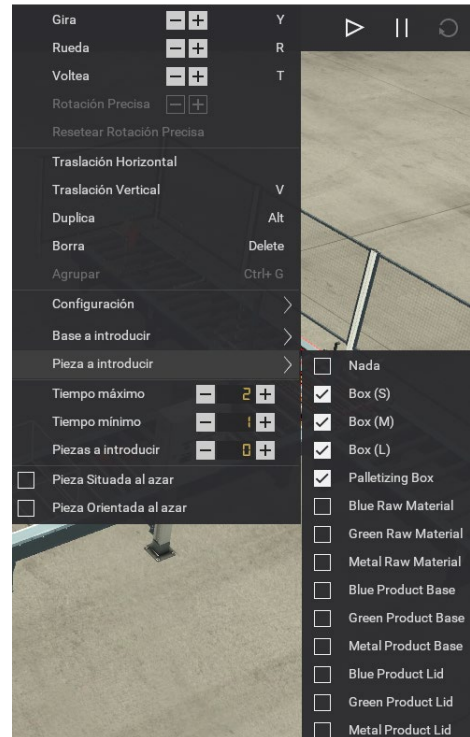
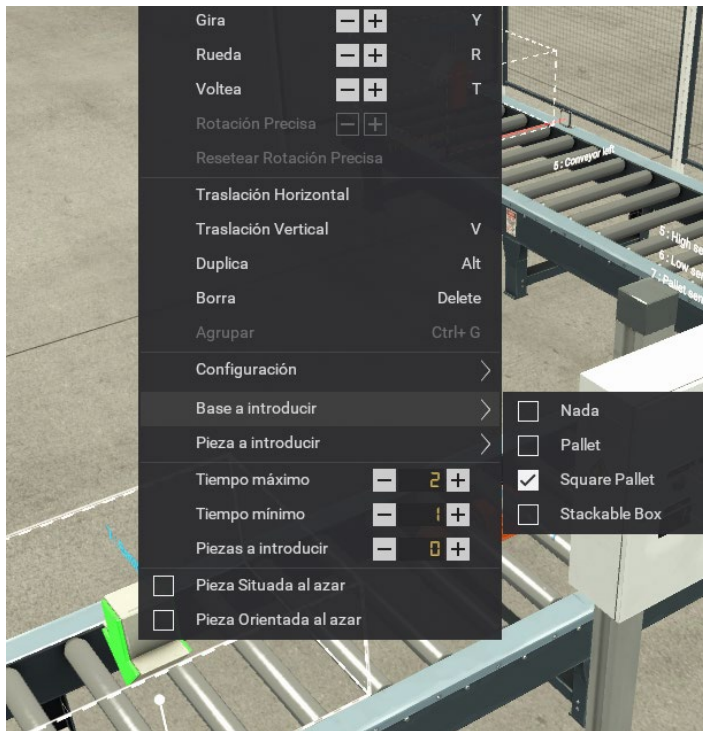
En nuestro caso, se seleccionaría Escenas, para abrir una escena de ejemplo grabada durante la instalación del paquete Factory IO. De entre ellas, se seleccionará la [Estación de selección por altura](#) (modo básico).



## 1.2 Elementos de la escena de selección por altura

El escenario genérico es una nave industrial en la que se ubican los elementos a simular. Para cada escena en concreto, se dispondrán los elementos de que se compone la estación. En este

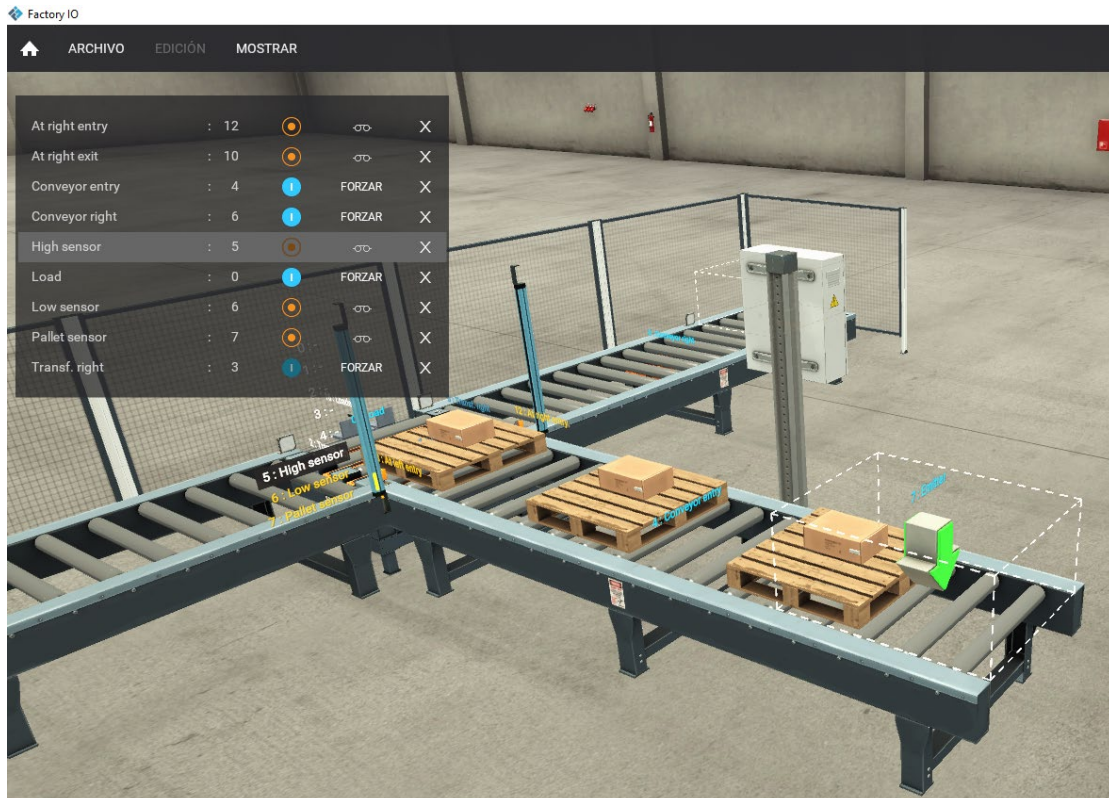
caso, se tiene una cinta (conveyor) de alimentación, que está asociado a un emisor o alimentador de piezas. Este elemento es configurable, y podemos elegir el tipo de piezas que se van sucediendo, y la forma en que aparecen. En esta escena, tal como aparecen en el ejemplo cargado, los elementos aparecen periódicamente cuando la cinta está en movimiento sobre un palé de madera. Los elementos que aparecen son de tres tamaños seleccionados de forma aleatoria.



Existen botones en la parte superior derecha que permiten visualizar las etiquetas de los distintos elementos de salida (actuadores y visualizadores) y entrada (sensores y botones de mando). Pulsando sobre dichos botones del menú, se visualizan las etiquetas que identifican dichas salidas y entradas. Pulsando sobre dichas etiquetas se incorpora dicha salida o entrada a una pequeña ventana que permite visualizar su estado, o incluso forzarlo. Si, por ejemplo, se fuerza a 1 la salida “Conveyor entry”, se activaría manualmente la cinta transportadora de alimentación de piezas.



Lo mismo se puede hacer con el resto de salidas (actuadores), con lo que podríamos visualizar la dinámica de la planta. Simultáneamente, se pueden observar cómo los sensores implicados van activándose y desactivándose a medida que los palés o las cajas van pasando por los sensores. Concretamente se puede observar que los sensores “At right entry” o “At right exit” están por defecto activos, y solo se desactivarán cuando pase un objeto por delante de ellos.



En general, la tarea de diseño pasaría por incluir y conectar los elementos necesarios para implementar la estación a simular, configurándolos de acuerdo a los requerimientos específicos de nuestra estación. En este proyecto de prueba, simplemente se mantendrán estos componentes tal como aparecen en la escena precargada.

Nota: las siguientes explicaciones suponen que la práctica se realizará con PLCSim. Si se opta por utilizar un PLC real, revisar preferentemente el documento “Enlace FIO Tia Portal.pdf”.

## 2 Codificación del programa de control en el entorno de programación

### 2.1 Listado de variables de entrada y salida a enlazar con el PLC

La siguiente tarea sería el planteamiento y programación del código que lleve a cabo las tareas encomendadas a la estación. Para esto, se utilizarán (algunas de) las variables de entrada/salida que entran en juego en la escena con la estación a simular. A este conjunto de variables se accede seleccionando Archivo > Driver > Siemens S7-PLCSim

High sensor	I0.0	Q0.0	Conveyor entry
Low sensor	I0.1	Q0.1	Load
Pallet sensor	I0.2	Q0.2	Unload
Loaded	I0.3	Q0.3	Transf. left
At left entry	I0.4	Q0.4	Transf. right
At left exit	I0.5	Q0.5	Conveyor left
At right entry	I0.6	Q0.6	Conveyor right
At right exit	I0.7	Q0.7	Start light
Start	I1.0	Q1.0	Reset light
Reset	I1.1	Q1.1	Stop light
Stop	I1.2	(INT) QW30	Counter
Emergency stop	I1.3		
Auto	I1.4		
FACTORY I/O (Running)	I1.5		

Para esta primera práctica introductoria, no se tendrá en cuenta el carril izquierdo, ni se aplicará ninguna distinción en función de la altura. El conjunto de entradas y salidas sería

Entradas	Dirección	Salidas	Dirección
Pallet sensor: sensor existe palé	I0.2	Conveyor entry: cinta de alimentación	Q0.0
Loaded: detector de palé cargado	I0.3	Load: actuador de carga de palé	Q0.1
At right entry: sensor de barrera en la entrada de la cinta derecha	I0.6	Transf. Right: desplazador a derecha	Q0.4
At right exit: sensor de barrera en la entrada de la cinta derecha	I0.7	Conveyor right: cinta derecha de salida	Q0.6
Start: botón de mando de inicio	I1.0		

Reset: marcado inicial del Grafcet	I1.1		
------------------------------------	------	--	--

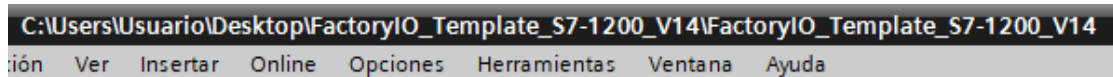
A tener en cuenta:

- Aunque no se hará así en esta práctica, en ocasiones puede ser preferible no utilizar las direcciones que empiezan por el byte 0, y en su lugar se utilizarían direcciones que empiecen por 10. De esta forma, se dejan libres las direcciones de entrada y salida con las que trabajamos por defecto cuando utilizamos el PLC real (%I0.0, %I0.1... %Q0.0, %Q0.1...).
- Hay que prestar atención al estado de los sensores. Para los detectores de presencia fotoeléctricos, como “at right entry” o “at right exit”, Factory IO suele utilizar sensores fotorreflejos que utilizan catadióptricos. Estos sensores proporcionan 1 cuando no hay presencia de pieza y 0 en caso contrario (**lógica negativa**). Sin embargo, el sensor de presencia para el palé funciona con lógica positiva.

Factory IO consume gran cantidad de recursos, lo que ralentizará la ejecución de otros programas. Se recomienda cerrar Factory IO durante la preparación del proyecto en TIA Portal.

## 2.2 Plantilla de partida

Factory IO proporciona, para cada versión de TIA Portal, un proyecto vacío que realiza la conexión básica entre TIA Portal y Factory IO. Dicho proyecto se llama [FactoryIO\\_Template\\_S7-1200\\_v17](#).



Después de desarchivarlo y abrirlo, es conveniente renombrarlo de acuerdo a las características de nuestra estación o proyecto. Llamarlo, por ejemplo, DobleCinta o ProyectoIntroduccion.

Esta plantilla consta únicamente de una llamada a un bloque, en OB1, de enlace con Factory IO.

En OB1, existe una llamada a una función “MHJ\_PLC\_Lab\_Functions-S71200”, que NO se deberá borrar. A continuación, se añadirá el código correspondiente a la automatización planteada.

## 2.3 Edición de las variables de proyecto

Aunque se pueden utilizar otros nombres de variables, rellenaremos la tabla de variables con los mismos nombres del proyecto de Factory IO, aunque limitándonos a los de la tabla anterior, que serán los que únicamente se utilicen en este proyecto.

## 2.4 Tarea a realizar

El alumno realizará el Grafcet y codificará en [FC\\_SCL](#) la siguiente secuencia de tareas:

0. El sistema está en reposo hasta que se pulse Start. En ese momento se inicia el proceso.
1. Activar la cinta de alimentación ([Conveyor entry](#)) hasta que se active el sensor de palé ([Pallet sensor](#))
2. Seguir activando la cinta de alimentación, junto con la salida [Load](#), hasta que se detecte el palé cargado ([Loaded](#))
3. Activar el desplazador a derecha ([Transf. Right](#)) hasta que el sensor de barrera en la entrada de la cinta derecha ([At right entry](#)) detecte objeto.
4. Seguir activando el desplazador a derecha, junto con la cinta transportadora derecha ([Conveyor right](#)), hasta que el sensor de barrera en la entrada de la cinta derecha deje de detectar objeto.

5. Seguir activando la cinta transportadora derecha hasta que el sensor de barrera en la salida de la cinta derecha ([At right exit](#)) detecte objeto.
6. Seguir activando la cinta transportadora derecha hasta el sensor de barrera en la salida de la cinta derecha deje de detectar objeto.
7. Volver a la etapa 0, donde esperará de nuevo la pulsación de Start.

Notas:

- 1.- Tener en cuenta la lógica negativa de los sensores [At right entry](#) y [At right exit](#) en la codificación de la solución.
- 2.- A fin de no tener que crear el OB100, ni tener que pasar de STOP a RUN para reiniciar el proceso, utilizar el botón Reset tal como se muestra en las primeras líneas del siguiente código

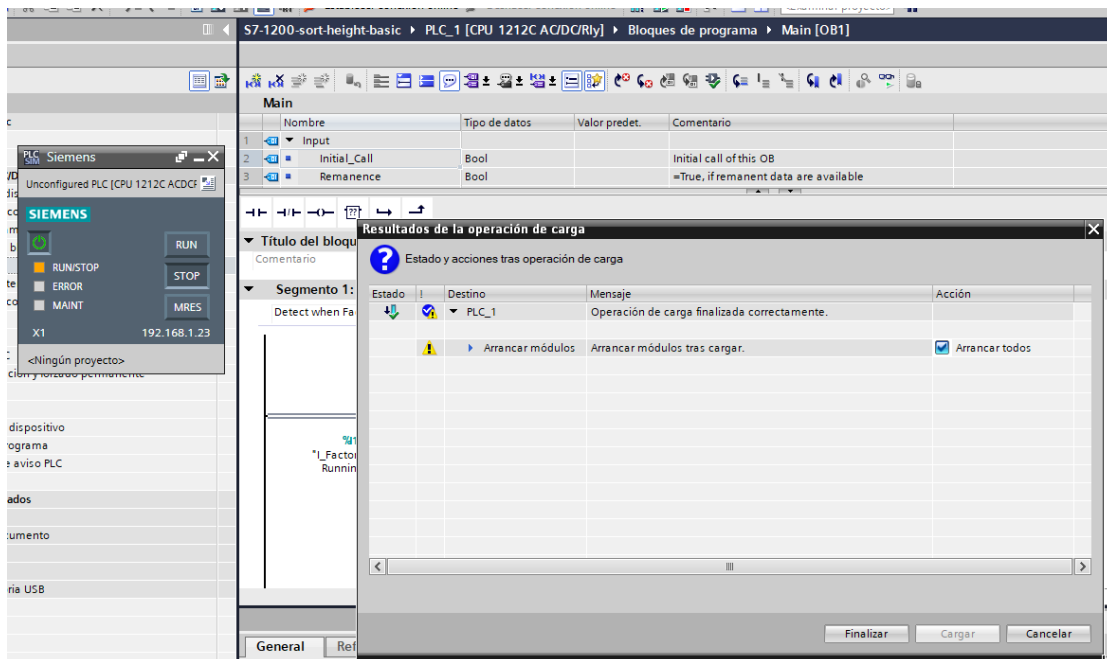
Para codificarlo en SCL, hacerlo de acuerdo a la siguiente estructura

1. IF "Reset" Then
2.     "X0" := 1;
3.     "X1" := "X2" := "X3" := "X4" := "X5" := "X6" := 0;
4. END\_IF;
- 5.
6. // Espera llegada de palé
7. IF "X0" AND "Start" Then
8.     "X0" := 0;
9.     "X1" := 1;
10. END\_IF;
11. --- A continuación el alumno completará el resto de evolución del Grafcet
  
40. // Asignación de salidas
41. "Conveyor entry" := "X1" OR "X2";
42. --- A continuación el alumno completará el resto de asignaciones de salidas

Incluir un segundo Grafcet con dos etapas (XParo y XMarcha), que controle el modo de arranque y parada a través de un pulsador de START normalmente abierto y un pulsador de STOP normalmente cerrado. Sustituir la consulta a START de la línea 7 por una consulta a una de las etapas de este nuevo grafcet. En las líneas 2 y 3 se deberá incluir el marcado inicial de este segundo grafcet (XParo inicialmente activo y XMarcha inicialmente inactivo).

Cargar en PLATEA un fichero con extensión .txt con el código SCL de la función FC\_SCL (no incluir los números de línea). Sólo un fichero por grupo. Como nombre del fichero, poner los primeros apellidos de cada integrante del grupo.

Una vez creada la tabla de variables, y definida la función FC\_SCL con la tarea a realizar, se iniciará el simulador PLC\_Sim, y se cargará el proyecto en él.



## 2.5 Tabla de observación de variables

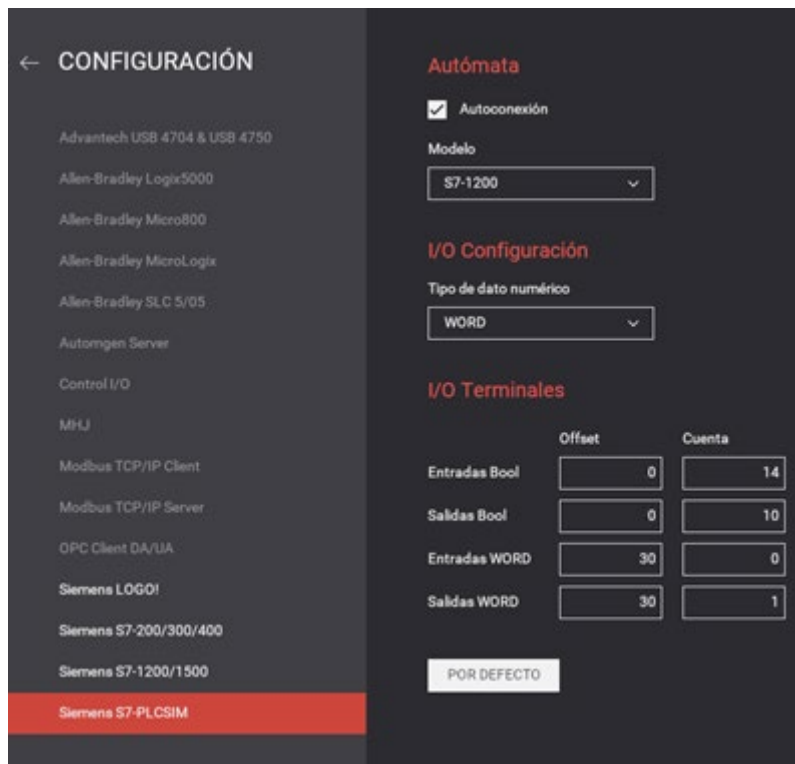
El lenguaje SCL es muy rápido en su edición, potente y cómodo en su exportación. Sin embargo, la depuración no resulta muy intuitiva con las herramientas habituales que se usan para KOP. Para facilitar esta tarea, es casi imprescindible rellenar una tabla de observación de variables con los estados de nuestro Grafcet y con algunas de las entradas y salidas.

	Nombre	Dirección	Formato visualiza..	Valor de observac..
1	"X0"	%M0.0	BOOL	FALSE
2	"X1"	%M0.1	BOOL	FALSE
3	"X2"	%M0.2	BOOL	FALSE
4	"X3"	%M0.3	BOOL	FALSE
5	"X4"	%M0.4	BOOL	FALSE
6	"X5"	%M0.5	BOOL	FALSE
7	"X6"	%M0.6	BOOL	FALSE
8	"Conveyor entry"	%Q 0.0	BOOL	FALSE
9	"Load"	%Q 0.1	BOOL	FALSE
10	"Transf. right"	%Q 0.4	BOOL	FALSE
11	"Conveyor right"	%Q 0.6	BOOL	FALSE
12	"Pallet Sensor"	%I 0.2	BOOL	FALSE
13	"Loaded"	%I 0.3	BOOL	FALSE
14	"At right entry"	%I 0.6	BOOL	TRUE
15	"At right exit"	%I 0.7	BOOL	TRUE

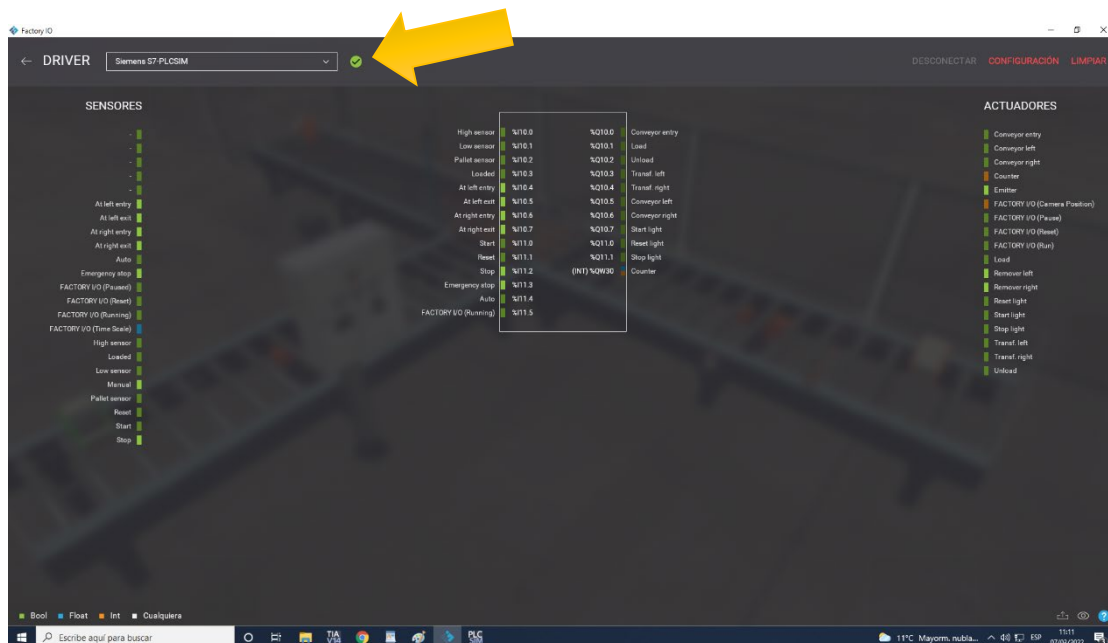
Esta ventana deberá desanclarse del editor de TIA Portal y ajustar la anchura de sus columnas para que pueda visualizarse al mismo tiempo que la simulación de Factory IO. Es muy **importante** realizar este paso para comprobar que el Grafcet va pasando por todas las etapas.

## 3 Conexión entre TIA Portal y Factory IO

Estando PLC Sim iniciado y en modo ejecución (Run), se lanzará Factory IO y se seleccionará Archivo > Driver > Siemens S7-PLCSim. En la ventana que aparece, ajustar la configuración como se muestra a continuación, aunque realmente la única ventana que habrá que modificar será la especificación del modelo **S7-1200**.



Al volver a la ventana anterior (Driver), entrará en un estado de intento de conexión, y pasado un tiempo y si la conexión se ha establecido, aparecerá un tick verde.



Una vez enlazados ambos programas, ya se puede lanzar la simulación en Factory IO y comprobar que el código consiga llevar a cabo la tarea encomendada.

Puede resultar que los palés no sean transferidos de forma fluida. En ese caso, el alumno deberá modificar el código para mejorar el funcionamiento.