



# **TRABAJO DE DIPLOMA**

**En opción al Título de Ingeniera en Automática**

**Autora**

**Talieny Cabrera Méndez**

**Tutor**

**MSc. Michel Sanz Pérez**

**Noviembre, 2023**



**UNIVERSIDAD  
DE ORIENTE**

**Facultad de Ingeniería Eléctrica  
Departamento de Automática**

# **TRABAJO DE DIPLOMA**

**Título**

**Control de procesos industriales simulados en Factory I/O**

**Autora**

**Talieny Cabrera Méndez**

**Tutor**

**MSc. Michel Sanz Pérez**

**Noviembre, 2023**

Hago constar que el presente Trabajo de Diploma fue realizado en la Universidad de Oriente como parte de la culminación de estudios de la especialidad de Ingeniería en Automática, autorizando a que el mismo sea utilizado por la Institución para los fines que estime convenientes, tanto de forma parcial como total, y que además no podrá ser presentado en eventos, ni publicados sin autorización de la Universidad.

---

Nombre y firma del autor

Los abajo firmantes certificamos que el presente trabajo ha sido realizado según acuerdo de la dirección de nuestro centro y el mismo cumple con los requisitos que debe tener un trabajo de esta envergadura referido a la temática señalada.

---

Nombre y firma del autor

---

Nombre y firma del Tutor

---

Fecha

---

Nombre y firma del Jefe de Carrera

---

Fecha

---

Nombre y firma del Jefe de Departamento

---

Fecha

## **Dedicatoria**

“La vida es tan extraña que no vale la pena reducirla a ideas,  
hay que someterla.”

Tomado del libro María Toda.

Dedicado especialmente a mis padres que son la inspiración para mi vida. A mi madre Odalis porque es mi ejemplo de constancia, a mi padre Pablo porque me enseñó el valor de la ingeniería y a no rendirme ante nada.

A los estudiantes de la carrera Ingeniería en Automática para que este trabajo les sirva como herramienta de aprendizaje durante sus estudios, y a quien pueda interesarle para que por este medio enriquezca sus conocimientos y crezca a nivel profesional.

## **Agradecimientos**

Agradezco infinitamente a mis padres porque hicieron posible el transcurso por esta carrera, a mi hermana Tayany por estar presente siempre que la necesité, a mi Disniel David por su tiempo dedicado durante la realización de este proyecto y a mi familia por apoyarme incondicionalmente.

A mi tutor Msc. Michel Sanz Pérez por asistirme en este reto enorme y por todo lo que aprendí de él. A todos aquellos profesores y compañeros que fomentaron mis conocimientos. A mis amistades más cercanas por su incesante preocupación.

## Listado de símbolos, términos especiales y abreviaturas no normalizadas

DAQ	Adquisición de Datos ( <i>Data Acquisition</i> ).
DI	Entradas Digitales ( <i>Digital Input</i> ).
DO	Salidas Digitales ( <i>Digital Output</i> ).
E/S	Entrada/Salida.
FPGA	Matriz de Compuertas Programable (Field Programmable Gate Array).
Grafcet	Gráfico de control con etapas y transiciones.
LD	Diagrama Escalera ( <i>Ladder Diagram</i> ).
LED	Diodo Emisor de Luz ( <i>Light Emitting Diode</i> ).
OPC	Protocolo ( <i>Open Protocol Communication</i> ).
PLC	Controlador Lógico Programable ( <i>Programmable Logic Controller</i> ).
TTL	Lógica Transistor-Transistor ( <i>Transistor-Transistor Logic</i> ).
USB	Bus Universal en Serie ( <i>Universal Serial Bus</i> ).

## Listado de imágenes

Figura 1 Planta piloto para el control de nivel.....	7
Figura 2 Planta piloto de un manipulador neumático. ....	8
Figura 3 Planta piloto para el control de presión. ....	8
Figura 4 Planta piloto para el control de flujo. ....	9
Figura 5 Interfaz PC-SIMU (separado de cajas).....	12
Figura 6 Interfaz FACTORY IO (separado de cajas).....	13
Figura 7 Interfaz Machines Simulator.....	14
Figura 8 Interfaz FESTO fluidSim.....	15
Figura 9 Logo del programa. ....	16
Figura 10 Entorno proporcionado para trabajar.....	17
Figura 11 Elementos. ....	18
Figura 12 Carga pesada.....	19
Figura 13 Carga ligera.....	20
Figura 14 Sensores. ....	21
Figura 15 Operadores ....	21
Figura 16 Estaciones.....	22
Figura 17 Dispositivos de advertencia.....	22
Figura 18 Pasarelas. ....	23
Figura 19 Configuración del protocolo de comunicación utilizado, Modbus TCP/IP Client (ejemplo Estación Buffer). ....	28
Figura 20 Logo del programa SoMachine Basic.....	31
Figura 21 Entorno de trabajo en el software SoMachine Basic. ....	32
Figura 22 Escena para el control de movimiento de objetos. ....	34
Figura 23 Modelado Grafcet del funcionamiento de la escena.....	35
Figura 24 Código LD implementado con: a) bobinas sin retención y b) bobinas con retención (Set y Reset). ....	36
Figura 25 Escena para el control de movimiento de objetos con panel. ....	37
Figura 26 Modelado Grafcet del funcionamiento de la escena.....	38
Figura 27 Escena para el control de estación separadora. ....	39
Figura 28 Modelado Grafcet del funcionamiento de la escena.....	42
Figura 29 Vista superior de la línea de llenado de botellas. ....	43
Figura 30 Escena correspondiente a la Zona1.....	45

Figura 31 Grafcet funcional del control de nivel. ....	46
Figura 32 Escena correspondiente a la Zona 2. ....	46
Figura 33 Cajas empleadas para simular botellas: a) Cajas del transportador 1 b) Cajas del transportador 2. ....	47
Figura 34 Grafcet funcional del llenado. ....	47
Figura 35 Escena correspondiente a la Zona 3. ....	48
Figura 36 Posibles casos según el orden de llegada de las cajas: a) Caso 1 b) Caso 2 y c) Caso 3. ....	50
Figura 37 Grafcet funcional de la estación de convergencia. ....	51

## **Resumen**

La automatización se ha expandido a tales niveles que van desde el sector industrial hasta el desarrollo de laboratorios virtuales para su comprensión especialmente en el sistema educacional. Más allá de los conceptos y principios, los laboratorios virtuales ayudan a los estudiantes a desarrollar habilidades profesionales fundamentales tales como la solución de problemas, el diseño de aplicaciones y la identificación de fallas. La Facultad de Ingeniería Eléctrica de la Universidad de Oriente debido a la escasez de medios para montar procesos automatizados en el laboratorio de Control de Procesos que se empleen en la realización de prácticas, ha optado por implementar un laboratorio virtual usando una herramienta de simulación ya que esta vía ofrece comodidad y ahorro de recursos, de esta manera los estudiantes podrán realizar prácticas asemejándose bastante a la realidad.

Con el presente trabajo se pretende demostrar que el entorno de simulación de procesos industriales Factory I/O es una vía para el estudio y experimentación en temáticas relacionadas con la automatización industrial. Para ello se elaboraron varias escenas con el fin de simular algoritmos de control desde un autómeta.

En un principio se escogieron, diseñaron, y construyeron los modelos 3D de las escenas en Factory I/O, yendo desde procesos sencillos con algoritmos de programación básicos hasta procesos más complejos representando líneas de producción completas. Posteriormente se utilizó el PLC M221 cuyo algoritmo de programación se escribió en el lenguaje *Ladder* en el *software* de programación SoMachine Basic para demostrar la veracidad de las escenas propuestas.

### **Palabras claves:**

Laboratorios virtuales, *software* de simulación, Factory I/O, automatización industrial, control de procesos.

## **Abstract**

*Automation has expanded to such levels that range from the industrial sector to the development of virtual laboratories for its understanding, especially in the educational system. Beyond concepts and principles, virtual labs help students develop critical professional skills such as problem solving, application design, and fault identification. The Faculty of Electrical Engineering of the Universidad de Oriente, due to the scarcity of means to set up automated processes in the Process Control laboratory that are used in carrying out practices, has chosen to implement a virtual laboratory using a simulation tool since This route offers convenience and savings of resources, in this way students will be able to carry out internships that are quite similar to reality.*

*This work aims to demonstrate that the Factory I/O industrial process simulation environment is a way for study and experimentation on topics related to industrial automation. To do this, several scenes were created in order to simulate control algorithms from an automaton.*

*Initially, the 3D models of the scenes were chosen, designed, and built in Factory I/O, ranging from simple processes with basic programming algorithms to more complex processes representing complete production lines. Subsequently, the M221 PLC was used, whose programming algorithm was written in the Ladder language in the SoMachine Basic programming software to demonstrate the veracity of the proposed scenes.*

### **Key-words:**

*Virtual laboratories, simulation software, Factory I/O, industrial automation, process control.*

## ÍNDICE

Listado de símbolos, términos especiales y abreviaturas no normalizadas .....	I
Listado de imágenes .....	II
Resumen.....	IV
<i>Abstract</i> .....	V
INTRODUCCIÓN .....	1
CAPITULO 1. MARCO TEÓRICO REFERENCIAL.....	5
1.1 Prácticas de laboratorio en la enseñanza .....	5
1.2 Instalaciones en el laboratorio de Control de Procesos .....	6
1.3 <i>Software</i> de simulación .....	10
1.3.1 Software de simulación de procesos industriales .....	11
1.4 <i>Software</i> de simulación Factory I/O. ....	15
1.4.1 Elementos disponibles.....	17
1.4.2 Posibilidades de comunicación .....	24
1.5 Controladores lógicos programables.....	29
1.5.1 Autómata programable ModiconM221. ....	30
1.6 Sistema de control SoMachine Basic.....	31
Conclusiones Parciales.....	33
CAPITULO 2. CONTROL DE PROCESOS INDUSTRIALES PREDEFINIDOS Y/O MODIFICADOS EN EL SOFTWARE FACTORY IO.....	34
2.1 Control de movimiento de objetos.....	34
2.1.1 Componentes de la escena .....	34
2.1.2 Modelado y programación del control .....	35
2.2 Control de movimiento de objetos con panel de control.....	36
2.2.1 Componentes de la escena .....	37
2.2.2 Modelado y programación .....	38
2.3 Control de estación separadora. ....	39
2.3.1 Componentes de la escena .....	39

2.3.2 Modelado y programación .....	42
2.4 Control de una línea de llenado de botellas. ....	43
2.4.1 Zona 1: control de nivel en un tanque.....	44
2.4.2 Zona 2: control del llenado de botellas .....	46
2.4.3 Zona 3: control de estación de convergencia .....	48
2.4.4 Programación .....	51
Conclusiones Parciales.....	52
CONCLUSIONES.....	53
RECOMENDACIONES .....	54
BIBLIOGRAFIA .....	55
ANEXOS .....	57

## INTRODUCCIÓN

El uso de sistemas de automatización industrial ha aumentado de forma constante en los últimos años, y por una buena razón. La capacidad de la automatización para procesar, montar, inspeccionar y manipular productos físicos en el proceso de fabricación beneficia a las empresas aumentando su productividad, reduciendo la fatiga y esfuerzo de los trabajadores, así como los plazos de fabricación, mejorando la seguridad del personal y la calidad del producto final, realizando procesos imposibles de controlar manualmente y aumentando la eficiencia [1].

La automatización industrial es en sí misma la aplicación de maquinarias y equipos de tecnología (*hardware* y *software*), que sean capaces de llevar todo tipo de procesos de producción, selección y control de manera autónoma, es decir, con la mínima participación de personal humano posible [2].

Usualmente los sistemas productivos se componen de una parte operativa que actúa físicamente sobre la planta y otra de mando o control encargada de realizar las órdenes programadas por el operador, y que tradicionalmente se puede implementar mediante tecnología cableada o programada.

Dentro de la tecnología programada se encuentra el controlador lógico programable (PLC) o autómeta programable como popularmente también se le conoce, que no es más que un equipo electrónico programable en lenguaje no informático diseñado para controlar sistemas secuenciales [3], cuyas características especiales y su constante evolución de *hardware* y *software* lo han convertido en el medio por excelencia para la automatización industrial a nivel global.

En el campo de la educación en ingeniería, Cuba, en la especialidad de Automática concibe un plan de estudios donde asignaturas como Medios Técnicos de Automatización, Autómatas Programables y Sistemas de Automatización son esenciales para la formación del estudiante en el trabajo con estos equipos con el objetivo de incrementar la automatización de la empresa estatal socialista, principal fuente de riqueza, bienestar y prosperidad del pueblo. El Departamento de Automática de la Universidad de Oriente en Santiago de Cuba reconoce que la ciencia es una actividad eminentemente práctica, además de teórica, lo cual hace que en su enseñanza el laboratorio sea un elemento

indispensable. Sin embargo, a pesar de su papel relevante, en la actualidad los laboratorios relacionados a estas asignaturas, especialmente en lo que a controladores lógicos programables se refiere en instalaciones físicas se han visto afectados debido a la ruptura de equipos, falta de mantenimiento, deterioro debido al uso prolongado y la obsolescencia de algunos componentes.

En búsqueda de alternativas se ha recurrido al empleo de simuladores de procesos industriales que a pesar de ser funcionales aún presentan limitaciones en cuanto a la conectividad con *software* de diferentes familias de controladores. De igual manera las visualizaciones obtenidas principalmente en dos dimensiones carecen de realismo y resulta escaso el número de componentes, lo que en ocasiones desestimula al uso de los mismos por parte de los estudiantes.

Por todo lo anteriormente expuesto el **problema de la investigación** del presente trabajo lo constituyen las limitaciones existentes en el uso de procesos físicos y/o simulados en el desarrollo de prácticas dirigidas a la programación de controladores lógicos programables en la carrera de Ingeniería Automática de la Universidad de Oriente.

Por lo que se define como **objeto de estudio** los entornos virtuales de simulación de procesos industriales compatibles con *software* de programación y simulación de controladores lógicos programables, y como **campo de acción** la simulación de algoritmos de control en controladores lógicos programables para procesos industriales en entornos virtuales.

De esta forma se propone como **objetivo general** el desarrollo de algoritmos de control para distintos escenarios de procesos industriales dentro del entorno virtual Factory I/O utilizando el *software* SoMachine Basic de Schneider.

Como **idea a defender** se plantea que el desarrollo de algoritmos de control para escenas de procesos industriales sobre el entorno virtual Factory I/O ofrece potencialidades para el desarrollo de prácticas en la programación de controladores lógicos programables.

Para alcanzar el objetivo propuesto se plantean las siguientes **tareas de la investigación**:

1. Caracterizar desde el punto de vista histórico y gnoseológico los entornos virtuales de simulación de procesos industriales existentes en la actualidad.

2. Estudiar las herramientas y formas de trabajo existentes dentro del simulador de procesos escogido.
3. Realizar una breve descripción del *software* de programación y simulación SoMachine Basic del fabricante Schneider.
4. Escoger y/o diseñar escenarios de procesos industriales de muestra en el entorno virtual del simulador escogido.
5. Desarrollar el modelado y programación del control para los diferentes escenarios.
6. Validar por medio de la simulación los algoritmos de control programados sobre los distintos escenarios escogidos y/o diseñados.

Durante el desarrollo de la investigación fueron empleados **métodos teóricos** como el histórico-lógico y el de análisis y síntesis fundamentalmente para la búsqueda y procesamiento de información. Estos permitieron el estudio de los fenómenos relacionados con la simulación y las diferentes herramientas empleadas a nivel internacional, así como su aplicación de acuerdo al tema en cuestión.

De igual manera fueron empleados los **métodos empíricos** de observación y experimentación que posibilitaron la descripción y análisis de situaciones sobre la realidad estudiada, así como el diseño, programación y simulación en el entorno empleado.

Como contribución son demostradas las posibilidades que brinda la plataforma Factory I/O al desarrollar cuatro escenas de procesos industriales simulados en dicho entorno virtual controladas por algoritmos programados en el controlador lógico programable M221 usando el *software* SoMachine Basic de Schneider.

El trabajo de diploma está estructurado en introducción, dos capítulos, conclusiones, recomendaciones, bibliografía y anexos.

En el primer capítulo son abordados brevemente la importancia de las prácticas de laboratorio en la enseñanza. Se describe la situación actual de las instalaciones existentes y las soluciones adoptadas para las prácticas relacionadas con la programación de controladores lógicos programables. Se realiza una búsqueda y evaluación de diferentes simuladores de procesos industriales con el fin de ser controlados, caracterizando el más pertinente. Y se hace alusión a los PLCs de forma general, introduciendo el *software* Somachine Basic y el PLC M221.

En el segundo capítulo son presentados un conjunto de procesos industriales desarrollados utilizando el simulador Factory I/O describiendo en cada caso sus componentes y modelando en Grafcet el control de los mismos, lo que sirve de base al desarrollo de los algoritmos programados en lenguaje LD que fueron validados a través de la simulación.

## **CAPITULO 1. MARCO TEÓRICO REFERENCIAL.**

En este capítulo se argumenta brevemente la importancia, objetivo y consecuencias de las prácticas de laboratorio en la enseñanza. Se describe la situación actual de las instalaciones en el laboratorio de control de procesos en el Departamento de Automática, así como las soluciones adoptadas hasta la época para las prácticas relacionadas con la programación de controladores lógicos programables. Se realiza una búsqueda y evaluación de los diferentes simuladores virtuales de procesos industriales existentes con el fin de ser controlados caracterizando el más pertinente. Se hace alusión a los PLCs de forma general, introduciendo el *software* SoMachine Basic y el PLC M221.

### **1.1 Prácticas de laboratorio en la enseñanza**

A comienzos del siglo XX, la enseñanza del laboratorio de ciencias tuvo un particular auge con énfasis en los trabajos experimentales, pero entró en conflicto en los años veinte y treinta debido a la importancia que se le comenzó a otorgar a las demostraciones sin evidencias pedagógicas justificables. No obstante, la época del lanzamiento del Sputnik, en 1957, le dio un empuje a la enseñanza de las ciencias en los años sesenta, resurgiendo la enseñanza experimental del laboratorio, ahora con énfasis en el método por descubrimiento. A pesar de este renacimiento experimental de la enseñanza de la ciencia en los años sesenta, se observa una declinación en el interés por los laboratorios en general y se comienza a cuestionar su efectividad y objetivos. Parte de este desánimo estaba asociado a los desacuerdos existentes sobre los objetivos del trabajo del laboratorio, poniéndose de manifiesto una situación que no era realmente nueva, ya que desde finales del siglo XIX se había reportado “el caótico trabajo de laboratorio”. No obstante, esta situación de incertidumbre abrió el camino para la investigación sobre su verdadero rol en la enseñanza de las ciencias y los objetivos que persigue.

Hasta finales de los años cincuenta del pasado siglo, la enseñanza del laboratorio se centró principalmente en actividades verificativas discutidas en las clases de teoría, planteadas en los libros de texto o sugeridas en manuales de laboratorio. Esta situación se trató de cambiar con el nuevo currículo de los años sesenta, dándosele a la enseñanza del laboratorio la función importante de

desarrollar habilidades de alto nivel cognitivo, mediante actividades centradas en los procesos de la ciencia a través del método indagatorio.

Un problema general con relación a los objetivos del trabajo de laboratorio detectado en los años sesenta es que los mismos no se correspondían con objetivos propios del trabajo práctico. Esto condujo a que en los años setenta esta situación se tratara de mejorar, pero resultó un fracaso, en virtud de que los objetivos elaborados respondían a los de un curso de ciencia, en general, en el que se enfatizaba el reforzamiento y comprobación de teorías. Hasta mediados de los años noventa, se señalaba que los trabajos de laboratorio tenían como objetivos principales generar motivación, comprobar teorías y desarrollar destrezas cognitivas de alto nivel. Sin embargo, muchos estudiantes piensan que el propósito del trabajo de laboratorio es seguir instrucciones y obtener la respuesta correcta, por lo que se concentran en la idea de manipular instrumentos más que manejar ideas.

El objetivo fundamental de los trabajos en los laboratorios es fomentar una enseñanza más activa, participativa e individualizada, donde se impulse el método científico y el espíritu crítico. De este modo favorece a que el alumno desarrolle habilidades, aprenda técnicas elementales y se familiarice con el manejo de instrumentos y equipos. Por otra parte, el enfoque que se va a dar a los trabajos prácticos va a depender de los objetivos particulares que se deseen conseguir tras su realización. Además, las prácticas de laboratorio en la educación de un ingeniero formado en esta especialidad de Automática, es muy importante ya que con las prácticas es donde este pueden ejercitar los conocimientos aprendidos teóricamente, además de que les brinda habilidades para poder desarrollarse en la industria una vez egresados, dándoles una mejor visión de cómo resolver los problemas a los que se enfrente [4].

## **1.2 Instalaciones en el laboratorio de Control de Procesos**

En el departamento de Automática perteneciente a la Facultad de Ingeniería Eléctrica en la Universidad de Oriente posee un laboratorio de control de procesos en cuyo interior existen disímiles instalaciones o plantas piloto, que consisten en partes específicas ensambladas que operan como un todo armónico con el propósito de reproducir, a escala, procesos productivos.

En estas instalaciones las asignaturas Medios Técnicos de Automatización, Autómatas Programables y Sistemas de Automatización realizaban tradicionalmente las prácticas relacionadas con la programación de controladores lógicos programables y automatización, entre ellas se pueden encontrar:

- **Instalación de laboratorio para el control de nivel** ver figura 1 compuesta básicamente por dos tanques, uno elevado y otro que hace función de reservorio en un nivel inferior; una bomba que permite mediante su accionamiento trasegar el líquido en el reservorio hacia el tanque elevado; y una electroválvula que permite el vaciado del tanque elevado hacia el reservorio. El control de esta instalación se realiza por un panel de control a través de lógica cableada o programada por medio de los controladores lógicos programables TSX1720 y TSX Momentum.



Figura 1 Planta piloto para el control de nivel.

Fuente: Autora, 2023.

- **Instalación de laboratorio para control de un manipulador neumático** formada por dos cilindros neumáticos de doble efecto que proporcionan desplazamiento horizontal y vertical; un conjunto de diafragma, eyector y ventosa que sirve para coger o soltar objetos y el conjunto de válvulas electro-neumáticas necesarias para activar dichos componentes,

accionadas por el controlador lógico programable TSX Micro 3722 ver figura 2.



Figura 2 Planta piloto de un manipulador neumático.

Fuente: Autora, 2023.

- **Instalación de laboratorio para el control de presión** formado básicamente por uno o dos tanques cerrados según se seleccione a los que se le introduce aire comprimido por medio de una válvula de regulación con posicionador SIEMENS ver figura 3. El control de esta instalación se realiza por un controlador lógico programable TSX Momentum.



Figura 3 Planta piloto para el control de presión.

Fuente: Autora, 2023.

- **Instalación de laboratorio para el control de flujo** que consta de forma general de un reservorio de agua, una bomba y una válvula de regulación con posicionador SIEMENS para el control del flujo ver figura 4. El control de esta instalación se realiza por un controlador lógico programable TSX Momentum.



Figura 4 Planta piloto para el control de flujo.

Fuente: Autora, 2023.

El hecho de programar y poder comprobar el correcto funcionamiento de estas instalaciones empleando medios de medición y actuación reales brinda sin lugar a dudas una experiencia enriquecedora al estudiante, al mismo tiempo que lo motiva. Sin embargo, en los últimos años debido a la ruptura de equipos auxiliares necesarios para el funcionamiento de las mismas como el compresor de aire y la bomba, la falta de mantenimiento, el deterioro por el uso prolongado y la obsolescencia de algunos componentes estas prácticas han dejado de realizarse.

- Soluciones de simulación adoptadas.

En búsqueda de alternativas estas asignaturas siempre en su afán de lograr un egresado con sólidas habilidades han empleado plataformas como son:

- **ISaGRAF**: *software* que permite crear sistemas de control distribuidos o locales, compatibles con todos los lenguajes de control del estándar internacional IEC 61131-3 y simularlos.

- **CoDeSys:** entorno de desarrollo para la programación de controladores conforme con el estándar industrial internacional IEC 61131-3 que permite la simulación.

Tanto en ISaGRAF inicialmente, como posteriormente en CoDeSys se realizaban ejercicios propios de programación de controladores lógicos programables que luego podían ser simulados, pero que al probarlos debían activarse los interruptores teniendo en cuenta el orden lógico según el funcionamiento del proceso controlado y a través de los indicadores de salida evaluar el algoritmo, proceso tedioso si el programa contaba con muchas entradas y salidas.

No obstante, para evitar esto y como ambos *softwares* contaban dentro de sus herramientas con la posibilidad de desarrollar visualizaciones, las que resultaban muy básicas en ISaGRAF y un poco más desarrolladas en CoDeSys, se recreaba el funcionamiento de la planta o proceso controlado. Esto causaba confusión en los estudiantes al tener que desarrollar ambos códigos sobre el controlador, un algoritmo de control y otro para simular el proceso controlado.

Con el tiempo se ha comenzado a emplear el *software* PC-SIMU que permite simular procesos automáticos de forma gráfica intercambiando las entradas salidas con otros programas como el CADE-SIMU, utilizado principalmente para elaborar o realizar esquemas de mando o potencia (lógica cableada) además de otras opciones, y el simulador del PLC S7-200 empleado para ejecutar los programas realizados en el *software* STEP7 Microwin de SIEMENS.

### **1.3 Software de simulación**

Siempre se ha buscado la manera de evaluar sistemas complejos y tal como se define la simulación es la ejecución de un modelo representado por un programa de computadora que permite recrear entornos de red, ahorrando tiempo y dinero. Comercialmente existen simuladores de tiempo continuo y tiempo discreto; la simulación en tiempo discreto modela sistemas que cambian en el tiempo de acuerdo con los diferentes estados que una variable puede tener, algo muy útil para sistemas de comunicación. Los simuladores de tiempo continuo, por su parte, avanzan en el tiempo y constantemente revisan si ha ocurrido algún evento, con el fin de actualizar las variables correspondientes, para, solo en ese caso, realizar la modificación de valores.

Los *software* de simulación, sobre todo los usados para procesos automatizados ofrecen un material invaluable para el diseño, análisis y optimización de sistemas de fabricación y procesos industriales. Estos programas pueden simular entornos completos de plantas industriales y sistemas de producción, permitiendo a los ingenieros y operadores visualizar, probar y mejorar sus procesos antes de la implementación física. Además de que en la educación se pueden implementar como una herramienta de aprendizaje [5].

### **1.3.1 Software de simulación de procesos industriales**

Para la elección del *software* donde se realizará la simulación se hizo una investigación y se analizaron algunas posibles opciones válidas para plantas industriales, teniendo en cuenta los siguientes requisitos:

- ✓ Elegir un programa que permitiera la comunicación con el SoMachine Basic, el *software* de programación del autómatas con el que se va a trabajar.
- ✓ Que fuese un simulador que motivara a los estudiantes a trabajar con el entorno que ofreciera.
- ✓ Que el entorno de trabajo que ofreciera fuera lo más parecido posible a una planta piloto.
- ✓ Que se pudieran recrear procesos parecidos a los que se controlaban en el antiguo laboratorio de la facultad.
- ✓ Que los requisitos solicitados por el programa para ser ejecutado en un computador no fuesen demasiado altos.
- ✓ Que fuese un programa moderno y de mejor calidad que los que anteriormente se usaban en las asignaturas afines con el control automático.

A continuación se describen brevemente las características de las mejores alternativas encontradas.

#### **PC-SIMU (V3.0)**

Se trata de un programa gratuito que, junto con el simulador del s7-200, permite realizar procesos automáticos de forma gráfica mediante un código exportado en AWL. Es un *software* que permite crear y simular modelos en 2D con piezas prediseñadas, pero fue descartado porque tiene muchas limitaciones a la hora

de realizar alguna planta virtual específica y presenta problemas a la hora de conectarlo con el simulador del autómeta [6].

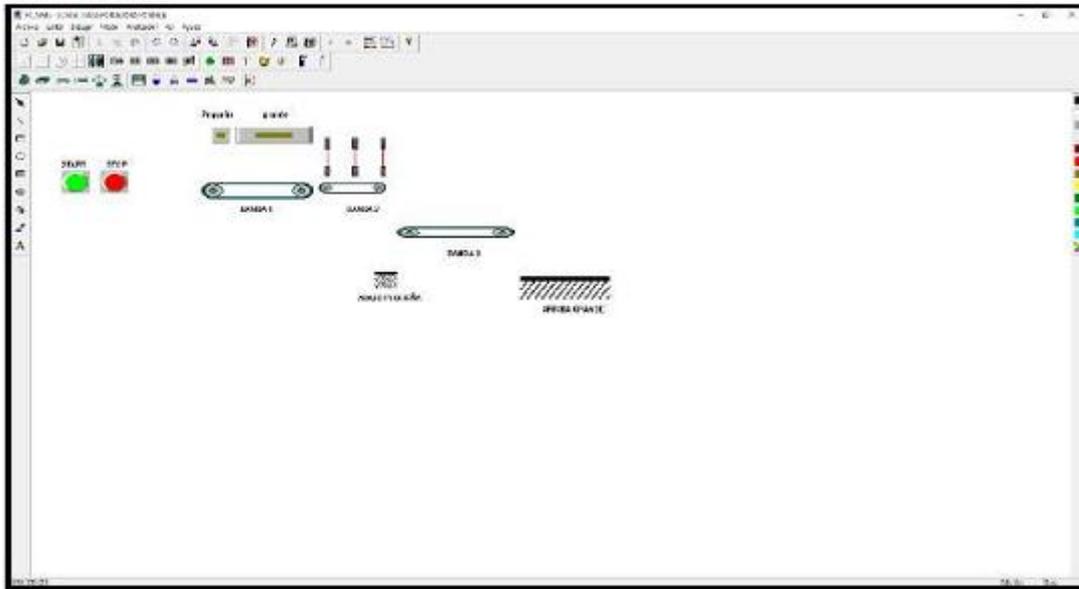


Figura 5 Interfaz PC-SIMU (separado de cajas).

### Factory I/O

Se trata de un *software* que permite crear y simular modelos en 3D. Es un programa bastante interactivo, que nos permite diseñar plantas con una alta calidad visual y de sonido con un alto grado de realismo. Además, permite la conexión con equipos externos como PLCs, microcontroladores, FPGA, así como la conexión con *software* de programación de autómetas. El usuario puede realizar su propio código KOP para la planta simulada y, posteriormente, comprobar su funcionamiento [7].

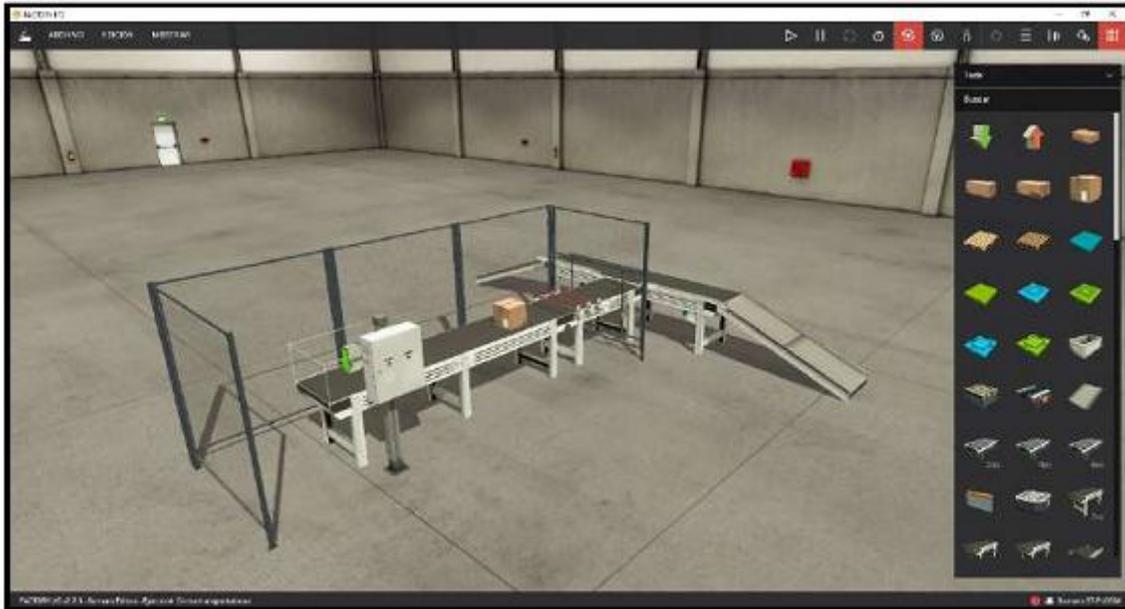


Figura 6 Interfaz FACTORY IO (separado de cajas).

### **Machines Simulator**

Se trata de un *software* de simulación 3D de Nitrec orientado a poder programar PLCs usando un entorno virtual que represente una línea de producción física. También se puede replicar sistemas físicos reales para ser programados de manera virtual y evitar dañar el sistema real. Una de las características y fortalezas que distingue a este simulador de otras alternativas es que se pueden crear objetos 3D personalizados para usarlos en el *software*. Fue descartado ya que no se encontró el instalador del programa [8].

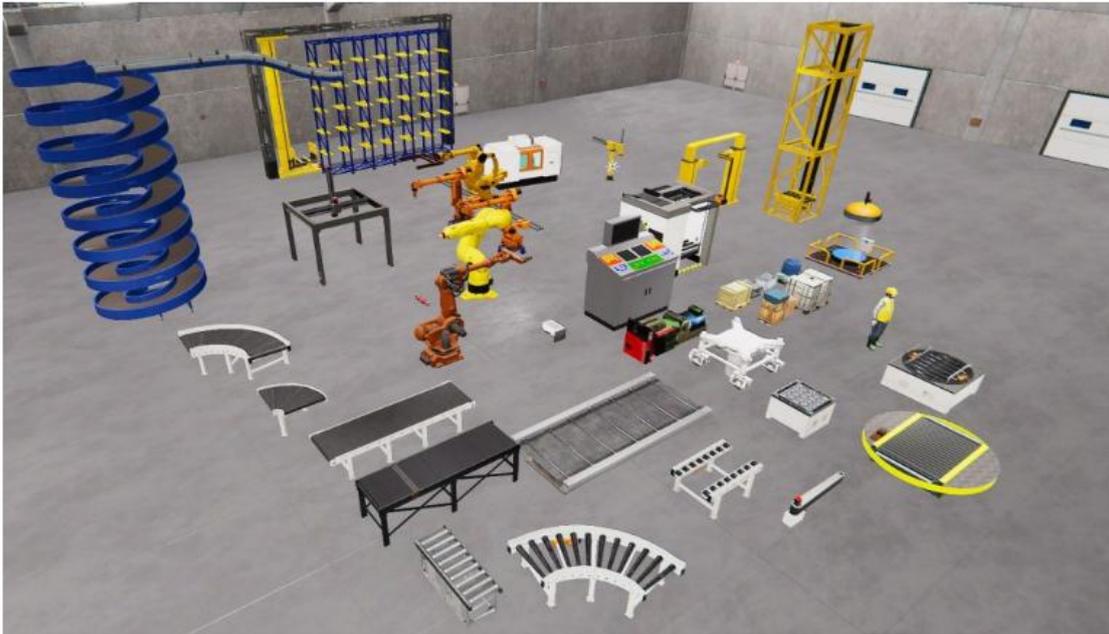


Figura 7 Interfaz Machines Simulator.

### **FESTO fluidSim**

Es un programa informático demostrativo 2D para el entrenamiento en sistemas neumáticos. Creado por la empresa alemana FESTO líder mundial en construcción y ventas de elementos neumáticos. Este programa cuenta con todas las herramientas necesarias para realizar sistemas neumáticos, electroneumáticos, así como la posibilidad de poder monitorear y simular en tiempo real la operación del sistema neumático. Permite exportar e importar diagramas de circuitos neumáticos, su uso desarrolla los conocimientos del usuario sobre el funcionamiento de válvulas, actuadores y accionamientos. Se descartó por el hecho de que es un simulador solo para procesos neumáticos y electroneumáticos, lo cual sería una limitante si se desea controlar otro tipo de sistema [9].

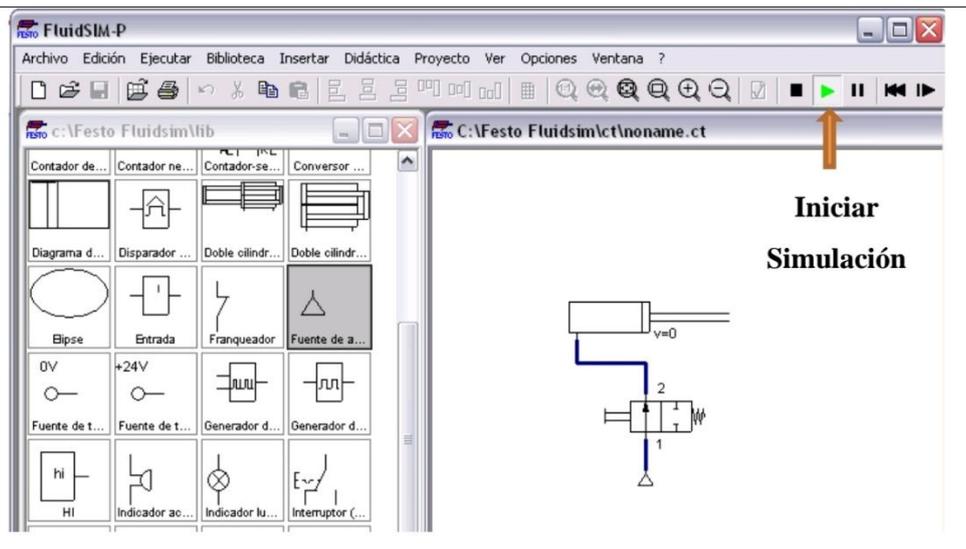


Figura 8 Interfaz FESTO fluidSim.

El simulador escogido fue el Factory I/O puesto que fue el más completo. Una de las prioridades era que fuese una herramienta educativa, ligera y fácil de comprender. Como es un *software* 3D permite recrear y controlar un entorno industrial automatizado llamativo para las nuevas generaciones, además de que se puede conectar con los principales autómatas que se estudian durante la carrera, como lo son Siemens, Schneider, Allen-Bradley. Los requisitos para ejecutarse en un ordenador no son tan exigentes por lo que se puede instalar en las computadoras destinadas al laboratorio virtual de la facultad. Además de que supera a los simuladores anteriores empleados en asignaturas afines con el control de procesos en cuanto a calidad de gráficos, velocidad de ejecución y capacidad de conexión con varios simuladores de PLCs.

#### 1.4 Software de simulación Factory I/O.

Factory I/O es un simulador de fábricas 3D muy útil para el aprendizaje de tecnologías para automatización lanzado por primera vez el 1 de julio del 2014 [8]. Su desarrollador, Real Games, aboga por una accesibilidad sencilla y realista, en especial para la gente joven ya que incluye mecánicas de trabajo (gráficas de alta calidad y sonido) similares al resto de sus productos tan consumidos por personas de dicho rango de edad. Facilita además un gran catálogo de manuales y descripciones de sus elementos, así como una detallada guía online muy útil para consultar tanto trabajando como aprendiendo [10].



Figura 9 Logo del programa.

El programa en su versión 2.5.2 (que fue la que se utilizó para el desarrollo del proyecto) trae consigo una serie de escenas predefinidas que ayudan a la introducción a todo tipo de elementos, pues vienen ya configuradas para su uso y preparadas para asociar a un bloque de Tia Portal, SoMachine, RSLogix, Codesys, Unity Pro, SIMATIC Step 7, TwinCat. En las escenas predefinidas se puede probar el funcionamiento de todas las estaciones y prácticamente todos los actuadores disponibles para después poder aplicarlos debidamente en el proyecto a realizar [10].

El simulador proporciona como condiciones iniciales un espacio en el que trabajar, en un entorno industrial dentro de una nave, con la iluminación y espacio predefinidos y no modificables. Los dispositivos se muestran en un menú lateral ordenados por tipología. La posición de la cámara se puede manejar de dos formas: por un control de navegación ubicado en la parte lateral inferior o con el teclado del computador; de esta forma el usuario se mueve en los ejes X, Y, Z y gira 360 grados.

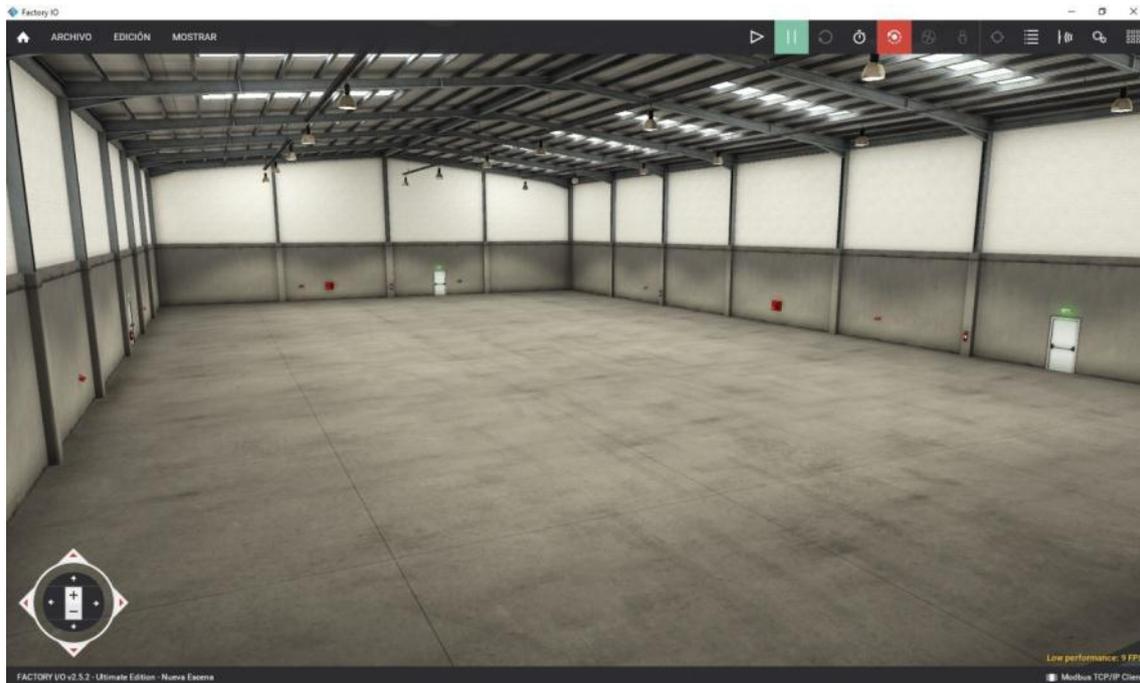


Figura 10 Entorno proporcionado para trabajar.

#### 1.4.1 Elementos disponibles.

Dispone de una amplia gama de elementos divididos en categorías que se describen a continuación [10]:

- Elementos: compuesto por todas las cajas y piezas que circularán por el proceso industrial, pudiendo trabajar con cajas de diversos tamaños, piezas de tres colores y tres tipos distintas o *pallet* de apoyo necesarios para la circulación en ciertas estaciones de trabajo.



Figura 11 Elementos.

- Carga pesada: lo forman todos los elementos de circulación de objetos de mayor peso, por lo que son robustos, anchos y poseen poca altura y velocidad de operación. Están relacionados directamente con los *ítems* como cajas o *pallets*.

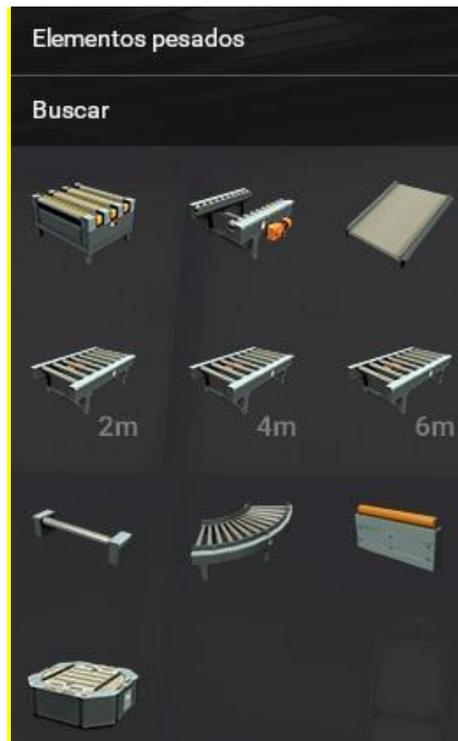


Figura 12 Carga pesada

- Carga ligera: se compone de elementos de circulación dedicados al transporte único de piezas o cajas pequeñas, lo cual permite el uso de maquinaria asociada como clasificadores en forma de brazo, ruedas giratorias o *pushers* que permiten su desplazamiento a distintas líneas de producción, o maquinaria de alineación que requiera de posiciones concretas de los elementos.

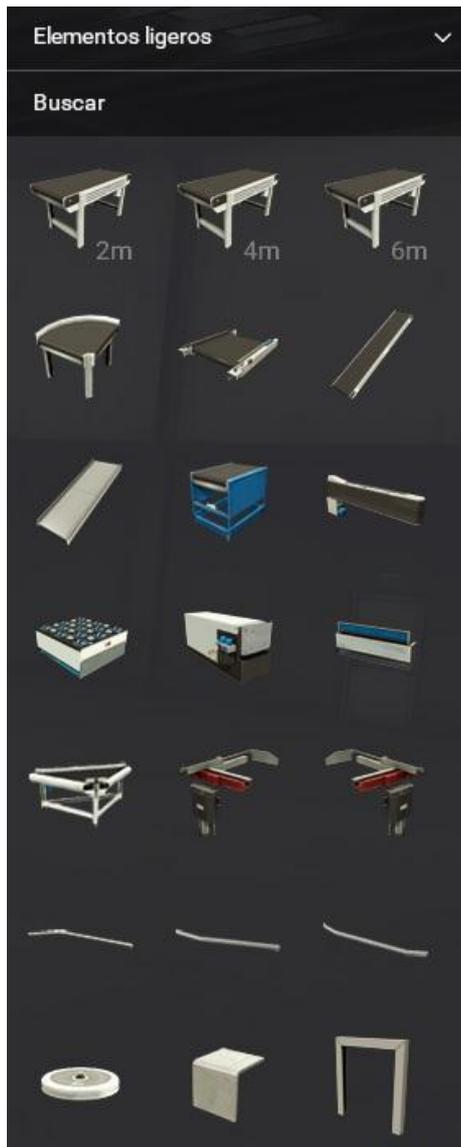


Figura 13 Carga ligera.

- Sensores: se dispone de una pequeña variedad de sensores, pero dedicados a procesos muy comunes como la detección de materiales, elementos conductores, o códigos de barras. Además, se permite la clasificación de los objetos según su altura, peso o color y tipo en el caso de las piezas.



Figura 14 Sensores.

- Operadores: engloba todos los elementos de actuación sobre la planta ya sean botones, potenciómetros o selectores, y de control, como contadores o indicadores.



Figura 15 Operadores

- Estaciones: poseen mayor complejidad que el resto de elementos ya que están dedicadas a la operación directa sobre los *ítems*. Se forman por un conjunto de

elementos cerrado que dispone de su propio sistema de entrada, salida, botonera y sensores.

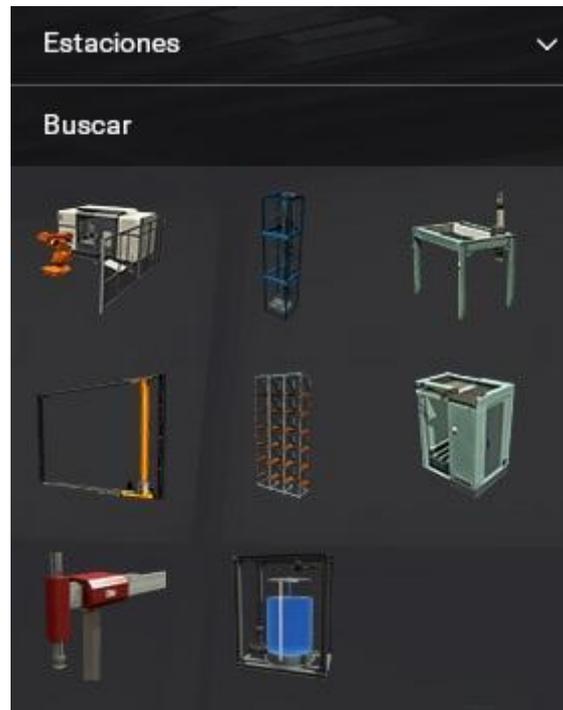


Figura 16 Estaciones.

- Elementos de alarma: una de las grandes ventajas de este programa es la posible introducción de errores o fallos en cada uno de los elementos, los cuales se comunican en planta mediante alarmas visuales o auditivas.

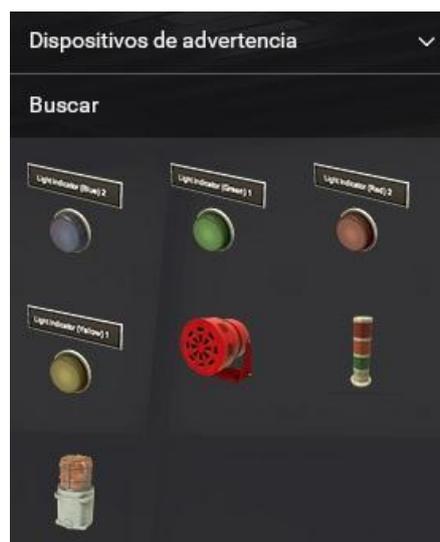


Figura 17 Dispositivos de advertencia.

- Pasarelas: Factory I/O añade estructuras de protección o de accesibilidad a ciertas zonas de la fábrica como medida de realismo y acercamiento a lo que

sería una planta de producción real. Estos elementos no disponen de ningún tipo de control desde el PLC.

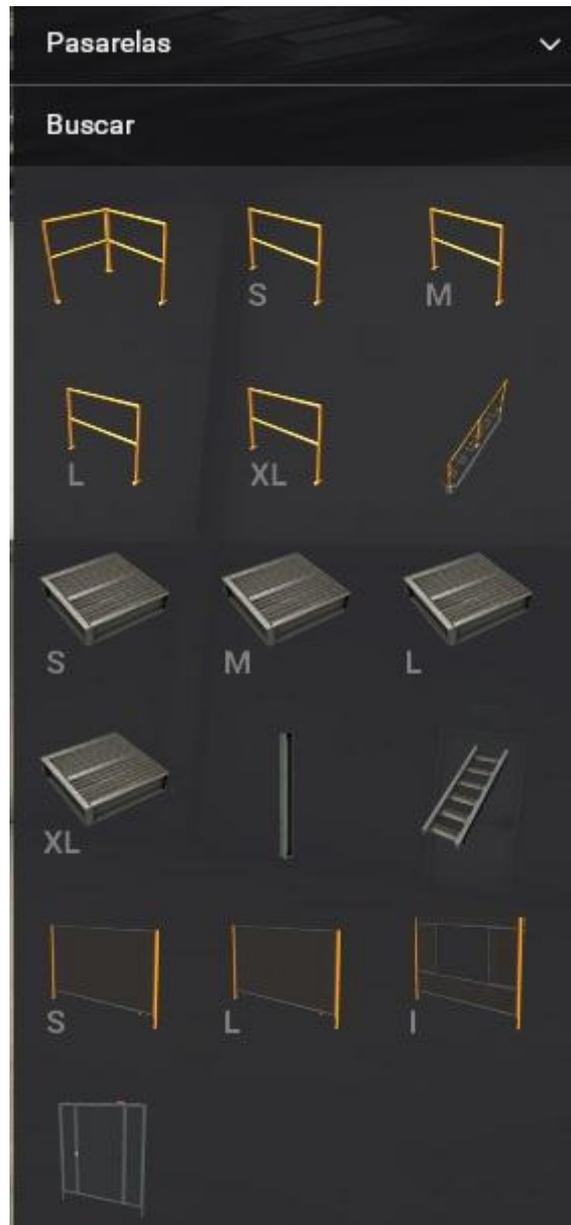


Figura 18 Pasarelas.

- Introducción y extracción de *ítems*: para introducir o extraer elementos se ha de hacer uso de estos sistemas, que simulan la entrada y salida de fábrica de los materiales.

Para ver una descripción de todos los componentes y dispositivos con sus respectivas especificidades, proporcionados por el simulador Factory IO ver Anexo 1.

El control de todos estos elementos se dividirá entre los dedicados a la detección (sensores introducidos manualmente o pertenecientes a las estaciones) o los actuadores (todos aquellos que producen algún efecto directo en la planta), y se podrán configurar de manera digital, analógica o híbrida entre ambos casos anteriores. De esta forma se realiza una simple asociación entre entradas y salidas del controlador, pero limita la lectura de las posibles marcas del sistema que quedarán como variables de operación, y no de lectura por parte de la simulación del Factory I/O [10].

Cabe añadir que las librerías de elementos son cerradas y no permite la introducción ni modificación de ninguno de los elementos, lo cual puede suponer una limitación en el proceso de diseño si se quiere recrear un proceso complejo, ya que según el desarrollador el objetivo del programa es que con lo que se ofrece se pueda aprender a programar procesos y no centralizarse netamente en el diseño de plantas industriales.

Permite 255 entradas digitales, 255 salidas digitales, 16 entradas analógicas y 16 salidas analógicas (para valores tipo real) [8].

Los requerimientos mínimos para el uso de Factory IO son: Windows Vista o superior, procesador Intel Core 2 Duo a 2Ghz, 1Gb de RAM y 500Mb de disco duro [11].

#### **1.4.2 Posibilidades de comunicación**

Un controlador de entradas y salidas es una característica incorporada del programa responsable de comunicarse con un controlador externo. Factory IO incluye muchos controladores de entradas y salidas, cada uno de los cuales está diseñado para usarse con una tecnología específica. El usuario es el que debe seleccionar el controlador que desea utilizar. A nivel de *driver* el *software* posee los más importantes: Modbus TCP/IP, OPC, Siemens (Logo, familia de los S7, PLCSIM), entre otros.

A continuación se detallan mucho mejor todos los *drivers* de los que dispone [10].

- Advantech USB 4750 & USB 4704

Este controlador de E/S proporciona una interfaz para las placas DAQ Advantech USB 4750/4704. Estas son placas DAQ alimentadas por USB que se pueden usar con computadoras de escritorio, portátiles y tabletas siempre que haya un puerto USB libre disponible y no se requiere una fuente de alimentación externa.

Por lo general, se utilizan como placas de interfaz entre Factory I/O y *hardware* externo como PLC, microcontroladores o cualquier otro dispositivo que pueda conectarse eléctricamente a la placa.

- Advantech USB 4750

Se utiliza para cablear PLC o cualquier otro *hardware* que requiera E/S digitales optoaisladas.

Compatible con USB 1.1/2.0.

16 canales DI aislados y 16 canales DO aislados.

Aislamiento de alto voltaje en todos los canales (2500 VDC).

Alta corriente de disipador en canales de salida aislados (100 mA/canales).

Rango de entrada: 5 - 50 VCC.

Rango de salida: 5 - 40 VCC.

Adecuado para montaje en carril DIN.

- Advantech USB 4704

Normalmente se utiliza como solución complementaria para el Advantech USB 4750 cuando se requieren valores analógicos (0-10 V). También incluye E/S digitales que pueden usarse para cablear microcontroladores o cualquier otro *hardware* TTL (5V).

Soporta USB 2.0

8 entradas analógicas (resolución de 12 bits)

Entradas analógicas bipolares (V) SE:  $\pm 10$ ; Diferencia:  $\pm 1$ ,  $\pm 1,25$ ,  $\pm 2$ ,  $\pm 2,5$ ,  $\pm 4$ ,  $\pm 5$ ,  $\pm 10$ ,  $\pm 20$

2 salidas analógicas (resolución de 12 bits)

Rango de salidas analógicas 0 - 5 V

Consumo de energía: 5 V a 100 mA; Máx.: 5 V a 500 mA

8 E/S digitales

Velocidad de muestreo de hasta 48 kS/s

- Allen-Bradley Logix5000

Este controlador proporciona una conexión Ethernet a Allen-Bradley ControlLogix, CompactLogix o SoftLogix PAC.

- Allen-Bradley Micro800

Este controlador proporciona una conexión Ethernet al PLC Allen-Bradley Micro800.

- Allen-Bradley MicroLogix

Este controlador proporciona una conexión Ethernet al PLC Allen-Bradley MicroLogix.

- Automgen Server

Este controlador de E/S proporciona conectividad con AUTOMGEN a través de un servidor TCP/IP. AUTOMGEN es una herramienta para el diseño/simulación de algoritmos de control y programación de PLC. Es especialmente adecuado para desarrollar controladores utilizando GRAFCET pero también permite el diseño de controladores utilizando lenguajes estándar como los definidos por los estándares IEC 61131-3 y SysML.

Está configurado de forma predeterminada para iniciar el servidor una vez que se selecciona este controlador. Puede cambiar este comportamiento desactivando la opción Conexión automática en el Panel de configuración, que permitirá INICIAR y DETENER el servidor manualmente.

- Control I/O

Control I/O es un Soft PLC simple pero potente diseñado exclusivamente para usarse con Factory I/O. El objetivo principal de Control I/O es proporcionar una herramienta independiente de la marca, lo suficientemente fácil de aprender para cualquiera que se inicie en el mundo de la automatización. Con Control I/O usted desarrolla programas con diagramas de bloques de funciones utilizando las funciones más comunes disponibles en un PLC real.

- MHJ

Este controlador permite utilizar Factory I/O junto con WinSPS-S7 y WinPLC-Engine, que son alternativas de bajo coste al *software* oficial de Siemens. Además, puede utilizar Grafcet-Studio, que le permite diseñar y ejecutar gráficos Grafcet.

WinSPS-S7 es una herramienta de programación de PLC para los S7 con un simulador de PLC integrado (S7-300/400). WinPLC-Engine es un simulador de PLC S7-300 que se puede programar con STEP7 V5.x y TIA Portal V1X. Grafcet-Studio es una herramienta para diseñar y ejecutar controladores especificados en Grafcet.

- Modbus TCP/IP Client

Este controlador implementa un cliente Modbus TCP/IP (maestro).

Puede definir desde dónde leer valores digitales (Entradas o Bobinas) y registrar valores (Registros de entrada o Registros de retención). Sin embargo, no puede cambiar dónde escribir los valores ya que un servidor Modbus TCP/IP solo permite operaciones de escritura en bobinas y registros de retención.

- Modbus TCP/IP Server

Este controlador implementa un servidor Modbus TCP/IP (esclavo).

- OPC Client DA/UA

Este controlador permite a Factory I/O intercambiar datos con PLC, SoftPLC o cualquier otra tecnología a la que se pueda acceder a través de un Servidor OPC DA/UA. Es compatible con las especificaciones OPC DA 1.0, 2.0, 3.0 (*Classic*) y UA (*Unified Architecture*).

- Siemens LOGO!

Este controlador proporciona una conexión Ethernet al LOGO! de Siemens. SOCIEDAD ANÓNIMA. Es compatible con los modelos 0BA7 y 0BA8.

- Siemens S7-200/300/400

Este controlador proporciona una conexión Ethernet al PLC Siemens S7-200, S7-200 SMART, S7-300 o S7-400.

- Siemens S7-1200/1500

Este controlador proporciona una conexión Ethernet a los PLC Siemens S7-1200 y S7-1500.

- Siemens S7-PLCSIM

Este controlador proporciona conectividad con Siemens S7-PLCSIM. Se puede utilizar para controlar Factory I/O con S7-PLCSIM v5/13-16/Advanced a través de TIA Portal.

El *driver* utilizado para realizar la conexión con el simulador del autómatas que se va a emplear es Modbus TCP/IP Client, ya que los PLCs virtuales de Schneider solo se conectan con el Factory IO a través de este protocolo. Esto significa que el Factory IO será quien inicie las conexiones, demandando una lectura o escritura de las variables en sus correspondientes espacios de memoria provenientes del PLC de SoMachine que toma el rol de servidor [12].

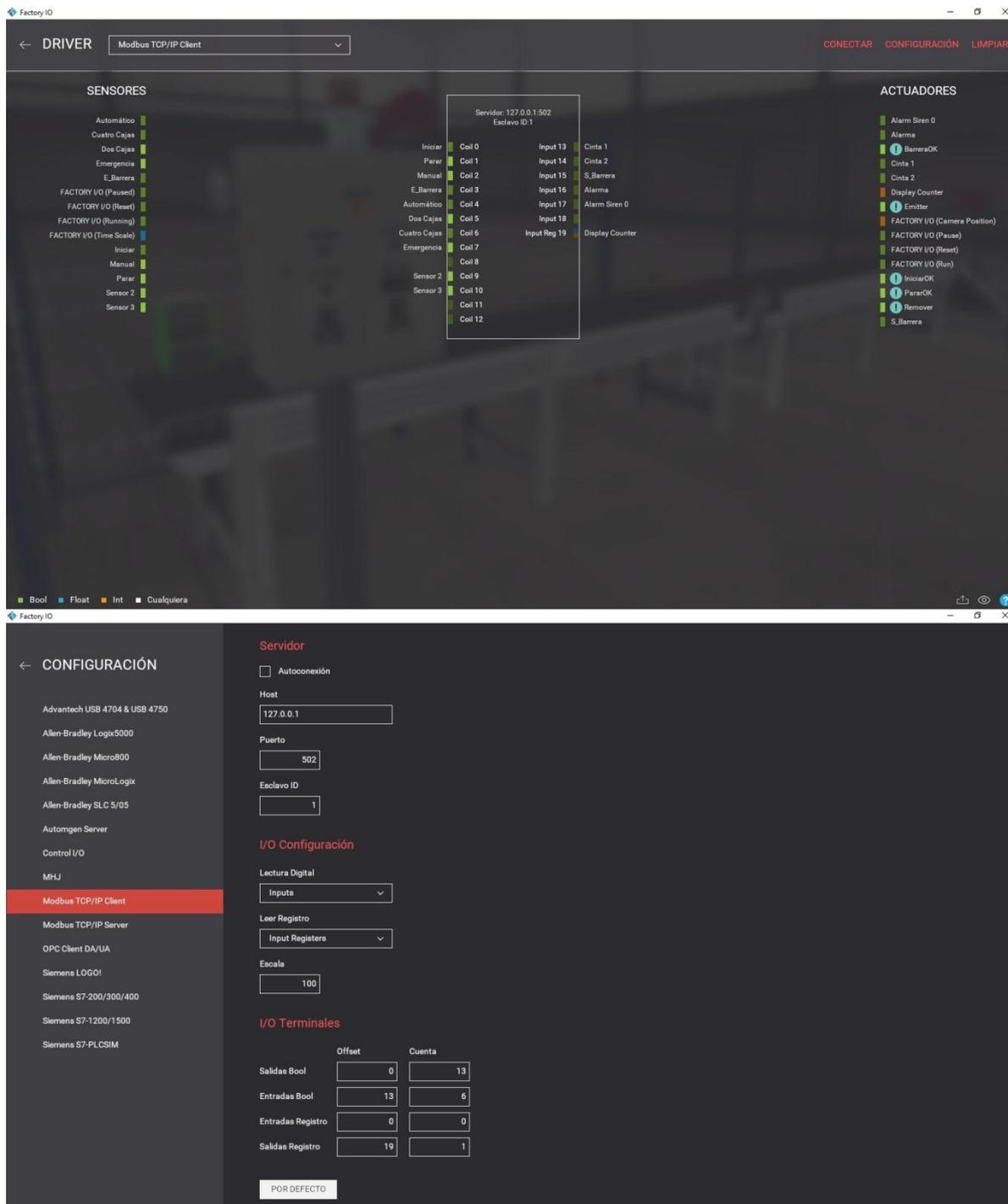


Figura 19 Configuración del protocolo de comunicación utilizado, Modbus TCP/IP Client (ejemplo Estación Buffer).

Una de las desventajas es que el Factory I/O no permite que las direcciones de sus entradas y salidas simuladas coincidan con las físicas añadidas al PLC ya que provocan superposición de valores en la escritura, por lo que se direccionará a cada uno de los módulos a un valor que no afecte a este caso, además se debe adquirir su licencia para ser usado.

Actualmente Real Games publicó el 30 de enero de 2023 la última actualización [10] hecha del software la cual se hizo pensando en arreglar problemas del *software* y manteniendo la idea de que el Factory IO continúe teniendo los mismos elementos que se le diseñaron desde la primera vez que salió al mercado.

### **1.5 Controladores lógicos programables.**

Entre las grandes innovaciones autómatas del mercado industrial se encuentra el PLC o Controlador Lógico Programable; se trata de una computadora industrial que tiene como objetivo principal la automatización de procesos en una línea de producción, por lo que está a cargo del control de los sistemas de las máquinas. Se compone de un sistema operativo (firmware) y de un entorno de programación que puede soportar uno o varios lenguajes. Estos dispositivos se vienen empleando para la automatización de procesos desde 1970. Desde entonces su tecnología ha ido mejorando y hoy en día ofrecen grandes prestaciones para la industria.

Un controlador lógico programable básico, cuenta con entradas y salidas digitales integradas. A este tipo de entradas, se le conectan elementos como pulsadores, interruptor, sensores, entre otros, para medir diversas variables como temperatura, presión, nivel, humedad, por mencionar algunos. A las salidas digitales por su parte, se conectan dispositivos que controlan la potencia, ejemplo de ellos, motores, válvulas, lámparas o resistencias eléctricas. Es así que se les llama digitales porque solo puede tomar dos estados (todo o nada, on off, activo inactivo). Para la activación de estas entradas y salidas se requiere entonces de un control lógico; es decir, que las salidas dependan del estado de las entradas en un momento determinado. Es de este modo que el sistema a controlar por el PLC puede tomar dos configuraciones [13]:

-Bucle abierto: las salidas digitales del PLC solo se utilizan para controlar el proceso y tienen la posibilidad de ser visualizadas.

-Bucle cerrado: las salidas digitales del proceso actúan sobre el PLC para influir en las señales de control generadas por él.

Control Analógico:

Permiten realizar el control de variables físicas en tiempo real. Es así que se debe estar en capacidad de adquirir señales, manipularlas y entregar la señal

adecuada a la planta, determinando así, su correcto funcionamiento. Cabe destacar que es importante la selección del tiempo de muestreo con el cual se va a realizar la adquisición [13].

La norma IEC 61131-3 es la única norma válida a nivel mundial para los lenguajes de programación de PLCs. [14]. A continuación se mencionan los cinco lenguajes existentes para la programación de autómatas:

- Lista de instrucciones (IL).
- Diagrama de escalera (LD).
- Diagrama de bloques funcionales (FBD).
- Texto estructurado (ST).
- Gráfico de funciones secuenciales (SFC).

#### **1.5.1 Autómata programable ModiconM221.**

El PLC escogido para realizar la conexión con el Factory I/O y demostrar la validez del funcionamiento de las escenas que se diseñaron como objetivo del trabajo es el ModiconM221 cuya CPU es la TM221CE24R del fabricante Schneider Electric. Aunque la conexión se va a hacer con el simulador del SoMachine Basic, se seleccionó este autómata ya que se encuentra físicamente en el laboratorio de la facultad lo que permitirá cargar la programación en él y realizar futuras pruebas en el Factory I/O, además de que es un autómata usado para trabajar en las asignaturas de Medios Técnicos de la Automatización, Autómatas Programables y Sistemas de Automatización.

Es un PLC compacto, se programa en el software SoMachine Basic que es un entorno de programación fácil de usar, admite diversas opciones de comunicación como puertos Ethernet incorporados, USB y puertos serie, lo que permite integración con otros dispositivos y sistemas en la red. Cuenta con funciones de seguridad integradas como cortocircuito en las salidas y protección contra sobrecargas, lo que ayuda a garantizar un funcionamiento seguro y contra daños a los dispositivos y personas. Se le incorporó un cartucho de tipo TMC2AQ2V con 2 salidas de tensión analógica (de 0 a 10V), 12 bits y un módulo de ampliación tipo TM2AMM6HT con 4 entradas y 2 salidas de (0-10V, 4-20 mA), 12 bits. Los datos del PLC se muestran en la siguiente tabla [15]:

Tabla 1. Datos del PLC compacto M221.

Alimentación	Entradas	Salidas	Conexión
100 a 240 V AC	14 entradas digitales	10 salidas de relé NA (2A)	1 puerto línea serie
	2 entradas analógicas (0...10V)		1 puerto Ethernet

Elaborado por la autora

### 1.6 Sistema de control SoMachine Basic.

SoMachine Basic V1.5 es el *software* que se utilizó para programar el autómeta M221, ya que es el *software* de programación para los controladores del fabricante Schneider. Además de que es uno de los programas que se da como herramienta de trabajo en asignaturas de la carrera como “Autómatas Programables” y “Sistemas de Automatización”.

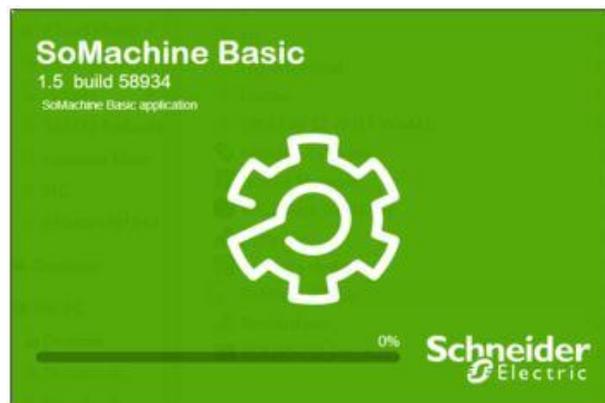


Figura 20 Logo del programa SoMachine Basic.

SoMachine Basic es una herramienta de programación gráfica diseñada para facilitar la configuración, el desarrollo y la puesta en marcha de programas para controladores lógicos que permite abrir aplicaciones creadas para controladores programables Twido y convertirlas en archivos de proyecto de SoMachine. Realiza la incorporación de funciones especiales como el control PID en controladores físicos. En el siguiente diagrama se muestran las fases típicas de desarrollo de un proyecto en SoMachine Basic (fichas Configuración, Programación y Puesta en marcha) [16]:

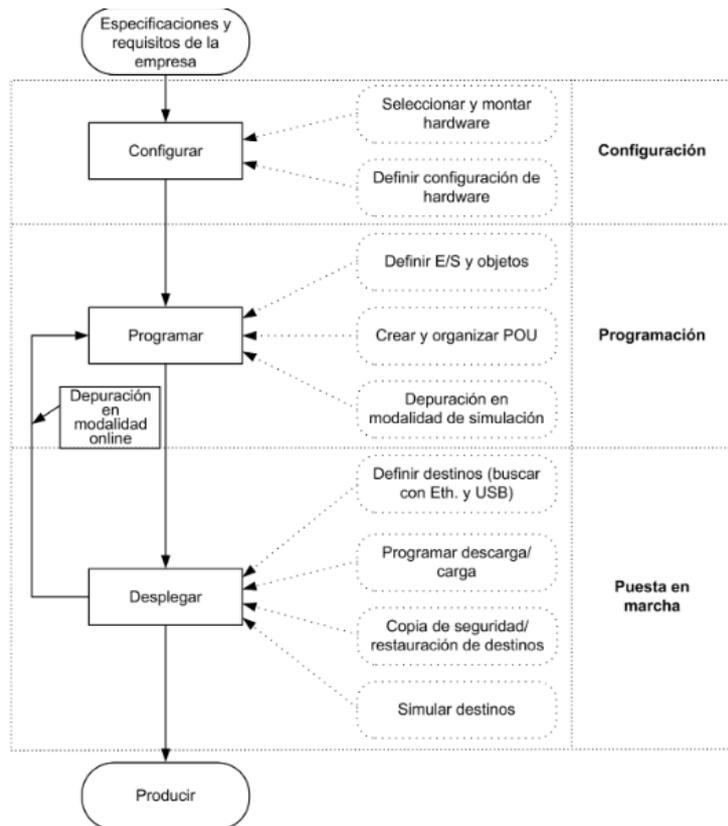


Figura 21 Entorno de trabajo en el *software* SoMachine Basic.

En SoMachine Basic se puede programar en tres tipos de lenguajes como son *Ladder*, Lista de Instrucciones y Gráficos de Funciones Secuenciales [12].

El lenguaje en el que se escribió el código fue en Diagrama de Escalera (*Ladder* o LD) ya que es uno de los lenguajes más usados para la programación de autómatas y uno de los enseñados en la carrera.

El diagrama en escalera, también *Ladder* o *Ladder Diagram*, es un lenguaje de programación gráfico muy popular dentro de los autómatas programables debido a que está basado en los esquemas eléctricos de control clásicos. De este modo, con los conocimientos que todo técnico o ingeniero eléctrico posee, es muy fácil adaptarse a la programación en este tipo de lenguaje. *Ladder* es uno de los diferentes lenguajes de programación para los controladores lógicos programables PLCs estandarizados con IEC 61131-3. En *Ladder*, la energía se desplaza de izquierda a derecha en lugar de arriba hacia abajo como en los esquemas eléctricos. En un circuito típico aparecen los contactos en la parte izquierda y una bobina en la parte derecha. La lógica de control que representa

dicho circuito puede verse como una inferencia lógica que tiene como antecedente la lógica de los contactos y como concluyente la bobina [16].

### **Conclusiones Parciales**

En este capítulo se realizó una investigación sobre el surgimiento de las prácticas de laboratorio así como la importancia de su desarrollo. Se analizaron desde un punto de vista conceptual los *softwares* de simulación y se llevó a cabo una búsqueda y selección de un entorno de simulación que permitiera recrear procesos automatizados industriales. Se describió de forma detallada el *software* Factory I/O, que fue el seleccionado para la realización del proyecto, además de que se hizo alusión brevemente al PLC M221 y al programa SoMachine Basic.

## CAPITULO 2. CONTROL DE PROCESOS INDUSTRIALES PREDEFINIDOS Y/O MODIFICADOS EN EL SOFTWARE FACTORY IO.

En este capítulo se describen las escenas escogidas, predefinidas o creadas empleando el programa Factory I/O, teniendo en cuenta los sensores y actuadores presentes en cada una. De igual manera se realiza el modelado funcional en Grafcet de estas atendiendo a la lógica de la solución propuesta y se programan en lenguaje *Ladder* los algoritmos de control empleando el *software* SoMachine Basic tomando en consideración la cantidad de variables necesarias de entrada/salida y su tipo.

### 2.1 Control de movimiento de objetos.

Esta primera escena a desarrollar constituye la segunda predefinida en el Factory I/O con el nombre “*From A to B (Set and Reset)*” y su objetivo es mover un objeto desde un punto A hasta un punto B, como se puede observar en la figura 18 desde una vista superior, haciendo uso de bobinas tipo *Set* y *Reset*, aunque puede ser resuelta empleando bobinas normales con su respectivo enclavamiento.

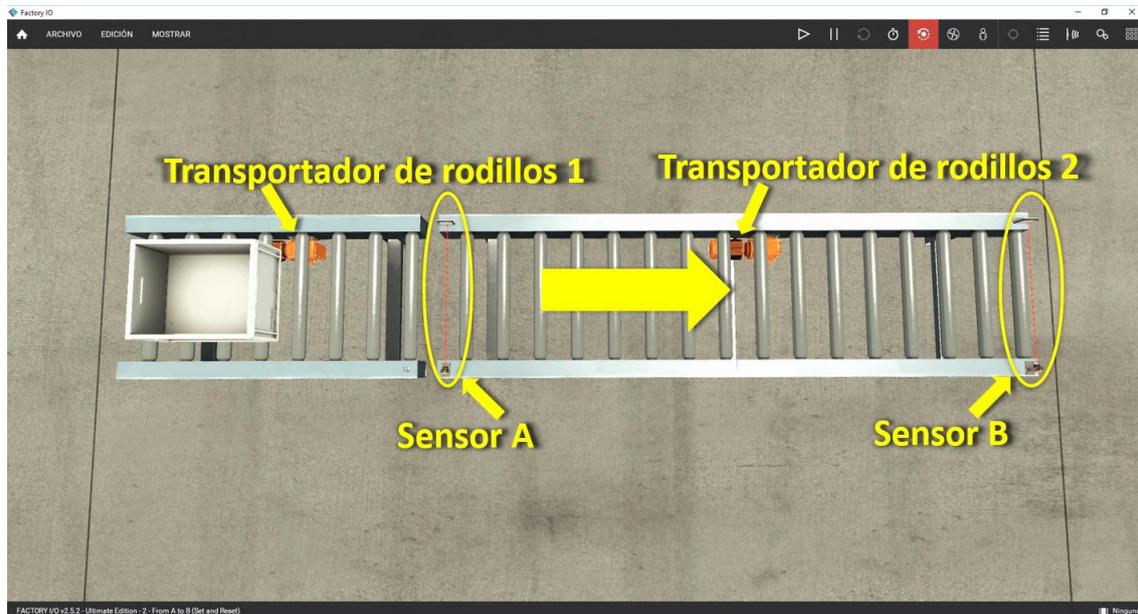


Figura 22 Escena para el control de movimiento de objetos.

Fuente: Autora, 2023.

#### 2.1.1 Componentes de la escena

En la figura 18 también pueden ser apreciados los componentes que conforman la escena, dentro de los que se encuentran:

- Transportador de rodillos 1: transportador pequeño situado a la izquierda, que en este caso se encuentra forzado (activo) por defecto para simular la entrada de la caja apilable al transportador 2.
- Transportador de rodillos 2: transportador más largo que el anterior situado a la derecha, a través del cual se moverá el objeto, en este caso la caja apilable, desde la posición A hasta la B.
- Sensores A y B: Sensores ópticos retroreflectivos ubicados al inicio y final del transportador de rodillos 2 que determinan la posición de la caja sobre el transportador de rodillos 2.

Teniendo en cuenta la explicación anterior, para este caso solo se tiene el transportador de la derecha como elemento a controlar y que será referido como **cinta**, y los sensores **A** y **B** como entradas, los cuales son negados, es decir entregan un valor lógico falso cuando detectan presencia de caja.

### 2.1.2 Modelado y programación del control

Con el fin de realizar el control de la misma se plantea el Grafcet de la figura 19 que describe la siguiente lógica de funcionamiento: cuando el sensor A detecta la caja proveniente del transportador 1 se debe poner en funcionamiento el 2 hasta que la caja llegue a donde se encuentra el sensor B.

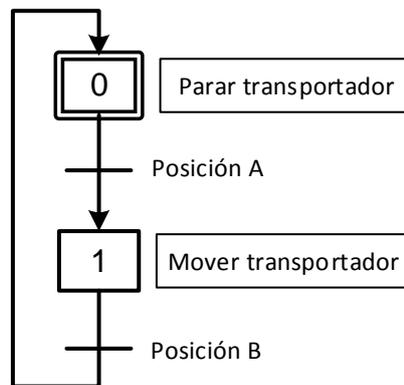


Figura 23 Modelado Grafcet del funcionamiento de la escena.

Fuente: Autora, 2023.

Es importante mencionar que dada la naturaleza del *driver* Modbus TCP/IP utilizado para intercambiar los datos entre el código implementado en el simulador de PLC del *software* SoMachine Basic de Schneider y el entorno virtual

Factory I/O, son empleadas marcas internas como entradas/salidas como se muestra en la siguiente tabla.

Tabla 2.1 Relación de variables entradas/salidas.

Sensores (Entradas)	Actuadores (Salidas)
A (M0)	Cinta (M2)
B (M1)	

Fuente: Autora, 2023.

Para esta escena se desarrollaron dos códigos en lenguaje LD empleando bobinas sin retención de estado en un caso y bobinas con retención de estado (*Set* y *Reset*) en otro, como se puede apreciar en la figura 20 a) y b) respectivamente.

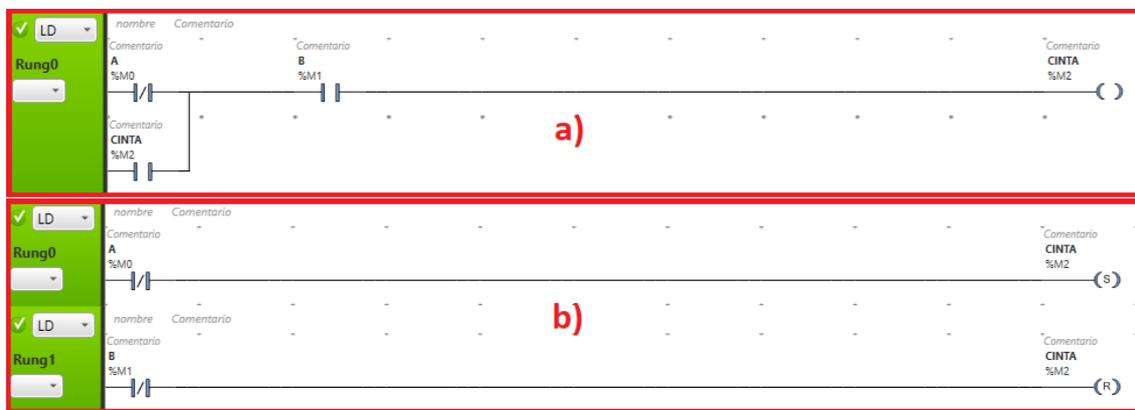


Figura 24 Código LD implementado con: a) bobinas sin retención y b) bobinas con retención (*Set* y *Reset*).

Fuente: Autora, 2023.

## 2.2 Control de movimiento de objetos con panel de control.

Esta segunda escena no es más que el resultado de la modificación de la anterior como se puede apreciar en la figura 21 donde se mantendrá el flujo de cajas y se utilizará un panel de control permitiendo probar el tránsito por los diferentes estados de un proceso de manera simple.

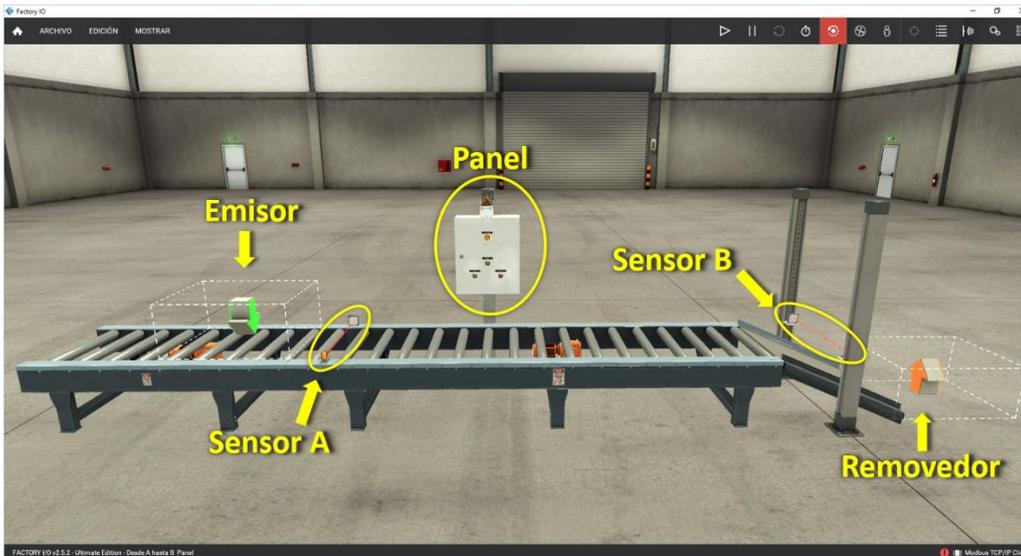


Figura 25 Escena para el control de movimiento de objetos con panel.

Fuente: Autora, 2023.

### 2.2.1 Componentes de la escena

Esta escena contiene los componentes de la escena anterior además de otros que fueron añadidos dentro de los que se pueden encontrar:

- Emisor: Elemento ficticio que introduce objetos, en este caso un contenedor tipo caja apilable, encima del transportador de rodillos 1 que continua al igual que la escena anterior forzado manteniendo el flujo de dichas cajas al transportador 2.
- Soporte: estructura al final del transportador de rodillos 2 en el que se posicionó el sensor B por fuera del transportador.
- Rampa: estructura ubicada a la derecha del transportador de rodillos 2 para que la caja al llegar al final descienda hacia el removedor.
- Removedor: elemento ficticio que remueve objetos ubicado al pie de la rampa para que la caja al llegar ahí desaparezca, creando un lazo infinito siempre y cuando el sistema esté en funcionamiento.
- Cuadro eléctrico: panel de control que consta a su vez de los siguientes elementos:
  - Parada de emergencia: pulsador normalmente cerrado con función mecánica de enclavamiento para detener la máquina ante cualquier situación anómala.
  - Iniciar: pulsador normalmente abierto para el arranque del sistema.

- Parar: pulsador normalmente cerrado para la parada del sistema.
- Rearme: pulsador normalmente abierto para reiniciar el sistema antes de volver arrancarlo luego de una parada de emergencia.
- Luz de advertencia: indicador lumínico empleado para cuando el sistema se encuentra en parada de emergencia.

## 2.2.2 Modelado y programación

Luego para la solución de esta escena se diseña el Grafset principal mostrado a la izquierda de la figura 22, donde se describe la lógica relacionada a los modos de funcionamiento del sistema según los pulsadores en el panel de control y en el cual existe una macroetapa que contiene el funcionamiento normal del sistema descrito por el Grafset desarrollado en la escena anterior.

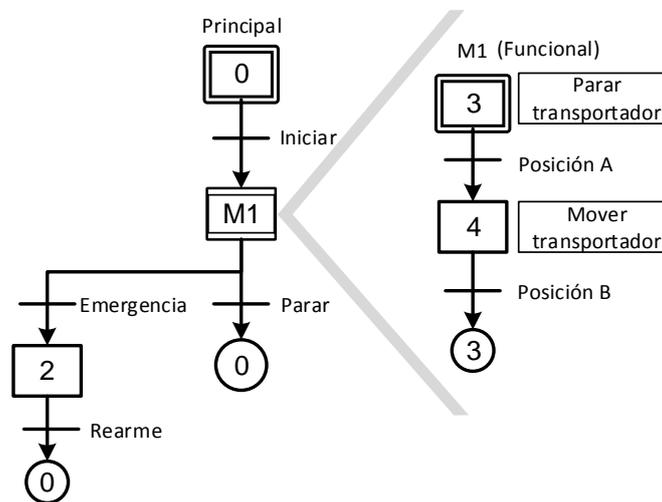


Figura 26 Modelado Grafset del funcionamiento de la escena.

Fuente: Autora, 2023.

Tomando en consideración todo lo añadido a la escena, y que cada pulsador tiene además su propio indicador piloto (P\_Inicio, P\_Parada, P\_Rearme) la relación entradas/salidas quedaría como se muestra en la tabla a continuación.

Tabla 2.2 Relación de variables entradas/salidas.

Entradas		Salidas	
Sensores	Pulsadores	Actuadores	Indicadores
A (M0)	Inicio (M2)	Cinta (M6)	P_Inicio
	Parada (M3)		P_Parada
B (M1)	Rearme (M4)		P_Emergencia (M9)
	Emergencia (M5)		P_Rearme

Fuente: Autora, 2023.

En este caso el código LD desarrollado que cumple con esta lógica se puede encontrar en el anexo 2.

### 2.3 Control de estación separadora.

Esta escena constituye la octava predefinida en el Factory I/O y como su nombre en inglés muestra “*Buffer Station*” el objetivo es separar hasta 5 cajas según la descripción del propio *software*. En un entorno industrial este tipo de estación es utilizada para ralentizar el flujo del proceso desagrupando las cajas que vienen una detrás otra, sabiendo que la estación a continuación posee una velocidad de procesamiento inferior, por ejemplo, un paletizador.

La figura 23 muestra los componentes de esta escena que fue modificada básicamente en lo que refiere a su panel de control y los modos de operación y funcionamiento con el objetivo de mostrar el uso de contadores en ciclos repetitivos, además de proporcionar el empleo de distintos modos de operación (manual o automático) y darle al usuario la libertad de tomar decisiones a la hora separar las cajas en grupos (de dos, o de cuatro).

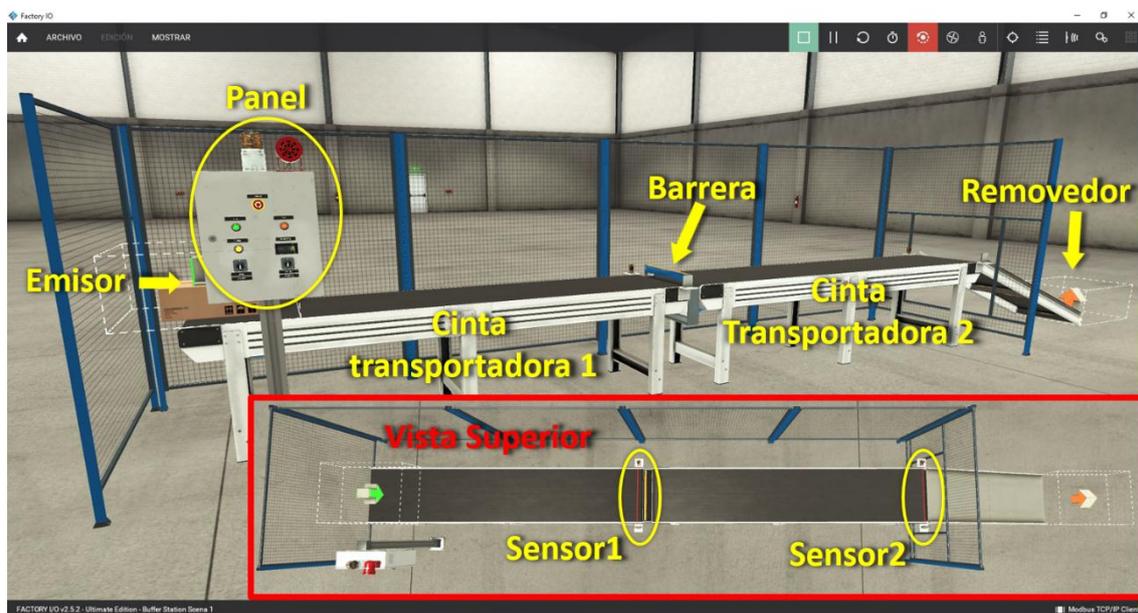


Figura 27 Escena para el control de estación separadora.

Fuente: Autora, 2023.

#### 2.3.1 Componentes de la escena

En esta escena de forma general son utilizados los componentes descritos a continuación:

- Emisor: elemento ficticio que introduce objetos, en este caso cajas, encima de la cinta transportadora 1.
- Cinta transportadora 1: cinta situada a la izquierda encargada de acumular las cajas provenientes del emisor.
- Barrera: elemento situado entre las cintas transportadoras 1 y 2 que detiene el paso de cajas acumulándolas en la cinta 1 hasta que la 2 sea liberada.
- Cinta transportadora 2: cinta situada a continuación de la primera por donde saldrán los grupos de cajas según se la cantidad de cajas seleccionada.
- Removedor: elemento ficticio que remueve objetos ubicado al pie de la rampa para que la caja al llegar ahí desaparezca, creando un lazo infinito siempre y cuando el sistema esté en funcionamiento.
- Sensor 1: sensor óptico retroreflectivo ubicado al final de la cinta transportadora 1, justo antes de la barrera, para controlar el número de cajas que se dejan pasar.
- Sensor 2: sensor óptico retroreflectivo ubicado al final de la cinta transportadora 2, justo antes de la rampa, para asegurarse de que dicha cinta este limpia.
- Cuadro eléctrico: panel de control que consta a su vez de los siguientes elementos:
  - Iniciar: pulsador normalmente abierto para el arrancar el sistema.
  - Parar: pulsador normalmente cerrado para detener el sistema antes de hacer cualquier cambio a algún modo o después de haber ocurrido alguna emergencia, hasta que se vuelva a iniciar desde cero.
  - Parada de emergencia: pulsador normalmente cerrado con función mecánica de enclavamiento para detener el sistema ante cualquier situación anómala.
  - Manual/Automático: selector para escoger el modo de operación.
  - Dos cajas/Cuatro cajas: selector para establecer la cantidad de cajas que debe salir cada vez.

- E Barrera: un pulsador normalmente abierto para manipular la barrera en modo manual. Con cada pulso de este la barrera es activada si se encontraba inactiva o desactivada si se encontraba activa.
- Luz de advertencia y Sirena de alarma: indicadores lumínico y sonoro respectivamente empleados para cuando el sistema se encuentra en parada de emergencia.
- Pantalla digital: indicador numérico en el que se muestra según la cantidad de cajas seleccionadas el número que transitan.

El funcionamiento normal de esta escena puede ser en modo manual o automático, cuyas funciones son descritas a continuación:

- Modo manual: se debe seleccionar el grupo de cajas que se desee formar e iniciar el proceso, activándose las dos cintas transportadoras. Inicialmente la barrera se encuentra desactivada, dejando pasar las cajas que ocupan la cinta transportadora 1 a la 2. En este caso, el operador debe estar pendiente al número de cajas según la cantidad que se haya seleccionado, para bloquear el paso presionando el botón E\_Barrera, lo que detiene la primera cinta. Al salir las cajas de la segunda cinta el operador debe presionar el mismo botón que hará que la barrera se desactive y se ponga en marcha la primera cinta, continuando el tránsito de cajas.

Si el operador no activa la barrera a tiempo y pasan más cajas que las previstas según la opción seleccionada desde la primera cinta hasta la segunda inmediatamente se detiene el proceso y se enciende la luz indicadora, hasta que se presione el botón de parada.

- Modo automático: se debe seleccionar la cantidad de cajas que se desee (dos o cuatro) e iniciar el proceso, activándose las dos cintas transportadoras. Igual que en el modo manual inicialmente la barrera se encuentra desactivada, dejando pasar las cajas que ocupan la cinta transportadora 1 a la 2, que son contabilizadas por el sensor ubicado al final de la primera cinta. Al alcanzar la cantidad seleccionada se activa automáticamente la barrera impidiendo el paso de más cajas y deteniendo la primera cinta. El sensor ubicado al final de la segunda cinta comprueba que esta esté vacía al contabilizar la misma cantidad seleccionada,

desactivando la barrera y poniendo en marcha la primera cinta para repetir el ciclo.

En caso de que ocurrir algún fallo se presiona el botón de emergencia y se detiene el proceso hasta que se presione el botón de parada y se comience desde cero. Durante el estado de emergencia la luz de advertencia y la sirena de alarma se mantienen activas hasta que el fallo sea eliminado y el proceso vuelva a su estado de espera presionando nuevamente el botón de emergencia.

### 2.3.2 Modelado y programación

Con el propósito de modelar el funcionamiento de esta escena fue desarrollado el conjunto de Grafcet funcionales que se muestran en a figura 24 descritos a continuación:

**PRINCIPAL:** donde se modela el transito del sistema por los posibles estados según los controles del panel.

**AUTOMATICO:** modela la secuencia en que son separadas automáticamente las cajas según la cantidad seleccionada.

**MANUAL:** modela la posibilidad de acción del operador sobre la barrera bloqueando o no el paso de cajas.

**ENTRADA/SALIDA:** modelan el funcionamiento de los contadores de cajas a la entrada y salida de la cinta transportadora 2 que funcionan en modo manual de manera independiente como asistencia al operador.

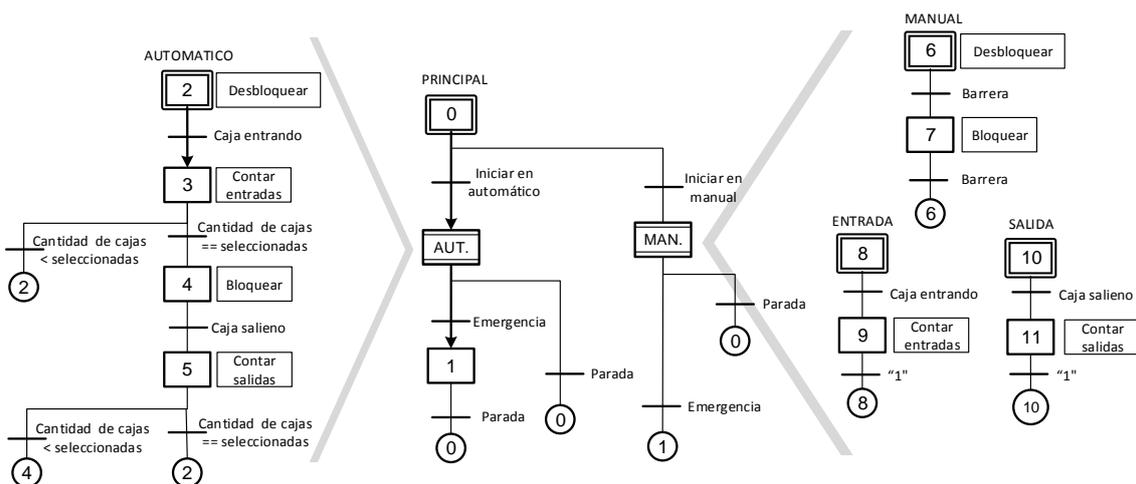


Figura 28 Modelado Grafcet del funcionamiento de la escena.

Fuente: Autora, 2023.

El código LD desarrollado que cumple con esta lógica en este caso se puede encontrar en el anexo 3.

Tabla 2.3 Relación de variables entradas/salidas.

Entradas		Salidas	
Sensores	Pulsadores	Actuadores	Indicadores
Sensor1 (M9)	Iniciar (M0)	Cinta 1 (M13)	Alarma (M17)
	Parar (M1)		
	Manual (M2)	Cinta 2 (M14)	Alarma Sirena (M17)
	E_Barrera (M3)		
Sensor2 (M10)	Automático(M4)	S_Barrera (M15)	Pantalla (MW19)
	Dos Cajas(M5)		
	Cuatro Cajas(M6)		
	Emergencia(M7)		

Fuente: Autora, 2023.

## 2.4 Control de una línea de llenado de botellas.

Esta escena constituye una prueba de cómo se puede a partir de escenas ya predefinidas, de elementos brindados por el Factory /IO y un poco de imaginación crear sistemas más complejos.

En la figura 25 se muestra una vista superior de lo que se denominó una “línea de llenado de botellas” donde se identifican escenas predefinidas ya existentes en el Factory IO como las enmarcadas en las zonas 1 y 3, y otra desarrollada a partir de los componentes que ofrece el software enmarcada en la zona 2.



Figura 29 Vista superior de la línea de llenado de botellas.

Fuente: Autora, 2023.

El objetivo de esta línea de llenado de botellas es realizar el control de variables analógicas y digitales, teniendo en cuenta tanto los sistemas puramente secuenciales como aquellos en que se comparte un recurso y se debe tomar decisiones según determinadas reglas. Luego para una mejor comprensión esta escena híbrida será analizada por partes, tal y como se pudo observar en la figura anterior, donde:

- **Zona 1:** corresponde a la escena 12 del Factory I/O de nombre “*Level Control*” cuyo objetivo es el control del nivel de líquido o flujo en un tanque usando un controlador proporcional integral derivativo (PID).
- **Zona 2:** corresponde a una escena totalmente diseñada con los componentes ofrecidos en el programa.
- **Zona 3:** corresponde a la escena 9 del Factory I/O de nombre “*Converge Station*” donde dos caminos independientes convergen en uno que resulta de uso común, haciendo que su objetivo sea el control eficiente del recurso compartido de acuerdo a determinadas reglas de prioridad.

#### **2.4.1 Zona 1: control de nivel en un tanque.**

De forma general esta escena predefinida está constituida por un tanque translúcido a través del cual se puede observar el líquido y que en una de sus paredes presenta un indicador de nivel de barra como se muestra en la siguiente figura.

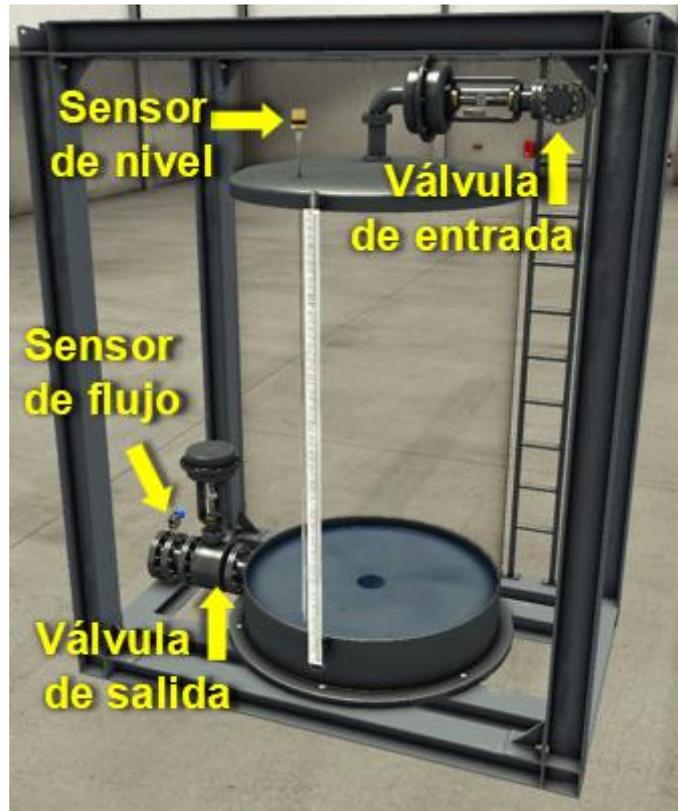


Figura 30 Escena correspondiente a la Zona1.

Fuente: Autora, 2023.

La instrumentación presente en esta escena se compone de los siguientes elementos:

- Una válvula a la entrada del tanque que controla el flujo de líquido.
- Una sonda de nivel situada de manera vertical en la cima del tanque que provee de una medición continua del nivel de líquido en el tanque.
- Una válvula a la salida del tanque que controla el flujo de líquido.
- Un sensor de flujo a la salida del tanque que provee de una medición continua de caudal de líquido.

Originalmente esta escena del tanque viene por defecto con un panel que no fue utilizado al formar parte de una línea de producción mayor. A pesar de que en el Factory I/O se promueve el reto de controlar el nivel utilizando un PID, es necesario aclarar que el simulador de PLC correspondiente al software SoMachine Basic empleado para controlar los procesos virtuales en el Factory I/O en este trabajo no permite la ejecución del PID en modo simulación a no ser que el PLC esté conectado físicamente, por lo que se diseñó un control ON/OFF modelado según el Grafcet funcional que se muestra a continuación.

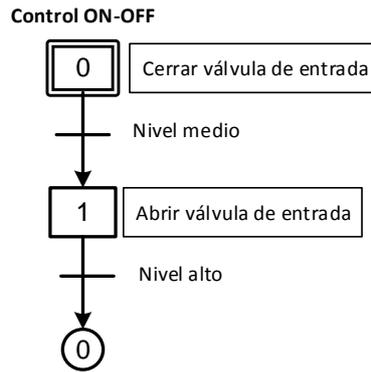


Figura 31 Grafcet funcional del control de nivel.

Fuente: Autora, 2023.

### 2.4.2 Zona 2: control del llenado de botellas

De forma general esta escena completamente creada con los componentes ofrecidos por el Factory I/O está constituida por dos líneas de sistemas idénticos en cuanto a componentes, situados una al lado de la otra, como se puede apreciar en la figura 28. Teniendo en cuenta que el Factory I/O no tiene botellas y no permite añadir o modificar sus componentes es válido aclarar que son empleadas cajas para simularlas.

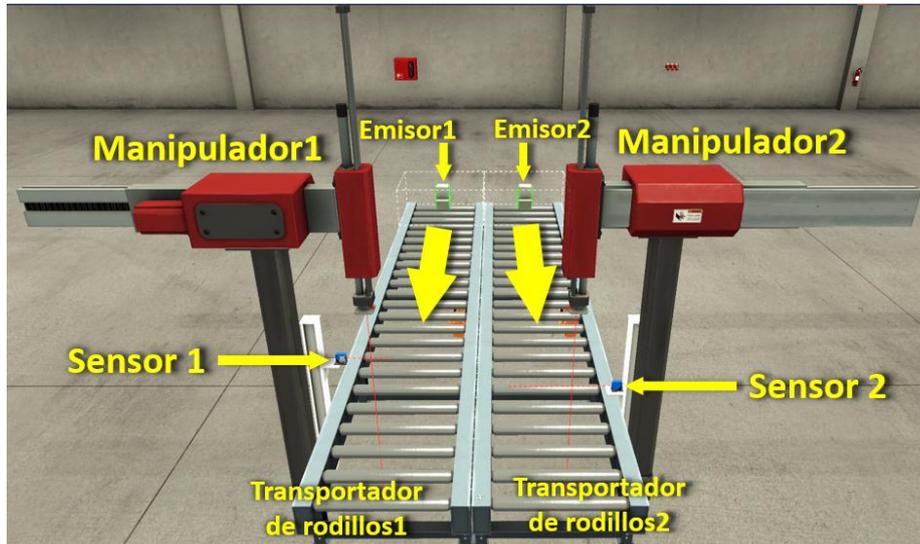


Figura 32 Escena correspondiente a la Zona 2.

Fuente: Autora, 2023.

Cada uno de estos sistemas está compuesto por los siguientes elementos:

- Transportador de rodillos: transportador encargado de mover las cajas desde su entrada a la posición de llenado, y de la posición de llenado a la zona 3.

- Emisor: elemento ficticio que introduce objetos, en este caso cajas, encima del transportador de rodillos.
- Sensor: sensor fotoeléctrico difuso empleado para detectar cuando la caja está en la posición de llenado.
- Manipulador: mecanismo cuyo objetivo es simular un cabezal de llenado.

El funcionamiento de esta escena tomando como ejemplo solo una de las líneas de llenado comienza con el transportador de rodillos moviéndose en sentido del cabezal de llenado o manipulador. Al detectarse la botella debajo del cabezal se detiene el transportador y este baja según la altura de la botella, permaneciendo así el tiempo necesario para llenarla. Transcurrido ese tiempo el cabezal vuelve a su posición original y se arranca el transportador hasta que llegue una nueva botella para ser llenada.

Como se aprecia en la figura 29, por el transportador de la izquierda pasarán botellas (cajas) de mayor altura que las botellas (cajas) que pasan por el derecho, aunque menos anchas, pudiendo diferenciar el tiempo necesario para ser llenadas.



Figura 33 Cajas empleadas para simular botellas: a) Cajas del transportador 1 b) Cajas del transportador 2.

Fuente: Autora, 2023.

Según el funcionamiento descrito anteriormente del proceso de llenado, se diseñó el Grafcet funcional que se muestra a continuación.

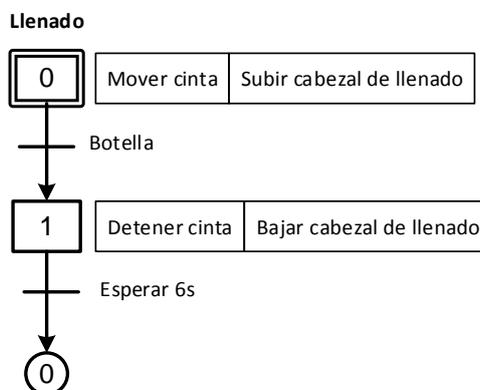


Figura 34 Grafcet funcional del llenado.

Fuente: Autora, 2023.

### 2.4.3 Zona 3: control de estación de convergencia

Esta última escena predefinida aparece en el Factory I/O en el lugar 9 con el nombre de “*Converge Station*”, y como su nombre en inglés indica, es una estación donde dos caminos convergen en uno solo como se puede apreciar en la siguiente figura.

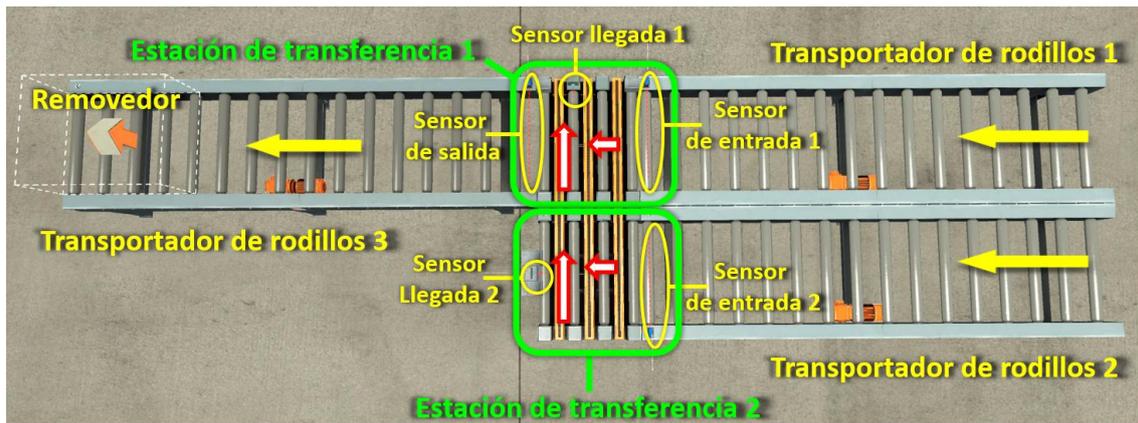


Figura 35 Escena correspondiente a la Zona 3.

Fuente: Autora, 2023.

Dentro de los elementos que componen esta escena podemos encontrar (ver figura 31):

- Transportador de rodillos 1: encargado de mover las botellas llenas provenientes del transportador 1 de la línea de llenado en la zona 2.
- Sensor de entrada 1: sensor óptico difuso empleado para detectar cuando la botella llena en el transportador de rodillos 1 está a la entrada de la estación de transferencia 1.
- Transportador de rodillos 2: encargado de mover las botellas llenas provenientes del transportador 2 de la línea de llenado en la zona 2.
- Sensor de entrada 2: sensor óptico difuso empleado para detectar cuando la botella llena en el transportador de rodillos 2 está a la entrada de la estación de transferencia 2.
- Estación de transferencia 2: responsable por recibir la botella proveniente del transportador de rodillos 2 y trasladarla a la estación de transferencia 1.

- Sensor de llegada 2: sensor capacitivo empleado para detectar cuando la botella proveniente del transportador de rodillos 2 se encuentra en la estación de transferencia 2.
- Estación de transferencia 1: responsable por recepcionar la botella proveniente del transportador de rodillos 1 y trasladarla al transportador de rodillos 3, o de recepcionar la caja proveniente de la estación de transferencia 2 y trasladarla al transportador de rodillos 3.
- Sensor de llegada 1: sensor capacitivo empleado para detectar cuando la botella proveniente de la estación de transferencia 2 se encuentra en la estación de transferencia 1.
- Sensor de salida: sensor óptico difuso empleado para detectar cuando las botellas abandonan la estación de transferencia 1 hacia el transportador de rodillos 3.
- Transportador de rodillos 3: encargado de recepcionar las botellas llenas provenientes de la estación de transferencia 1.
- Removedor: elemento ficticio que remueve objetos ubicado al final del transportador de rodillos 3 para que las botellas desaparezcan al llegar ahí, creando un lazo infinito siempre y cuando el sistema esté en funcionamiento.
- Cuadro eléctrico: panel de control que consta a su vez de los siguientes elementos:
  - Iniciar: pulsador normalmente abierto para el arrancar el sistema.
  - Parar: pulsador normalmente cerrado para detener el sistema.
  - Normal/Vaciar: selector que permite una vez apagado el sistema vaciar el tanque.
  - Luz de advertencia: indicador lumínico empleado para señalar cuando están ocupadas cualquiera de las estaciones de transferencia.
  - Pantalla digital: indicador numérico en el que se muestra el nivel de líquido en el tanque.

En el funcionamiento de esta escena es necesario tener en cuenta los posibles casos mostrados en la figura 32 que pueden originarse según el orden detectado de las cajas provenientes de los transportadores de rodillos 1 y 2.

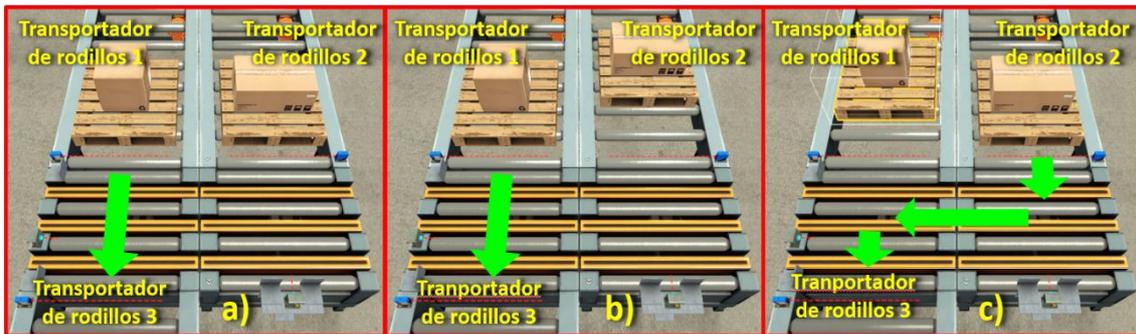


Figura 36 Posibles casos según el orden de llegada de las botellas: a) Caso 1 b) Caso 2 y c) Caso 3.

Fuente: Autora, 2023.

En esta figura es posible apreciar que el caso 1 corresponde a cuando las botellas llegan simultáneamente por el transportador de rodillos 1 y 2, donde tendrá prioridad la proveniente del primero, que será transferida directamente a la salida. El caso 2 corresponde a cuando llega la botella del transportador de rodillos 1 primero, siendo transferida igualmente a la salida. El caso 3 se pone de manifiesto cuando la botella del transportador de rodillos 2 llega primero, momento en que harán falta unos 3 movimientos hasta poder colocar dicha botella en la salida.

Es importante aclarar que, si la estación de transferencia 1 o 2 se encuentra ocupada, cualquier otra botella detectada a la entrada deberá esperar a que el proceso de transferencia haya concluido, momento en el cual el sistema continuará transfiriendo dando prioridad siempre a la botella proveniente del transportador de rodillos 1.

Siguiendo la lógica del funcionamiento descrito anteriormente para esta escena se diseñó el Graficet mostrado en la figura 33 donde se representan varios procesos interrelacionados entre sí y no secuenciales en su totalidad, debido a la característica de esta estación en la que se comparte un mismo recurso. Luego se pueden apreciar un diagrama principal denominado casos del cual dependen las secuencias correspondientes de transferencias 1 y 2, y dos diagramas nombrados transportador 1 y 2 que representan los dos posibles estados de estos, y las condiciones para transitar de uno a otro en cada caso.

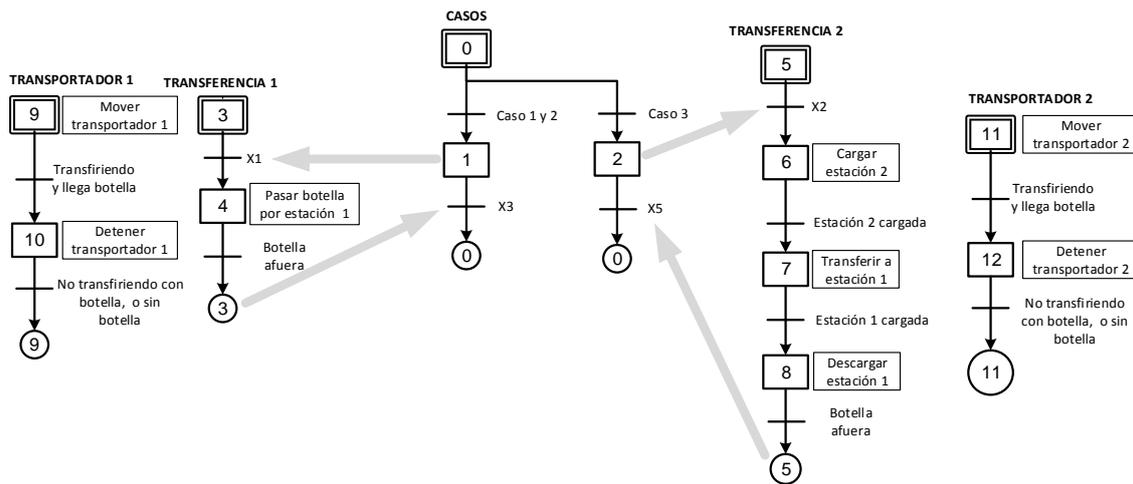


Figura 37 Grafcet funcional de la estación de convergencia.

Fuente: Autora, 2023.

#### 2.4.4 Programación

Para el control de la línea de llenado de botellas se realizó la programación en lenguaje LD teniendo en cuenta los Grafcet modelados para cada una de las zonas anteriormente descritas y tomando en consideración las interrelaciones existentes entre ellas que se enuncian a continuación:

- Las cintas que introducen las botellas en la zona 2 solo son activadas una vez que el líquido en el tanque de la zona 1 alcanza el nivel medio al arrancar el sistema.
- Durante el tiempo que permanecen los cabezales de llenado abajo una vez que se detecta botellas la válvula de salida del tanque en la zona 1 es abierta en según el tamaño de la botella.
- Una vez arrancado el sistema la zona 3 se mantiene en funcionamiento esperando por las botellas provenientes de la zona 2 hasta que se detenga el proceso.

En la tabla 2.4 se muestra la relación de variables entrada/salida empleadas en el desarrollo del programa, donde no están incluidas las variables internas o auxiliares. El código correspondiente para el control de esta escena híbrida puede ser encontrado en el anexo 4.

Tabla 2.4 Relación de variables entradas/salidas.

Entradas		Salidas	
Sensores	Controles	Actuadores	Indicadores
S_Cajas1 (M3)	Iniciar (M0)	C_Entrada1 (M18)	Ocupado (M27)
S_Cajas2 (M4)	Parar (M1)	C_Entrada2 (M19)	Nivel (MW37)
Posición_M1 (MW33)	Normal (M12)	Cinta1 (M27)	
Posición_M2 (MW34)	Vaciar (M13)	Cinta2 (M27)	
S_Nivel (MW32)		Carrete1 (M22)	
SE_Cajas1 (M7)		Carrete2 (M23)	
SE_Cajas2 (M8)		Riel1 (M24)	
S_Carrete1 (M9)		Riel2 (M25)	
S_Carrete1 (M10)		C_Salida (M26)	
S_Salida (M15)		Piston1 (MW38)	
		Piston2 (MW39)	
		V_Entrada (MW40)	
		V_Salida (MW41)	

Fuente: Autora, 2023.

### Conclusiones Parciales

En este capítulo se desarrollan cuatro escenas que muestran desde un nivel básico hasta uno más avanzado las posibilidades que brinda el Factory I/O en la creación de sistemas industriales virtuales de automatización para el entrenamiento en la programación de controladores lógicos programables. Fue realizado el modelado funcional en Grafcet de cada una de estas según el funcionamiento propuesto y se elaboró el programa en lenguaje LD para el PLC M221 en el software SoMachine Basic.

La posibilidad de comunicación entre el Factory I/O y el simulador del SoMachine Basic por medio del driver de comunicación Modbus TCP/IP sirvió para validar el control efectuado sobre los procesos virtuales desarrollados, evidenciando las fortalezas de una plataforma de simulación 3D en la enseñanza en Ingeniería Automática.

## CONCLUSIONES

La realización de este trabajo investigativo arrojó las siguientes conclusiones:

- Los simuladores no solo permiten acelerar el proceso de desarrollo y validación de algoritmos sin riesgo de dañar cualquier máquina, sino que constituyen una manera eficaz de desarrollar prácticas incluso de forma no presencial.
- El entorno virtual Factory I/O se presenta como una herramienta útil que permite la práctica y experimentación sin restricciones de tiempo y espacio a través de escenas que recrean procesos industriales que pueden ser controlados por algoritmos programados sobre controladores lógicos programables.
- La posibilidad de creación de escenas a la imaginación del usuario del simulador Factory I/O abre un gran espectro de posibilidades en el entrenamiento en la programación de controladores lógicos programables, al mismo tiempo que garantiza motivación por parte del estudiante debido al entorno realista que este ofrece.

## RECOMENDACIONES

- Incorporar el uso del Factory I/O a la docencia diseñando guías según las habilidades específicas a lograr en los estudiantes en asignaturas como Medios Técnicos de Automatización, Autómatas Programables y Sistemas de Automatización.
- Realizar el control PID de nivel o flujo de líquido en la escena del tanque con el objetivo de aplicar y comprobar los conocimientos adquiridos en asignaturas como Ingeniería de Control y Control de Procesos.
- Continuar usando el software para el desarrollo de simulaciones que requieran algoritmos más complejos aplicando la guía de estudio de los modos de marchas y paradas, experimentando además la conexión con otros *software* de programación de controladores lógicos programables.

## BIBLIOGRAFIA

- [1] Sicma21, Automatización Industrial: qué es y por qué es tan importante, <https://www.sicma21.com> , Actualizado (12/3/2021), Accedido (1/10/2023).
- [2] EdsRobotics, La importancia de la automatización de procesos industriales, <https://www.edsrobotics.com> , Actualizado (27/7/2020), Accedido (1/10/2023).
- [3] Lacor, Guía para saber más sobre los PLC, <https://www.lacor.com> , Actualizado (2023), Accedido (1/10/2023).
- [4] Guilarte Gonzáles Orleans, (2016), Trabajo de diploma “Diseño e implementación de Prácticas de Laboratorio para asignaturas de la carrera de Ingeniería Automática en base al Arduino Uno y MatLab.”
- [5] Digital Iseci.  
[https://repository.icesi.edu.co/biblioteca\\_digital/bitstream/10906/68447/1/eleccion\\_herramienta\\_simulacion.pdf](https://repository.icesi.edu.co/biblioteca_digital/bitstream/10906/68447/1/eleccion_herramienta_simulacion.pdf)
- [6] PC-SIMU, PC SIMU: Usos y procesos de instalación, <https://cade-simu.com/pc-simu/>, Actualizado (2023), Accedido (5/10/2023).
- [7] Factory IO, *Next-Gen PLC Training 3D Factory Simulation*, <https://factoryio.com/>, Actualizado (2023), Accedido (5/10/2023).
- [8] Machines Simulator, Machines Simulator vs Factory IO, <https://www.programacionmultidisciplinar.com/machines-simulador-vs-factory-io/>, Actualizado (2023), Accedido (7/10/2023).
- [9] FESTO fluidSim, *Software de simulación fluidSim*, <https://www.festo.com/mx/es/e/educacion/aprendizaje-digital/simulacion-virtual-y-modelacion/>, Actualizado (2022), Accedido (20/10/2023)
- [10] Bárcena Martínez Alejandro, (Septiembre, 2021), Proyecto fin de grado “Automatización y visualización de un proceso industrial empleando a TIA Portal y Factory IO”.
- [11] Revista +Ciencia, Factory IO: Simulación 3D de Fábrica, <https://www.anahuac.mx/mexico/noticias/Factory-IO-Simulacion-3D-de-fabrica>, Actualizado (2023), Accedido (28/10/2023).
- [12] Prades Batalla Ramón, (2019-2020), Trabajo de fin de grado “Desarrollo de la automatización de un proceso de clasificación de cajas de una empresa logística con implementación mediante la herramienta de programación IEC 61131-3 Codesys y con simulación de la planta a través del *software* Factory IO”.

[13] Canales Cruz Bryan Joel, (Diciembre, 2022), Proyecto de Graduación “Programación de línea de llenado de botellas en TIA Portal con control PID”.

[14] Factory IO, Latest news, <https://factoryio.com/news/>, Actualizado (2023), Accedido (29/10/2023).

[15] Schneider Electric, Catálogo del PLC M221CE24R.

[16] *Software* SoMachine versión 1.5, Schneider Electric, [www.schneiderelectric.com](http://www.schneiderelectric.com), SoMachine Basic Guía de funcionamiento (12/2017).

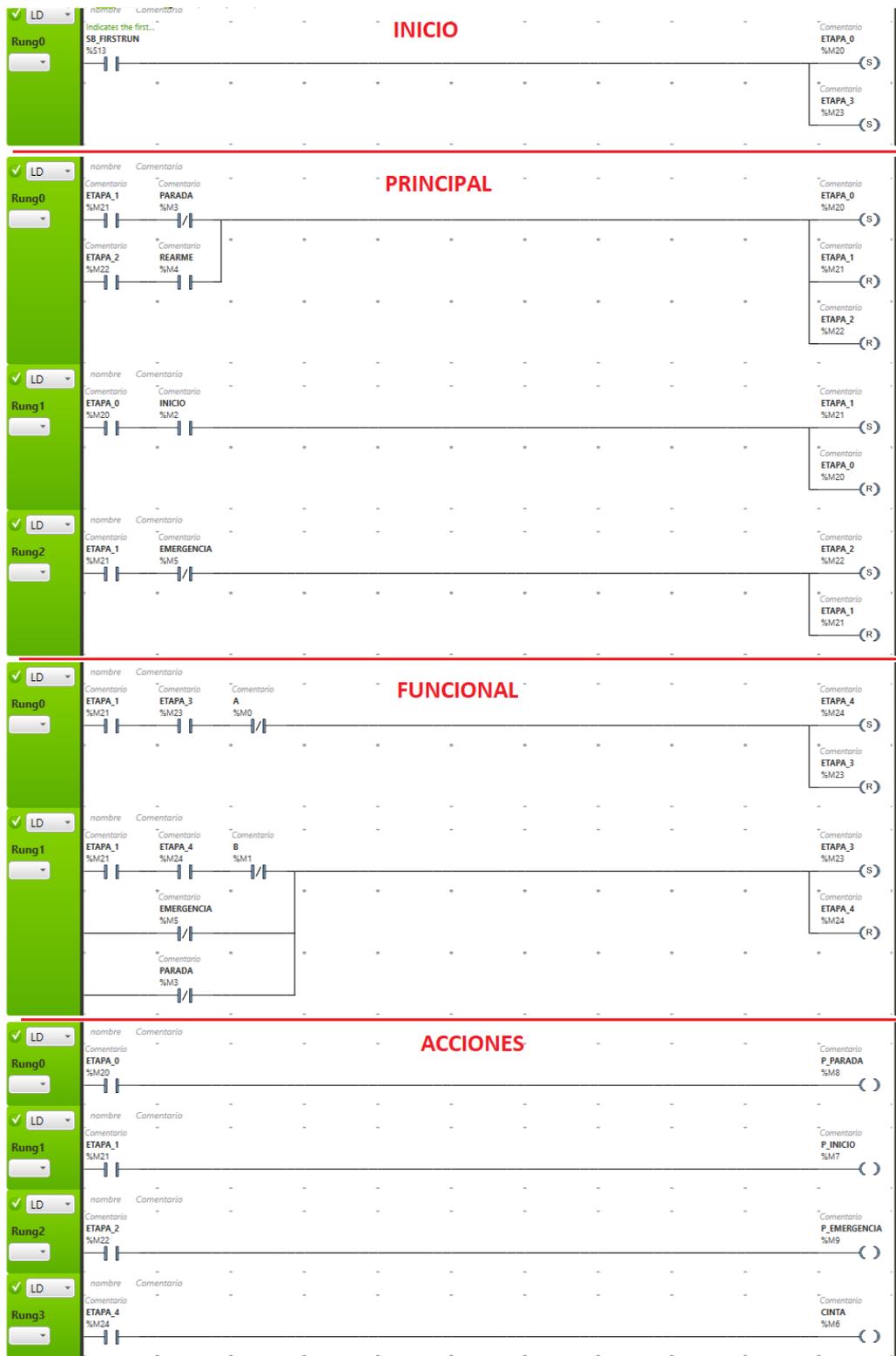
## ANEXOS

### Anexo 1. Especificaciones de los componentes del Factory IO.

Denominación en Factory IO	Denominación en el proyecto	Cantidad (Ud.)	Descripción
Emitter	Emisor	6	Emite un objeto para ser usado en la escena
Remover	Salida	14	Retira un objeto de la escena
Palletizing box	Caja pequeña	-	3 kg
Box (S)	Caja mediana	-	8 Kg
Box (L)	Caja grande	-	15 kg
Pallet	Pallet	-	20 kg
Blue raw material	Material crudo azul	-	Material para ser tratado color azul, 8 Kg
Green raw material	Material crudo verde	-	Material para ser tratado color verde, 8 Kg
Stackable box	Caja apilable	-	15 kg, puede almacenar material
Chain transfer	Cinta de transferencia	8	Cinta para desplazar carga a cintas adyacentes
Loading conveyor	Cinta de carga	5	Cinta para cargar la grúa
Chute conveyor low	Rampa	10	Rampa para desechar carga
Roller conveyor (2m)	Cinta de rodillos	7	Cinta transportadora de pallets y cajas sobre pallets
Roller conveyor (4m)	Cinta de rodillos	2	
Roller conveyor (6m)	Cinta de rodillos	4	
Belt conveyor (2m)	Cinta transportadora	2	Cinta transportadora de material
Belt conveyor (4m)	Cinta transportadora	4	
Aligner 2	Guía	21	Estructura de metal para guiar y alinear objetos sobre cintas
Aligner 3			
Wheel aligner	Rueda guía	6	Rueda para guiar objetos
Metal corner	Esquina metálica	7	Usada para situar sensores
Pusher	Empujador	8	Empujador neumático de objetos
Bracket	Arco	1	Estructura metálica para situar sensores
Diffuse sensor	Sensor	29	Puede detectar cualquier sólido
Vision sensor	Cámara	1	Reconoce las características de materiales
Light array emitter	Sensor de altura	1	Conjunto de láseres paralelos para detectar diferentes alturas
Light array receiver			
Pick & place	Pick & place	2	Puente grúa de 3 ejes para mover cajas o material
Stacker crane	Grúa	3	Grúa situada sobre railes para almacenar pallets y cajas sobre pallets
Rack	Almacén	6	Almacén de pallets y cajas sobre pallets
Two-axis Pick & place	Two-axis pick & place	3	Brazo de 2 ejes para mover cajas o material
Platform (S)	Plataforma	-	Estructuras de metal utilizadas para elevar dispositivos
Platform (M)			
Platform (L)			
Platform (XL)			

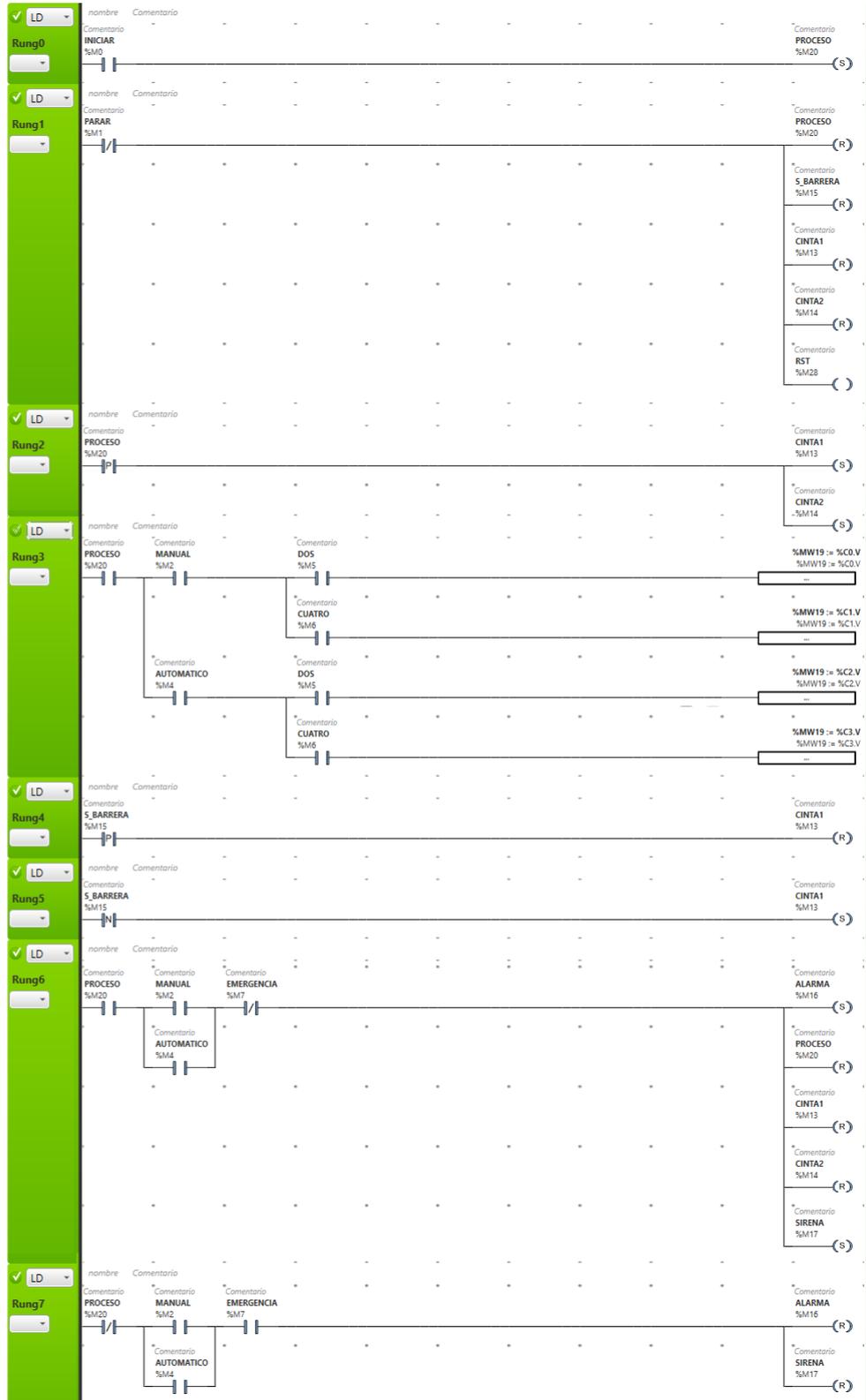
# Anexo 2 Programación ESCENA 2

- Tarea maestra
  - 1 - INICIO
    - Rung0
  - 2 - PRINCIPAL
    - Rung0
    - Rung1
    - Rung2
  - 3 - FUNCIONAL
    - Rung0
    - Rung1
  - 4 - ACCIONES
    - Rung0
    - Rung1
    - Rung2
    - Rung3
- Tarea periódica
- Eventos
- POU disponibles

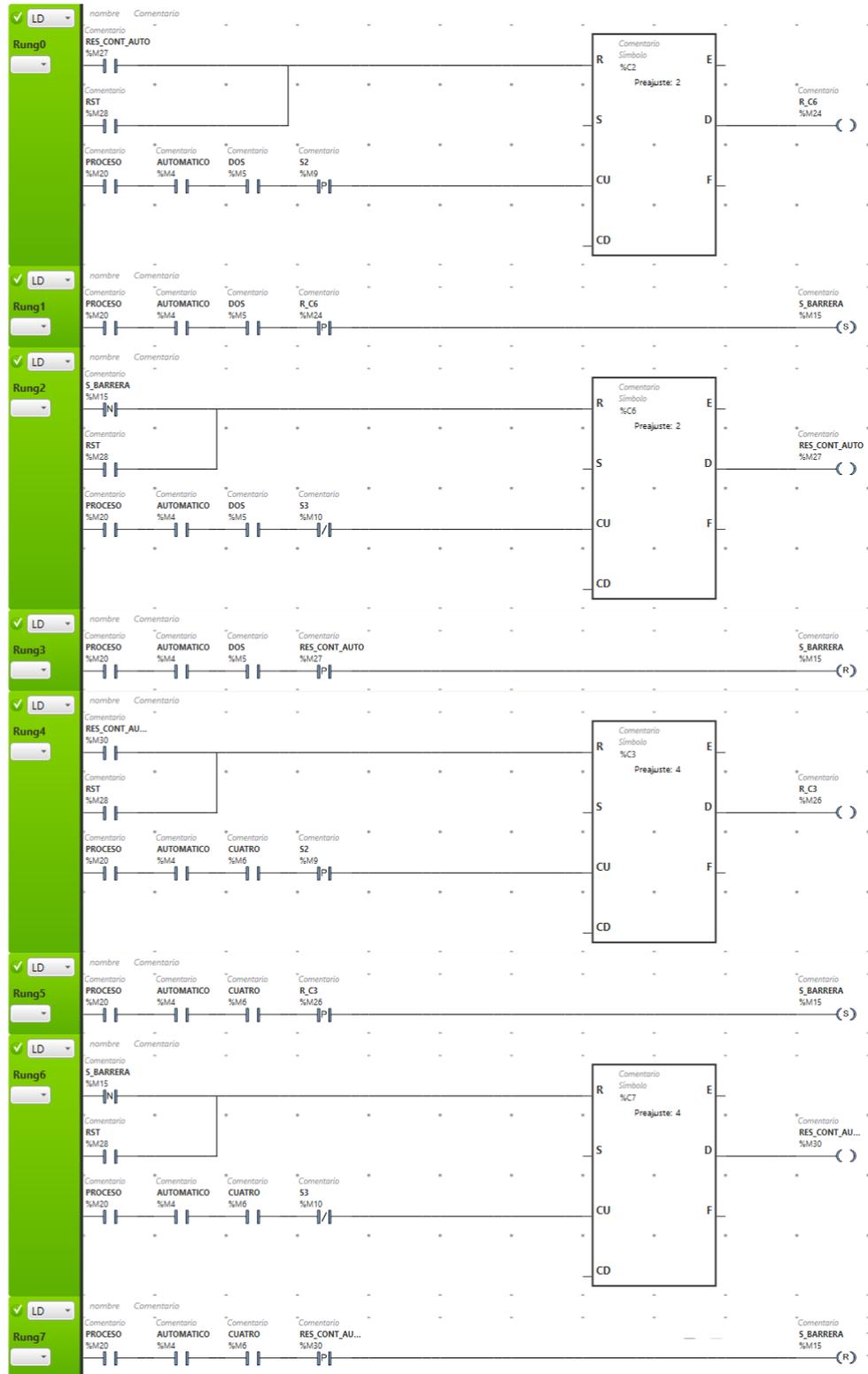


# Anexo 3 Programación ESCENA 3

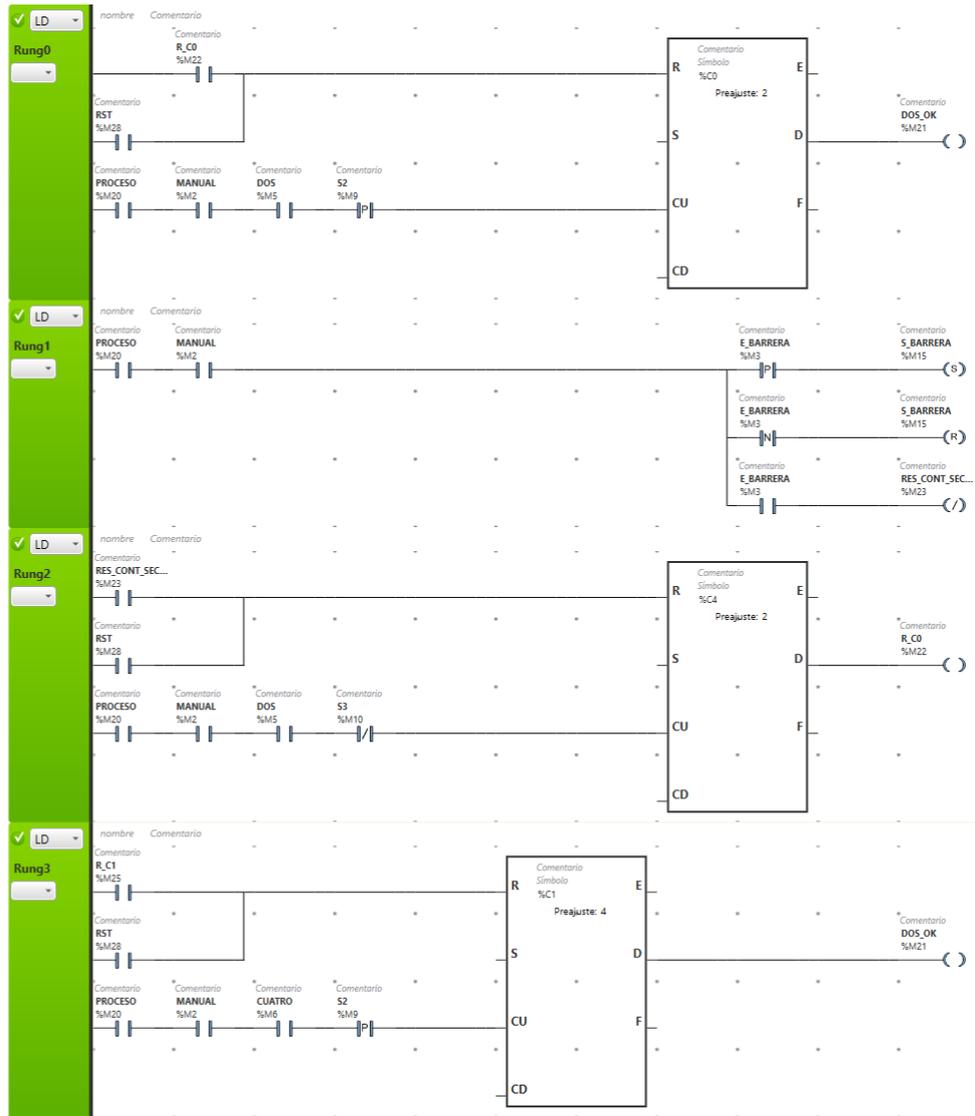
- ☐ Tarea maestra
- ☑ 1 - Manual
- ☑ 2 - Principal
- ☑ 3 - Automático
- Tarea periódica
- Eventos
- POU disponibles

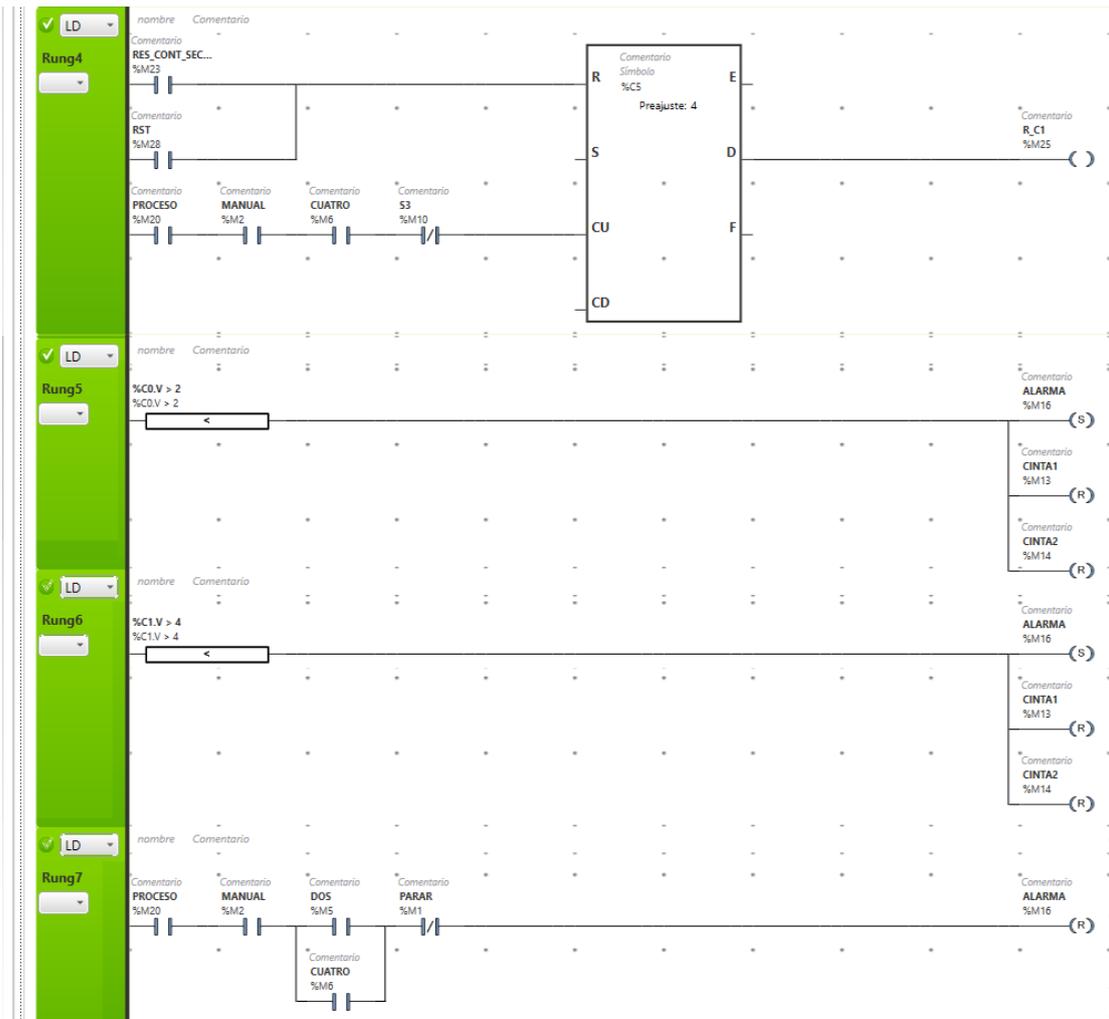


- Tarea maestra
- 1 - Manual
- 2 - Principal
- 3 - Automatico
- Tarea periódica
- Eventos
- POU disponibles



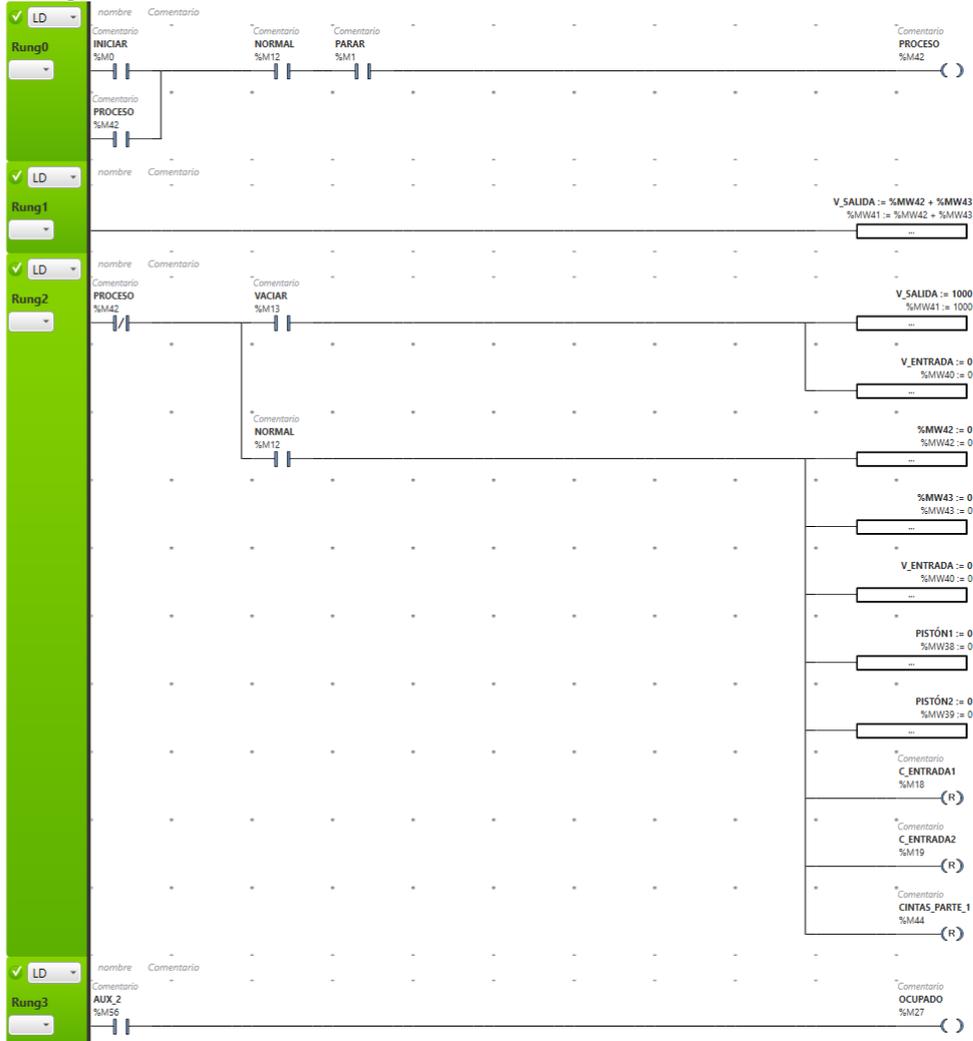
- Comportamiento
- Tarea maestra
    - 1 - Manual
      - Rung0
      - Rung1
      - Rung2
      - Rung3
      - Rung4
      - Rung5
      - Rung6
      - Rung7
    - 2 - Principal
    - 3 - Automatico
    - Tarea periódica
    - Eventos
    - POU disponibles



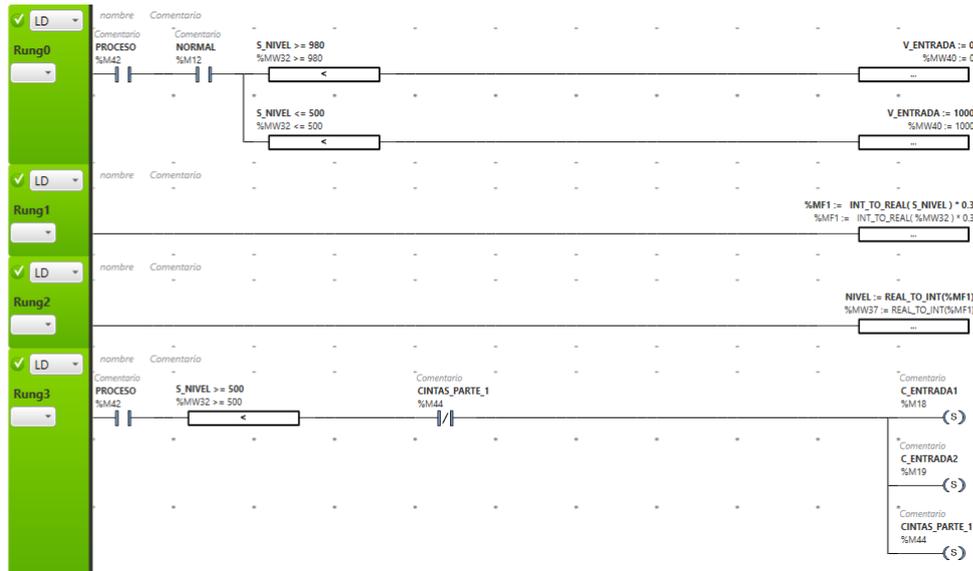


# Anexo 4 Programación ESCENA 4

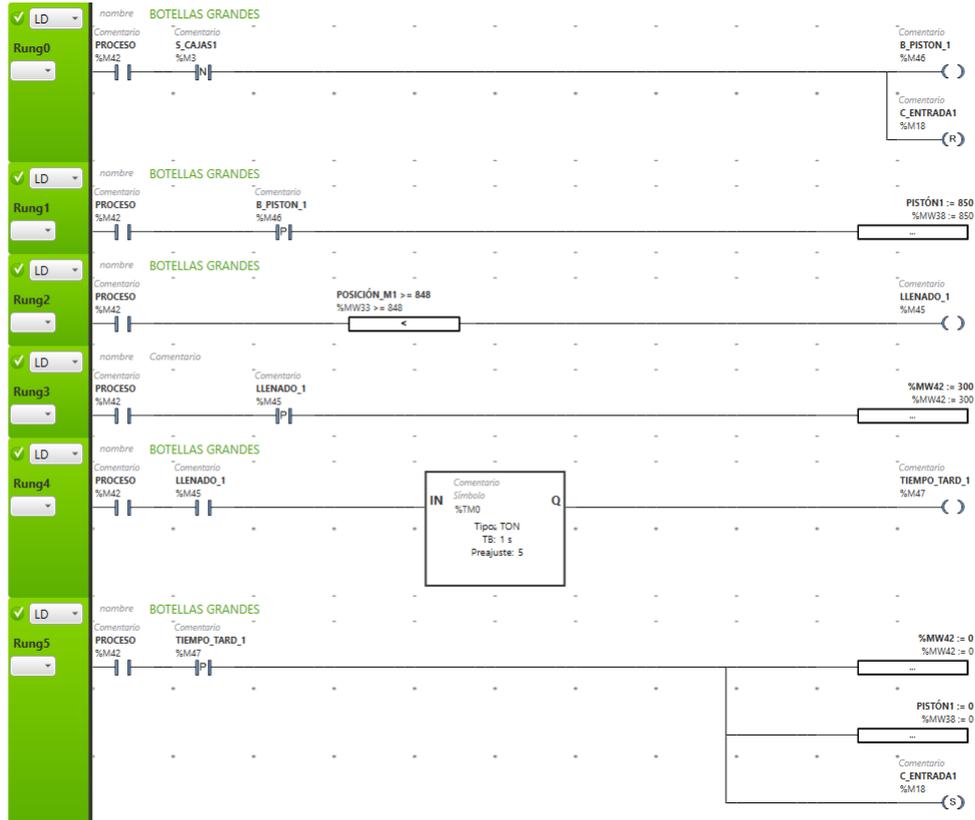
- Tarea maestra
- 1 - Principal
- 2 - Tanque
- 3 - Llenado 1
- 4 - Llenado 2
- 5 - Bloque1
- 6 - Bloque2
- 7 - Bloque3
- 8 - Bloque4
- 9 - Bloque5
- Tarea periódica
- Eventos
- POU disponibles



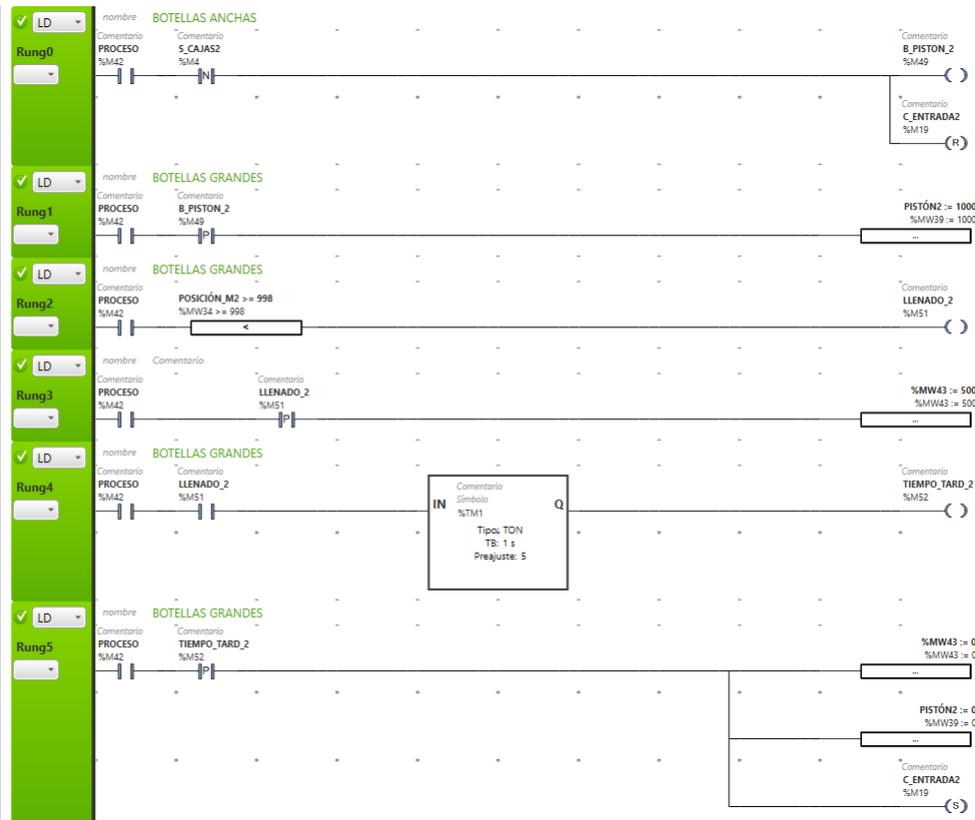
- Tarea maestra
- 1 - Principal
- 2 - Tanque
- 3 - Llenado 1
- 4 - Llenado 2
- 5 - Bloque1
- 6 - Bloque2
- 7 - Bloque3
- 8 - Bloque4
- 9 - Bloque5
- Tarea periódica
- Eventos
- POU disponibles



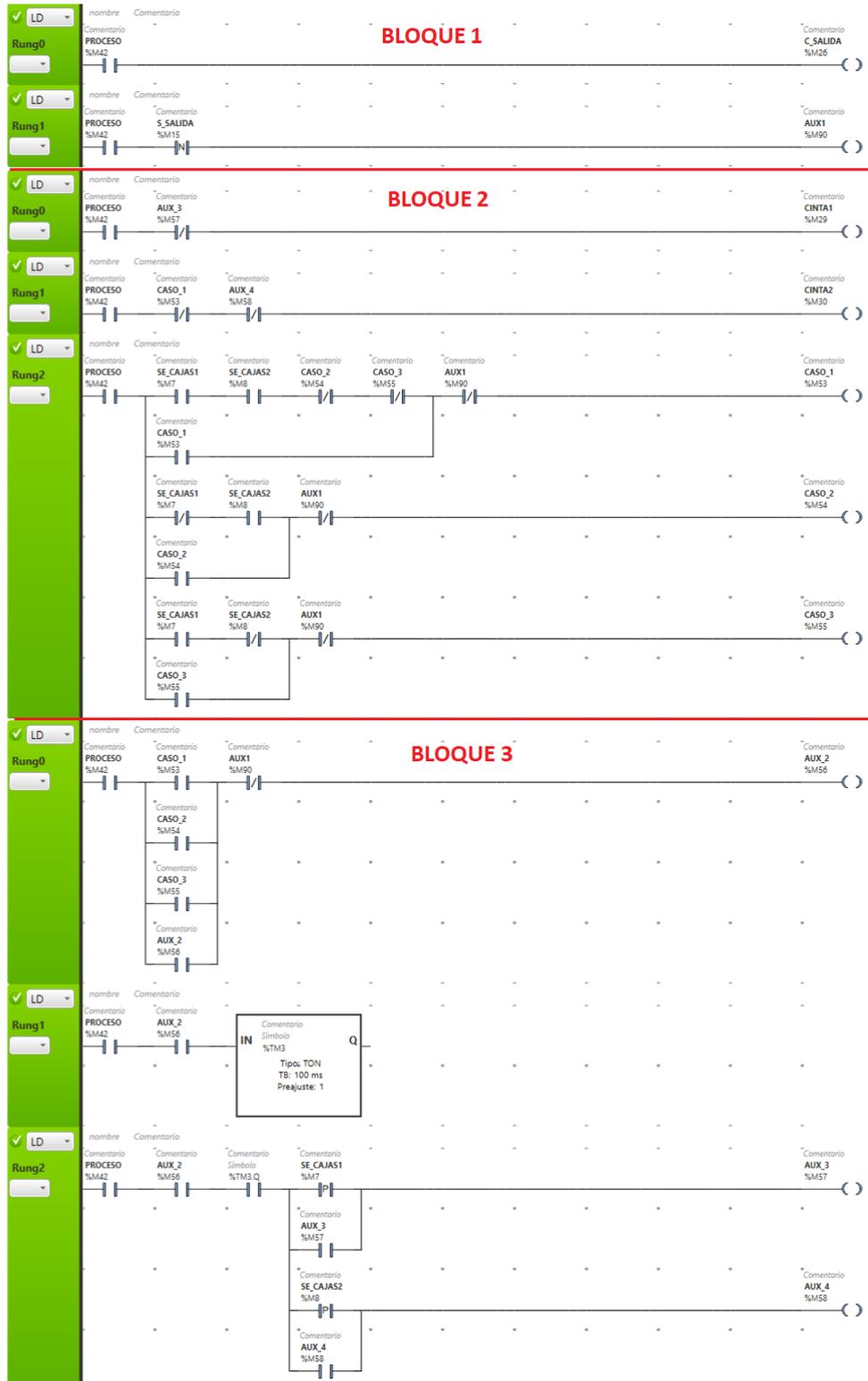
- Tarea maestra
- 1 - Principal
- 2 - Tanque
- 3 - Llenado 1
- 4 - Llenado 2
- 5 - Bloque1
- 6 - Bloque2
- 7 - Bloque3
- 8 - Bloque4
- 9 - Bloque5
- Tarea periódica
- Eventos
- POU disponibles



- Tarea maestra
- 1 - Principal
- 2 - Tanque
- 3 - Llenado 1
- 4 - Llenado 2
- 5 - Bloque1
- 6 - Bloque2
- 7 - Bloque3
- 8 - Bloque4
- 9 - Bloque5
- Tarea periódica
- Eventos
- POU disponibles



- Tarea maestra
- 1 - Principal
- 2 - Tanque
- 3 - Llenado 1
- 4 - Llenado 2
- 5 - Bloque1
- 6 - Bloque2
- 7 - Bloque3
- 8 - Bloque4
- 9 - Bloque5
- Tarea periódica
- Eventos
- POU disponibles



- ☐ Tarea maestra
- ☒ 1 - Principal
- ☒ 2 - Tanque
- ☒ 3 - Llenado 1
- ☒ 4 - Llenado 2
- ☒ 5 - Bloque1
- ☒ 6 - Bloque2
- ☒ 7 - Bloque3
- ☒ 8 - Bloque4
- ☒ 9 - Bloque5
- ☐ Tarea periódica
- ☐ Eventos
- ☐ POU disponibles

