



DISEÑO DE PRACTICAS VIRTUALES DE AUTOMATIZACIÓN BASADO EN LA CONECTIVIDAD ENTRE FACTORY I/O Y CODESYS

**ANDRÉS MAURICIO REYES PORRAS
ANDRÉS FELIPE NIETO MORALES**

Universidad Antonio Nariño
Facultad de Ingeniería Mecánica, Electrónica y Biomédica
Villavicencio, Colombia
2020

DISEÑO DE PRACTICAS VIRTUALES DE AUTOMATIZACIÓN BASADO EN LA CONECTIVIDAD ENTRE FACTORY I/O Y CODESYS

**ANDRÉS MAURICIO REYES PORRAS
ANDRÉS FELIPE NIETO MORALES**

Proyecto de grado presentado como requisito parcial para optar al título de:

Ingeniero Electromecánico

Director (a):

MSc. Alberto Alfonso Villarraga Baquero

Universidad Antonio Nariño

Facultad de Ingeniería Mecánica, Electrónica y Biomédica

Ciudad, Colombia

Año

Nota de aceptación:

Firma del presidente del jurado

Firma del jurado

Firma del jurado

Villavicencio (, ,)

(Dedicatoria)

Dedicamos esta tesis a todos aquellos que creyeron en nuestras capacidades, de igual forma a las que no creyeron en nosotros, y aquellas que esperaban con ansias nuestra culminación de este gran peldaño del proceso universitario, a nuestros maestros quienes nunca desistieron al enseñarnos, aun sin importar las dificultades que cada uno encontramos en nuestro proceso de aprendizaje, aquellos que aun incansablemente confiaron y depositaron su esperanza en nosotros.

ANDRÉS MAURICIO REYES PORRAS
ANDRÉS FELIPE NIETO MORALES

Agradecimientos

Agradezco la confianza, apoyo y dedicación de su tiempo al ingeniero Alberto Alfonso Villarraga Baquero, por haber prestado de su tiempo, conocimientos y oportunidad de superarnos a nosotros mismos.

ANDRÉS FELIPE NIETO MORALES

Agradezco a Dios por este gran logro, a mis padres por su apoyo incondicional, por darme ánimos cuando más lo necesitaba. A su dedicación, confianza, apoyo, carisma y de su valioso tiempo al ingeniero Alberto Alfonso Villarraga Baquero, por habernos compartido parte de su conocimiento y experiencia. Gracias a él, la oportunidad de superarnos y obtener nuestro título Universitario.

ANDRÉS MAURICIO REYES PORRAS

Resumen

En este documento se desarrolla la conectividad entre los programas Factory I/O y Codesys con el fin de dar una herramienta que permita realizar prácticas de automatización de manera virtual, ya que el programa de Factory I/O es una plataforma que cuenta con procesos industriales en 3D que permiten realizar prácticas en una forma casi real, sin problemas al tener errores en la programación, permitiendo al estudiante dedicarse a la programación sin tener complicaciones si presentan fallos en el proceso se prueban, además se pueden hacer modificaciones en línea, y esto es algo que en la industria es importante ya que después de realizar la automatización se debe poner a punto un proceso industrial.

El software Codesys es una de las herramientas que se basa en estándar IEC-61131-3 que establece el lenguaje de programación de los PLC, además incluye muchos de estos diferentes tipos de lenguaje por lo cual es muy completo y pertinente para la enseñanza de la automatización.

Palabras clave: (Automatización, Factory I/O, Codesys).

Abstract

In this document, the connectivity between the Factory iIO and Codesys programs is developed in order to provide a tool for virtual automation practices, since the Factory I / O program is a platform that has 3D industrial processes that allow practicing in an almost real way, without problems when having programming errors, allowing the student to dedicate themselves to programming without complications if they have faults in the testing process, in addition, online modifications can be made, and this is something This is important in the industry, since after carrying out automation, an industrial process must be developed.

The Codesys software is one of the tools that is based on the IEC-61131-3 standard that establishes the programming language of the PLCs, it also includes many of these different types of language, making it very complete and relevant for teaching automation.

Key words: (Automation, Factory I / O, Codesys).

Contenido

| | |
|---|-----------|
| 1. AUTOMATIZACIÓN..... | 3 |
| 1.1 Elementos del Automatismo | 4 |
| 1.1.1 Accionadores | 4 |
| 1.1.2 Preactuadores | 6 |
| 1.1.3 Sensores..... | 7 |
| 1.2 Unidad de control..... | 8 |
| 1.2.1 Sistemas de Control Distribuido (DCS)..... | 8 |
| 1.2.2 Controladores lógicos programables (PLC)..... | 8 |
| 2. CODESYS | 10 |
| 2.1 Lenguajes de programación | 10 |
| 2.1.1 Lista de Instrucciones..... | 10 |
| 2.1.2 Texto Estructurado | 11 |
| 2.1.3 Diagrama de bloques de funciones (FBD)..... | 12 |
| 2.1.4 Diagrama secuencial de bloques (SFC) | 12 |
| 2.1.5 Diagrama escalera (LD). | 13 |
| 2.2 Servidor OPC | 14 |
| 3. FACTORY I/O..... | 16 |
| 4. INTEGRACIÓN ENTRE FACTORY I/O Y CODESYS | 19 |
| 4.1 Procedimiento de conectividad entre Codesys y Factory I/O | 19 |
| 4.1.1 Configuración de Codesys | 19 |
| 4.1.2 Configuración de Factory I/O | 24 |
| 5. CREACIÓN DE ESCENAS..... | 28 |
| 5.1 Escena contador de cajas de A a B..... | 28 |
| 5.2 Escena Selección de Materia Prima..... | 30 |

| | |
|--|-----------|
| 6. USO DE ESCENAS PREDISEÑADAS | 35 |
| 6.1 Escena estación de separación | 35 |
| 6.2 Escena estación de ensamble..... | 37 |
| 7. CONCLUSIONES Y RECOMENDACIONES | 41 |
| 7.1 Conclusiones | 41 |
| 7.2 Recomendaciones | 42 |
| 8. BIBLIOGRAFÍA..... | 52 |

Lista de figuras

| | |
|---|----|
| Figura 1–1 | 3 |
| Figura 1–2 Motor Eléctrico..... | 4 |
| Figura 1–3 Cilindro neumático..... | 5 |
| Figura 1–4 Motor Hidráulico | 6 |
| Figura 1–5 Contactores | 7 |
| Figura 1–6 Sensores | 7 |
| Figura 1–7 Controladores Lógicos Programables | 9 |
| Figura 2–1 Ejemplo código FBD..... | 12 |
| Figura 2–2 ejemplo código SFC..... | 13 |
| Figura 2–3 ejemplo código LD..... | 13 |
| Figura 2–4 Servidor OPC | 14 |
| Figura 3–1 Escenas Prediseñadas | 17 |
| Figura 3–2 OPC Client Data Access | 18 |
| Figura 3–3 Cámara peatón..... | 18 |
| Figura 4–1 Control Win V3 | 20 |
| Figura 4–2 selección de lenguaje de programación | 20 |
| Figura 4–3 variables globales..... | 21 |
| Figura 4–4 propiedades de variables globales | 22 |
| Figura 4–5 Editor de programa..... | 22 |
| Figura 4–6 visualización | 23 |
| Figura 4–7 symbol configuración..... | 23 |
| Figura 4–8 ruta de enlace..... | 24 |
| Figura 4–9 Driver Factory I/O | 25 |
| Figura 4–10 configuración del servidor | 26 |
| Figura 4–11 Conexiones de entrada y salida del OPC..... | 27 |

| | |
|--|----|
| Figura 5–1 Escena contador de cajas de A a B..... | 29 |
| Figura 5–2 Código contador de cajas de A a B | 30 |
| Figura 5–3 Escena selector de material..... | 31 |
| Figura 5–4 Código Escena selector de material | 33 |
| Figura 6–1 Escena estación de separación | 36 |
| Figura 6–2 código escena estación de separación..... | 37 |
| Figura 6–3 Escena estación de ensamble | 39 |
| Figura 6–4 código estación de ensamble | 40 |

Pág.

Lista de tablas

| | Pág. |
|--|-------------|
| Tabla 1 variables escena contador de cajas de A a B..... | 28 |
| Tabla 2 variable escena selección de materia prima..... | 31 |
| Tabla 3 variables escena estación de separación..... | 35 |
| Tabla 4 variables escena estación de ensamble..... | 38 |

Introducción

El presente documento se da a conocer dos programas que están haciendo que la automatización se pueda llevar a lo virtual, ya que uno de ellos Factory I/O es un programa con escenas tridimensionales (3D) de la industria prediseñadas y la posibilidad de realizar tus propias escenas industriales con los elementos que cuenta el programa.

El programa de Codesys tiene la ventaja que en su versión demo sin caducidad 100% libre y puede ser usado para la enseñanza de la automatización, cuenta diferentes lenguajes de programación, así como de un servidos OPC que permite la conectividad con otros programas.

Se da una guía de cómo realizar la conexión ente los programas Factory I/O y Codesys a través de un OPC, así como la realización del paso a paso de la creación de variables y la interconexión entre los dos programas.

Se realizo escenas en el programa Factory I/O que ayuden a desarrollar la habilidades en la programación de controladores lógicos programables (PLC) y que interactúan con el Codesys para llevar las prácticas a este nueva forma de manera que el estudiante de los programas de ingeniería electromecánica, tecnológica en mantenimiento electromecánico industrial e ingeniería electrónica puedan adelantar estos laboratorios en sus casas, y poder mejorar cada día mas atreves se sus propias practicas o su intriga por aprender a programas en este grandioso mundo de la automatización.

Adicional a las escenas realizadas se tomaron dos escenas que complementan el aprendizaje en la automatización, y ayuda a involucrar al estudiante y al docente en estas herramientas que son importantes en este momento para la modalidad distancia o virtual que pudiese tener la universidad en las carreras a fines a la automatización.

Se espera que la universidad realiza la adquisición de los programas para poder llevar este proyecto a la practica en las materias de mando y control, y sistemas mixtos de los programas que actualmente cuenta la universidad Antonio Nariño sede Villavicencio y que se pueda implementar en toras sedes a nivel nacional.

1.AUTOMATIZACIÓN

El control de sistemas trata de controlar o regular un proceso en todas las áreas en donde se requiere un avance tecnológico con el fin de dar un continuo trabajo industrializado, con la mínima intervención humana en los procesos donde puede tener riesgos de accidentes, esto mediante el comportamiento dinámico de un sistema mediante órdenes de mando que son enviadas por un controlador. (Flower Leiva, 2008)

La automatización pretende transferir las tareas de gran dificultad o riesgo para una línea de producción que son realizadas manualmente por operadores humanos para ser realizada por una maquina capaz de reproducir el control de este. Esta automatización consiste en diseñar un modelo que se pueda realizar un dispositivo y que genere una respuesta del sistema con el fin de responder ante las órdenes que el usuario.

Figura 1-1



Fuente: el autor

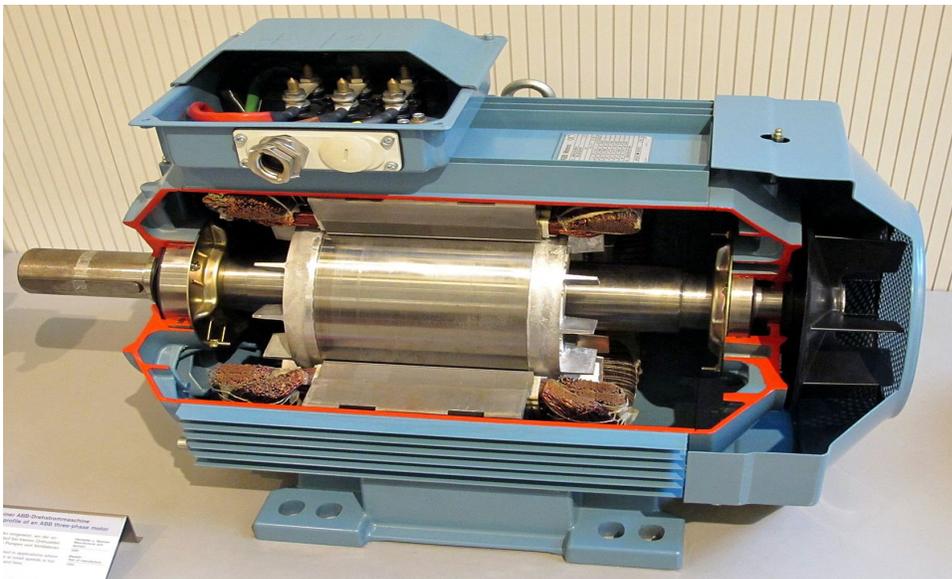
1.1 Elementos del Automatismo

1.1.1 Accionadores

Los actuadores son los dispositivos finales en los sistemas de Automatización su fin de proporcionar potencia a sistema final, ya sea moviendo una banda transportadora, moviendo un fluido entre otros. Las fuentes de trabajo pueden ser de presión hidráulica, neumática, eléctrica, térmica y otras que pueden convertir este tipo de energía en trabajo útil.

Los accionadores eléctricos usan la energía eléctrica como fuente principal convirtiéndola en energía mecánica, dentro de los cuales se encuentra el actuador más utilizado a nivel mundial en motor eléctrico y sus distintas aplicaciones, dentro de los cuales sus principales características son la velocidad, potencia. Pueden ser de corriente continua (DC) o de corriente Alterna (AC).

Figura 1–2 Motor Eléctrico



[Esta foto](#) de Autor desconocido está bajo licencia [CC BY-SA](#)

Los actuadores Neumáticos SU fuente de Energía es aire comprimido el cual es usado para a través de actuadores neumáticos, como los cilindros, motores neumáticos para convertir este tipo de energía en trabajo mecánico.

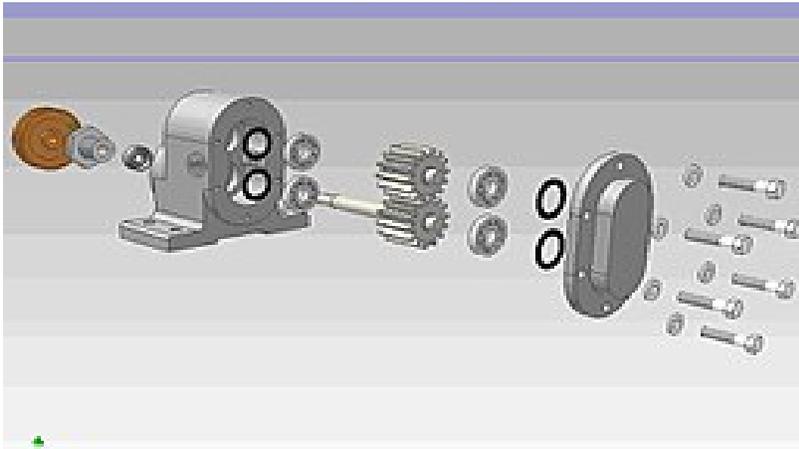
Figura 1-3 Cilindro neumático



[Esta foto](#) de Autor desconocido está bajo licencia [CC BY-SA](#)

Otro tipo de actuadores son los hidráulico, en los cuales se utiliza la fuerza fluida para la generación de trabajo mecánico, son usados para mover grandes cantidades de carga, a bajas velocidades, existen cilindros hidráulicos, motores hidráulicos.

Figura 1–4 Motor Hidráulico



[Esta foto](#) de Autor desconocido está bajo licencia [CC BY-SA](#)

1.1.2 Preactuadores

Son los elementos de interconexión entre la parte de mando y la parte operativa, toman una señal ya sea eléctrica, neumática, hidráulica para accionar los actuadores, este tipo de Preactuadores se pueden ser.

los Preactuadores eléctricos dentro de los cuales se encuentran los contactores, variadores de velocidad, arrancadoras suaves, relés, electroválvulas y otros.

Figura 1–5 Contactores

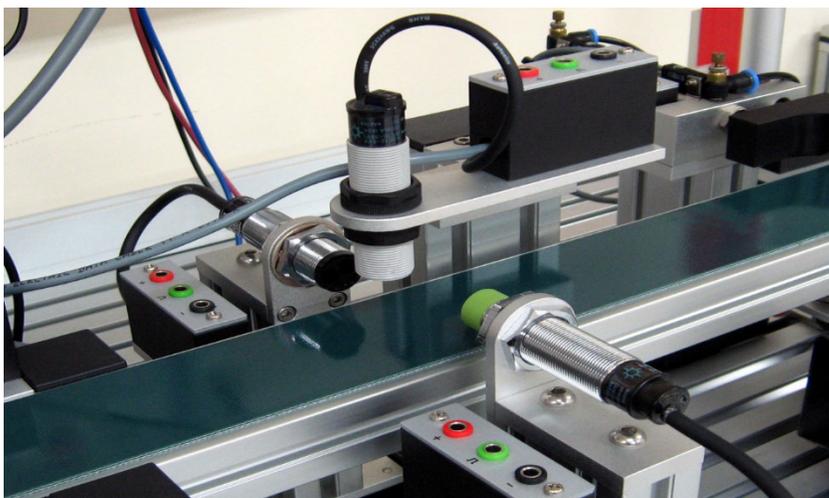


Esta foto de Autor desconocido está bajo licencia [CC BY-NC-ND](https://creativecommons.org/licenses/by-nc-nd/4.0/)

1.1.3 Sensores

Son elementos que convierten una magnitud física en una señal de magnitud eléctrica con la cual podemos determinar el estado de un sistema. Son usados para conocer los estados de los procesos, los cuales pueden medir distintas magnitudes físicas, dentro de las más importantes se encuentra la temperatura, presión, posición, caudal, humedad entre otros.

Figura 1–6 Sensores



Esta foto de Autor desconocido está bajo licencia [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/)

1.2 Unidad de control

En las unidades de control se tiene diferentes formas y equipos para lograr el control y el gobierno de un sistema dentro de los cuales podemos encontrar procesadores, controladores lógicos programables (PLC) y Sistemas de Control Distribuido (DCS), dentro de los cuales en la industria se encuentran los dos últimos como grandes opciones en el momento de tomar la decisión de Automatizar un proceso.

1.2.1 Sistemas de Control Distribuido (DCS)

Estos sistemas surgieron del desarrollo de los microprocesadores en los años 70. Estos sistemas son un conjunto de elementos de hardware y software que permiten lograr la funcionalidad requerida en los sistemas de control y en la adquisición de datos de un proceso.

Se caracterizan por la existencia de varios lazos de control con capacidad de procesamiento de la información, los cuales están conectados en la red. Las unidades pueden comunicarse y sincronizarse para lograr un objetivo común. (Izaguirre Castellanos, 2012)

1.2.2 Controladores lógicos programables (PLC)

Es un dispositivo capaz de almacenar, estructurar y procesar información para controlar un proceso. Se caracterizan por tener lenguajes estructurados de raíces secuenciales, por lo cual tiene grandes ventajas en tratamiento de automatismos secuenciales y mediciones analógicas.

En aplicaciones donde se emplean los PLC podemos realizar con procesos sencillos modificaciones en los sistemas automáticos, lo cual es una gran ventaja a la hora de explorar esta tecnología.

Actualmente existen desarrollos y perfeccionamientos de diferentes lenguajes de programación para autómatas como son lista de instrucciones, diagrama de contactos, gráficos secuenciales, Grafcet Texto estructurado que ayudan a realizar cada día funciones más complejas en los sistemas de automatización.

Las principales características que distinguen a los PLC son, que son modulares y expandibles, flexibilidad en la configuración de software y hardware, alta capacidad de realización de tareas (Multitareas) y operación en tiempo real y otras.

Además, su gran demanda que generan este tipo de autómatas hace que sean de gran importancia a la hora de tomar decisiones en el momento de automatizar un proceso. (Izaguirre Castellanos, 2012)

Figura 1–7 Controladores Lógicos Programables



[Esta foto](#) de Autor desconocido está bajo licencia [CC BY](#)

2.CODESYS

Es un software para la automatización de sistemas de control bajo la norma IEC 61131-3 la cual describe los lenguajes de programación de los PLC. Además, cuenta con opciones de visualizaciones del proceso mediante pantallas gráficas, elementos de comunicación y diferentes tipos de lenguajes en los cuales se puede programar, como son:

- IL: Lista de instrucciones (Instruction List).
- ST: Texto estructurado (Structured Text).
- FBD: Diagrama de bloques (Function Block Diagram).
- SFC: Diagrama secuencial de bloques (Sequential Function Chart)
- LD: Diagrama escalera (Ladder).

Actualmente es un software dispone de una versión demo 100% operativa que permite programar autómatas basado en la norma IEC 61131-3 y poder simular los códigos, por lo cual puede ser utilizado para la enseñanza de la Automatización. (CODESYS Group, 2020)

2.1 Lenguajes de programación

2.1.1 Lista de Instrucciones

Es un lenguaje de nivel bajo con características particulares a los lenguajes ensamblador, se usa para soluciones sencillas y que son secuenciales, y es tomado como un lenguaje de referencia a los demás lenguajes estandarizados. (Orozco Gutierrez , Guarnizo Lemus, & Holguin Londoño, 2008)

A continuación, se muestra en ejemplo de programación en este tipo de lenguaje.

LD a (se carga a en el acumulador)

MUL a (se carga a*a en el acumulador)
SUB(b (resta del acumulado la suma de b y c)
AND c (suma b+c)
)
ST d (almacena el valor del acumulador en d)

2.1.2 Texto Estructurado

Es un lenguaje de alto nivel usado para aplicaciones de control industrial, con una sintaxis similar a los lenguajes basados en texto. El cual cuenta con sentencias como (IF -----THEN -----ELSE), CASE, (WHILE ---DO), REPEAT ---UNTIL, RETURN, Asignaciones, Operadores donde le valor de la variable se asigna Val1:= val2; donde el valor de Val2 es asignado a la variable Va1, otros operadores com AND, OR, XOR, NOT, *,/,MOD, +, -, <, >, <= entre otros a continuación se da un ejemplo de una programación con (Orozco Gutierrez , Guarnizo Lemus, & Holguin Londoño, 2008)

```
IF (GVL.Start= TRUE OR GVL.Motor= TRUE) AND GVL.Stop= TRUE THEN GVL.Motor:= TRUE;
```

```
ELSE GVL.Motor:= FALSE;
```

```
END_IF;
```

```
IF GVL.Sensor= FALSE AND FLAG00= FALSE THEN
```

```
    FLAG00 := TRUE;
```

```
    CONTA:= CONTA+1;
```

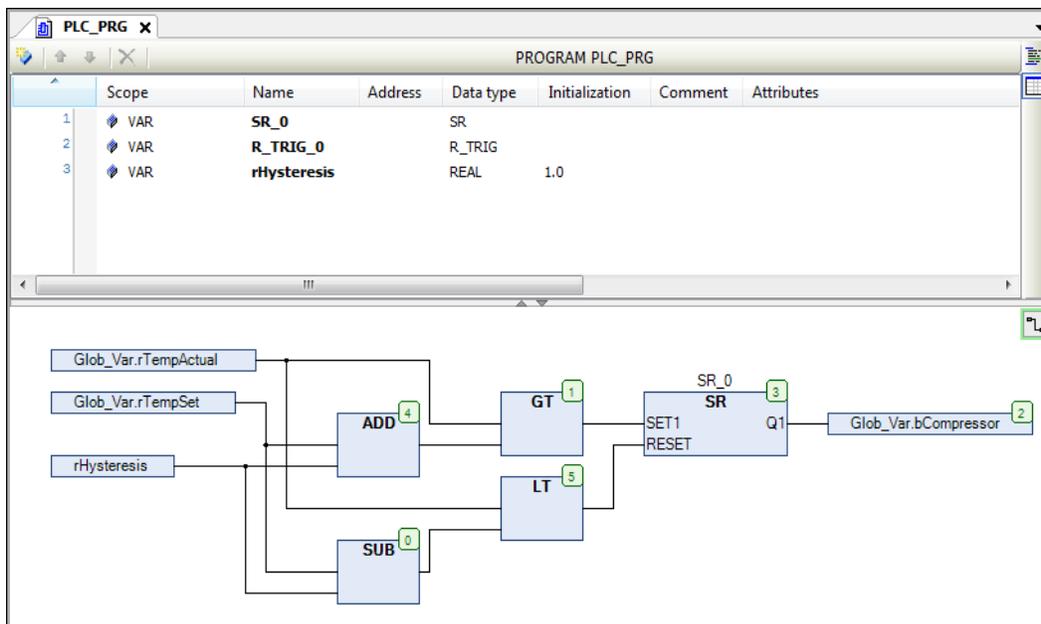
```
    GVL.Digital:= CONTA;
```

```
END_IF;
```

2.1.3 Diagrama de bloques de funciones (FBD)

Lenguaje utilizado para describir el funcionamiento de un sistema mediante un conjunto de bloques interconectados. Su funcionamiento tiene un fundamento al flujo de señales entre los diferentes elementos

Figura 2–1 Ejemplo código FBD

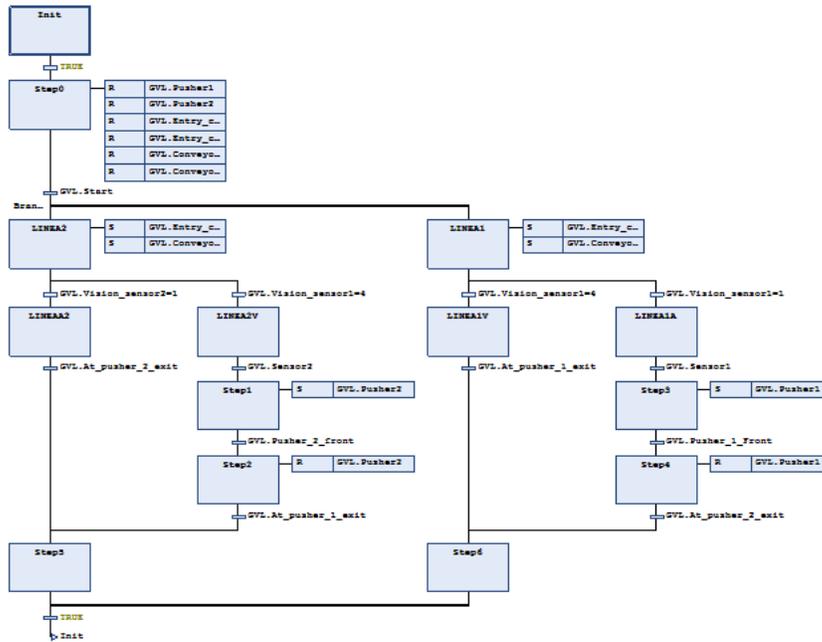


[Esta foto](#) de Autor desconocido está bajo licencia [CC BY-SA](#)

2.1.4 Diagrama secuencial de bloques (SFC)

Tiene un fundamento metodológico para sistemas secuenciales, principalmente en las redes de Petri. Es una metodología grafica que permite la estructuración e implementación mediante estados que se ejecutan a medida que las condiciones se cumplan las cuales dependen de las entradas del sistema.

Figura 2–2 ejemplo código SFC

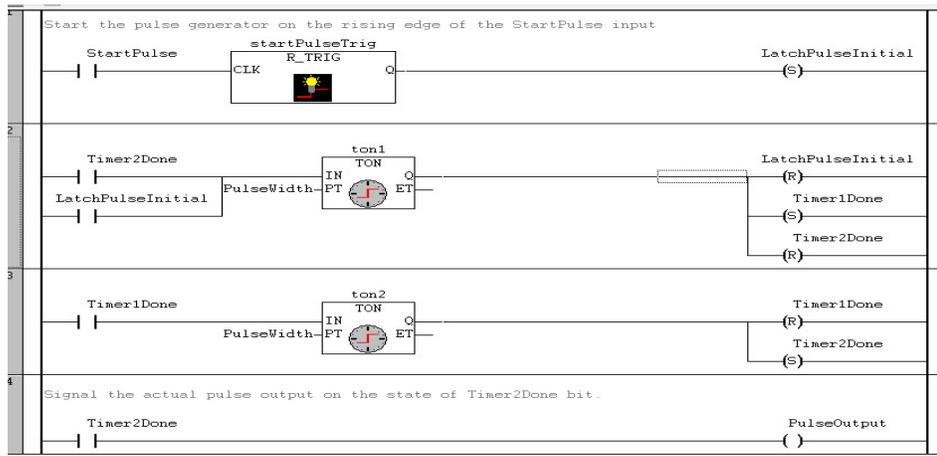


Fuente: el autor

2.1.5 Diagrama escalera (LD).

Es el lenguaje más utilizado en la industria, debido a que parece a un diagrama de contactos eléctricos, debido a su origen, lo que hace que una persona sin experiencia en programación, pero con conocimientos eléctricos pueda realizar automatizaciones simples.

Figura 2–3 ejemplo código LD



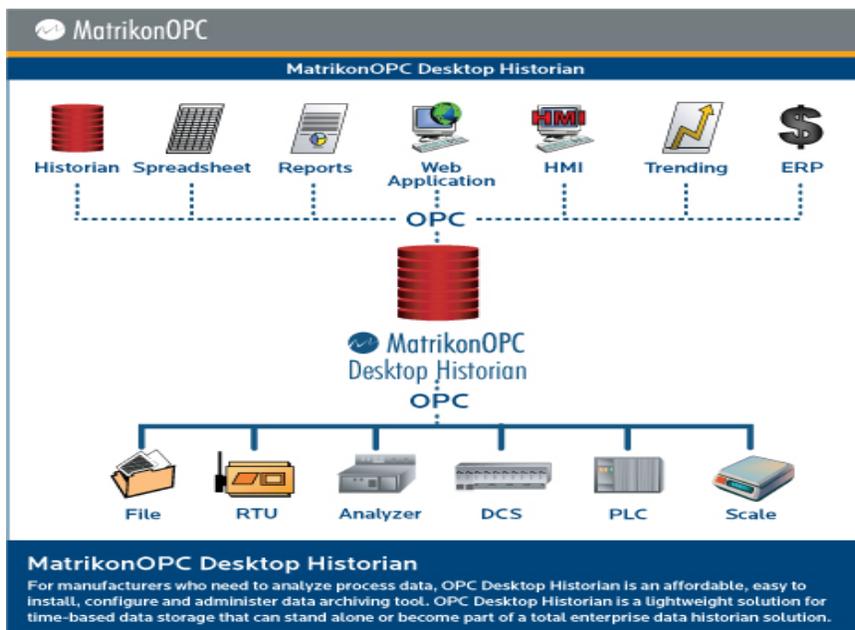
Esta foto de Autor desconocido está bajo licencia [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/)

2.2 Servidor OPC

Es una aplicación de software que se encarga de hacer una interfaz de comunicación por fuentes de datos nativos como los PLC y en el otro extremo Clientes OPC que típicamente pueden ser SCADAs, HMIs, Aplicaciones entre otros, de los cuales existen cuatro tipos de servidores OPC definidos.

- Servidor OPC DA: basado en datos de acceso especialmente diseñado para la transmisión de datos en tiempo real.
- Servidor OPC HDA: Basado en el acceso a datos históricos que provee al cliente HDA de datos.
- Servidos OPC A&E server: basado en la especificación de Alarmas y Eventos desde un dispositivo hacia el cliente OPC A&E.
- Servidor OPC UA: basado en la especificación de arquitectura unificada, permite a los servidores OPC trabajar con cualquier tipo de datos. (Matrikon, 2020)

Figura 2–4 Servidor OPC



[Esta foto](#) de Autor desconocido está bajo licencia [CC BY-SA](#)

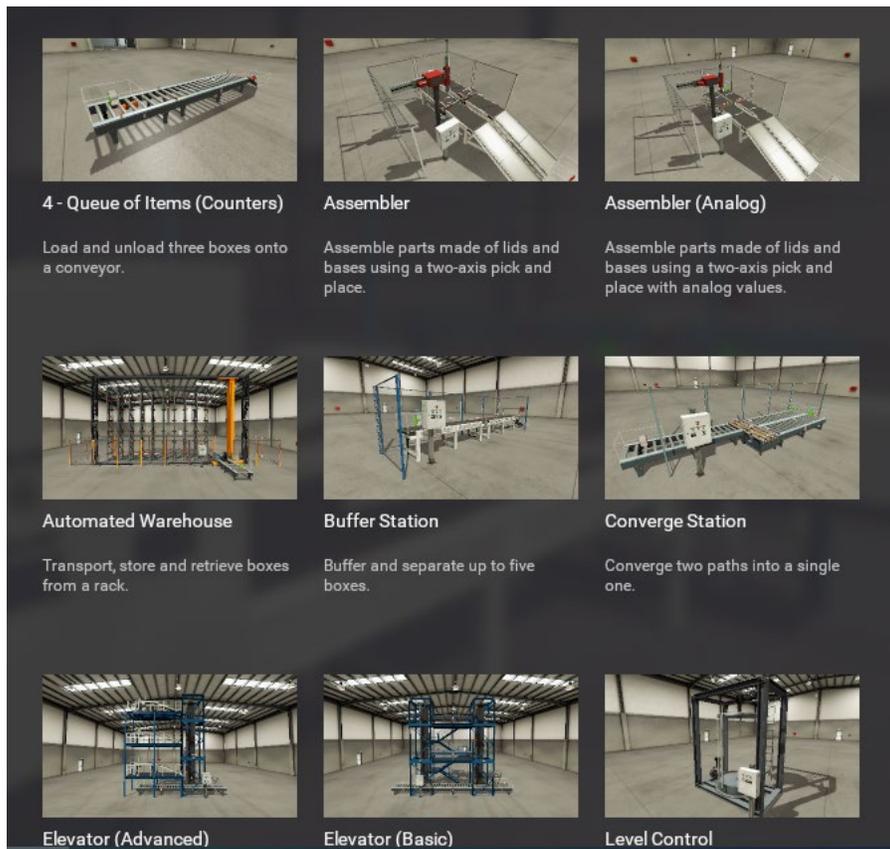
El servidor OPC de CODESYS es una interfase de acceso a los datos de proceso IEC 61131-3 de un controlador a través de una plataforma de comunicación, que permite el intercambio de datos. Además, cuenta con algunas características de funcionalidad como inicio automático al establecer una conexión de cliente, disipador automático cuando cambian los datos, soporte multi-cliente y soporte multiPLC, entre otros. (CODESYS Group, 2020)

3. FACTORY I/O

Factory I/O es un simulador 3D de fabrica diseñado para la enseñanza de la automatización. Fácil de utilizar, cuenta con escenas prediseñadas industriales además de elementos industriales para que construir tu propio escenario industrial, con lo cual puedes empezar a programar desde el nivel básico hasta programación avanzada. (games, 2020)

Cuenta con una biblioteca de piezas industriales incluidos sensores, cintas trasportadoras, ascensores, estaciones entre otros, además 20 escenas listar para usar, y un modo instructor, con el fin de que el docente comparta el sistema a los estudiantes, para que encuentren el problema a solucionar.

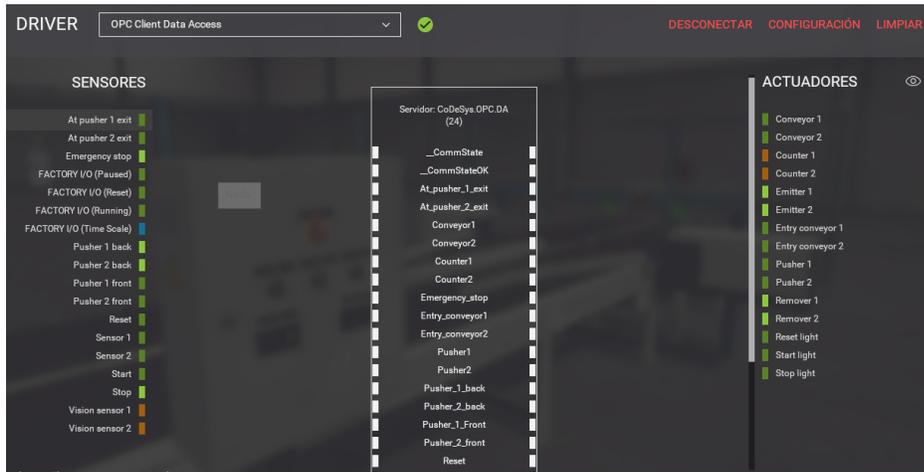
Figura 3–1 Escenas Prediseñadas



Fuente: El autor

La conectividad entre los PLC o Software de Programación y el software Factory I/O a través de un OPC UA hacen que se integre fácilmente, y de ahí su gran aumento en el ámbito de la enseñanza. (games, 2020)

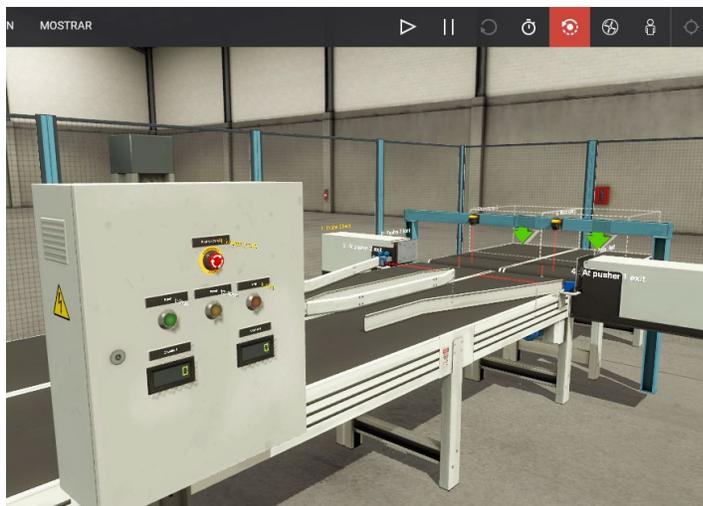
Figura 3–2 OPC Client Data Access



Fuente: El autor

Sus características principales tenemos la interfaz de usuario, con la cual podemos tener diferentes cámaras de observación como la de peatón que permite tener una visión mucho más real, además de la cámara tipo mosca la cual es como si tuvieras un dron el cual puedes maniobrar dentro de la industria, y una cámara orbital la cual cuenta con jiro de 360° para ver toda la fábrica.

Figura 3–3 Cámara peatón



Fuente: El autor

4.INTEGRACIÓN ENTRE FACTORY I/O Y CODESYS

La conectividad entre los programas CODESYS y el Software Factory I/O se realiza a través de un OPC DA el cual permite la comunicación bidireccional de todo tipo de datos entre los dos programas.

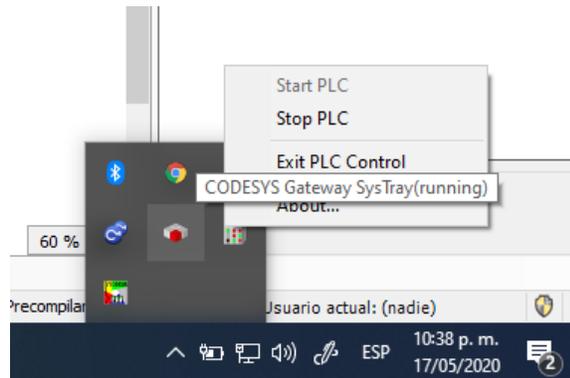
Para realizar la comunicación hay que tener en cuenta que el software CODESYS hasta la versión 3.5 SP12 incluye el OPC necesario para la comunicación, las versiones superiores El OPC es de pago por lo cual para este proyecto se utiliza la versión 3.5 SP10. Para el programa de Factory I/O se realizó en la versión de prueba gratuita Edición Ultimate por 30 días.

4.1 Procedimiento de conectividad entre Codesys y Factory I/O

4.1.1 Configuración de Codesys

Apertura de los programas Codesys 3.5 SP10, en la barra de tareas se debe colocar en modo Run el icono de Control Win V3 de CODESYS el cual simula la conectividad con un PLC.

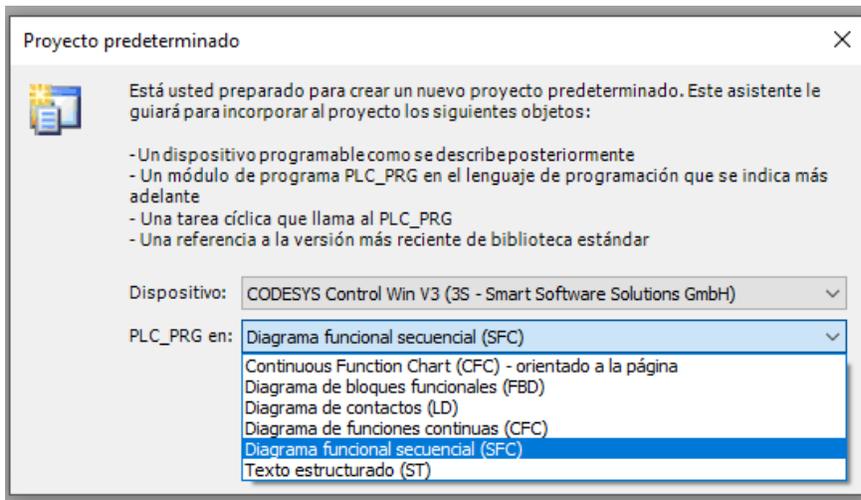
Figura 4–1 Control Win V3



Fuente: El autor

A continuación, se crea el proyecto en Codesys en el que se determina el nombre del proyecto, la dirección del proyecto y la plantilla a utilizar la cual se recomienda se seleccione un proyecto standard. Seguido debemos determinar el dispositivo y el lenguaje de programación el cual depende de la practica a realizar, como se muestra en la figura.

Figura 4–2 selección de lenguaje de programación



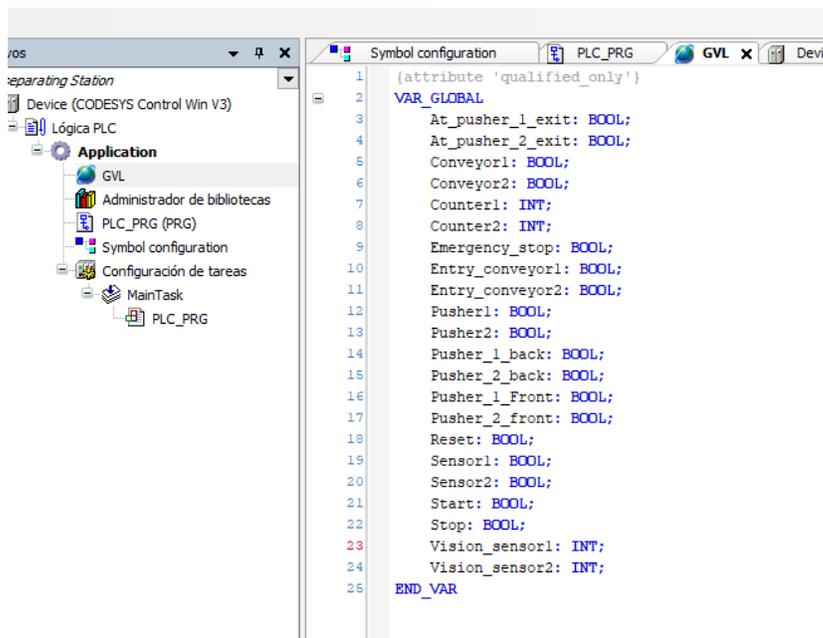
Fuente: el autor

Después de crear el proyecto se procede a crear las variables globales en Codesys, ya que este tipo de variables existen durante todo el proceso de la función existen desde que son declaradas hasta que termine el programa, las cuales se recomienda que tengan

nombres similares a las de la escena, se recuerda que los nombres de las variables no pueden tener espacio ni caracteres especiales.

Para crear las variables globales debemos ir al árbol del proyecto en Aplicaciones y dar clic derecho buscar agregar un objeto y seleccionar la opción lista de variables globales, las variables se pueden crear en forma textual o en forma de tabla.

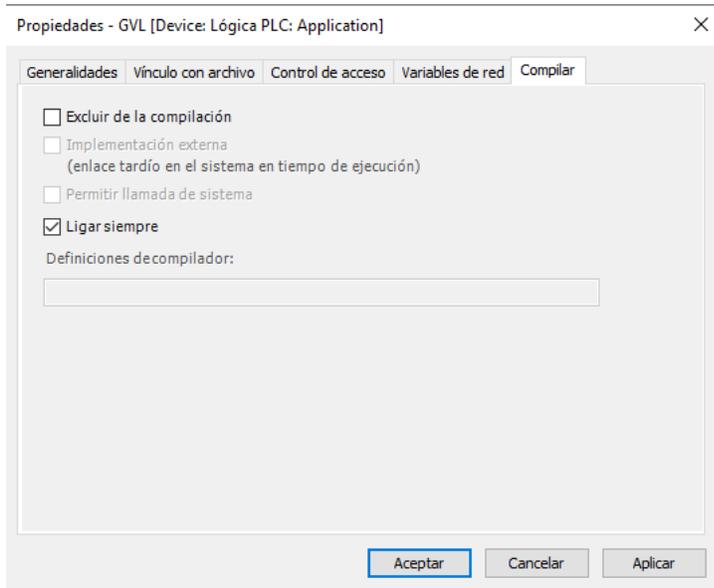
Figura 4–3 variables globales



Fuente: el autor

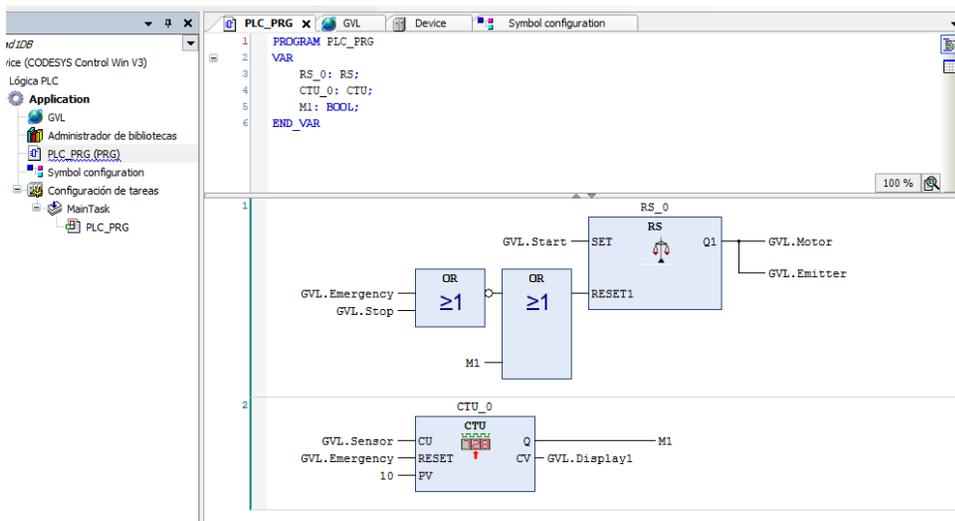
Después de terminar de editar las variables se debe configurara en las propiedades de ellas en la ventana de compilar se activa la casilla de ligar siempre.

Figura 4–4 propiedades de variables globales



A continuación, se puede describir el código en alguno de los lenguajes tomados en la configuración del proyecto, en el archivo PLC_PRG donde después de describir el código según sea la aplicación para automatizar, después de terminar el código se debe copiar para verificar que no cuente con errores de programación.

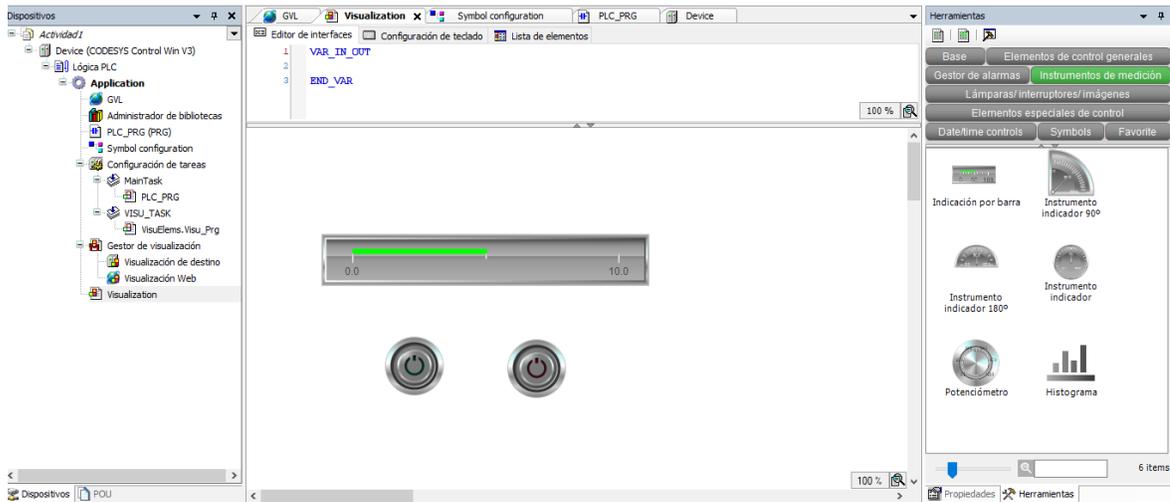
Figura 4–5 Editor de programa



Fuente: el autor

Dependiendo de la aplicación se puede crear una visualización para hacer una Interfase hombre maquina (HMI), con el fin de mejorar un proyecto o una aplicación industrial, según sea el caso.

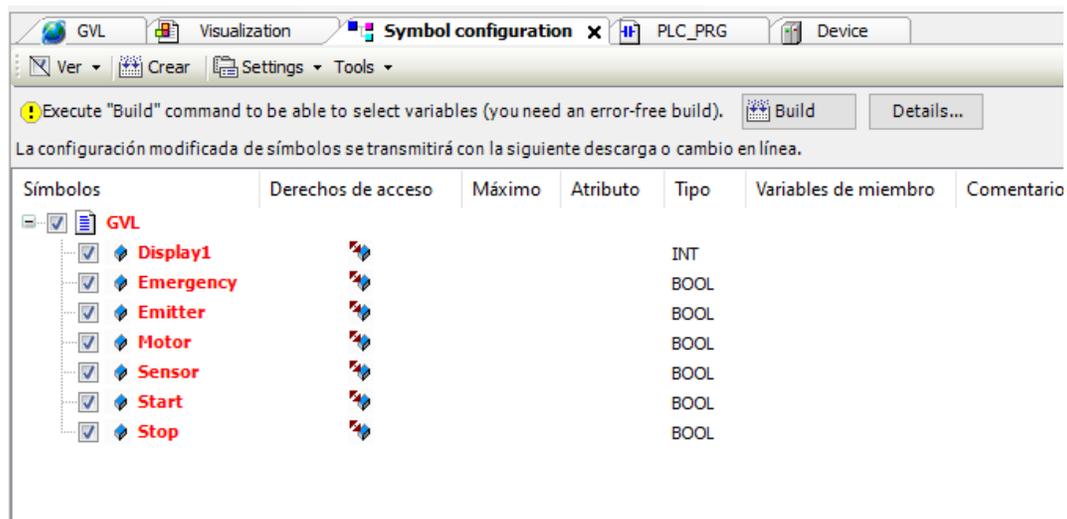
Figura 4–6 visualización



Fuente: el autor

A continuación, dando clic en aplicaciones buscar agregar objeto se agrega una configuración simbólica (symbol configuration), la cual solicita que se ejecute el comando build para realizar un barrido de las variables y a continuación activarlas para poder ser utilizadas por el OPC.

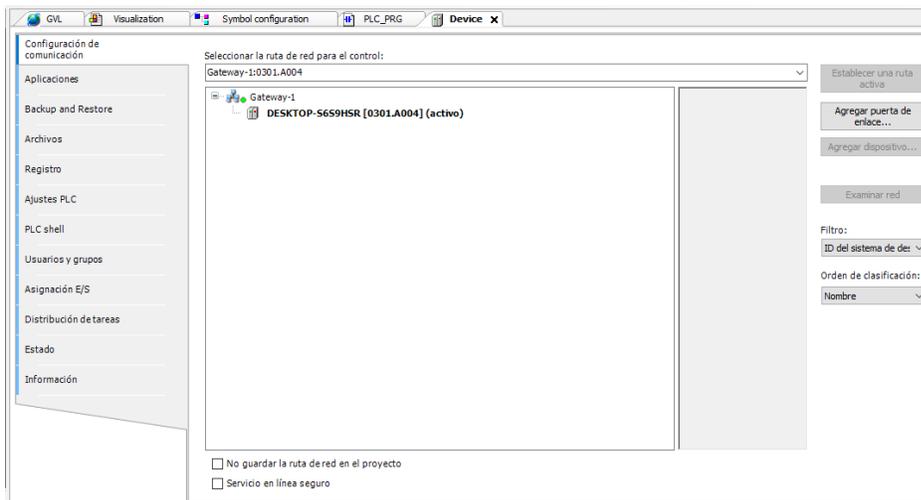
Figura 4–7 symbol configuración



Fuente: el autor

A continuación de abre la configuración del dispositivo el cual se encuentra en el árbol del proyecto, en el debemos configurar una ruta de red como activa, con el fin de usar el Gateway V3 el cual simula la conexión con un PLC.

Figura 4–8 ruta de enlace



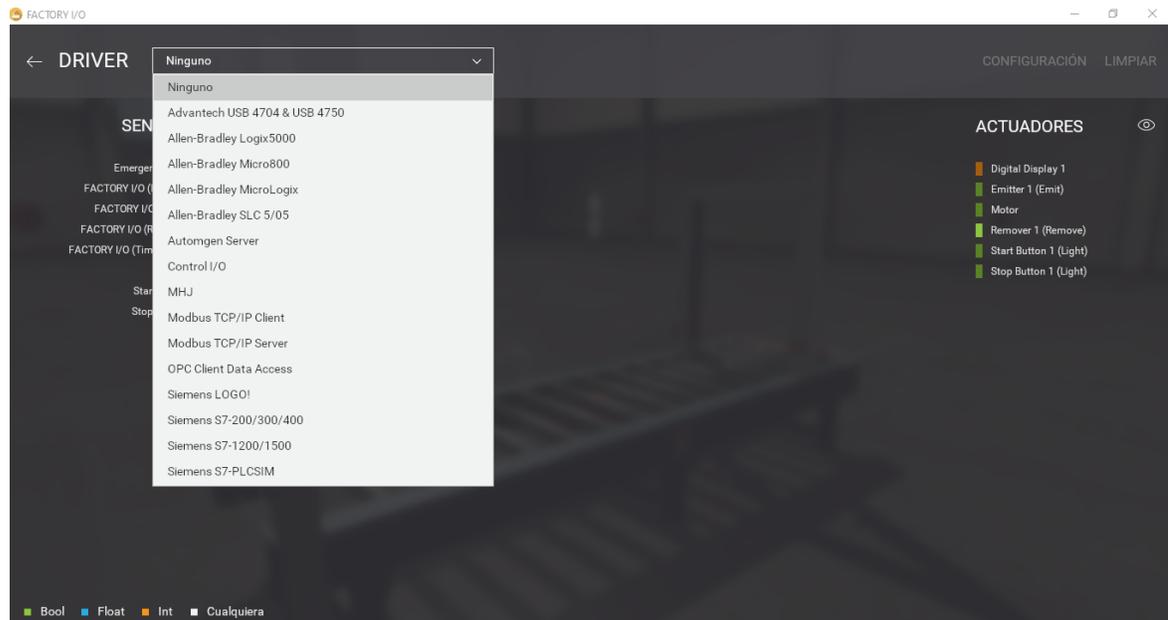
Fuente: el autor

Después de tener activo la ruta de enlace se activa el comando iniciar sección pulsando Alt + F8, y a continuación en iniciar para crear establecer comunicación con el servidor OPC.

4.1.2 Configuración de Factory I/O

Al iniciar el programa de debe abrir la escenas a automatizar, ya sea una prediseñada o creada con anterioridad, en archivo y driver o pulsar F4 se abre una ventada donde podemos seleccionar las diferentes opciones de comunicación con las que cuenta Factory I/O dentro de las cuales tenemos PLC Allen-Bradley, Siemens, Control I/O, y comunicaciones Modbus y OPC, en la cual la comunicación que realizaremos es por el driver de OPC Client data Access, que nos permite la comunicación con el OPC de Codesys.

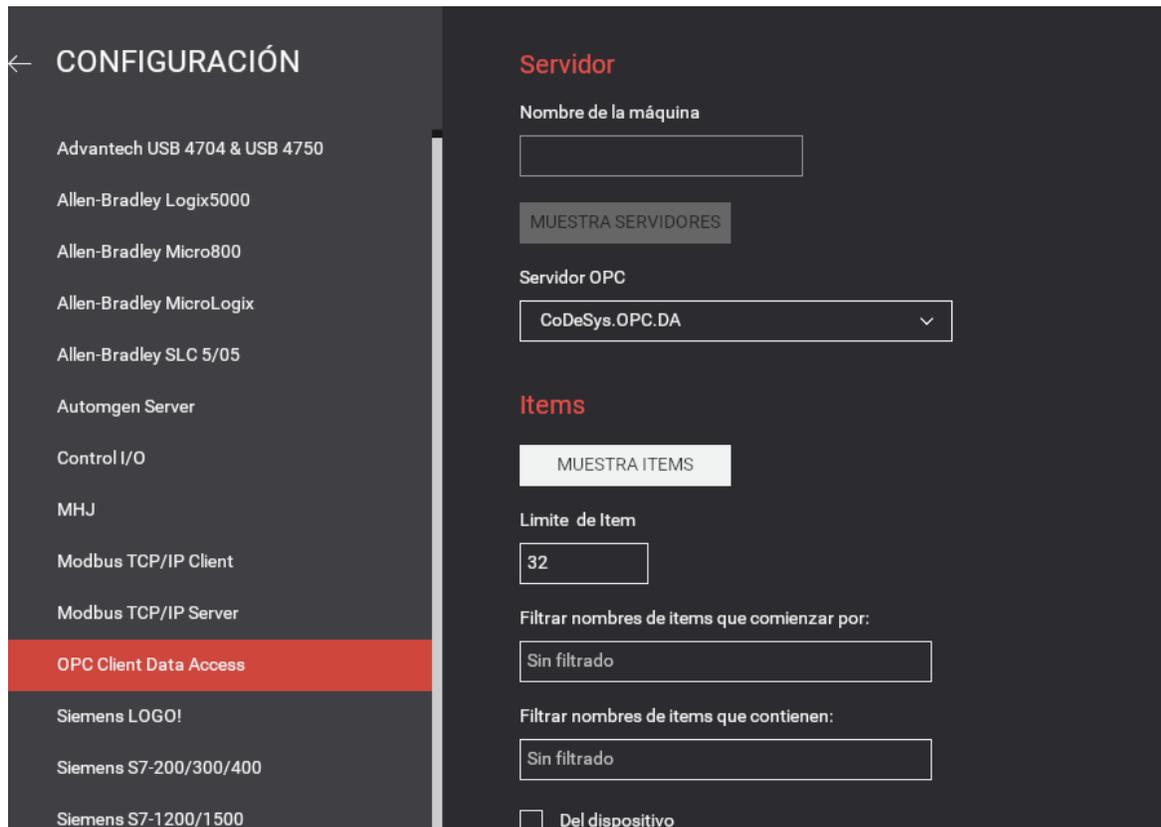
Figura 4–9 Driver Factory I/O



Fuente: el autor

Después de seleccionar el OPC client data Access en la parte superior derecha se encuentra la configuración en el damos mostrar servidores para que actualice la lista de los servidores a los que se puede conectar, y seleccionamos el CoDesys.OPC.DA el cual pertenece a Codesys. En Ítems damos clic en muestra ítems para que muestre los nombres de las variables globales creadas en Codesys.

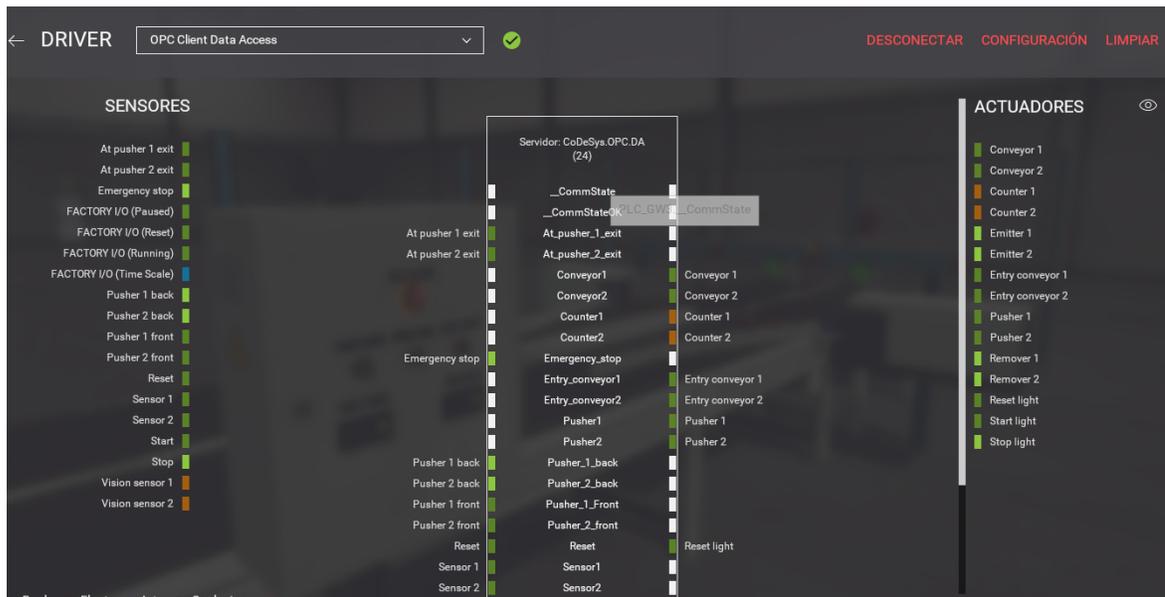
Figura 4–10 configuración del servidor



Fuente: el autor

Al dar atrás podemos ver un rectángulo con los nombres de las variables globales de entrada y salida y a manos izquierda las entradas o sensores y a la derecha los actuadores o salidas las cuales deben ser conectadas según corresponda y damos en conectar en la parte superior de la pantalla.

Figura 4–11 Conexiones de entrada y salida del OPC



Fuente: el autor

Y a continuación se prueba el buen funcionamiento de la programación realizada para la escena seleccionada.

5. CREACIÓN DE ESCENAS

A continuación, se dará la información de las escenas creadas para la guía de aprendizaje diseñadas para ser usadas en la materia de mando y control eléctrico o en sistemas mixtos que son las materias en las cuales se cuenta con el tema de automatización.

5.1 Escena contador de cajas de A a B

En este escenario se utilizó una banda transportadora para llevar cajas de un punto A a un punto B, en el punto se instaló un sensor reflectivo para la detección de cajas y un tablero de control con un display para mostrar el número de cajas y tres botones de inicio, parada y paro de emergencia.

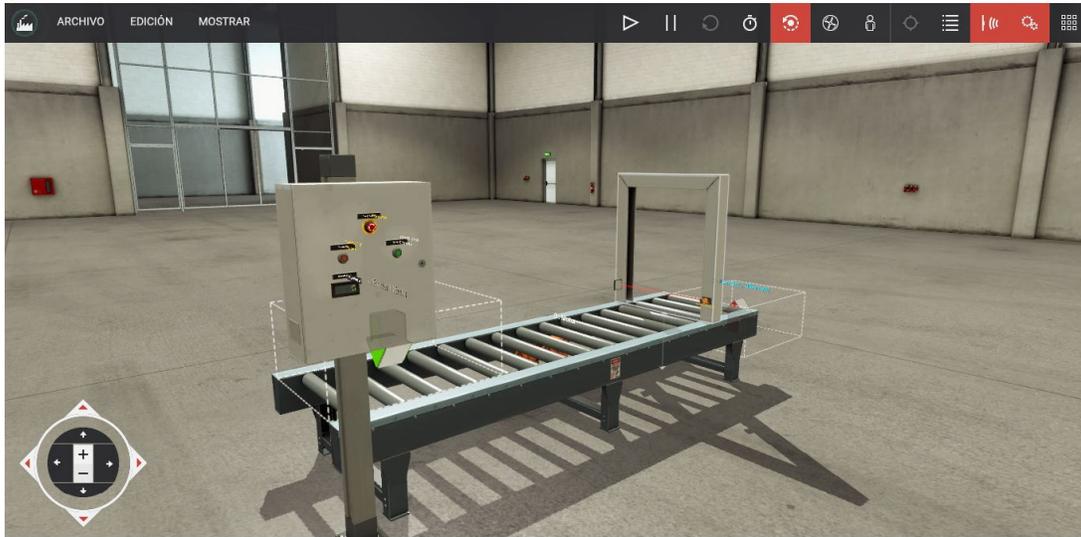
En la siguiente tabla se muestran los elementos a usar en la creación de la escena

Tabla 1 variables escena contador de cajas de A a B

| Nombre del elemento | Nombre de la variable | Tipo de dato |
|------------------------|-----------------------|--------------|
| Retroreflective sensor | Sensor | Booleano |
| Roller conveyer (4m) | Motor | Booleano |
| Start button | Star button | Booleano |
| Stop button | Stop button | Booleano |
| Remover | Remover | Booleano |
| Emitter | Emitter | Booleano |
| Platform Pillar | N/A | N/A |
| Electric swichboard | N/A | N/A |
| Bracket | N/A | N/A |

y a continuación se muestra una imagen de la escena ensamblada en el programa Factory I/O

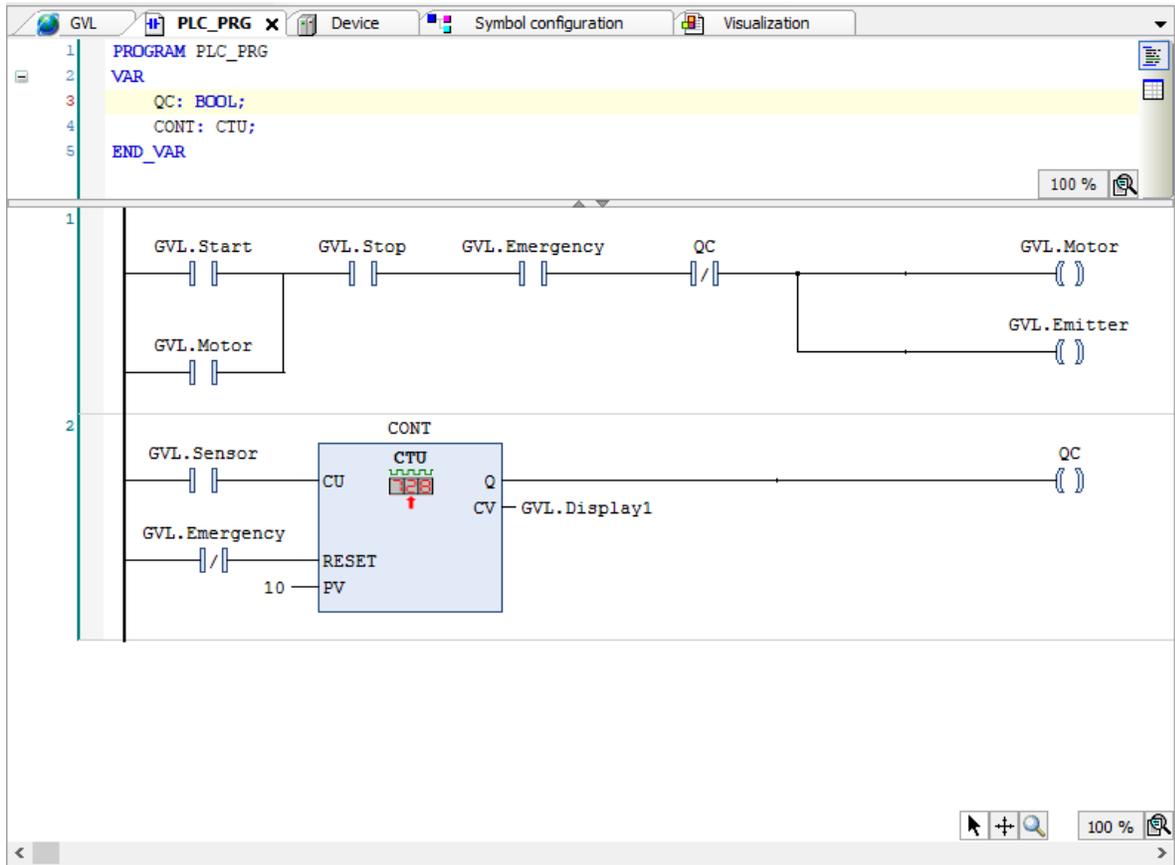
Figura 5–1 Escena contador de cajas de A a B



Fuente: el autor

Para verificar el buen funcionamiento de la escena se realizó la programación la cual para se realizó el código en el lenguaje ladder, en el cual es una escena inicial la cual es un lenguaje básico, el cual puede ser aplicado al inicio como introducción al lenguaje de programación de PLC, en la cual tenemos en la primera línea de programación el arranque de la banda con retención, el cual por medio de una entrada llamada Start, la cual inicia la banda transportadora, además cuenta con un botón Stop el cual detiene la carga, y un botón de emergencia el cual detiene la banda y restablece el contador a cero, además cuenta con un contador ascendente el cual incrementa al paso de las cajas por el sensor.

Figura 5–2 Código contador de cajas de A a B



Fuente: el autor

5.2 Escena Selección de Materia Prima

En este escenario contamos con una banda principal donde llegan 3 tipos de materiales y de distintos colores los cuales deben ser seleccionados por un sensor de visión el cual debe enviar la información a e PLC con el fin de enviar el material a 3 bandas las cuales están interconectadas por medio de un clasificador de rueda emergente el cual debe enviar a tres bandas las cuales envían cada material a una segunda etapa, el cuanta con 3 posiciones a la derecha, izquierda y central loas cuales deben activarse según el material que entre por la banda principal.

Tabla 2 variable escena selección de materia prima

| Nombre del elemento | Nombre de la variable | Tipo de dato |
|---------------------------|-----------------------|--------------|
| Banda | Banda base | Booleano |
| Banda | Banda Crudo | Booleano |
| Banda | Banda liquido | Booleano |
| Banda | Banda Principal | Booleano |
| Display | Display base | INT |
| Display | Display Crudo | INT |
| Display | Display Liquido | INT |
| Botón | Emergency | Booleano |
| Rueda emergente derecha | P base Derecha | Booleano |
| Rueda principal | P crudo | Booleano |
| Rueda emergente Izquierda | P liquido izquierda | Booleano |
| Botón | Start | Booleano |
| Sensor Difusión | Sensor base | Booleano |
| Sensor Difusión | Sensor crudo | Booleano |
| Sensor Difusión | Sensor Liquido | Booleano |
| Botón | Stop | Booleano |
| Sensor de visión | Vision Sensor 1 | INT |

Figura 5–3 Escena selector de material



Fuente: El autor

Nuevamente para verificar el funcionamiento de esta escena se realizó la programación del PLC en Codesys con el fin de confirmar que no cuente con errores, además de dar una guía de programación para los lectores de este documento.

El tablero de control cuenta con un botón de inicio, uno de parada y uno de emergencia, además de tres displays para mostrar el conteo de las piezas organizadas según el material.

El lenguaje utilizado para esta programación se realizó en Grafset (SFC), ya que además queremos mostrar la aplicación de los diferentes lenguajes de programación que se pueden usar en Codesys.

El programa inicia parando el proceso ya que inicialmente debe estar en esa condición, al pulsar el botón inicio debe activar la banda principal para enviar los materiales a la clasificadora que es gobernada por el PLC el cual dependiendo de la señal analógica que envíe el sensor de visión debe tomar la decisión en qué dirección debe enviar el material, además como debe en cualquier estado parar el proceso se adiciona en cada paso del programa un salto el cual puede ser a solo parar el proceso con el botón stop, o a parar el proceso y reiniciar el conteo el cual es con el botón de emergencia.

Además, cuenta con acciones programadas en otros lenguajes para comprobar el uso de otros lenguajes como texto estructurado, un lenguaje de alto nivel que puede ser implementado en su totalidad en el proyecto o en este caso para acciones programadas que se usaron para esta escena, para la cual el código descrito a continuación realiza en conteo de las piezas en cada una de las bandas se los diferentes tipos de material.

```
IF GVL.Sensor_base = TRUE AND FLAG02= FALSE THEN  
    FLAG02 := TRUE;  
    CONTA1:= CONTA1+1;  
    GVL.Display_base:= CONTA1;  
END_IF;
```

6. USO DE ESCENAS PREDISEÑADAS

También se realizó el código y la guía de aprendizaje en la cual se incluyeron escenas prediseñadas con la que cuenta Factory I/O las cuales se describen a continuación.

6.1 Escena estación de separación

Esta escena prediseñada cuenta con cuatro bandas dos de ingreso de material y dos para retirar las piezas ya seleccionadas, las cuales al pasar por un sensor de visión el cual dependiendo del color envía un dato tipo INT a el PLC el cual se encarga de ordenar a los cilindros neumáticos si deben accionar dependiendo del color del material a ingresar, con el fin de que en una de las bandas solo pase material de color verde y por la otra banda solo material azul.

Tabla 3 variables escena estación de separación

| Nombre del elemento | Nombre de la variable | Tipo de dato |
|---------------------|-----------------------|--------------|
| Sensor | At pusher 1 exit | Booleano |
| Sensor | At pusher 2 exit | Booleano |
| Banda | Conveyor 1 | Booleano |
| Banda | Conveyor 2 | Booleano |
| Display | Counter 1 | INT |
| Display | Counter 2 | INT |
| Botón | Reset | Booleano |
| Botón | Emergency stop | Booleano |
| Banda | Entry conveyor 1 | Booleano |
| Banda | Entry conveyor 2 | Booleano |
| Cilindro | Pusher 1 | Booleano |
| Cilindro | Pusher 2 | Booleano |
| Sensor | Pusher 1 back | Booleano |
| Sensor | Pusher 2 back | Booleano |

| | | |
|------------------|-----------------|----------|
| Sensor | Pusher 2 front | Booleano |
| Botón | Stop | Booleano |
| Sensor de visión | Vision Sensor 1 | INT |
| Botón | Start | Booleano |
| Sensor de visión | Vision sensor 2 | INT |

Figura 6–1 Escena estación de separación



Fuente: el autor

Se realizó la programación de esta escena con el fin de entregar un ejemplo de programación a los estudiantes que sirva de guía en el momento de realizar la programación en Codesys.

tapa superior y base inferior respectivamente en la primera llega al posicionador 1 para poder ser tomada por la estación de movimiento la cual al llegar la base al segundo posicionador 2 lleva la tapa a la segunda banda para realizar en ensamble al terminar el ensamble se levanta el posicionador 2 para que la pieza ensamblada sea llevada a otro proceso.

Tabla 4 variables escena estación de ensamble

| Nombre del elemento | Nombre de la variable | Tipo de dato |
|---------------------|-----------------------|--------------|
| Botón | Auto | Booleano |
| Sensor | Base at place | Booleano |
| Banda | Bases conveyor | Booleano |
| Banda | Bases emitter | Booleano |
| Display | Counter | INT |
| Sensor | Base clamped | Booleano |
| Botón | Reset | Booleano |
| Botón | Emergency stop | Booleano |
| Cilindro | Clamp base | Booleano |
| Cilindro | Clamp lid | Booleano |
| Generador de vacío | grap | Booleano |
| Sensor | Item detected | Booleano |
| Sensor | Lid at place | Booleano |
| Sensor | Lid clamped | Booleano |
| Sensor | Lids emitter | Booleano |
| Botón | Stop | Booleano |
| Selector | Manual | INT |
| Botón | Start | Booleano |
| Movimiento X | Move X | Booleano |
| Movimiento Z | Move Z | Booleano |

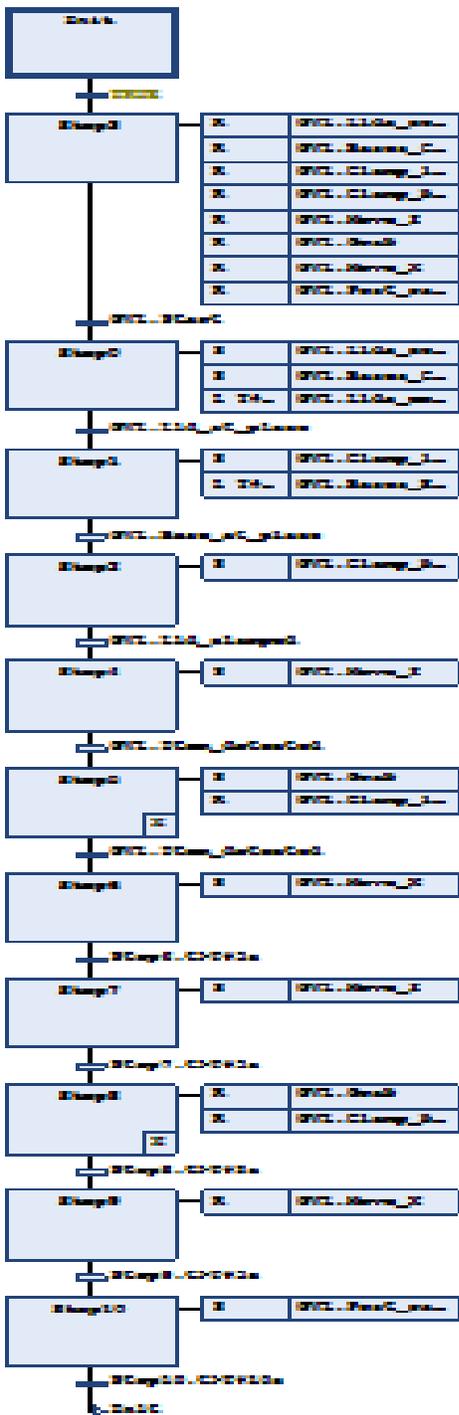
Figura 6–3 Escena estación de ensamble



Fuente: el autor

Ahora presentamos el código realizado para la confirmación de la escena la cual está basado en el lenguaje grafcet, y realizado bajo el programa Codesys.

Figura 6-4 código estación de ensamble



Fuente: el autor

7.CONCLUSIONES Y RECOMENDACIONES

7.1 Conclusiones

Se realizo la comunicación entre los programas Factory I/O y el Programa Codesys los cuales permitieron realizar una nueva forma de enseñar la programación de PLC.

El documento permite dar una guía para la conexión entre los programas y llevar a automatizar las diferentes escenas que se encuentran en el como lo son el contador de cajas de A a B, la Escena de selección de materia prima, estación de separación y la estación de ensamble.

Algunas escenas fueron tomadas directamente de las que cuenta el programa Factory I/O para dar a conocerlas y además brindan nuevas experiencias en el mundo de la automatización.

7.2 Recomendaciones

Se espera que la universidad en el futuro adquiera el programa de Factory I/O para poder hacer uso de las nuevas modificaciones que tiene el programa.

Se pueden tomar escenas prediseñadas y realizar su respectiva guía de laboratorio, con el fin de mejorar y actualizar las experiencias a través de ellas.

**A. Anexo: Guía de laboratorio
escena contador de cajas de A a B**

GUÍA LABORATORIO VIRTUAL

Escena contador de cajas de A a B

ASIGNATURA: _____

FECHA: _____

Objetivo: lograr que el estudiante se familiarice con la programación basada en el lenguaje ladder, y le cambio de lógica de contactos a lógica programable.

Materiales:

- Computador
- Software Factory I/O
- Software Codesys V3.5 SP10
- Archivo Escena Contador_A_To_B

Descripción:

En la escena se requiere que al pasar las cajas por el sensor posicionado en la extremo B debe mostrar la cantidad de cajas que ingresan, además si pulsamos el botón de emergencia el conteo debe reiniciarse.

Procedimiento:

- Programar el PLC en el software Codesys usando uno de los lenguajes de programación con los que cuenta Codesys.
- Realizar la conexión entre el OPC y Factory I/O.
- Poner a punto la automatización.

Duración:

(1 hora de programación y 1 de puesta a punto)

Entregables:

Debe entregar un informe donde explique el código y explique el funcionamiento.

Debe realizar un video exponiendo el proceso, así como el funcionamiento los posibles errores que solución.

**B. Anexo: Guía de laboratorio
escena selección de materia prima**

GUÍA LABORATORIO VIRTUAL

Escena selección de materia prima

ASIGNATURA: _____

FECHA: _____

Objetivo: manejo de diferentes tipos de datos introducción a graficet con convergencia y divergencia OR o ADD, y el uso de sensores de visión.

Materiales:

- Computador
- Software Factory I/O
- Software Codesys V3.5 SP10
- Archivo Escena Materia prima

Descripción:

se requiere seleccionar el material que ingresa a la planta en 3 tipos de materiales los cuales deben ir en tres direcciones dependiendo del tipo de material.

Procedimiento:

- Programar el PLC en el software Codesys usando uno de los lenguajes de programación con los que cuenta Codesys.
- Realizar la conexión entre el OPC y Factory I/O.
- Poner a punto la automatización.

Duración:

(4 hora de programación y 3 de puesta a punto)

Entregables:

Debe entregar un informe donde explique el código y explique el funcionamiento.

Debe realizar un video exponiendo el proceso, así como el funcionamiento los posibles errores que solución.

C. Anexo: Guía de laboratorio escena estación de separación

GUÍA LABORATORIO VIRTUAL

Escena Estación de separación

ASIGNATURA: _____

FECHA: _____

Objetivo: obtener conocimientos en la programación basada en el lenguaje grafcet, y el uso de sistemas neumáticos en la automatización.

Materiales:

- Computador
- Software Factory I/O
- Software Codesys V3.5 SP10
- Archivo Escena Estacion_separacion

Descripción:

Se requiere que las piezas que ingresen al proceso se separen según el color en dos bandas una para color azul y otra para el color verde.

Procedimiento:

- Programar el PLC en el software Codesys usando uno de los lenguajes de programación con los que cuenta Codesys.
- Realizar la conexión entre el OPC y Factory I/O.
- Poner a punto la automatización.

Duración:

(4 hora de programación y 3 de puesta a punto)

Entregables:

Debe entregar un informe donde explique el código y explique el funcionamiento.

Debe realizar un video exponiendo el proceso, así como el funcionamiento los posibles errores que solución.

D. Guía de laboratorio escena estación de ensamble

GUÍA LABORATORIO VIRTUAL

Escena estación de ensamble

ASIGNATURA: _____

FECHA: _____

Objetivo: manejo de diferentes tipos de datos introducción a graficet con convergencia y divergencia OR o ADD, y el uso de sistemas neumáticos y robóticos.

Materiales:

- Computador
- Software Factory I/O
- Software Codesys V3.5 SP10
- Archivo Escena estación_ensamble

Descripción:

se requiere ensamblar la base y la tapa de una pieza las cuales entrar a una banda transportadora al terminar el ensamble son enviadas a otro proceso.

Procedimiento:

- Programar el PLC en el software Codesys usando uno de los lenguajes de programación con los que cuenta Codesys.
- Realizar la conexión entre el OPC y Factory I/O.
- Poner a punto la automatización.

Duración:

(4 hora de programación y 3 de puesta a punto)

Entregables:

Debe entregar un informe donde explique el código y explique el funcionamiento.

Debe realizar un video exponiendo el proceso, así como el funcionamiento los posibles errores que solución.

8. BIBLIOGRAFÍA

CODESYS Group. (16 de 05 de 2020). *CODESYS*. Obtenido de <https://www.codesys.com/the-system.html>

Flower Leiva, L. (2008). *Controles y Automatismos*. Bogota: Panamericana Formas e Impresos S.A.

games, r. (17 de 05 de 2020). *real games*. Obtenido de factory I/O: <https://docs.factoryio.com/>

Izaguirre Castellanos, E. (2012). *Sistemas de Automatización*. Santa Clara : Feijóo.

Matrikon. (17 de 05 de 2020). *Matrikonopc*. Obtenido de <https://www.matrikonopc.es/opc-servidor/index.aspx>

Orozco Gutierrez , A. A., Guarnizo Lemus, C., & Holguin Londoño, M. (2008). *Automatismos Industriales*. Pereira: Universidad Tecnológica de Pereira.