

**Prepráctica 1**

**Tema:** Interfaz Hombre Maquina y Programación Ladder: Temporizadores, comparadores y contadores

**Objetivos**

**Objetivo general**

Elaborar programación ladder utilizando temporizadores, comparadores y contadores del software Connected Components Workbench para la automatización de aplicaciones industriales.

**Objetivos específicos**

1. Comprender los procesos industriales utilizando datos adjuntados en los gráficos de cada ejercicio tales como sensores y actuadores.
2. Identificar las variables de los procesos para la creación de estos en el software de programación.
3. Analizar el funcionamiento de temporizadores en la programación escalera para la generación de diferentes anchos de pulso en salidas digitales.
4. Elaborar la programación escalera utilizando el controlador Micro850 para la realización de aplicaciones industriales.

**¿Qué actividades realizarán?**

1. Investigar y analizar el diagrama de fuerza y control para la inversión de giro de un motor trifásico. Además, realizar el bosquejo del diagrama de conexiones para que pueda ser accionada desde un PLC micro850.
2. Identificar y describir brevemente el funcionamiento (nivel de voltaje, conexiones, entre otras) de sensores, botoneras, interruptores y finales de carrera de acuerdo con la siguiente escena en Factory I/O. (Descargar los [archivos de la prepráctica](#) para completar el ejercicio propuesto)
3. Realizar una programación en CCW diferente a los ejemplos de la escena anterior de Factory IO, utilizando los mismos sensores y actuadores identificado en el literal anterior



con contactos (NA – NC), bobinas (directa y negada), y algunos de los elementos nombrados a continuación:

- Temporizador con retraso a la conexión.
  - Temporizador con retraso a la desconexión.
  - Temporizador con retraso a la conexión y desconexión.
  - Temporización retenida.
  - Temporización del pulso.
  - Comparadores.
  - Contador ascendente
  - Contador descendente
  - Contador ascendente/descendente
  - Contacto de Pulso e instrucción R\_TRIG
  - Instrucción Set y Reset
  - Instrucciones de suma y resta con comparadores.
4. Descargar la programación anterior en el simulador del micro 850 y comprobar su funcionamiento con la escena de Factory IO.
5. Realizar un bosquejo de HMI en CCW con un botón de marcha, stop, sensores y actuadores que usted considere relevante.

### **¿Como lo realizaran?**

Programas por utilizar:

- Connected Components Workbench -CCW (se recomienda versión 13)
- RsLinx Classic
- Simulador micro850
- Factory IO

### **¿Cuáles son los entregables?**

Realizar una infografía de manera individual/grupal (pareja) de las actividades a desarrollar en formato A3 (horizontal o vertical), el cual debe tener los siguientes elementos:

- Un tema claro y conciso: Es decir, comunicar un tema específico de manera clara y fácil de entender.
- Datos relevantes y precisos: La información presentada en la infografía debe ser precisa y relevantes a su tema.
- Diseño atractivo y llamativo: La infografía debe ser visualmente atractiva y llamar la atención del espectador.



- Estructura fácil de seguir: La información debe presentarse en una estructura clara y fácil de seguir para que el espectador no tenga problemas para entender su contenido.
- Fuentes y referencias: La infografía debe incluir las fuentes y referencias utilizadas para obtener la información presentada en ella.

Nota: Todas las preprácticas serán entregados en el aula virtual en PDF, hasta la semana de la práctica, las faltas ortográficas serán penalizadas, así como la copia ya sea con otros reportes o de internet.

### Material de apoyo

**Nota:** Revisar en el siguiente orden el material de apoyo.

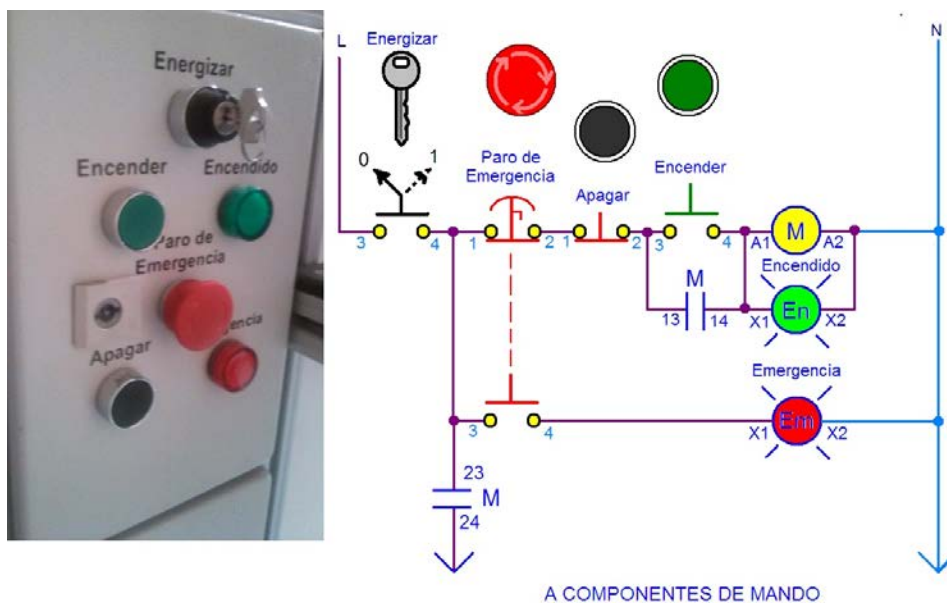
1. [Guías de prácticas del laboratorio de automatización de procesos industriales](#)
2. [Video - Como descargar e instalar CCW V12](#)
3. [Enlace de descarga de CCW](#)
4. [Video de descarga Factory IO](#)
5. [Enlace de descarga de Factory IO](#)
6. [Red Ethernet Devices en Rslinx](#)
7. [Introducción a CCW](#)
8. [Descargar la programación hacia el simulador micro800](#)
9. [Archivos de la prepráctica 1](#)
10. [Importar proyecto en CCW](#)
11. [Establecer conexión entre CCW, RSLinx y Factory IO](#)
12. [Solución de ejemplo 0 de la simulación](#)
13. [Errores más comunes que cometes en CCW](#)
14. [Temporizadores y contadores en CCW](#)
15. [Marcha y parada de un motor utilizando un PLC Micro850](#)
16. [Identificación de sensores y actuadores digitales](#)
17. [Realizar diagramas de fuerza utilizando CADE SIMU](#)
18. [Realizar diagramas de control utilizando CADE SIMU](#)
19. [Solución de ejemplo 0 de la simulación](#)
20. [Automatización de una línea de clasificación de cajas por peso utilizando un PLC MICRO850.](#)
21. <http://www.electroclub.com.mx/2020/07/programa-1-basico-en-un-plc-arranque-y.html>
22. [Configurar un PanelView en CCW](#)
23. [Descargar una pantalla HMI en un PanelView 800 utilizando CCW](#)
24. [Conexiones físicas para el control de un motor utilizando un PowerFlex 4M, PLC micro850 y PanelView 800.](#)
25. [Implementación de control de motor utilizando PowerFlex 4M, PLC micro850 y PanelView 800.](#)

**Anexo**

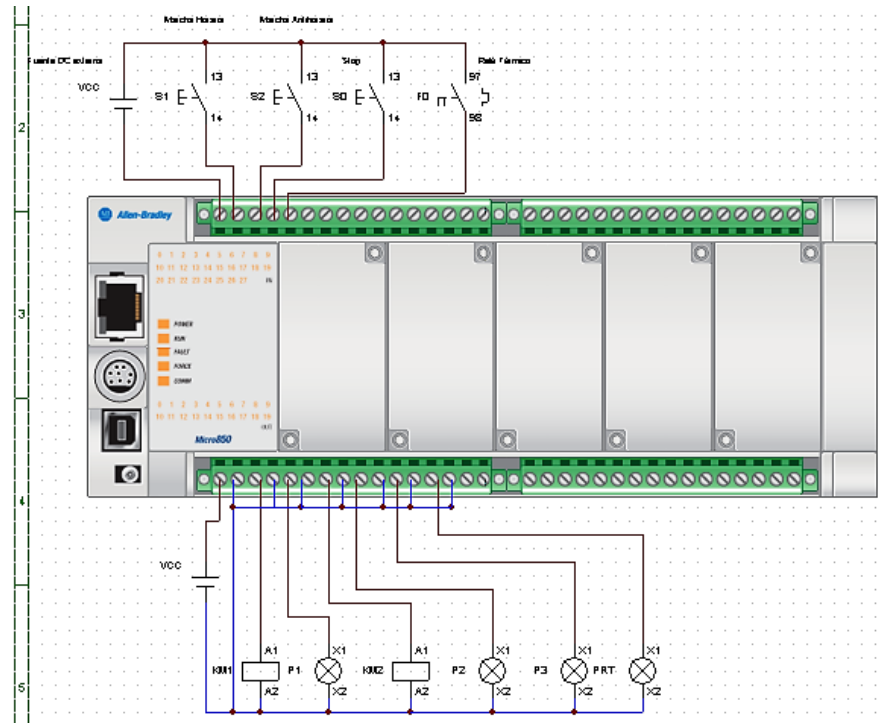
**Elementos de maniobra y control**

|                    |   |   |                     |   |   |
|--------------------|---|---|---------------------|---|---|
| Interruptor        |  |  | Conmutador unipolar |  |  |
| Pulsadores NA - NC |  |  | Conmutador bipolar  |  |  |
| Microinterruptor   |  |  | Relé                |  |  |

**Diagrama de circuito eléctrico de control con lógica cableada**



### Conexiones físicas del sistema de control utilizando un PLC micro850



### Anexos de actividades

Identificar y describir brevemente el funcionamiento (nivel de voltaje, conexiones, entre otras) de sensores, botoneras, interruptores y finales de carrera que se encuentra en la siguiente figura:



Figura. Escena de Factory I/O para el movimiento de una banda transportadora

## **1 Sensores**

### **1.1 Sensor reflectivo**

Se trata de un sensor fotoeléctrico que contiene un transmisor y un receptor en una caja única y emite un haz de luz pulsada, por medio de un filtro de polarización hacia un reflector distante, vuelve la luz reflejada al sensor, que pasan a través de un segundo filtro antes de llegar al receptor. Cuando se interrumpe el haz de luz con un objeto el receptor detecta la intensidad de luz reducida y activa el sensor.

Es decir que, con respecto a la imagen, una vez que la caja pase por el haz luminoso emitido, donde actúa el receptor captando la señal y así el sensor se va a activar, que es cuando se va a mandar la señal de salida. Lo que va a provocar es que la caja no se resbale por fuera de la banda transportadora.

Los sensores reflectivos, como es el caso Mini Beam, trabajan entre 10 y 30 voltios, con conexión integral QD rápida M12, salidas bipolar (npn y pnp), con tiempo de respuesta de salida de 1ms.

### **1.2 Sensor difuso 2**

El modo difuso, es un sensor fotoeléctrico reflectante, que contiene un transmisor y un receptor en una sola carcasa. El sensor emite un haz de luz hacia un objetivo distante, que actúa como un reflector, volviendo parte de la luz transmitida al sensor. El receptor detecta la cantidad de luz reflejada por el objetivo, activando el sensor cuando la intensidad de la luz alcanza un valor de umbral.

En este caso, el color de la caja va a influir para que este sensor se active, ya que al ser metálica la luz reflejada será mayor, permitiendo su activación. De tal forma lo que va a suceder es que también evitará que la caja resbale por la banda transportadora, cambiando la dirección del motor, una vez que el sensor se activa.

Un ejemplo está el sensor difuso de contrinex, con polaridad pnp con conexión M12 de 14 pines, tiene tensión de alimentación entre 10 y 36 voltios. Los sensores de difusión en el factory I/O el color del led es rojo para detectar, capaz de detectar materiales sólidos con un rango de detección entre 0 y 1.6 metros.

### **1.3 Sensor difuso 1**

Tiene la misma definición que el sensor difuso 2, es decir que contiene un transmisor y un receptor en una sola carcasa. El sensor emite un haz de luz hacia un objetivo distante, que actúa como un reflector, volviendo parte de la luz transmitida al sensor, pero en el caso de la imagen



va a detectar si la caja se resbaló de la banda por alguna falla o si existe algún operador alrededor de tal forma que avisa uno de los casos para evitar que alguien se acerque a la banda.

## **2 Botoneras**

El código utiliza los colores como expresión de lectura. De esta manera podemos saber cuál es el que cuenta con las condiciones necesarias según nuestra necesidad. Colores blancos, grises o negros son elegidos para los equipos de mando, aunque también está permitido utilizar el color verde.

El color rojo es utilizado para las paradas de emergencia. Los colores blanco y gris también son usados para paradas. También están los colores amarillo y azul, el primero se reserva para situaciones que indican advertencia o anomalía y el segundo para situaciones de carácter obligatorio. El color rojo se debe accionar en caso de condiciones peligrosas o de riesgo, un ejemplo de aplicación es en una parada de emergencia. El color amarillo se utiliza en presencia de situación anormal. El azul se activa cuando una acción es obligatoria, ejemplo, una función de rearme.

### **2.1 Botón paro de emergencia**

Los dispositivos de parada de emergencia deben ser instalados en todas aquellas máquinas en las cuales existan peligros de tipo mecánico durante las condiciones normales de trabajo. Hay que tener en cuenta que, dependiendo del tipo de máquina, puede ser necesaria la instalación de más de un dispositivo de parada de emergencia; por ejemplo, en máquinas con más de un puesto de mando y control, máquinas de características especiales con varios puntos de peligro separados del puesto de mando y control, etc.

La función esencial del dispositivo de parada de emergencia será la de interrumpir (en caso de peligro) el suministro de las fuentes de alimentación de energía (corriente eléctrica, aire a presión, etc.) y parar la máquina lo más rápidamente posible.

## **3 Selector**

El selector es un conmutador con dos o más posiciones estables, en las que permanece tras su accionamiento. Los selectores son similares a los interruptores y conmutadores en cuanto a funcionamiento, aunque para su actuación suelen llevar un botón, palanca o llave giratoria (que puede ser extraíble).

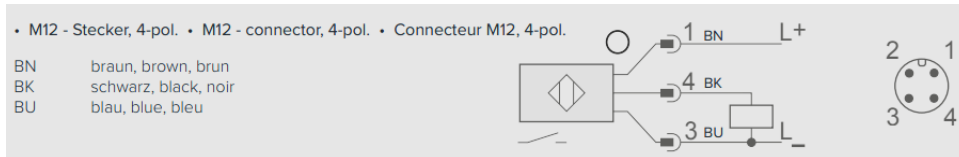
## **4 Torre de luz industrial**

Están diseñados para ser montados fácilmente sobre equipos industriales y pueden extenderse a cualquier altura que sea claramente visible para todos en el área circundante. La iluminación de las torres apiladas admite muchos colores diferentes y también puede proporcionar alertas audibles de estado, proporcionando un sistema de alerta fácil de ver para notificar a todo el

mundo en un entorno industrial ruidoso y concurrido sobre el estado actual de la máquina. Esto también puede ser importante si la máquina se controla o configura a distancia, cuando no hay un operador humano local que informe a los demás de la operación de la máquina que se está realizando.

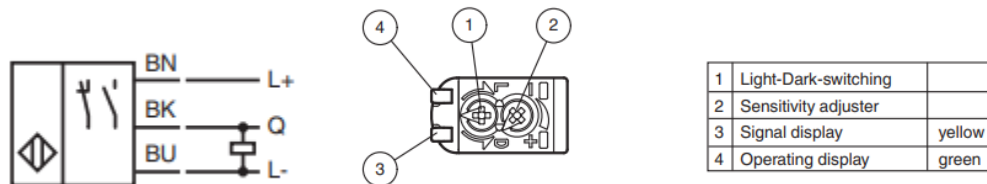
### 1. Sensor de proximidad óptico (Diffuse sensor 1)

Tiene un rango de detección de 0.05 a 1.4m con supresión de fondo, modo de encendido N.A. (normalmente abiertos) con grado de protección IP65. Se alimenta con 10VDC hasta máximo 36VDC, con capacidad de carga de corriente máx. de 200mA, consume max. 25mA, posee 3 pines de conexión chapados en oro. Conexión y asignación de pines:



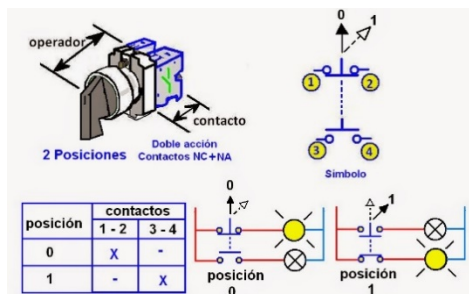
### 2. Sensor retro refléctate (retroreflective sensor)

Es un sensor de proximidad con rango de detección de 0 a 5m, distancia de reflector de 0.01 a 5m, se alimenta con 10 a 30VDC, frecuencia de conmutación 1kHz, corriente de conmutación máx. 100mA, tiempo de respuesta 0.5ms, tiene un grado de protección IP67, y tiene un LED indicador verde de encendido, conexión y ensamble:



### 3. Selector de dos posiciones.

Es una perilla selectora de dos posiciones a 90°, cierra o abre los contactos de acuerdo con una posición seleccionada manual, y tiene manecilla simétrica.



### 4. Botones pulsadores (Start)



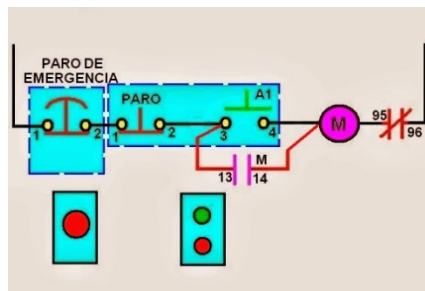
Estos pulsadores tienen metal chapado en cromo, presentan retorno por resorte de retorno si sus contactos son N.A. (normalmente abiertos), presentan protección contra cortocircuito con fusible de 10 A, la mayoría tiene grado de protección IP66, NEMA 13 o IK06. No posee materiales tóxicos, la corriente térmica convencional de aislamiento es de 10 A. Se presionan para cerrar el contacto, su conexión es a través de dos cables.

#### 5. Columnas luminosas o pilotos (Stacks light)

Son elementos de visualización óptica en forma de columnas, que indican el estado de un proceso. Se rigen bajo la norma IEC 60204 -1, que indica que el color rojo representa emergencias, amarillo una anomalía, blanco chequeo opcional, azul una acción obligatoria y verde funcionamiento normal.

#### 6. Botón de parada de emergencia o STOP.

Son botones pulsadores con 1 contacto NC (normalmente cerrado), con cabezal estándar tipo seta rojo de 40mm de diámetro, tienen protección contra cortocircuitos mediante fusible de 10 A, presentan protección normativa NEMA e IEC. Su accionamiento permite un enclavamiento mecánico, es resistente a vibraciones e impactos. Conexión y funcionamiento:



Investigar y analizar el diagrama de fuerza y control para la inversión de giro de un motor trifásico. Además, realizar el bosquejo del diagrama de conexiones para que pueda ser accionada desde un PLC micro850.

**Analizar el proyecto en diagrama de lógica escalera realizado en el software Connected Component Workbench, explicando cuales son los sensores y actuadores en las entradas y salidas del PLC de la figura 2 para que cada programa del proyecto pueda funcionar correctamente.**

Descargar los [archivos de la prepráctica](#) para completar el ejercicio propuesto.

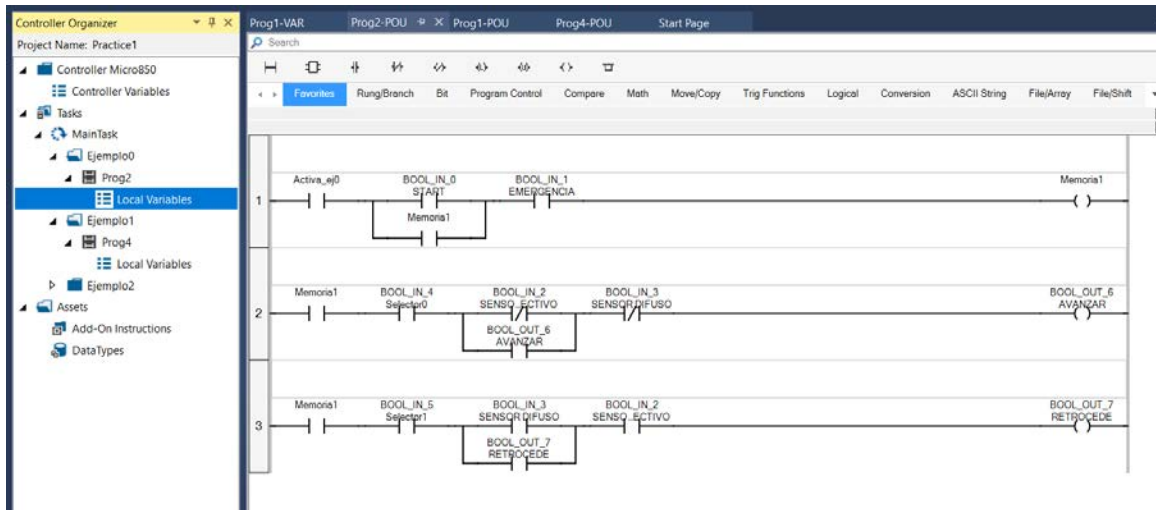


Figura 2: Programación en diagrama de escalera para el movimiento de una banda transportadora

En este diagrama escalera se presentan las siguientes entradas (color verde) y salidas (color naranja):

Entradas.

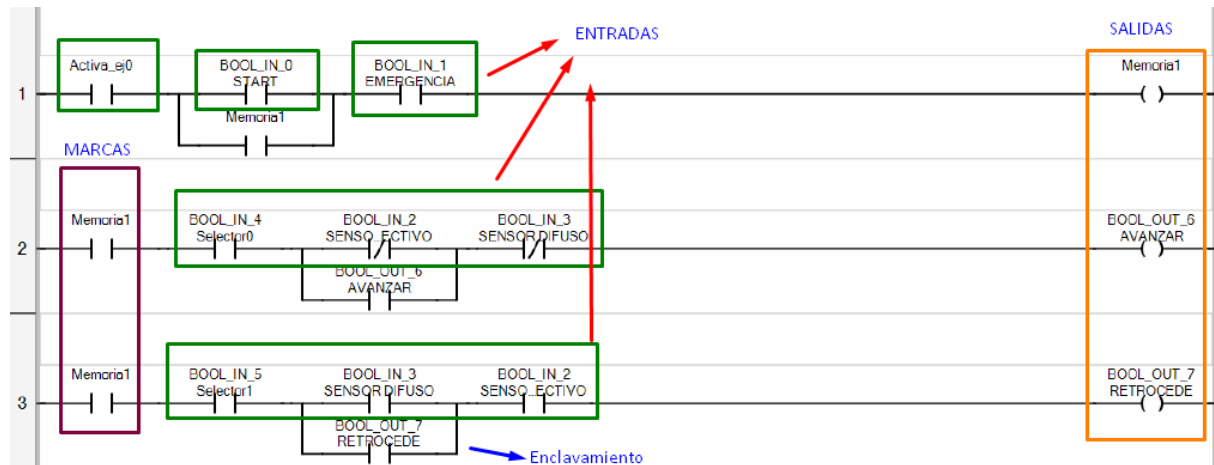
- START (BOOL\_IN\_0)
- EMERGENCIA (BOOL\_IN\_1)
- Selector0 (BOOL\_IN\_4)
- Selector1 (BOOL\_IN\_5)
- Sensor Reflectivo (BOOL\_IN\_2)
- Sensor difuso (BOOL\_IN\_3)

Salidas.

- AVANZAR (BOOL\_OUT\_6)
- RETROCEDE (BOOL\_OUT\_7)

Marcas/memorias.

- Memoria1



**Figura 3: Conexiones del PLC de Factory I/O**

## Resumen del análisis

El proyecto posee 6 entradas y 2 salidas según la figura 2, como entrada posee a un sensor reflectivo y uno difuso, además de botones como start, botón de emergencia y por último un selector donde sus estados están puestos como 2 entradas al PLC.

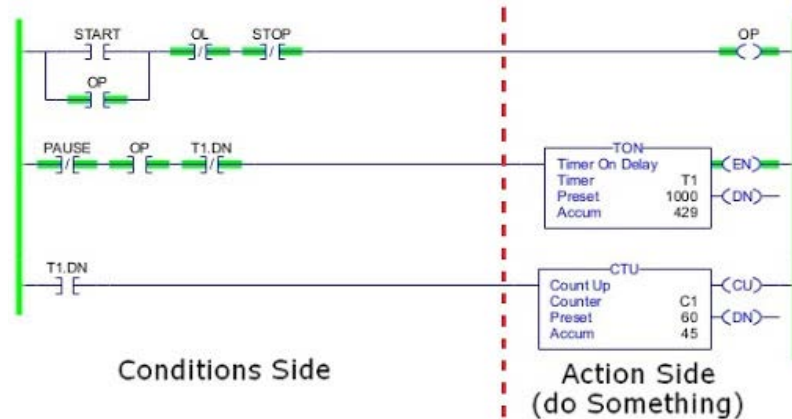
Las salidas son 2, ambas conectadas al motor, la salida 6 permite avanzar, mientras que la salida 7 permite retroceder, es decir, los actuadores.

Para que funcione el proyecto, se tuvo que cambiar la configuración de las salidas en el programa factory I/O, debido a que se estaban utilizando las salidas 0 y 1, sin embargo, las salidas previamente configuradas estaban en el 6 y 7.

## Anexo de marco teórico de la prepráctica

### Diagrama de Lógica Escalera

El diagrama de lógica de escalera (LD) es un lenguaje gráfico que utiliza la combinación de contactos con bobinas, organizado como un diagrama de cableado de lógica de escalera de relés, asemejándose a una representación gráfica de ecuaciones booleanas. El término "escalera" está relacionado con el concepto de peldaños conectados a raíles de potencia verticales a ambos extremos, en los que cada peldaño representa un circuito individual.

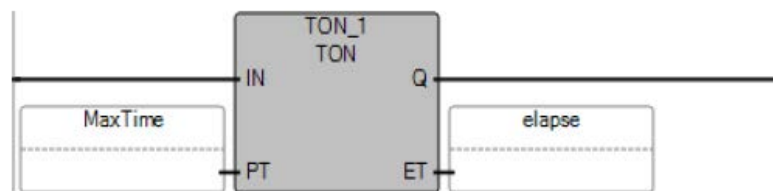


### Temporizadores

Son funciones de programación que permiten el control de acciones específicas en función del tiempo, principalmente cumple dos funciones relevantes: contar tiempo y poner una variable de salida asociada a nivel alto.

Las instrucciones de temporizador de Connected Components Workbench son las siguientes:

a) *Temporizador con retraso a la conexión.*

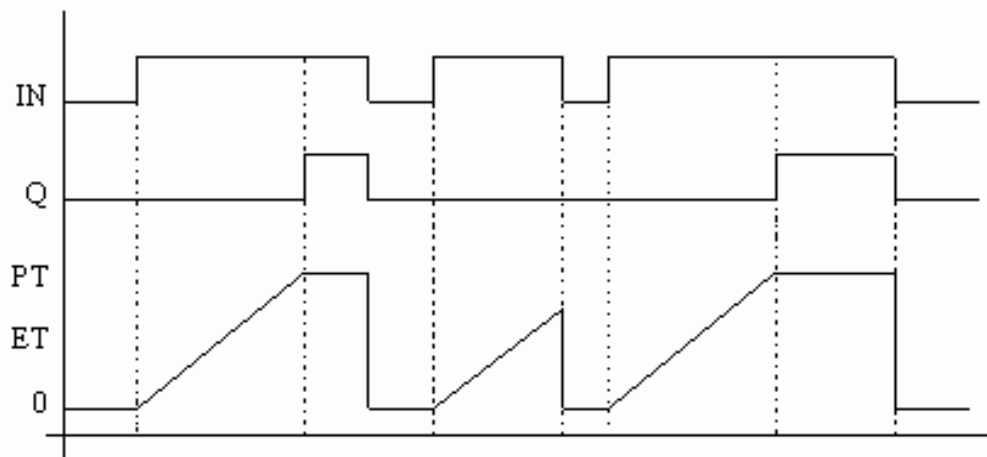


| Elemento | Descripción |
|----------|-------------|
|----------|-------------|

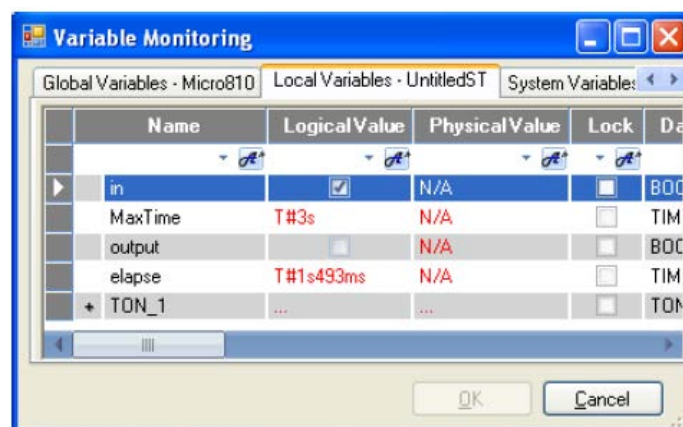


|                  |  |
|------------------|--|
| <b>IN (Bool)</b> | En el caso que reciba un flanco ascendente, empieza a aumentar el temporizador interno, en el caso contrario recibe un flanco descendente se detiene y restablece el temporizador interno. |
| <b>PT (Time)</b> | Tiempo programado máximo ejemplo (T#3S).   |
| <b>Q (Bool)</b>  | Si es Verdadero: el tiempo programado ha transcurrido.   |
| <b>ET (Time)</b> | Tiempo actual transcurrido.  |

A continuación, se muestra el diagrama de tiempo de la funcionalidad del temporizador con retraso a la conexión:

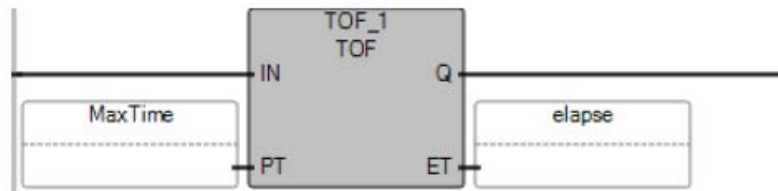


Resultados



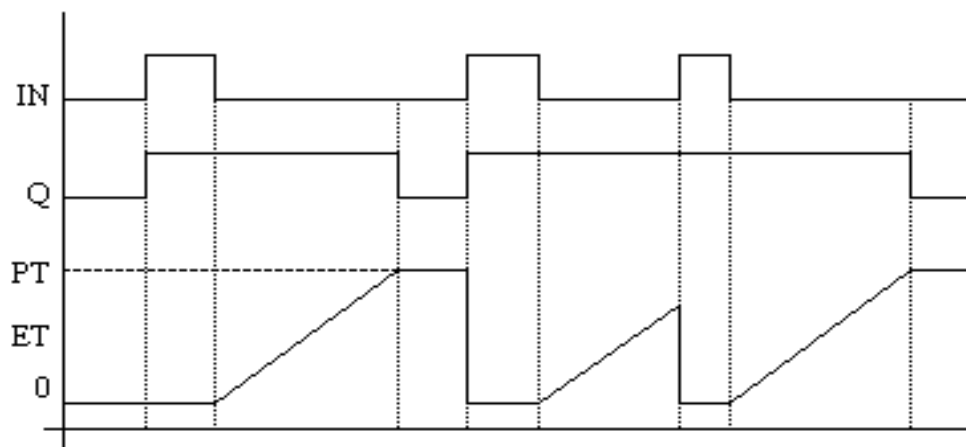
b) Temporizador con retraso a la desconexión.





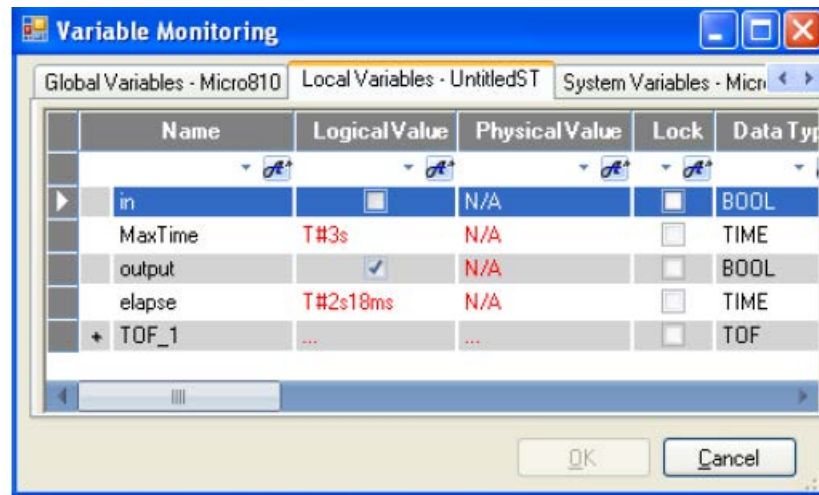
| Elemento         | Descripción  |
|------------------|--|
| <b>IN (Bool)</b> | En el caso que reciba un flanco descendente, empieza a aumentar el temporizador interno, en el caso contrario reciba un flanco ascendente se detiene y restablece el temporizador interno. |
| <b>PT (Time)</b> | Tiempo programado máximo.  |
| <b>Q (Bool)</b>  | Si es Verdadero: el tiempo programado ha transcurrido.   |
| <b>ET (Time)</b> | Tiempo actual transcurrido.  |

A continuación, se muestra el diagrama de tiempo de la funcionalidad del temporizador con retraso a la desconexión:



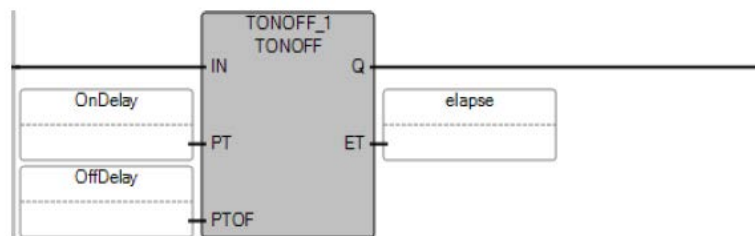
Resultados





c) *Temporizador con retraso a la conexión y desconexión*

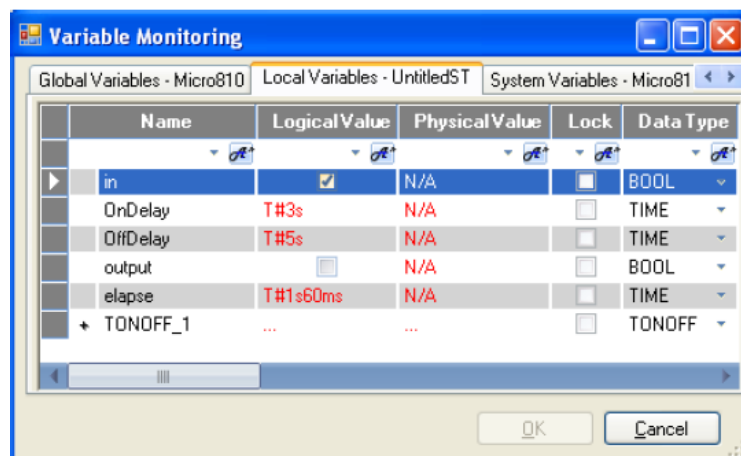
Este temporizador es el conjunto del temporizador con retraso a la conexión y desconexión.

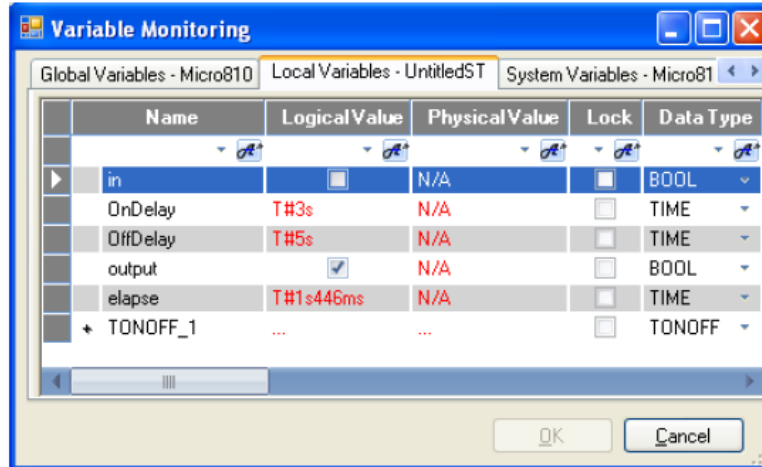


| Elemento         | Descripción   |
|------------------|---|
| <b>IN (Bool)</b> | <p>Si existe un flanco ascendente, comienza el temporizador de retraso. Si se supera el tiempo de retraso programado y existe un flanco descendente, el temporizador de retardo de desconexión se inicia y restablece el tiempo transcurrido.</p> <p>Si se supera el tiempo de retraso programado y existe un flanco descendente, el temporizador de retardo de desconexión se inicia. Si no se supera el tiempo de retraso programado y existe un flanco ascendente, el temporizador de retraso se inicia.</p> |

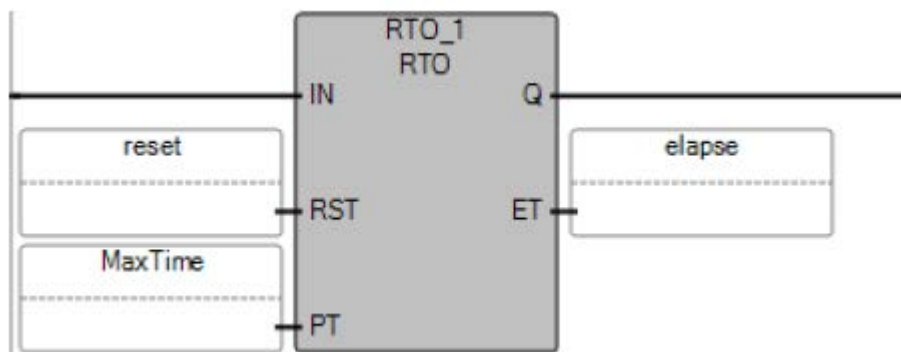
|                    |  |
|--------------------|--|
| <b>PT (Time)</b>   | Ajuste del temporizador de retraso a la conexión.  |
| <b>PTOF (Time)</b> | Ajuste del retraso de desconexión.   |
| <b>Q (Bool)</b>    | Si es Verdadero, el tiempo de retraso programado se supera y el temporizador de retardo de desconexión no se supera. |
| <b>ET (Time)</b>   | Tiempo actual transcurrido. Tipo TIME.   |

Resultado





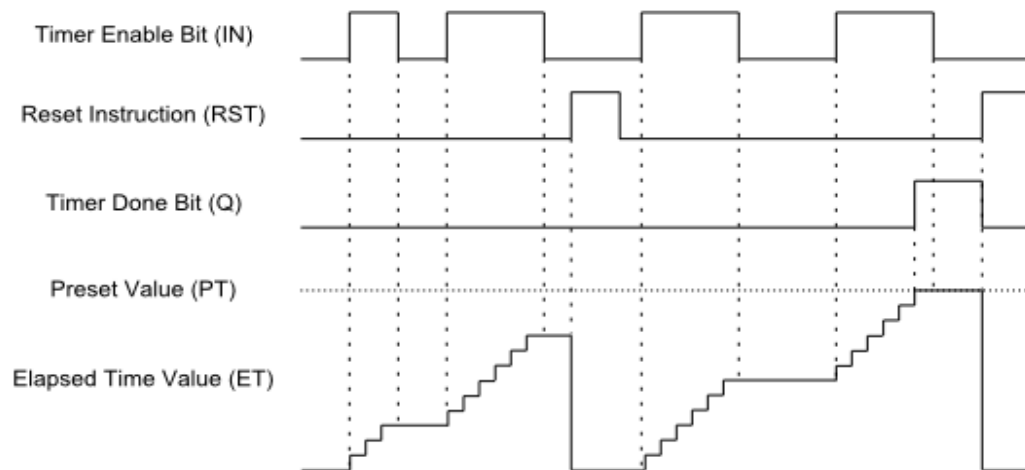
d) *Temporización retenida*



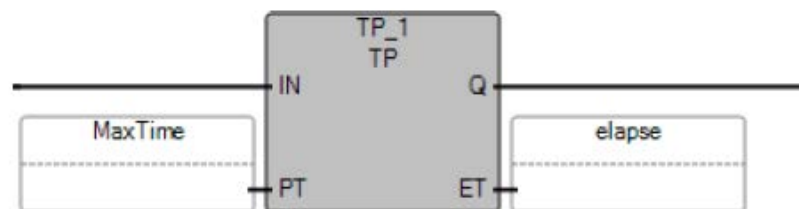
| Elemento          | Descripción  |
|-------------------|--|
| <b>IN (Bool)</b>  | En caso de un flanco ascendente, se empieza a aumentar el temporizador interno. En caso de flanco descendente, detiene el temporizador interno y no lo restablece. |
| <b>RST (Time)</b> | Si el flanco es ascendente, restablece el temporizador interno.  |
| <b>PT (Time)</b>  | Tiempo programado máximo.  |
| <b>Q (Bool)</b>   | Si es Verdadero: el tiempo programado transcurre.  |

|                  |                             |
|------------------|-----------------------------|
| <b>ET (Time)</b> | Tiempo actual transcurrido. |
|------------------|-----------------------------|

A continuación, se muestra el diagrama de tiempo de la funcionalidad del temporizador retenida:

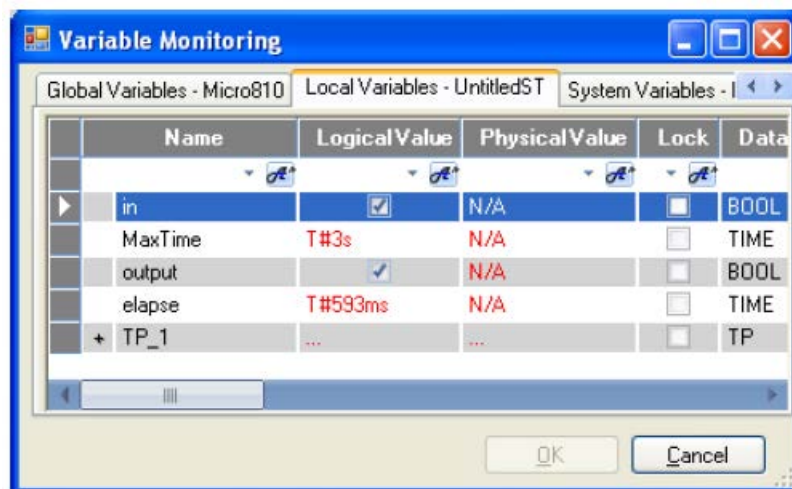
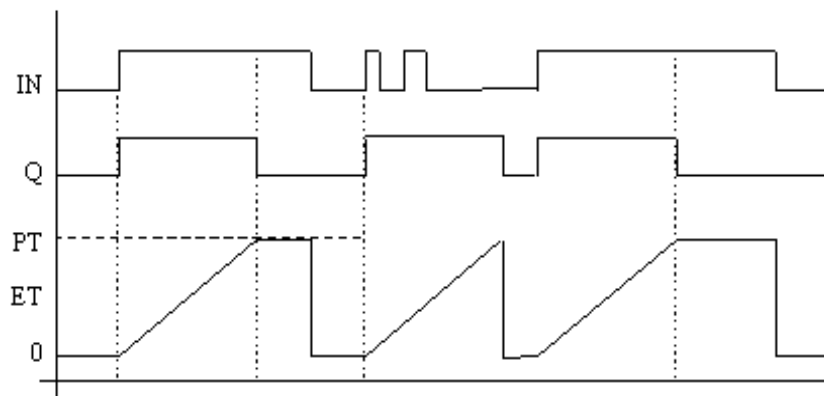


*e) Temporización del pulso.*



| Elemento         | Descripción   |
|------------------|---|
| <b>IN (Bool)</b> | En el caso que exista un flanco ascendente, se empieza a aumentar el temporizador interno (si no está aumentando ya), caso contrario que exista un flanco descendente y solo si transcurre el temporizador, restablece el temporizador interno.<br>Cualquier cambio de IN durante el recuento no tiene efecto alguno. |
| <b>PT (Time)</b> | Tiempo programado máximo.   |
| <b>Q (Bool)</b>  | Si es Verdadero: el temporizador está habilitado.   |
| <b>ET (Time)</b> | Tiempo actual transcurrido.   |

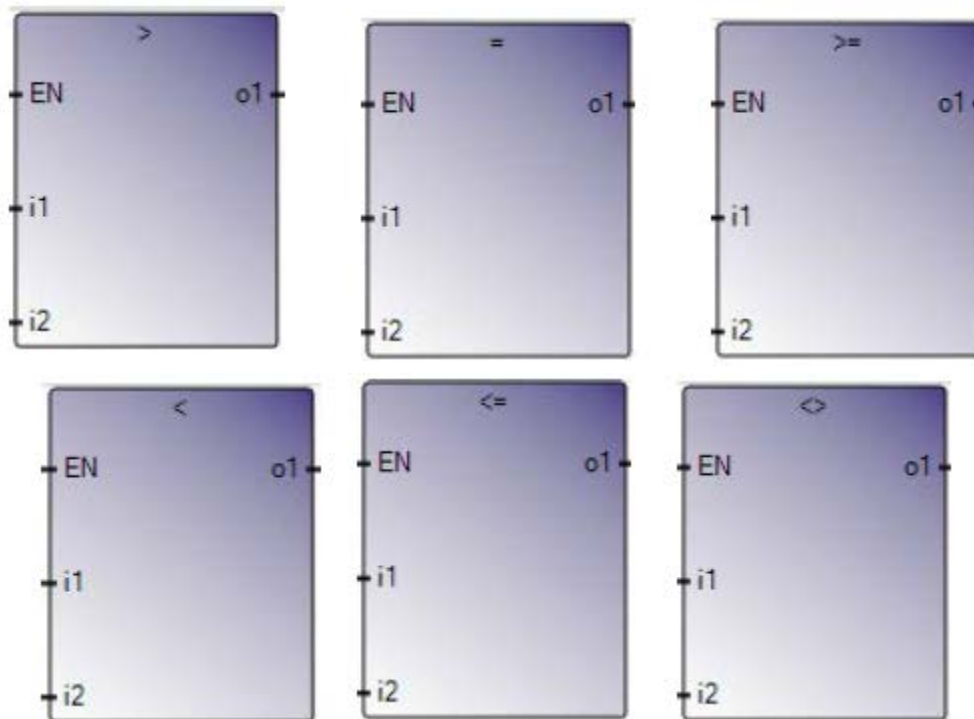
A continuación, se muestra el diagrama de tiempo de la funcionalidad del temporizador de pulso:



### Comparadores

Un comparador recepta dos señales en sus entradas para compararlas en una relación de igualdad o desigualdad a la función de comparación establecida para generar en su señal de salida de alto o bajo.

Las instrucciones de comparadores de Connected Components Workbench son las siguientes:



| Elemento                       | Descripción                                  |
|--------------------------------|--|
| <b>Menor que</b>               | Toma el valor de verdadero si $i1 < i2$ .    |
| <b>Menor que o igual que</b>   | Toma el valor de verdadero si $i1 \leq i2$ . |
| <b>No igual a</b>              | Toma el valor de verdadero si $i1 \neq i2$ . |
| <b>Igual a</b>                 | Toma el valor de verdadero si $i1 = i2$ .    |
| <b>Mayor que</b>               | Toma el valor de verdadero si $i1 > i2$ .    |
| <b>Mayor que o igual a que</b> | Toma el valor de verdadero si $i1 \geq i2$ . |



## Contadores

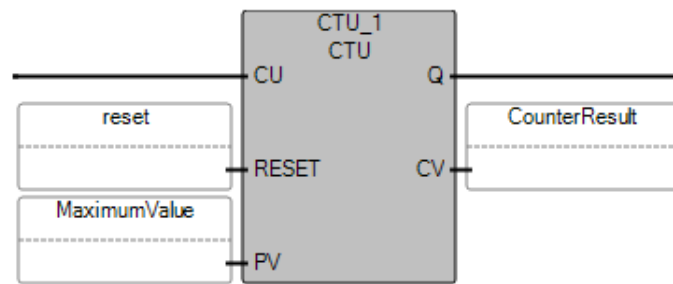
Un contador tiene la función de cómputo que permite efectuar la cuenta de impulsos. La cuenta se puede programar en forma progresiva (ascendente) o regresiva (descendente).

La operación de conteo consiste en incrementar 1 al contenido del contador, mientras que la operación de descuento consiste en decrementar 1 al contenido del contador, ambos al presentarse un pulso.

La función del contador permite activar salidas o memorias internas, en el momento que su registro de conteo coincide con el valor presente previamente definido. Por otro lado, si el registro es diferente al valor de presente la salida asociada al contador no se activará.

Las instrucciones de contadores de Connected Components Workbench son las siguientes:

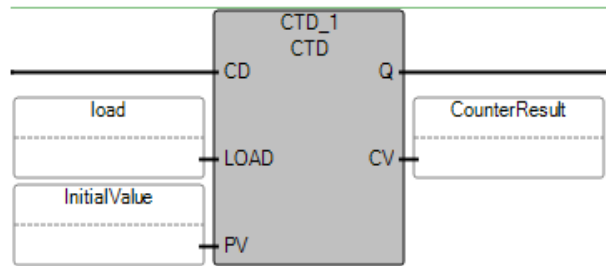
### a) Contador ascendente



| Elemento            | Descripción   |
|---------------------|---|
| <b>CU (Bool)</b>    | Contador ascendente (si CU recepta un flanco ascendente). |
| <b>RESET (Bool)</b> | Restablecimiento, encera el contador, comando dominante.  |
| <b>PV (DINT)</b>    | Valor máximo programado.                                  |
| <b>Q (Bool)</b>     | Toma el valor de verdadero si $CV \geq PV$ .              |
| <b>CV (DINT)</b>    | Valor actual del contador.                                |



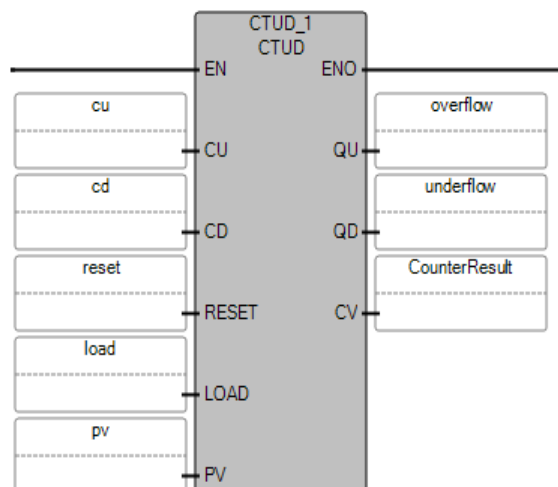
b) *Contador descendente*



| Elemento           | Descripción   |
|--------------------|---|
| <b>CD (Bool)</b>   | Entrada de recuento (cuenta atrás si CD es un flanco de bajada).  |
| <b>LOAD (Bool)</b> | Comando de carga, dominante. (CV= PV si LOAD es verdadero).       |
| <b>PV (DINT)</b>   | Valor inicial programado.   |
| <b>Q (Bool)</b>    | Toma el valor de verdadero si $CV \leq 0$ . Falso cuando $CV > 0$ |
| <b>CV (DINT)</b>   | Valor actual del contador.  |

c) *Contador ascendente/descendente*

Este contador es el conjunto del contador ascendente y contador descendente.



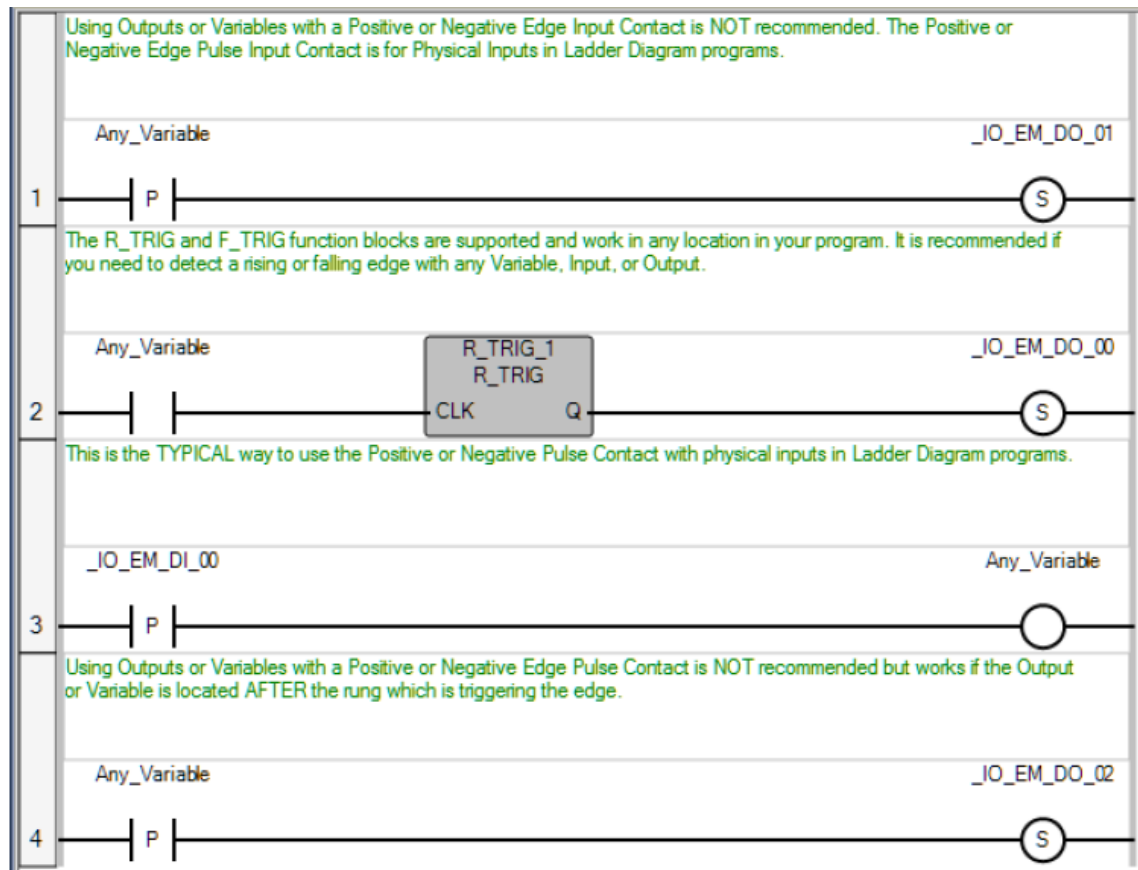
| Elemento            | Descripción  |
|---------------------|--|
| <b>CU (Bool)</b>    | Contador ascendente (si CU recepta un flanco ascendente).  |
| <b>CD (Bool)</b>    | Contador ascendente (si CD recepta un flanco descendente). |
| <b>LOAD (Bool)</b>  | Comando de carga. (CV= PV si LOAD es verdadero).           |
| <b>RESET (Bool)</b> | Restablecimiento, encera el contador, comando dominante.   |
| <b>PV (DINT)</b>    | Valor inicial programado.                                  |
| <b>QU (Bool)</b>    | Toma el valor de verdadero si $CV \geq PV$ .               |
| <b>QD (Bool)</b>    | Toma el valor de verdadero si $CV \leq 0$ .                |
| <b>CV (DINT)</b>    | Valor actual del contador.                                 |



**ANEXOS**

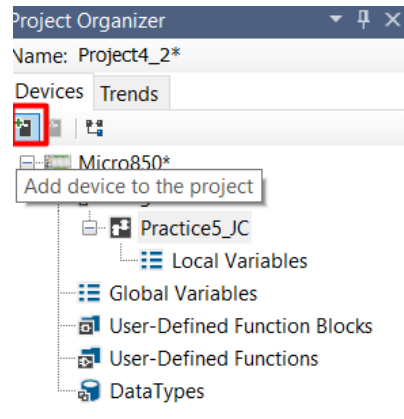
**R\_TRIG function block**

The following example program shows the recommended usage of an R\_TRIG function block used to detect an edge while connected to the controller.

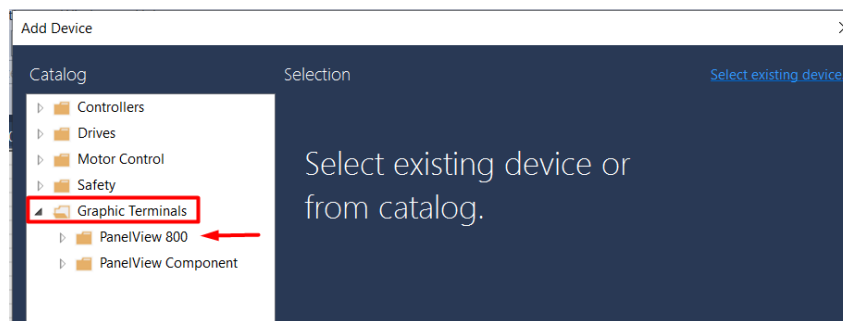


## Selección y programación del PanelView

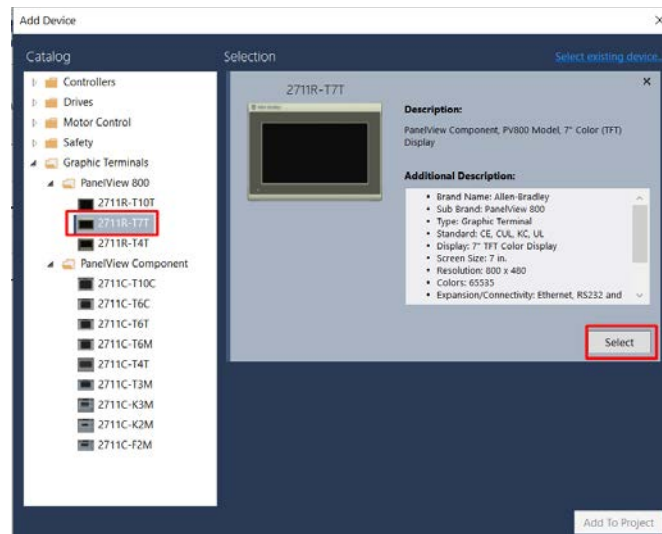
1. En la 25evice25 **Project Organizer** seleccionar **Add device to the project**.



2. En la ventana **Add 25evice**, seleccionar la carpeta **Graphic Terminals**.



3. En **PanelView 800** desplegarlo, dar clic en **2711R-T7T** y seleccionarlo para **Add to Project**.



4. En “Etiquetas” agregamos variables las cuales vincularemos con las variables globales del controlador, recordemos que sólo podemos vincular variables globales o salidas físicas del controlador.
5. En “Pantallas” podemos agregar o seleccionar pantallas de la aplicación en la que nos encontramos trabajando.



6. En el editor de etiquetas para la vinculación de variables describiremos el funcionamiento de los parámetros a llenar:
  - a. Nombre de etiqueta: Se asigna el nombre de la nueva variable creada.
  - b. Tipo de datos: Se selecciona el tipo de dato de la variable creada.

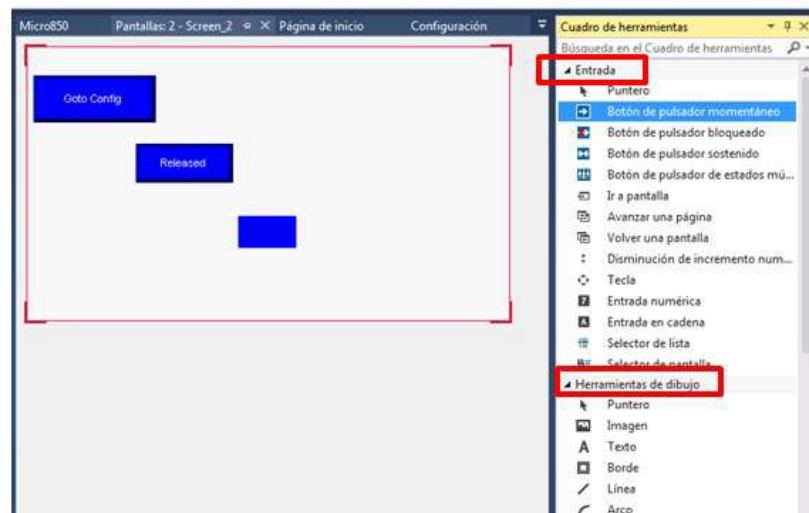




- c. Dirección: Vinculación de la variable del controlador (variable global y salida físicas).
- d. Controlador: Selección del controlador.
- e. Acceso: Selección de variable de lectura (salidas) o escritura (entradas).

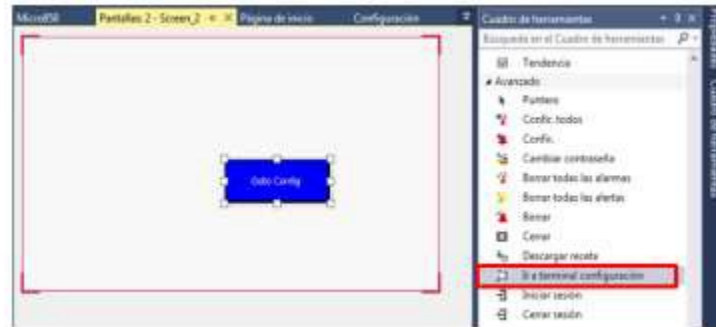


7. En el “Cuadro de herramientas” se puede acceder a una variedad de herramientas como por ejemplo para escritura (entradas), para indicadores (salidas), etc.

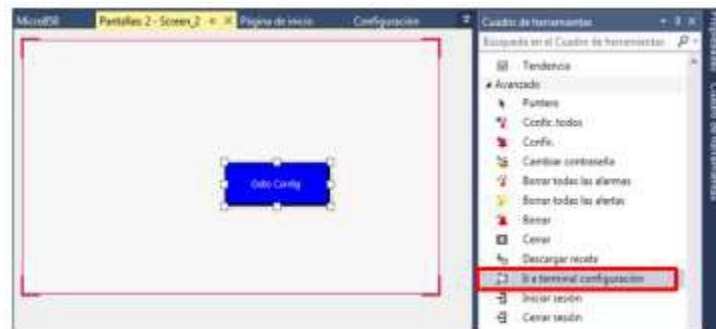


8. Es importante agregar en cada pantalla el botón “Ir a terminal configuración” del “Cuadro de herramientas”, nos permite dirigirnos a las configuraciones del Panel View, en caso de no colocar queda en un bucle.

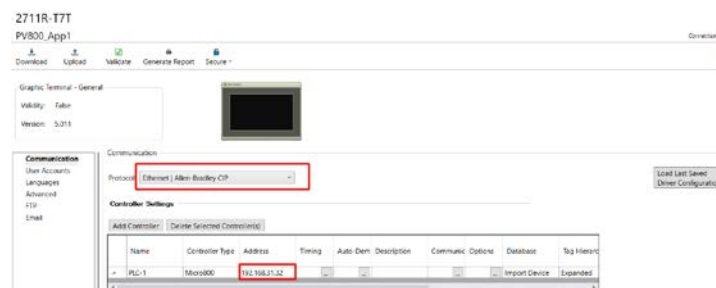




9. Para vincular alguna herramienta con una variable del controlador, damos clic derecho, seleccionamos "Propiedades", posteriormente aparecerá la ventana "Propiedades", en "Conexiones", seleccionamos la acción que queremos darle a la herramienta (indicador o visibilidad). Recordemos que para vincular anteriormente tuvimos que haber creado las etiquetas vinculadas con el controlador.



10. Para asignarle la dirección IP al PanelView, clic en el terminal gráfico en el "Organizador de proyectos", en "Protocolo" seleccionar "Ethernet| Allen-Bradley CIP", en dirección asignar la dirección IP del controlador.



11. Finalmente buscar la dirección IP del terminal gráfico en la ventana que se muestra, la dirección IP del gráfico debe coincidir con la asignada en el dispositivo.  
En el PanelView se debe seleccionar la aplicación cargada, luego presionar "ejecutar".

