

**PREPRACTICA**

**TEMA:** Introducción a diagrama secuencial de funciones (SFC)

**1. Objetivos**

**1.1. Objetivo general**

Elaborar la programación GRAFCET o SFC utilizando etapas, transiciones y acciones para la realización de aplicación en un entorno virtual en 3D.

**1.2. Objetivos específicos**

1. Analizar el funcionamiento de la programación GRAFCET mediante las herramientas del software Studio 5000.
2. Identificar las herramientas que posee el software FactoryTalk View Machine Edition para el manejo de pantallas HMI de la marca Rockwell Automation.
3. Desarrollar la vinculación de las variables globales del controlador con FactoryTalk View Machine Edition para la comunicación de los dispositivos a través de Ethernet.

**¿Qué actividades realizarán?**

1. Realizar el diagrama grafcet el proceso virtual adjuntado en Factory IO.
2. Realizar la programación en el lenguaje de programación de su preferencia en CCW.
3. Elaborar y simular una interfaz HMI en FactoryTalk View Studio de la aplicación del literal anterior.

**¿Como lo realizarán?**

Programas por utilizar:

- RsLinx Classic
- Factory IO
- CCW.



## ¿Cuáles son los entregables de la prepráctica?

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

Seguir el siguiente orden para la realización de la prepráctica.

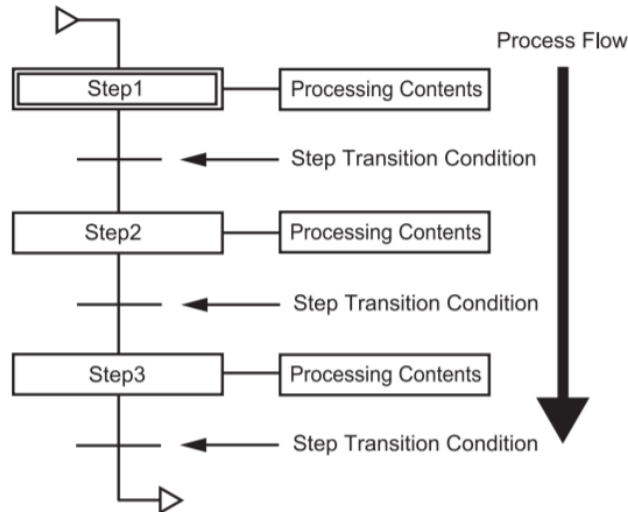
1. Tableros Midrange y Large project
2. introducción a Studio 5000
3. Diagrama secuencial de funciones en Studio 5000
4. Instrucción CTP
5. Configurar un PanelView en FactoryTalk View
6. Establecer comunicación entre CCW y FTV Studio
7. Restaurar runtime en FactoryTalk View ME y descargar aplicación hacia un PanelView Plus 1000  
Cambiar proyecto de un backplane virtual a un CompactLogix L33ERM
8. Cambiar proyecto de un backplane virtual a un ControlLogix L73
9. Configurar un PanelView en FactoryTalk View
10. Establecer comunicación entre CCW y FTV Studio
11. Restaurar runtime en FactoryTalk View ME y descargar aplicación hacia un PanelView Plus 1000



**Anexo - Marco teórico**

**Grafcet**

GRAFCET fue un modelo propuesto por investigadores y administradores industriales, donde el modelo tiene graficos basico. Ademas, realizado por AFCET (Association française de cybernétique économique et technique) asociacion cientifica que dio soporte al trabajo realizado.



GRAFCET, del frances Graphe Fonctionnel de Commande Etape-Transition que quiere decir diagrama de control con etapas y transiciones. En 1993, se propuso el nombre Sequential Function Chart (SFC) definido por el estándar IEC 61131-3, donde es un lenguaje de programacion grafica usado para controladores logicos programables (PLC). Adicionalmente, la ventaja es que puede ser usado para programar procesos que puedan ser divididos en estados.

En este diagrama cada nivel representa un comportamiento o acción de un sistema lógico, lo que va contemplándose en etapas, acciones que se esquematizan por sucesos medios. Es recomendable utilizar estos diagramas para describir modelos secuenciales. El Grafcet se estructura de:

- Etapas que están adjuntas a acciones.
- Transiciones en las cuales se receptan las acciones.
- Uniones que se encargan de unir las etapas con las transiciones.

Un **GRAFCET** está compuesto de:



**ETAPA:** Representa un estado específico del proceso automatizado. Es un punto en el cual el sistema realiza una acción o espera una condición antes de proceder. Las etapas iniciales se indican con un doble cuadrado para diferenciarlas de las demás, estas etapas iniciales son las que se activan al inicializar el GRAFCET. Una vez se ha inicializado el GRAFCET, las etapas iniciales actúan como etapas normales. Puede haber tantas etapas iniciales como se desee, pero como mínimo una. Pueden estar situadas en cualquier lugar dentro del GRAFCET.

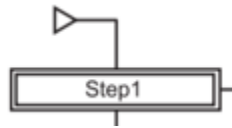


Ilustración 1 Etapa Inicial

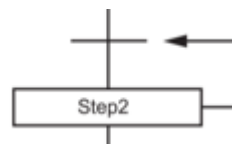
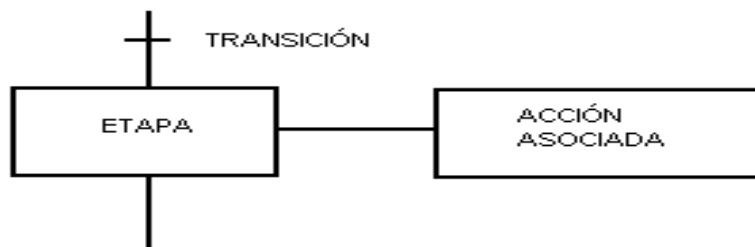


Ilustración 2 Etapa normal

**ACCIÓN ASOCIADA:** Es la tarea que se ejecuta en una etapa determinada. Esta puede incluir activar un contactor, desactivar una bobina, iniciar un motor, entre otras acciones. Es lo que el sistema hace mientras se encuentra en esa etapa.

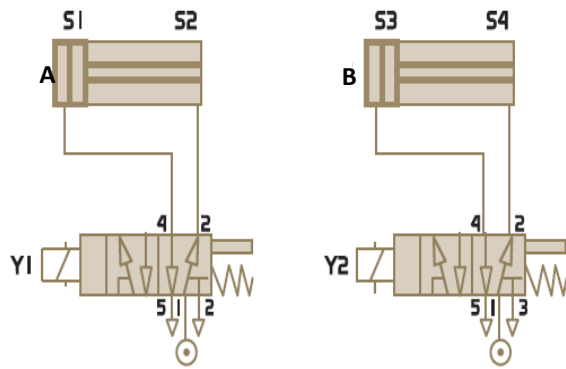
**TRANSICIÓN:** Es la condición o conjunto de condiciones que deben cumplirse para que el sistema pase de una etapa a otra. Estas condiciones pueden ser la activación de un pulsador, la detección de un sensor, la expiración de un temporizador, entre otras. La transición asegura que el proceso solo avance cuando se cumplen ciertos criterios específicos.



### Ejemplo

Realizar la siguiente programación en Grafcet

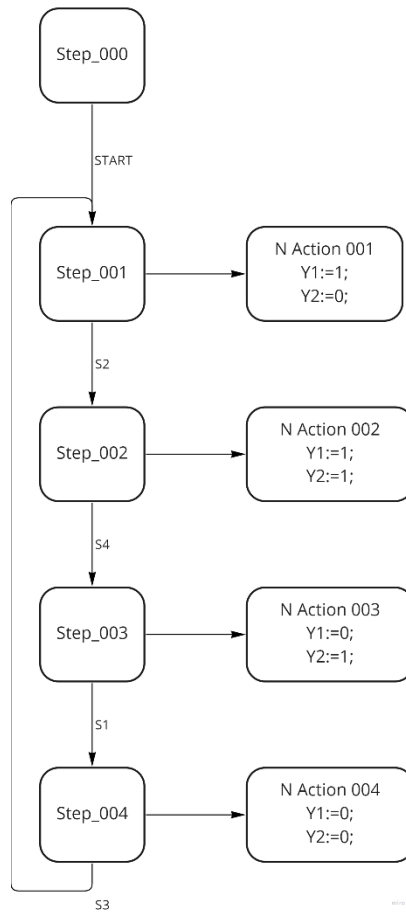




Component	Steps						
Cylinder A							
Cylinder B							

Símbolo	Descripción
S1	Detector de fin de carrera. Determina que el cilindro Z1 se halla en su posición inicial.
S2	Detector de fin de carrera. Determina que el cilindro Z1 se halla en su posición final.
S3	Detector de fin de carrera. Determina que el cilindro Z2 se halla en su posición inicial.
S4	Detector de fin de carrera. Determina que el cilindro Z2 se halla en su posición final.
Y1	Electroválvula. Activa al cilindro A.
Y2	Electroválvula. Activa al cilindro B.





### Explicación

1. **Step\_000** Etapa inicial. No está asociada a ninguna acción específica, pero está vinculada a una transición que lleva a la siguiente etapa. Esta transición ocurre cuando se presiona el pulsador START.
2. **Step\_001** Esta etapa si tiene una acción asociada, que es establecer Y1=1 y Y2=0. La transición a la siguiente etapa ocurre cuando se activa el fin de carrera S2.
3. **Step\_002** Esta etapa tiene una acción asociada, que es establecer Y1=1 y Y2=1. La transición a la siguiente etapa ocurre cuando se activa el fin de carrera S4.
4. **Step\_003** Esta etapa tiene una acción asociada, que es establecer Y1=0 y Y2=1. La transición a la siguiente etapa ocurre cuando se activa el fin de carrera S1.
5. **Step\_004** Esta etapa tiene una acción asociada, que es establecer Y1=0 y Y2=0. La transición a la siguiente etapa ocurre cuando se activa el fin de carrera S3.
6. Luego se retorna al **Step\_001** haciendo el bucle, repitiendo los pasos del 2 al 5.



**Identificación de sensores y actuadores.**

**Sensor de proximidad.**

Esta clase de sensores son dispositivos que detectan la presencia de objetos al estar cerca del sensor. En el mercado existen algunos tipos de sensores de proximidad acorde al principio físico que utilicen, entre ellos están los interruptores de posición, sensores inductivos, fotoeléctricos y capacitivos, como los que se muestran en la figura 4. Son utilizados para conocer el estado del proceso, para conocer la posición del producto a lo largo del proceso se utilizan esta clase de sensores que son capaces de detectar la distancia, ausencia o presencia de un objeto. Los sensores de proximidad son capaces de identificar la presencia de objetos sin necesidad de entrar en contacto directo con ellos, siendo frecuentemente utilizados en cintas transportadoras y procesos continuos; también se usan en vehículos autónomos para frenar automáticamente al detectar la proximidad de un objeto.



**Tipos**

Tipo de sensor	Ventajas	Desventajas
Inductivo	Detección sin contacto Considerablemente baratos Especializado en la detección de metales	Bajo rango de detección Limitado a detectar objetos metálicos
Capacitivo	Detección sin contacto Capaz de detectar una amplia gama de materiales Sensibilidad elevada	Relativo bajo rango de detección Precios más elevados
Fotoeléctrico	Detección sin contacto Capacidad de detectar objetos blandos	Disminuye su rendimiento a largas distancias Se puede ver afectado por el nivel de reflexión del objeto

**Sensores de color y contraste.**

Los sensores de color identifican los colores de los objetos mediante la emisión de luz en todo el campo visual sobre los objetos que deben ser analizados. Estos sensores calculan las proporciones de color a partir de la luz reflejada y las comparan con los valores cromáticos de referencia. Un sensor de color es un tipo de "sensor fotoeléctrico" que emite luz desde un transmisor mediante un receptor, detecta la luz reflejada por el objeto de detección. Este sensor puede evaluar la intensidad de luz recibida de los colores rojo, azul y verde, respectivamente, lo que posibilita la determinación del color del objeto en cuestión



**Tipos**

Tipo de sensor	Características	
Sensor cromático	Esta clase de sensores son capaces de sensor varios colores a la vez de forma fiable y rápida.	
Sensor de contraste	Los sensores de contraste detectan un color que es configurado por el usuario sobre una superficie de color diferente. Comúnmente utilizado en la detección de marcas	

**Actuador lineal**

Los actuadores lineales son dispositivos actuadores que como su nombre indica realizan un movimiento lineal transformando la energía que puede ser eléctrica, neumática o hidráulica, y la convierte en energía cinética. los actuadores lineales eléctricos transforman el movimiento rotatorio de los motores y los transforman en un movimiento lineal que se puede aprovechar para distintos propósitos dependiendo de la aplicación. Existen diferente tipo de cilindros dependiendo del número de vástagos, pistones o forma de accionamiento.

**Tipos**

Tipo de sensor	Ventajas	Desventajas
Neumatico	De accionamiento rápido Robustos De funcionamiento sencillo	Requiere la instalación de otros equipos Son ruidosos
Hidráulico	Capacidad de carga elevada Estabilidad en cargas estáticas	Requiere la instalación de otros equipos Precio elevado
Eléctrico	Instalación sencilla De fácil uso	Tienen limitaciones en su potencia





### **Anexo – Ejercicio propuesto**

Se requiere programar un sistema de detección y clasificación de colores para cajas utilizando el lenguaje de programación de su preferencia. El sistema deberá cumplir con las siguientes especificaciones:

1. **Detección de Colores:** El sistema debe ser capaz de detectar cajas de color rojo y azul. (Esta se hará mediante un sensor que es capaz de mandar una señal cuando el objeto procesado es azul)
2. **Acción del Pistón:** Si la caja detectada es de color azul, un pistón deberá empujarla. Si la caja es de color rojo, esta seguirá su curso sin intervención.
3. **Monitoreo:** El sistema debe monitorear continuamente el proceso y registrar la siguiente información:
  - El número total de cajas procesadas.
  - La cantidad de cajas verdes detectadas.
  - La cantidad de cajas azules detectadas.

A continuación se detalla el proceso de la planta

La planta contará con un botón de inicio (Start) y un botón de parada (Stop) que permitirán iniciar y detener el proceso. Al presionar el botón de inicio, se activará un relé que encenderá el motor de la banda transportadora.

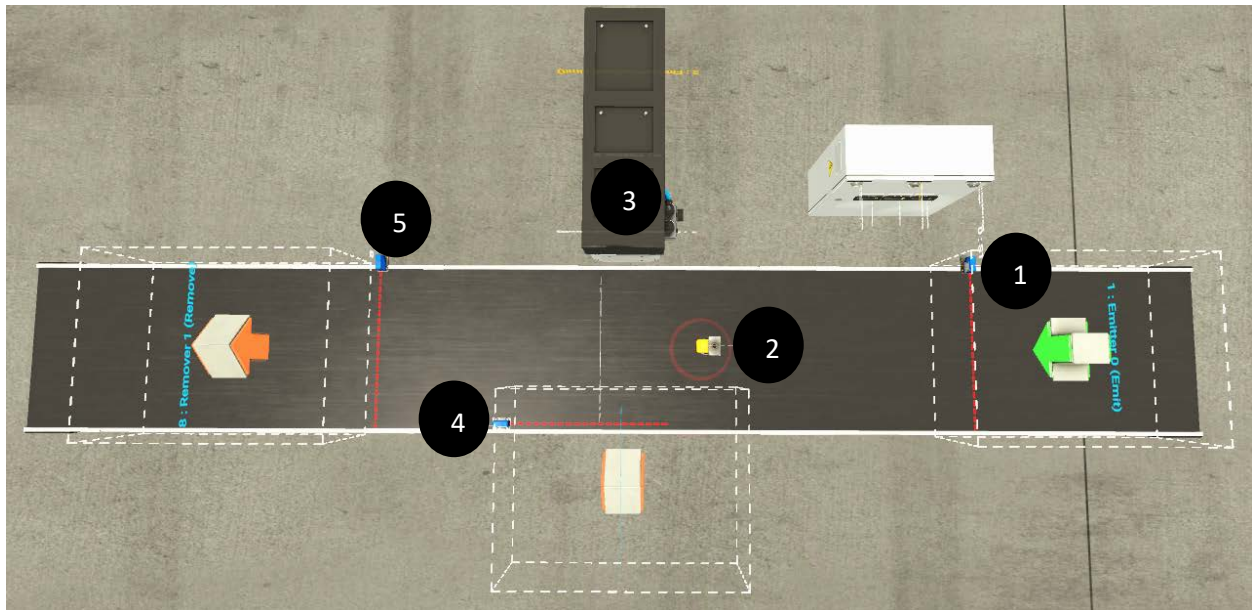
Al inicio de la banda, habrá un sensor encargado de detectar los objetos que ingresan a la banda transportadora. Seguido de este, habrá un sensor cuya función será detectar el color del objeto.

Si el objeto es azul, deberá ser empujado por un pistón y se registrará la cantidad de objetos azules mediante un sensor ubicado a las afueras de la banda para confirmar que el objeto fue efectivamente empujado por el pistón. Dado que el pistón estará a una determinada distancia del sensor de color, se deberá calcular el tiempo necesario para que el objeto se posicione frente al pistón. Para que el pistón pueda empujar el objeto, se deberá detener la banda transportadora. Una vez que el pistón haya cumplido su función y vuelva a su posición inicial gracias a un fin de carrera ubicado en el vástago del mismo, indicará que el proceso debe continuar. Este fin de carrera enviará una señal cuando el pistón esté retraído, y cuando el pistón se active, dejará de enviar la señal.

Por otro lado, si el objeto es verde, deberá seguir su curso hasta el final de la banda transportadora, donde habrá un sensor encargado de detectar cuántos objetos pasaron por allí.



### Imagen referencial del proceso



1. Sensor a la entrada (S1)
2. Sensor de contraste (S4)
3. Piston + Fin de carrera
4. Sensor para el objeto azul (S2)
5. Sensor para el objeto verde(S3)

### Observaciones

El pistón en factory IO tiene una entrada y dos salidas

- Entrada: Activar el pistón (Este se demora aproximadamente 1.1 segundos para llegar a su punto máximo)
- Salida1: Se activa cuando el pistón llega a su tope máximo
- Salida2: Se activa cuando el pistón está en su posición inicial

Para este desafío solo usaremos la **Entrada (Piston)** para activar el pistón y únicamente la **Salida2 (FinCarrera(Back Limit))** que será nuestro fin de carrera que nos ayudará en la programación.

El botón **Reset** servirá para resetear los contadores que sensan la cantidad de objetos totales, verdes y azules que han pasado por el proceso



**Direcciones que deberán usar para programar cada objeto del proceso**

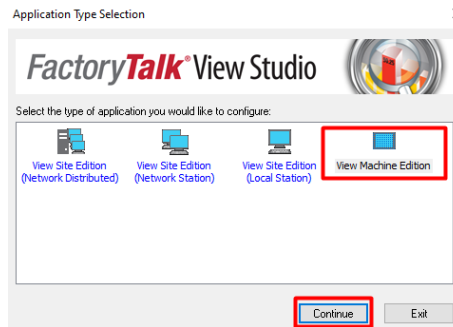
<b>Objeto</b>	<b>Dirección</b>
Sensor (S1)	DI_08
Sensor (S2)	DI_10
Sensor (S3)	DI_09
Sensor (S4)	DI_06
Start	DI_00
Stop	DI_01
Reset	DI_02
Fin de carrera	DI_07
Rele del piston	DO_01
Rele motor banda transportadora	DI_00



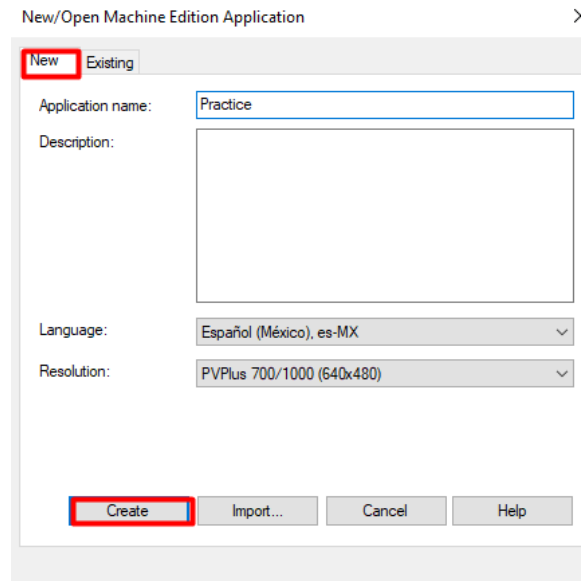
## FactoryTalk View

### Crear y configurar la comunicación de una aplicación en FactoryTalk View Machine Edition

1. Abrir el software **FactoryTalk View Studio**, luego nos aparecerá la ventana **Application Type Selection** donde se debe seleccionar **View Machine Edition**.

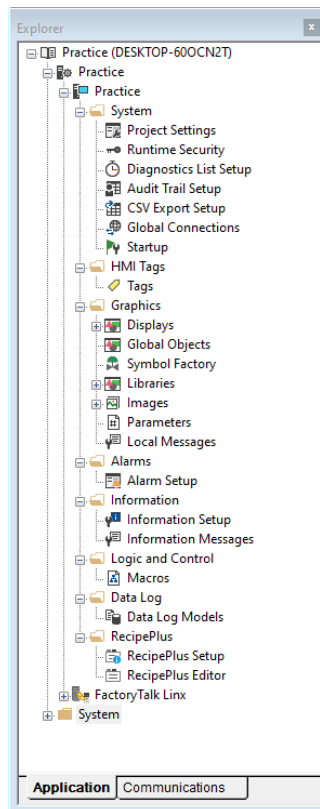


2. Aparecerá automáticamente la ventana **New/Open Machine Edition Application**, en la pestaña **New** buscar la sección **Application name** para escribir el nombre de la aplicación que realizaremos, luego seleccionar el lenguaje que queramos que este el programa y dar clic en **Create**.

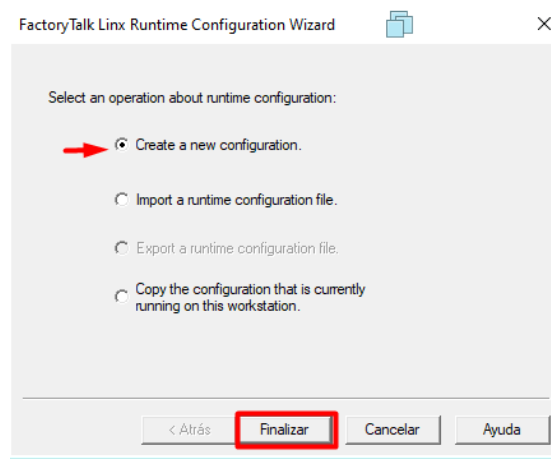


3. En la ventana **Explorer**, Desplegar la opción **FactoryTalk Linx** para establecer la comunicación y dar doble clic a la opción **Communications Setup**.

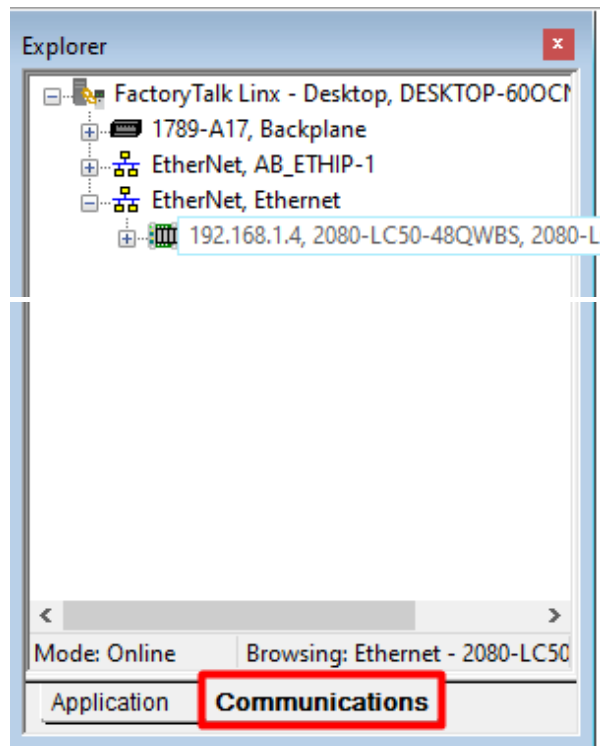




4. En la ventana de **FactoryTalk Linx Configuration Wizard**, seleccionar **Create a new configuration** y dar clic en **Finalizar**.

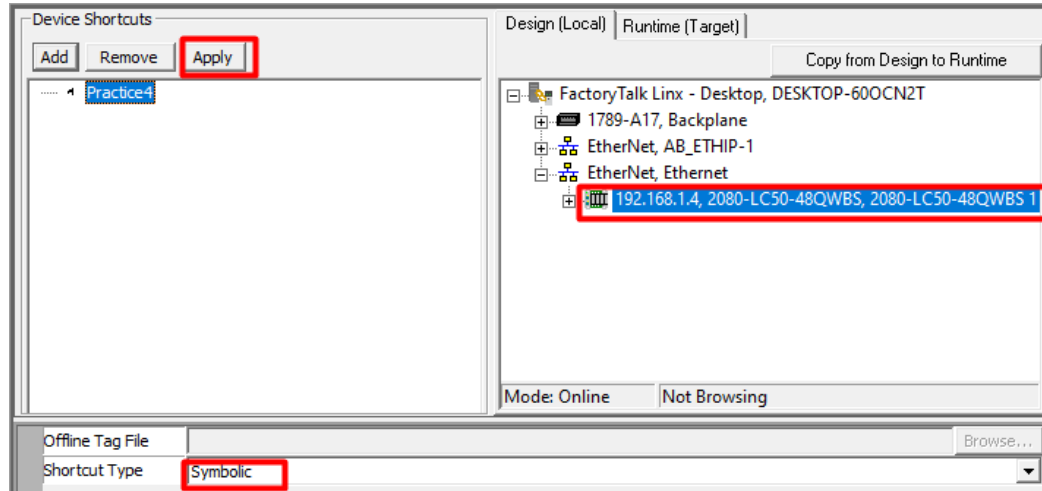


5. En la ventana **Explorer**, en la parte inferior se encuentra la pestaña **Communications** donde se muestra los dispositivos conectados a la red.

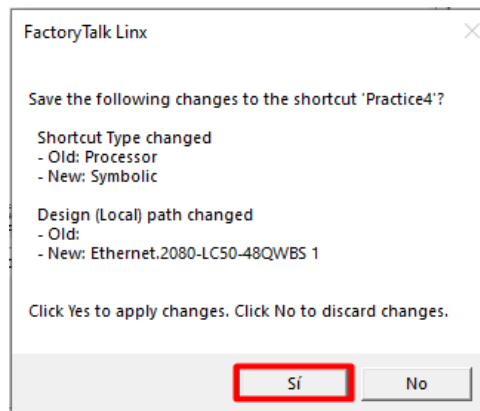


7. En la ventana **Device Shortcuts**, clic en **Add** para crear un **shortcut** que nos representa la ruta específica del dispositivo a comunicarnos, y posteriormente asignarle un nombre. Luego teniendo seleccionado el shortcut, en la pestaña **Design (Local)** escoger el controlador con que se va a trabajar dando clic en **Apply**, esto permite que el dispositivo seleccionado, es decir el controlador se asocie con la ruta anteriormente especificada. Además, en la sección de **Shortcut Type** se selecciona **Symbolic**.



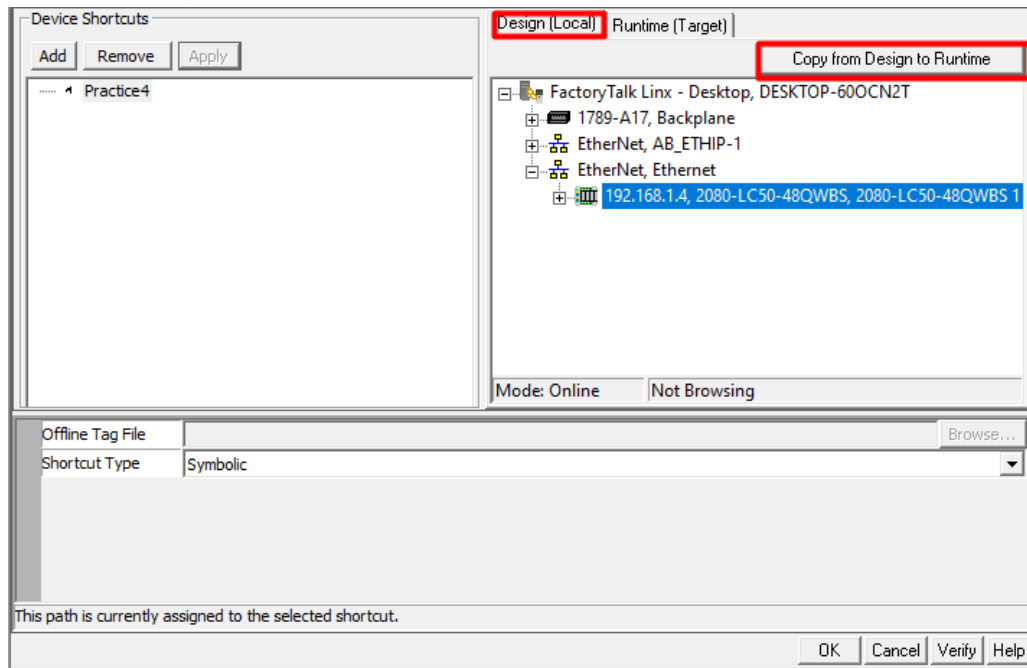


8. Aparecerá la siguiente ventana de información, dar clic en **Si**.

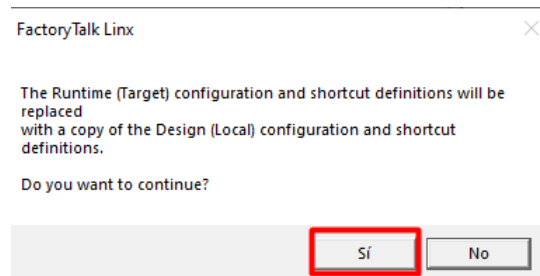


9. En la parte derecha de la pestaña **Design (Local)**, dar clic en **Copy from Design to Runtime** para copiar la topología en la pestaña **Runtime (Target)**.





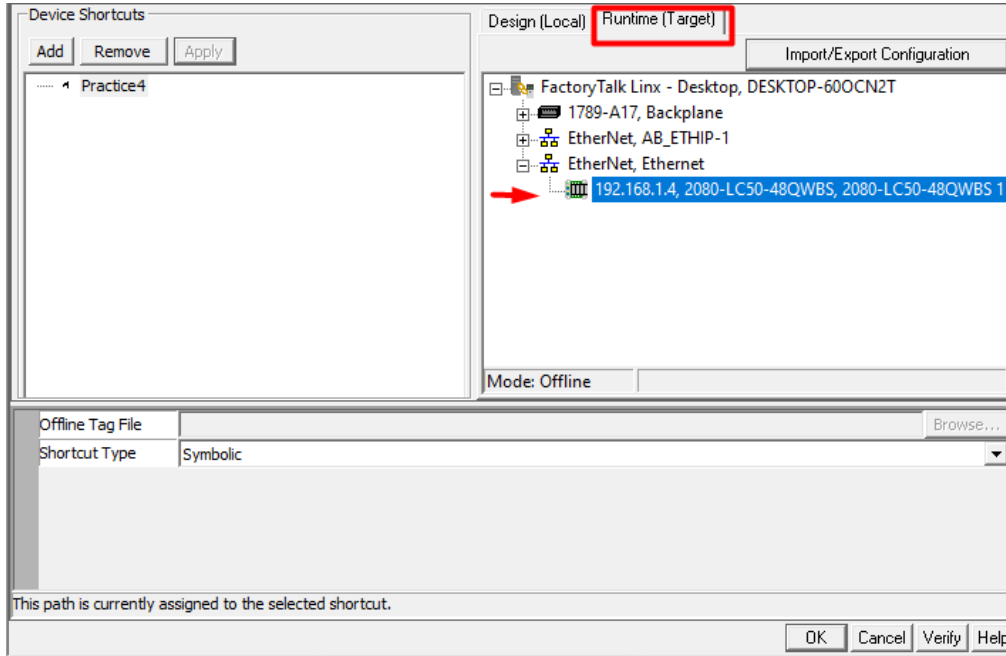
10. En la siguiente ventana, dar clic en **Si**.



11. En la pestaña **Runtime (Target)** se debe verificar que la copia de la ruta del controlador fue realizada con éxito.







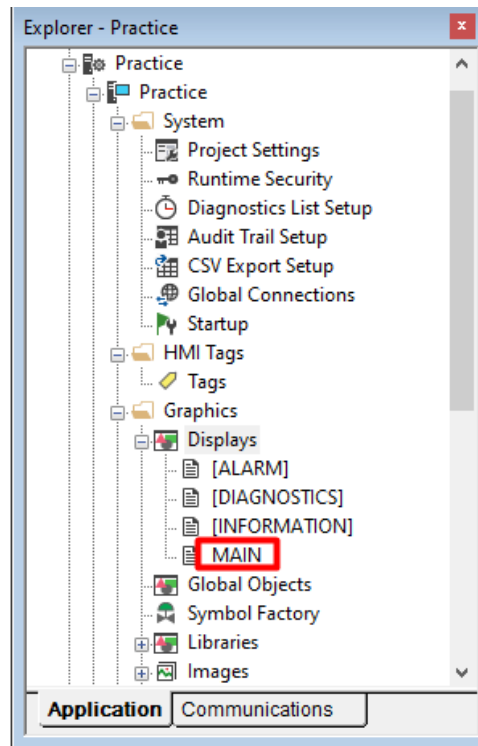
12. En la parte inferior de la pestaña **Design (Local)**, dar clic en **Verify** donde nos aparecerá la siguiente ventana y dar Close. Si todos los pasos fueron realizados correctamente, los enlaces de **Design** y **Rutime** fueron asignados y mostradas en el cuadro de verificación.



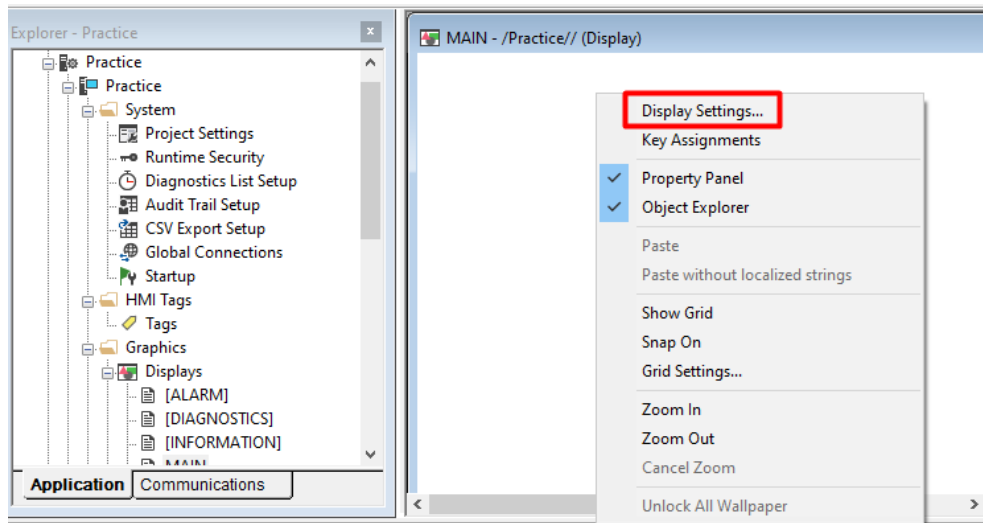
### Crear un display en FactoryTalk Machine Edition

1. En la ventana **Explorer** y en la pestaña **application**, desplegar la lista de opciones de **Displays** y dar doble clic en **MAIN** donde surgirá una ventana en blanco.



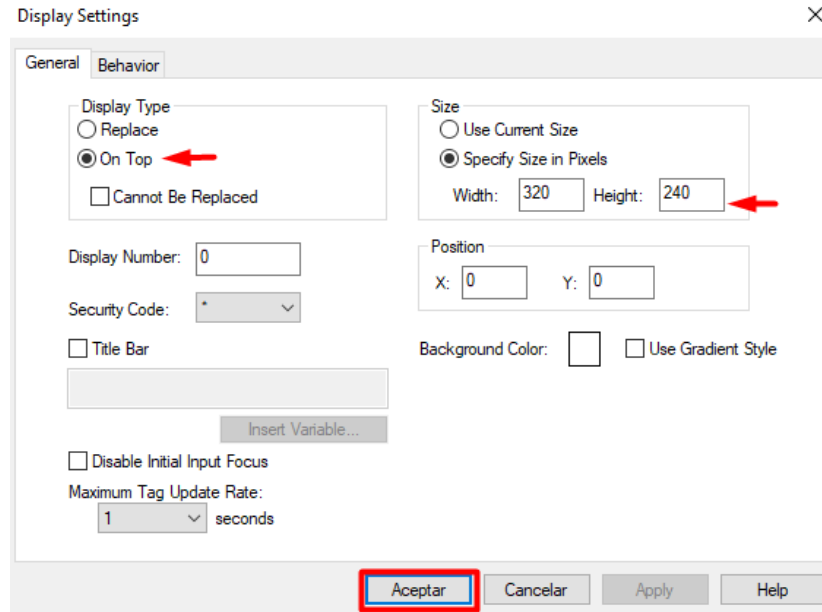


2. Dar clic derecho en un espacio en blanco de la pantalla MAIN y selecciona **Display Settings** para abrir las propiedades de la ventana de Display, en la cual se puede configurar ciertos parámetros de acuerdo con las necesidades de la aplicación.

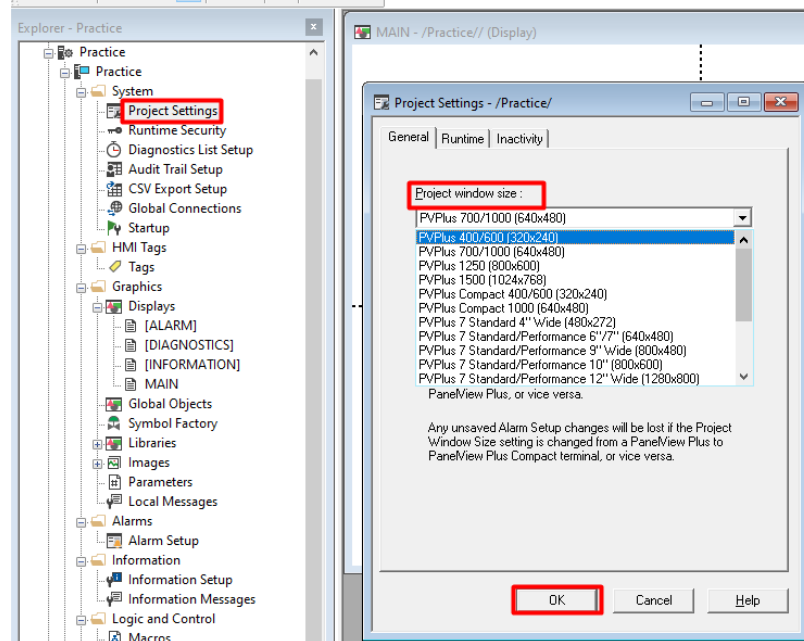


3. La ventana **Display Settings** aparecerá. En la pestaña **General**, en la opción **Display Type** seleccionar **On Top**, y en la opción **Size** podemos configurar el tamaño de la ventana de acuerdo con el Panel View que tengamos, en este caso 320x240. También en **Background Color** podemos elegir el color de fondo de la ventana. Además, configurar la opción **Position**

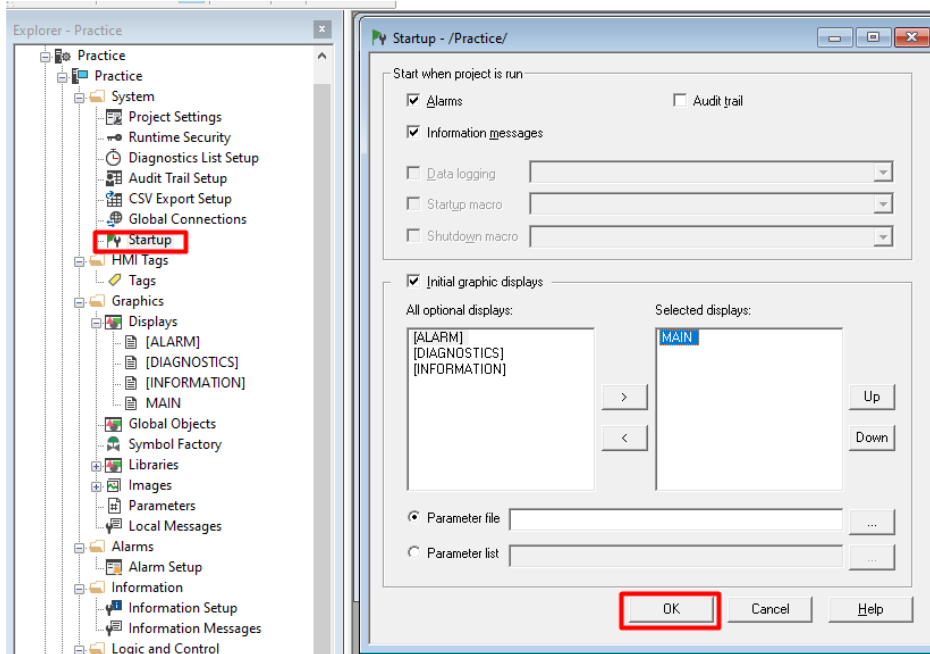
donde de acuerdo con los valores ingresados corresponden a la posición de donde surgirá la ventana en el Panel View.



4. En la ventana **Explorer**, doble clic en **Project Settings** y seleccionar la pestaña **General**, donde se configurará el tamaño de todas las ventanas del proyecto de acuerdo con el panel que utilizemos.

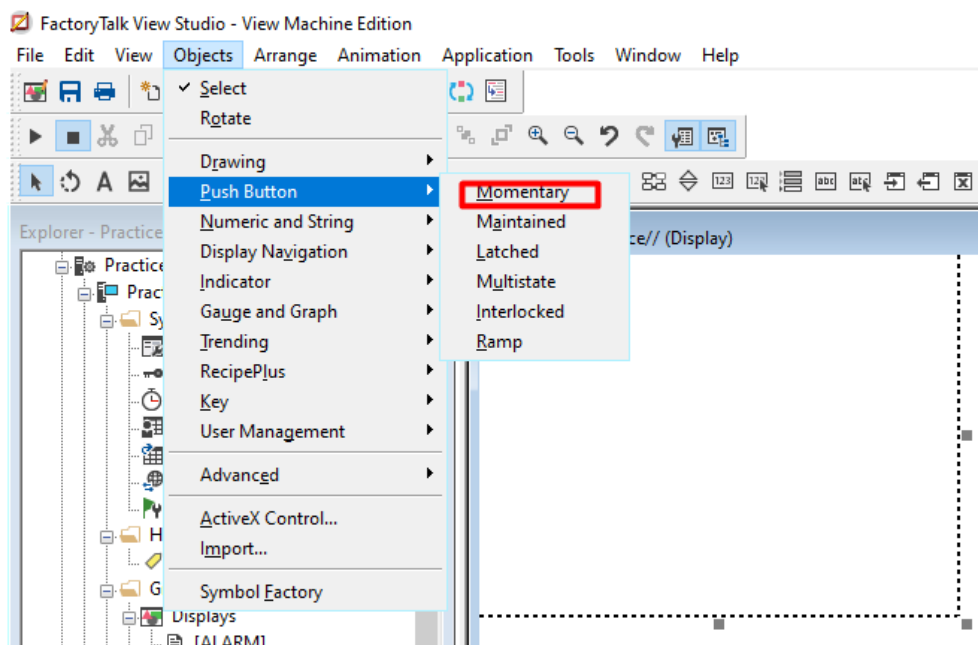


5. En la ventana **Explorer**, dar doble clic en **Startup** y marcar la casilla **Initial graphic displays** para que la ventana seleccionada aparezca inicialmente al cargar nuestro proyecto.

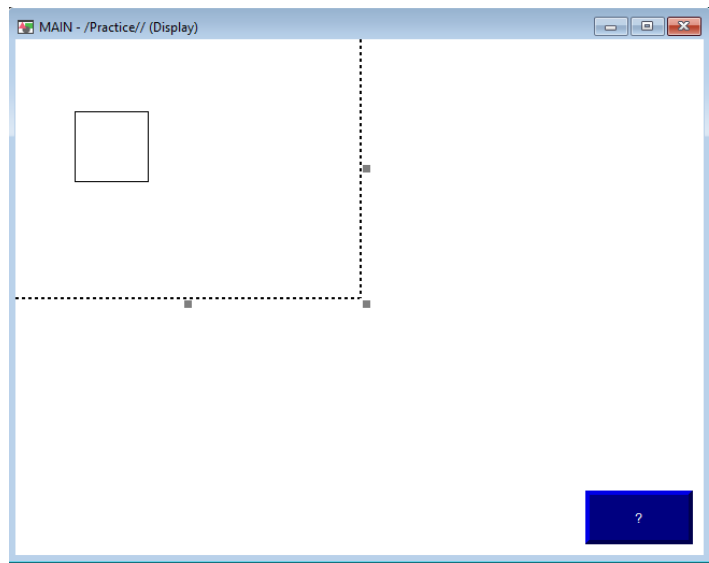


### Crear botones y asignaciones de tags en FactoryTalk Machine Edition

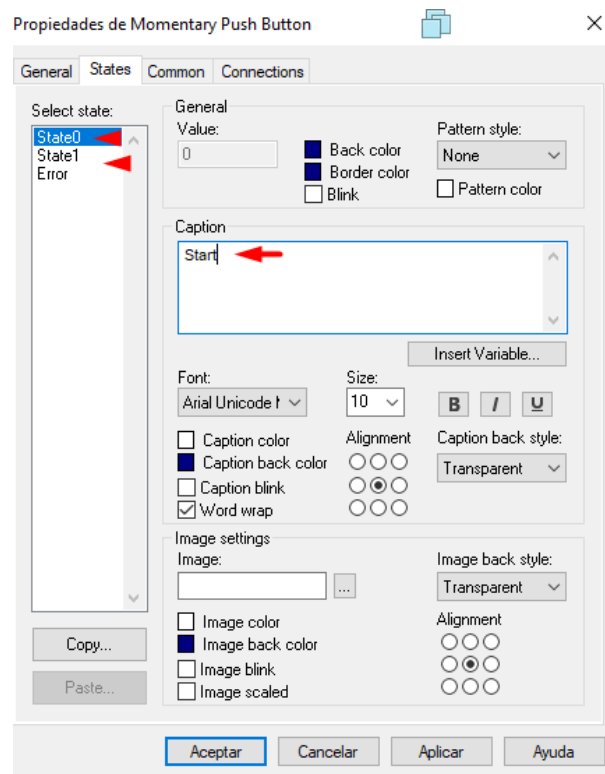
1. En la barra de menú de FTV, dar clic en **Objects** y seleccionar **Push Button**, luego escoger **Momentary**. Como podemos notar existen más opciones para el accionamiento mecánico de botón, éstos variarán de acuerdo con la elección del usuario o aplicación.



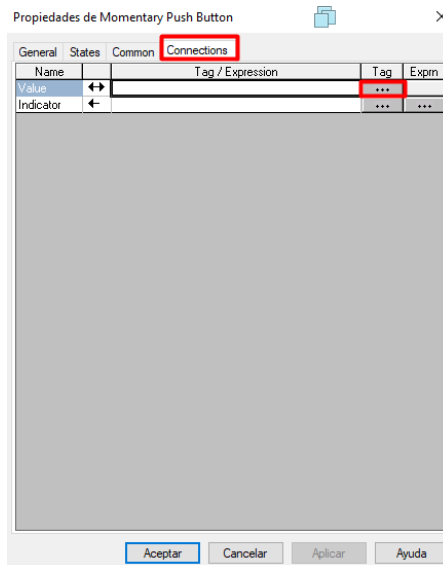
- Se debe “dibujar” el botón y a continuación se puede observar la figura agregada en nuestra aplicación.



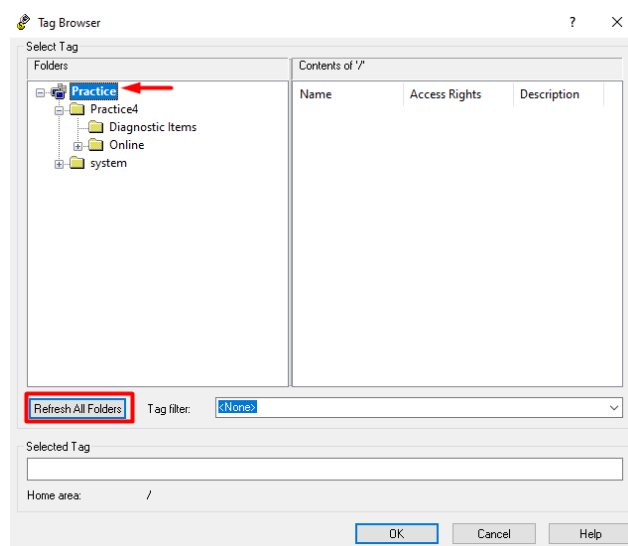
- Dar clic derecho al botón agregado y seleccionar **Properties** para abrir las propiedades de la ventana de Display. En la pestaña **States** de la ventana de **Propiedades de Momentary Push Button**, se puede colocar el texto para el botón en los estados del botón de acuerdo con el valor lógico.



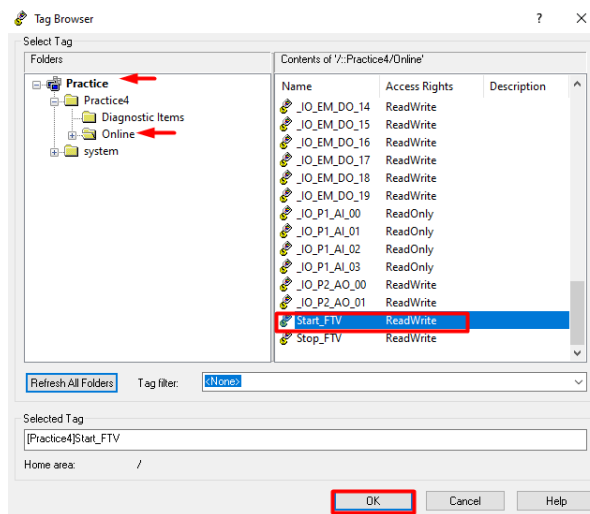
- En la pestaña **Connections**, dar clic en “...” de la fila **Value** y columna **Tag** para la asignación de una etiqueta al botón creado.



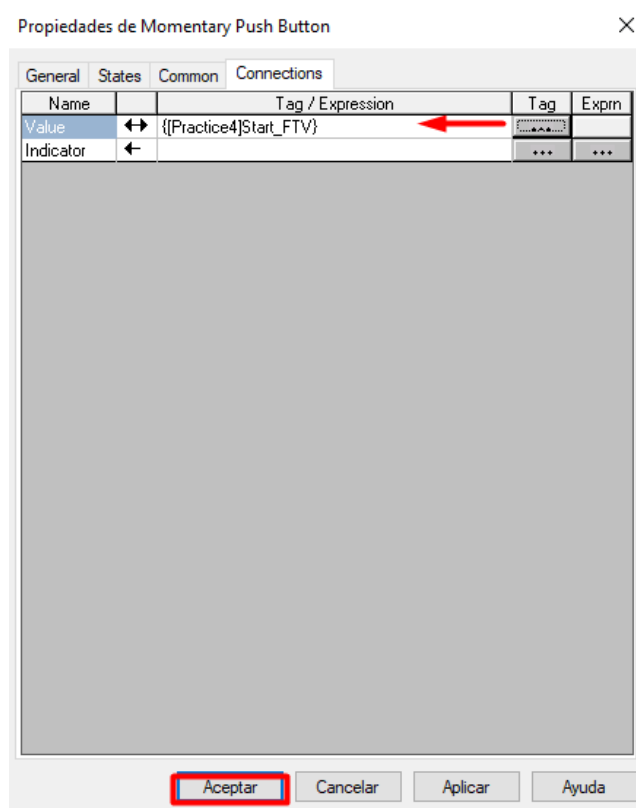
- La ventana **Tag Browser** aparecerá, donde se mostrará todos los tags creados tanto en Factory Talk View Machine Edition y Studio 5000. Cabe mencionar que los tags de Studio 5000 aparecerán sí la comunicación de una aplicación en FactoryTalk View Machine Edition se lo ha realizado con éxito. Dar clic en **Refresh All Folders** para actualizar los tags creados, luego clic en **Practice** que corresponde al nombre de nuestro proyecto de CCW descargado en el controlador.



- Desplazar **Practice** y dar clic en **Online** donde aparecerán las entradas y salidas físicas de nuestro controlador, así como las variables globales. Una vez seleccionado el tag dar clic en **OK**.



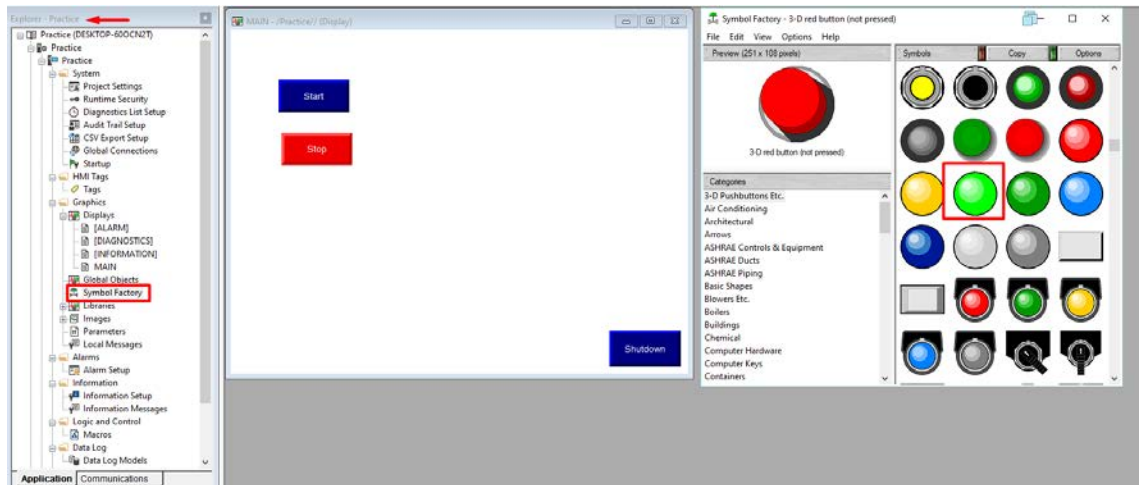
- A continuación, la ventana anterior de **Propiedades de Momentary Push Button** con la dirección del tag asignado, los mismos pasos lo realizaremos para los demás botones creados para nuestra aplicación.



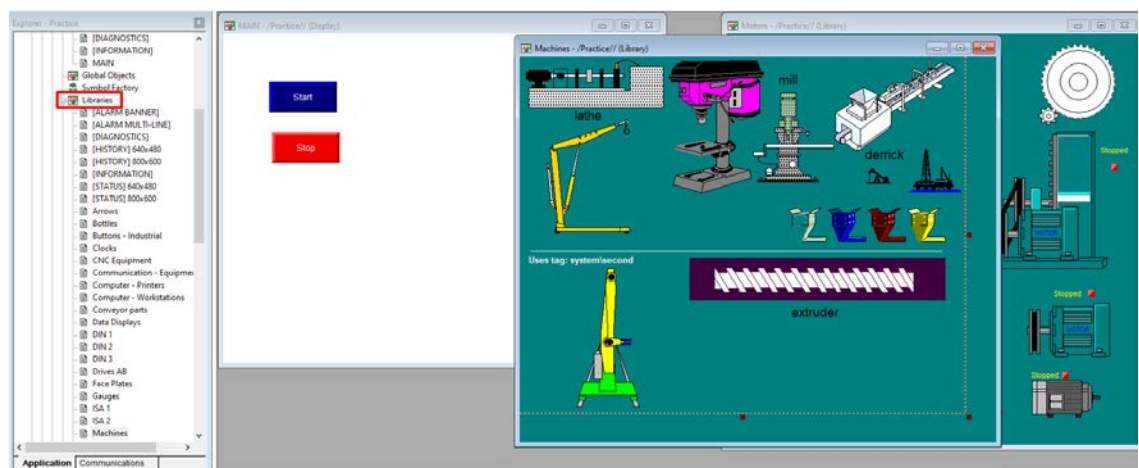


### Crear un símbolo u objeto en Factory Talk Machine Edition.

1. En la ventana **Explorer**, doble clic en **Symbol Factory** para colocar un objeto o símbolo como por ejemplo alguna máquina, motor, etc. A continuación, surgirá una ventana donde tendremos una variedad de símbolos que podemos elegir de acuerdo con nuestro proceso.



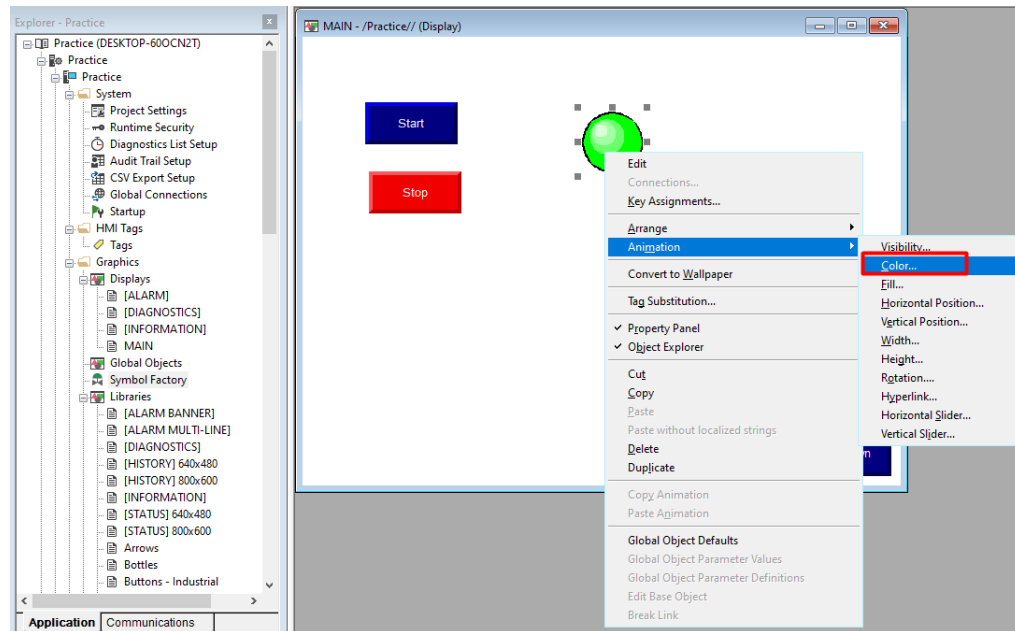
2. En la ventana **Explorer**, desplegar **Libraries** para obtener más elementos referentes a la aplicación del proceso que se está realizando.



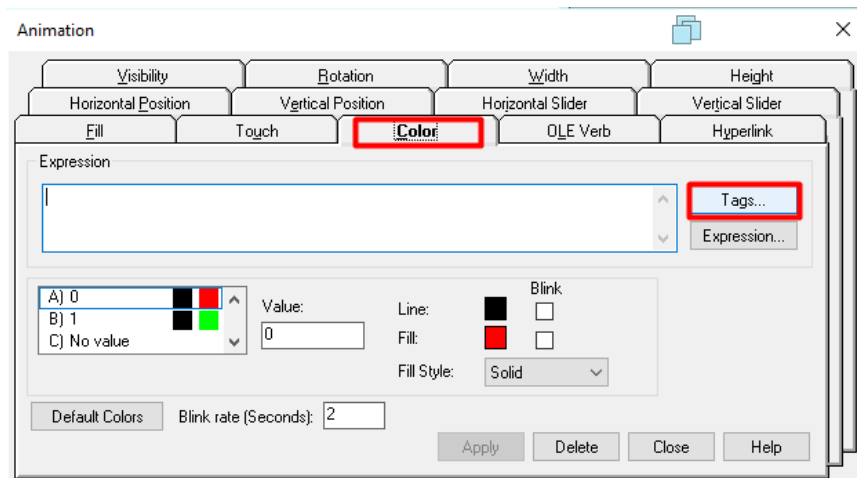
3. Arrastrar el objeto seleccionado a la ventana de la pantalla. Dar clic derecho y seleccionar **Animation**, aquí nos aparecerá una lista de animaciones como, por ejemplo, **visibility** donde el objeto será visible de acuerdo con el estado que anteriormente hallamos configurado ya sea falso o verdadero, **Color** hará cambiar de color al objeto de acuerdo con lo configurado, **Horizontal o Vertical Position** hará que el objeto se desplace la distancia y posición colocada, entre otros. En este caso seleccionar **Color**.





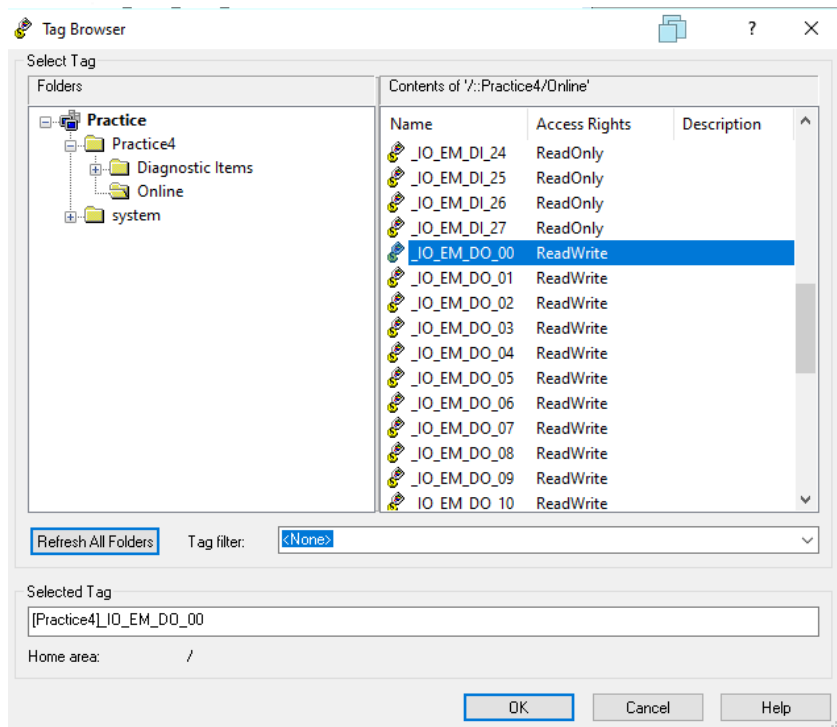


- La ventana de **Animation** aparecera. Seleccionar la pestaña de **Color** para cambiar algunos parámetros como el color de acuerdo al estado que se encuentre el tag, etc. Dar clic en **Tags...** para asignar un tag al simbolo.

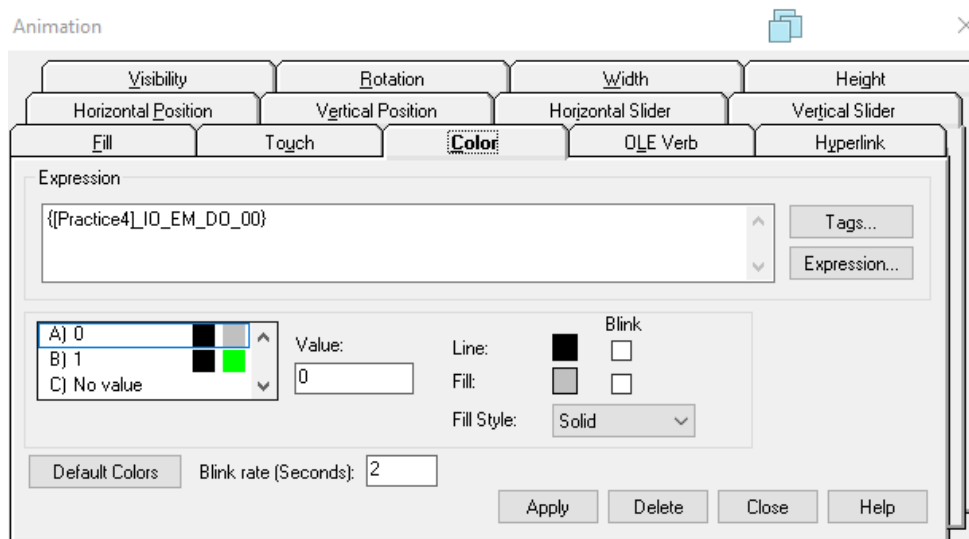


- La ventana de **Tag Browser** aparecerá. Escoger el tag con el cual queremos vincular al objeto como se realizó en pasos anteriores, luego dar clic en **OK**.



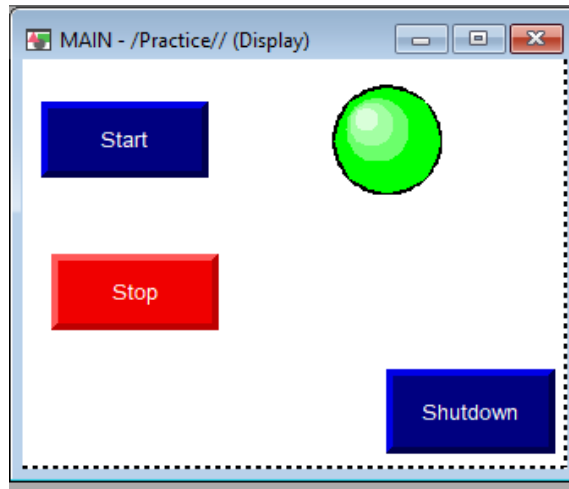


6. La ventana de **Animation** aparecerá con la dirección del tag asignado. Además, podemos cambiar el color de acuerdo con el estado del tag en la sección de **Value** y **Fill**. Luego, dar clic en **Apply** y en **Close**.



6. Finalmente, nuestro proyecto es como muestra la figura a continuación. El botón **Shutdown** está por defecto en el display main, cuya función permite salir de la aplicación en el PanelView y regresar al menú principal del mismo.





7. Dar clic en **Test Display** para simular la aplicación.

