

## PRÁCTICA

### Implementación de PID en Connected Component Workbench

#### 1. Objetivos

##### 1.1. Objetivo general

Ajustar las constantes del bloque PID para el control del nivel del tanque en Factory IO, evaluando su desempeño en Connected Component Workbench y Factory Talk View.

##### 1.2 Objetivos específicos

1. Configurar las constantes del bloque PID para el sistema de nivel de tanque.
2. Analizar el desempeño del bloque PID en Connected Component Workbench en la regulación de nivel de tanque de la planta de Factory IO.

#### 2. Equipos y herramientas

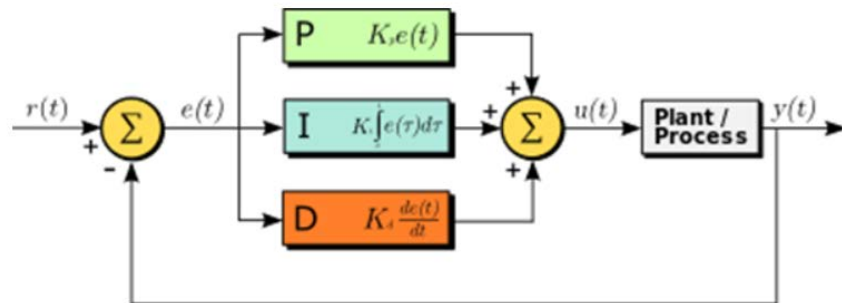
- Micro850
- Connected Components Workbench
- RsLinx Classic
- Computadora
- Factory IO

#### 3. Marco teórico

##### Controlador PID

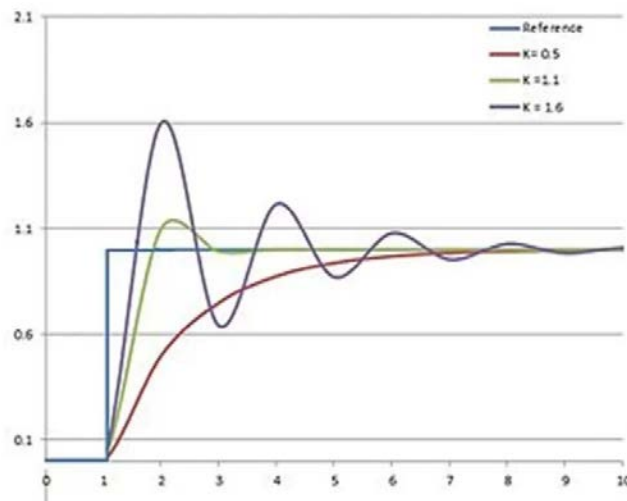
Un Controlador Proporcional Integral Derivativo (PID) es un sistema de control de circuito cerrado comúnmente utilizado en sistemas de control industrial. Este método es altamente efectivo en aplicaciones que requieren una modulación de control continua. El PID calcula un valor de error, que es la diferencia entre el valor de control actual y un valor de control objetivo. Utilizando este valor de error, el bucle de control proporciona correcciones al sistema de salida. El bucle de control utiliza tres términos de control: proporcional, integral y derivativo, los cuales influyen en cómo el sistema de control avanza hacia el valor objetivo. Para ajustar un bucle de control PID, cada término se multiplica por un coeficiente para modular su influencia en el comportamiento del sistema. Los tres coeficientes son  $K_p$ ,  $K_i$  y  $K_d$ . Cabe mencionar que el objetivo de los ajustes de los parámetros PID es lograr que el bucle de control corrija eficazmente y en el mínimo tiempo los efectos de las perturbaciones; se

tiene que lograr la mínima integral de error. Si los parámetros del controlador PID (la ganancia del proporcional, integral y derivativo) se eligen incorrectamente, el proceso a controlar puede ser inestable, por ejemplo, que la salida de este varíe, con o sin oscilación (Yang, 2022).



### **Coefficiente proporcional ( $K_p$ )**

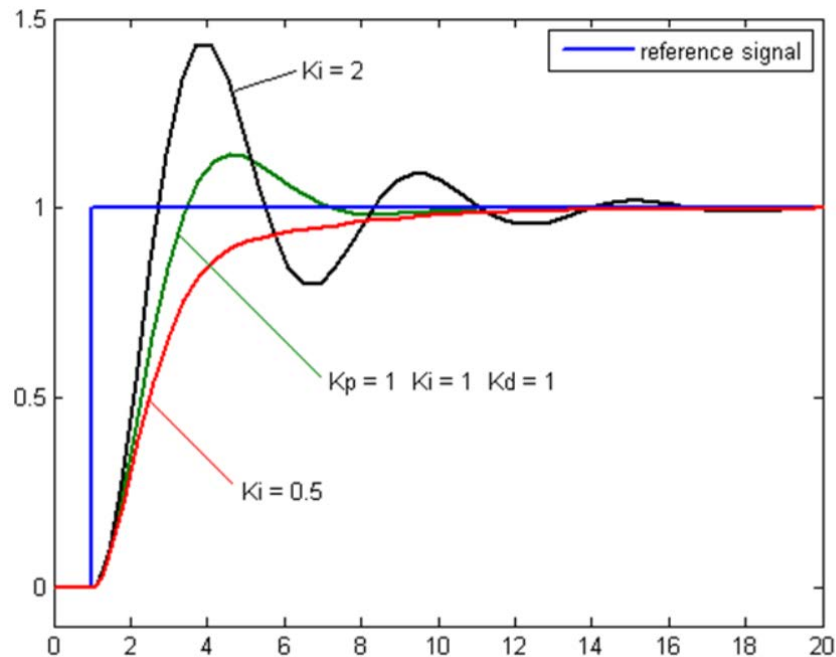
El término proporcional genera un valor de salida que es proporcional al error. Entonces, aumentar el coeficiente del término proporcional crea un circuito de control más sensible. Con un coeficiente alto, el controlador PID resultante responderá rápidamente. Sin embargo, este método provocará una mayor inestabilidad del sistema y provocará un mayor sobre nivel porcentual. Generalmente, el coeficiente proporcional es el coeficiente dominante de un lazo de control PID y debería contribuir con la mayor parte del comportamiento de salida.



### **Constante integral ( $K_i$ )**

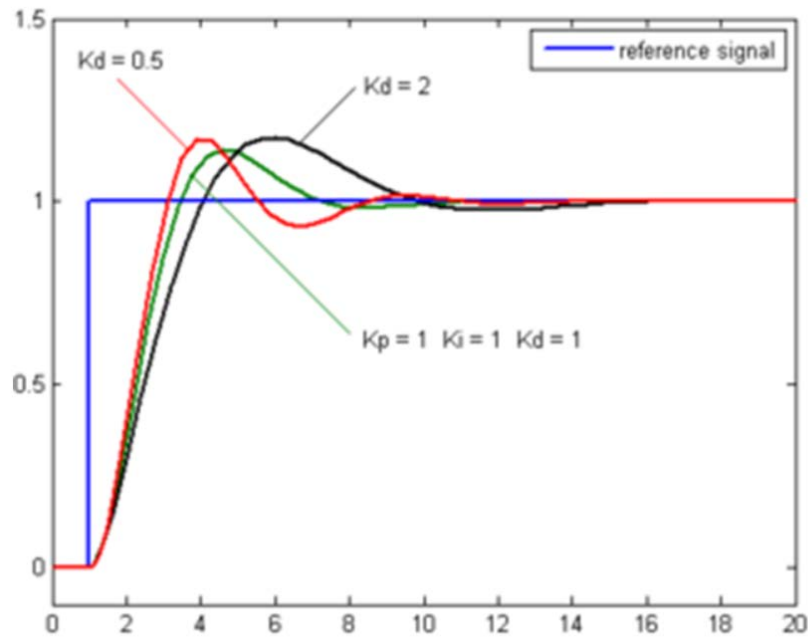
Toma la integral de los errores registrados por el lazo de control y, por lo tanto, tiene en cuenta tanto la magnitud como la duración de los valores de error. Debido a que la integral

considera el tiempo, puede tener en cuenta el valor de error actual y los valores de error detectados en el pasado, algo que el término proporcional por sí solo no puede hacer. Un coeficiente integral más alto ayuda a amortiguar el error de estado estable, y esto permite que el sistema se asiente más rápidamente en su objetivo. Sin embargo, los coeficientes integrales altos pueden causar un mayor sobrepaso cuando se acerca por primera vez al punto de operación.

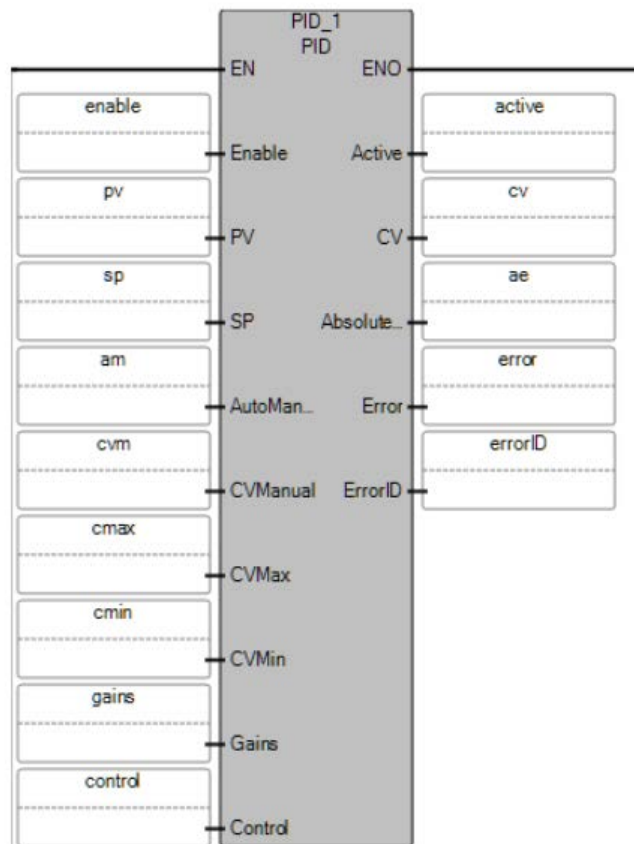


### **Coeficiente derivativo ( $K_d$ )**

Utiliza la pendiente del valor del error a medida que cambia con el tiempo. De esta manera, el término derivado rastrea el error a medida que cambia con el tiempo y puede usarse para predecir el comportamiento del sistema. El término derivado puede suavizar en gran medida el rendimiento del circuito de control, aumentar el tiempo de estabilización y la estabilidad. Sin embargo, el término derivado de bucle de control PID rara vez se utiliza en la práctica ya que sus efectos sobre el rendimiento del sistema pueden ser muy difíciles de predecir. Como resultado, ajustar un coeficiente de término derivado adecuado es generalmente muy difícil.



#### RA PID AUTOTUNE UDFB



Es un bloque de controlador de lazo cerrado que se utiliza para ajustar automáticamente los parámetros de un controlador PID en función de la respuesta del proceso en tiempo real. EL bloque cuenta con los siguientes parámetros:

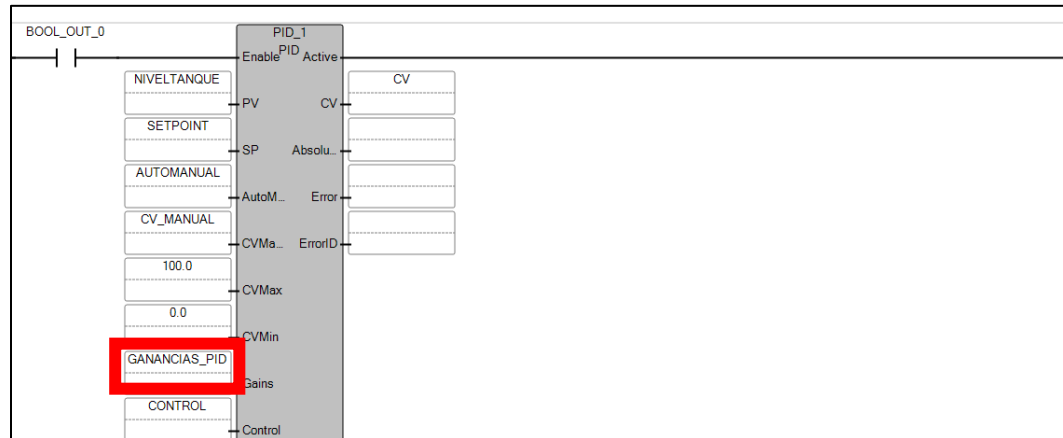
Parámetro	Tipo de parámetro	Tipo de dato	Descripción
Enable	Entrada	BOOL	Habilite la instrucción.  TRUE: inicia la ejecución con los parámetros de entrada actuales.  FALSE: PID no se ejecuta. Establezca CV en 0 y calcule AbsoluteError.
PV	Entrada	REAL	Valor del proceso. Normalmente, este valor se lee desde un módulo de entrada analógica.  La unidad SI debe ser la misma que el punto de ajuste.
SP	Entrada	REAL	El valor de consigna para el proceso.
AutoManual	Entrada	BOOL	Selección de modo automático o manual.  TRUE: CV está controlado por PID.  FALSE - PID se está ejecutando y CV está controlado por la entrada CVManual.
CVManual	Entrada	REAL	Entrada de valores de control definida para el funcionamiento en modo manual.  El rango válido para CVManual es:  $CVMin < CVManual < CVMax$
CVMin	Entrada	REAL	Límite mínimo del valor de control.  Si $CV < CVMin$ , entonces $CV = CVMin$ .  Si $CVMin > CVMax$ , se produce un error.
CVMax	Entrada	REAL	Límite máximo del valor de control.  Si $CV > CVMax$ , entonces $CV = CVMax$ .  Si $CVMax < CVMin$ , se produce un error.
Gains	Entrada	PID_GAINS	Ganancias de PID para el controlador.

			Utilice el tipo de datos PID_GAINS para configurar el parámetro Ganancia.
Control	Entrada	BOOL	Dirección de control del proceso:  TRUE - Actuación directa, como Enfriamiento.  FALSE - Acción inversa, como Calentamiento.
Active	Salida	BOOL	Estado del controlador PID.  TRUE: PID está activo.  FALSE - PID está detenido.
CV	Salida	REAL	La salida del valor de control.  Si se ha producido algún error, CV es 0.
Error absoluto	Salida	REAL	El error absoluto es la diferencia entre el valor de proceso (PV) y el valor de consigna (SP).
Error	Salida	BOOL	Indica la existencia de una condición de error.  TRUE: la operación encontró un error.  FALSE - La operación se ha completado correctamente o la instrucción no se está ejecutando.
ErrorID	Salida	USINT	Un número único que identifica el error. Los errores se definen en códigos de error PID.  0 PID funciona normalmente. 1 Kc no es válido. 2 Ti no es válido. 3 Td no es válido. 4 FC no es válido. 5 CVMin > CVMax o CVMax < CVMin 6 CVManual < CVMin  CVManual no es válido. 7 CVManual > CVMax  CVManual no es válid

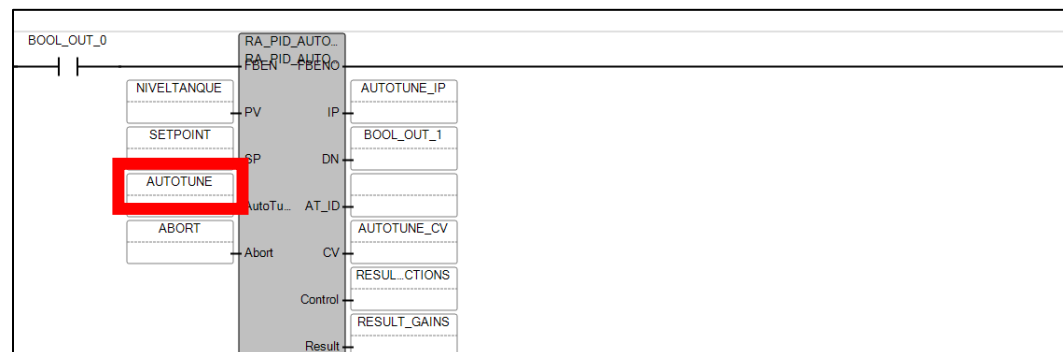


## Funcionamiento del controlador sintonizado

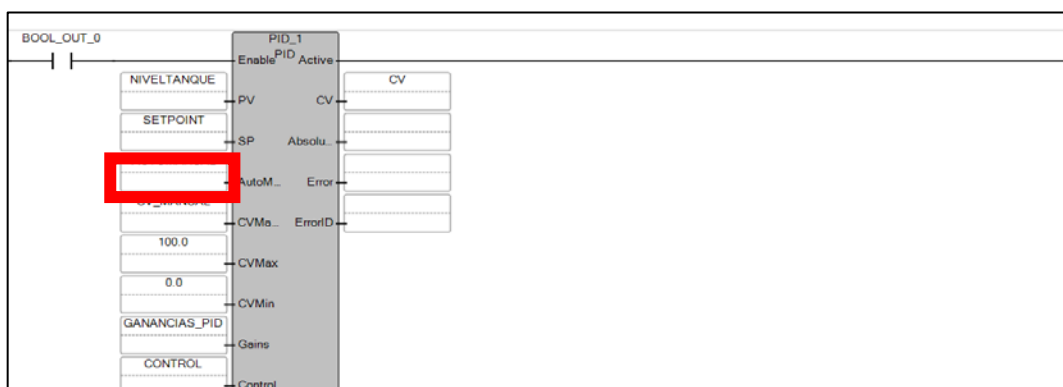
1. Las ganancias obtenidas anteriormente como salida del bloque “RA\_PID\_AUTOTUNE” se las procede a escribir como las ganancias del bloque “PID”.



2. Se verifica que la entrada “AutoTune” del bloque “RA\_PID\_AUTOTUNE” sea falsa puesto que ya está sintonizado el controlador.



3. Se activa la entrada “AutoManual” del bloque “PID” y el control con PID inicia a funcionar.



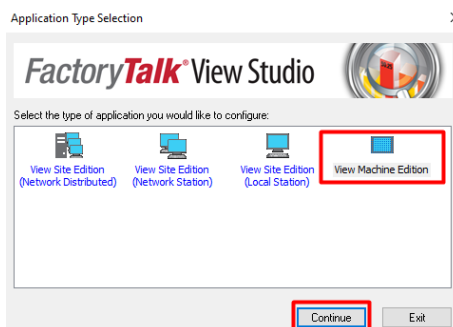


4. Se evidencia el correcto funcionamiento del controlador PID.



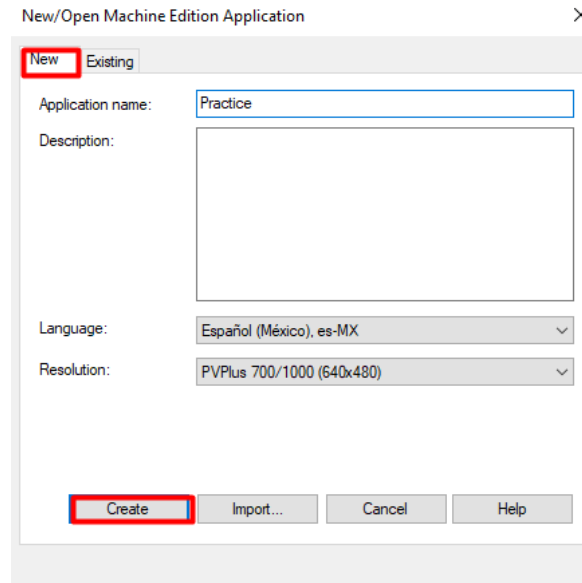
### Crear y configurar la comunicación de una aplicación en FactoryTalk View Machine Edition

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

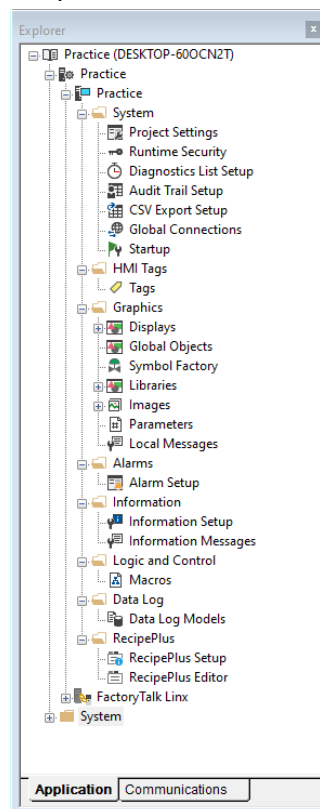


2. Aparecerá automáticamente la ventana **New/Open Machine Edition Application**, en la pestaña **New** buscar la sección **Application name** para escribir el nombre de la aplicación que realizaremos, luego seleccionar el lenguaje que queramos que este el programa y dar clic en **Create**.



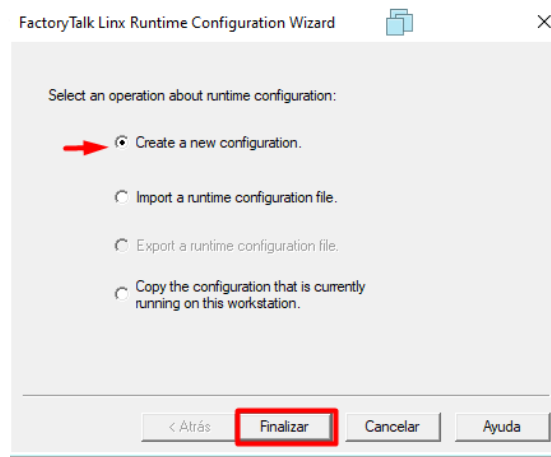


- En la ventana **Explorer**, Desplegar la opción **FactoryTalk Linx** para establecer la comunicación y dar doble clic a la opción **Communications Setup**.

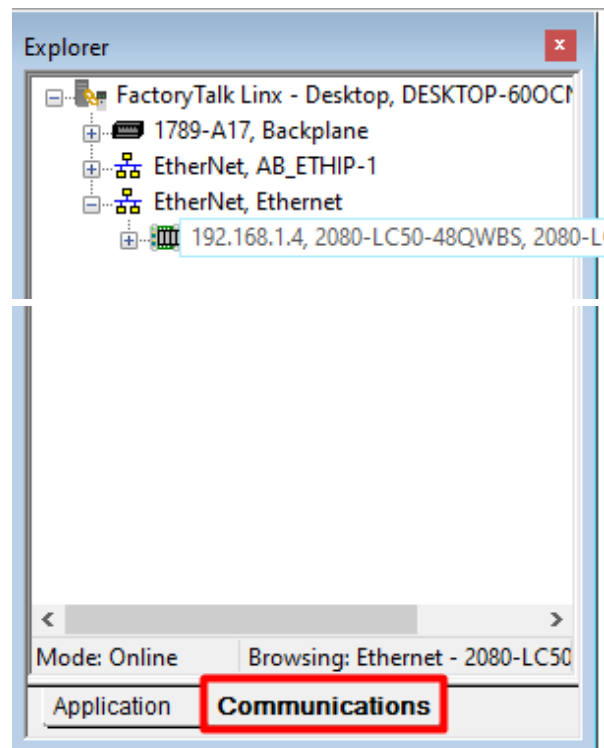


- En la ventana de **FactoryTalk Linx Configuration Wizard**, seleccionar **Create a new configuration** y dar clic en **Finalizar**.



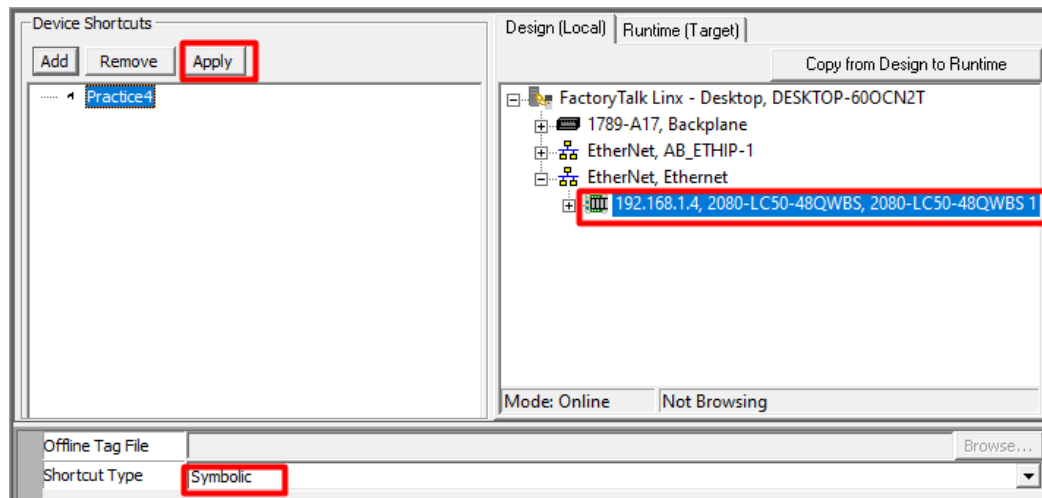


5. En la ventana **Explorer**, en la parte inferior se encuentra la pestaña **Communications** donde se muestra los dispositivos conectados a la red.

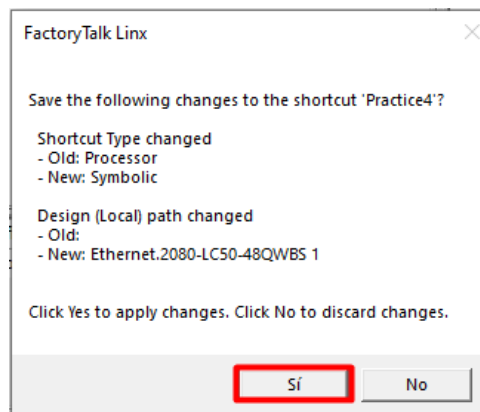


7. En la ventana **Device Shortcuts**, clic en **Add** para crear un **shortcut** que nos representa la ruta específica del dispositivo a comunicarnos, y posteriormente asignarle un nombre. Luego teniendo seleccionado el shortcut, en la pestaña **Design (Local)** escoger el controlador con que se va 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**.

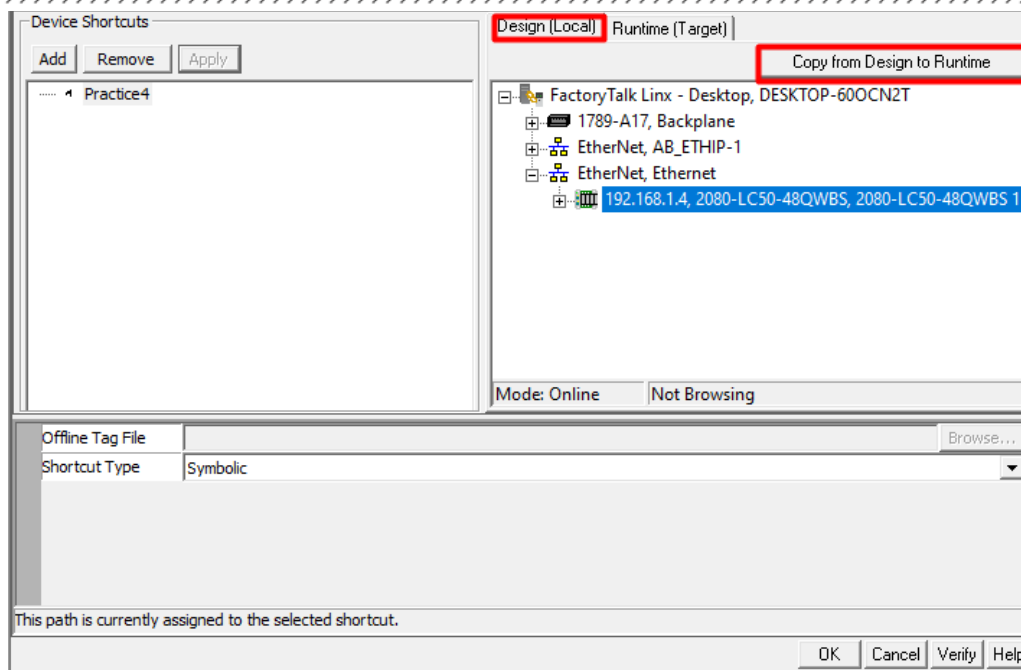




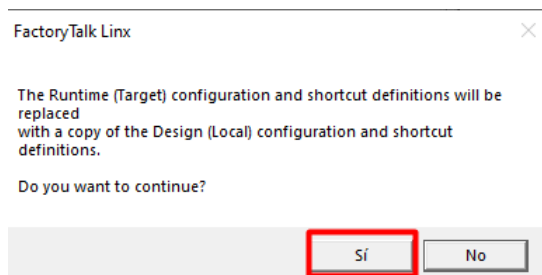
8. Aparecerá la siguiente ventana de información, dar clic en **Si**.



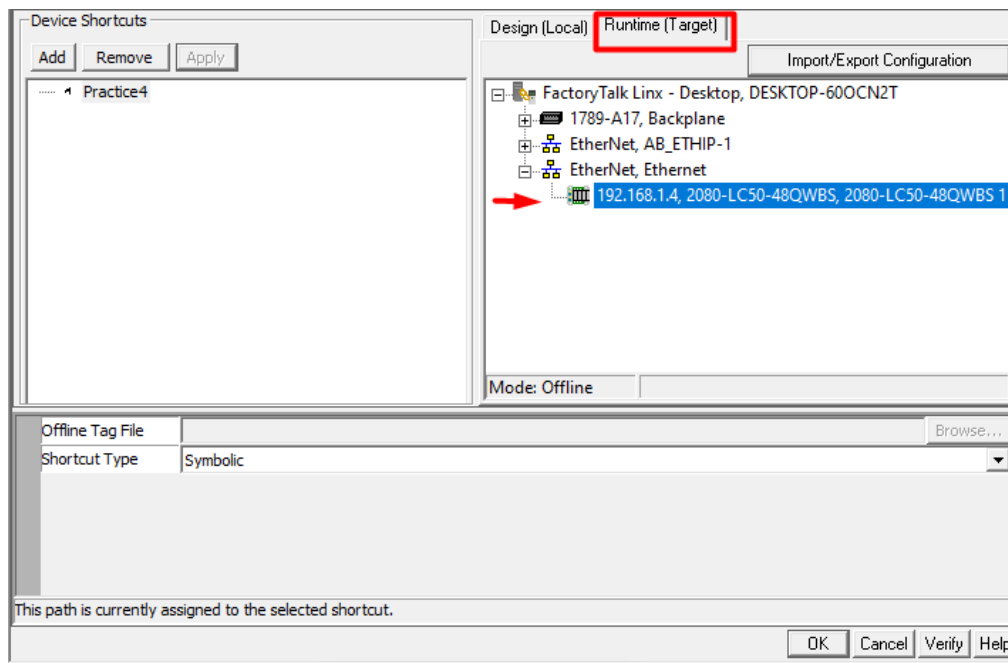
9. En la parte derecha de la pestaña **Design (Local)**, dar clic en **Copy from Design to Runtime** para copiar la topología en la pestaña **Runtime (Target)**.



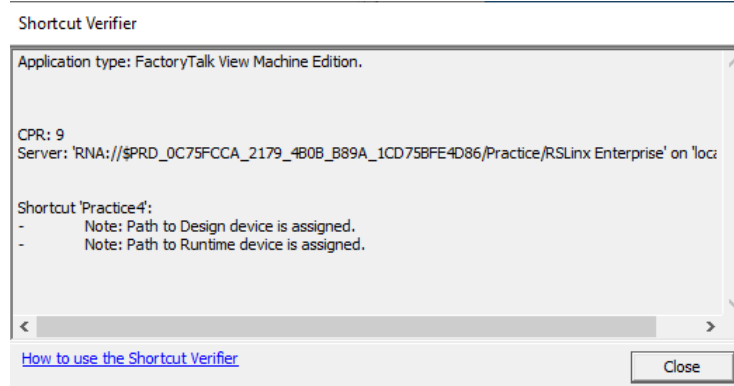
10. En la siguiente ventana, dar clic en **Si**.



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

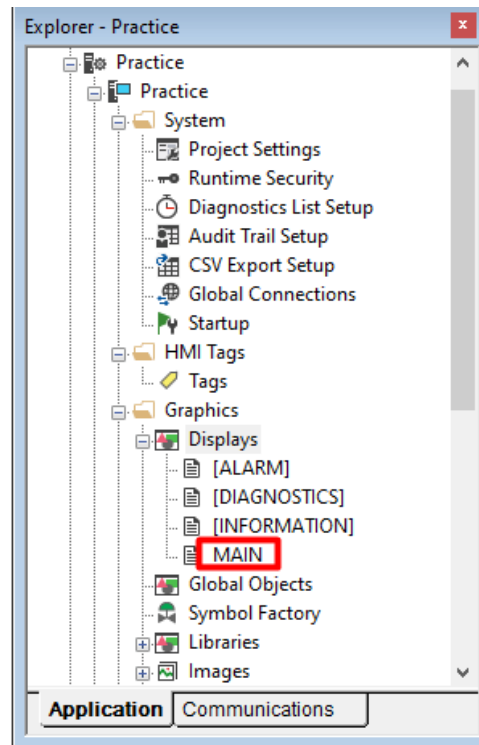


12. En la parte inferior de la pestaña **Design (Local)**, dar clic en **Verify** donde nos aparecerá la siguiente ventana y dar Close. Si todos los pasos fueron realizados correctamente, los enlaces de **Design** y **Runtime** fueron asignados y mostradas en el cuadro de verificación.

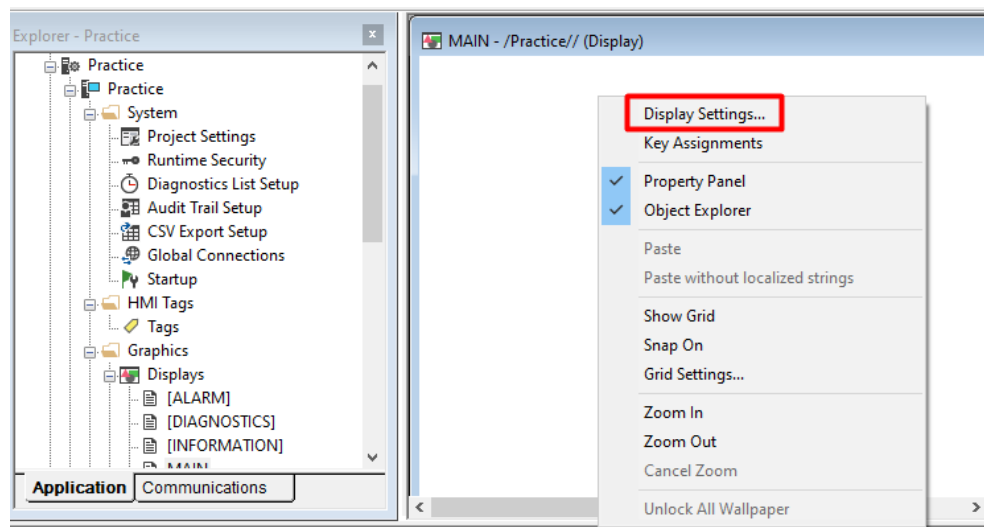


### Crear un display en FactoryTalk Machine Edition

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.

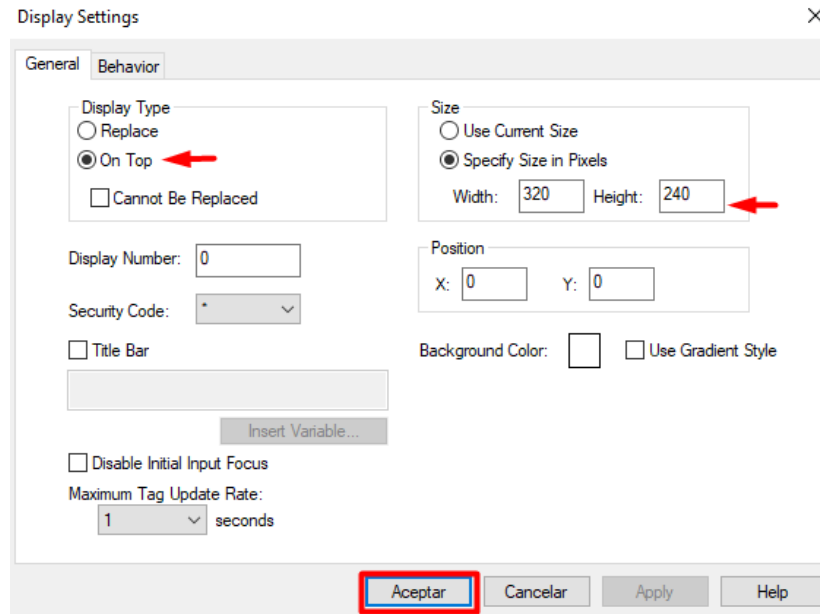


2. Dar clic derecho en un espacio en blanco de la pantalla MAIN y selecciona **Display Settings** para abrir las propiedades de la ventana de Display, en la cual se puede configurar ciertos parámetros de acuerdo con las necesidades de la aplicación.

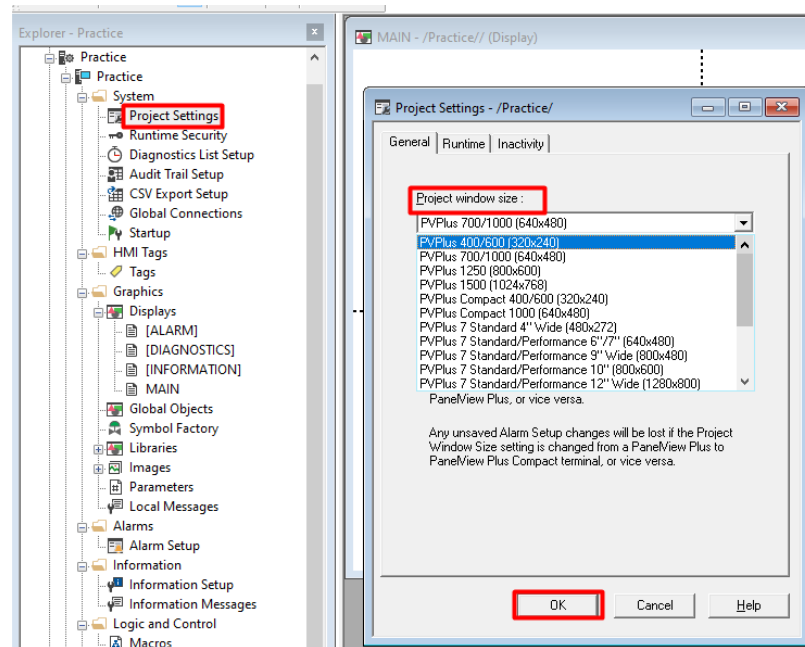


3. La ventana **Display Settings** aparecerá. En la pestaña **General**, en la opción **Display Type** seleccionar **On Top**, y en la opción **Size** podemos configurar el tamaño de la ventana de acuerdo con el Panel View que tengamos, en este caso 320x240. También en **Background Color** podemos elegir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

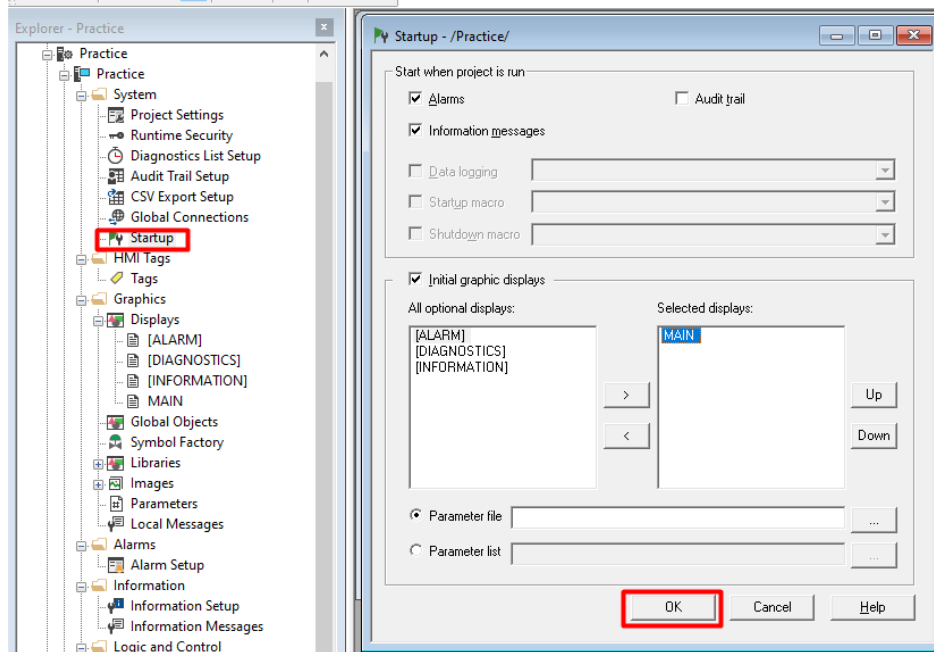


- En la ventana **Explorer**, doble clic en **Project Settings** y seleccionar la pestaña **General**, donde se configurará el tamaño de todas las ventanas del proyecto de acuerdo con el panel que utilizemos.



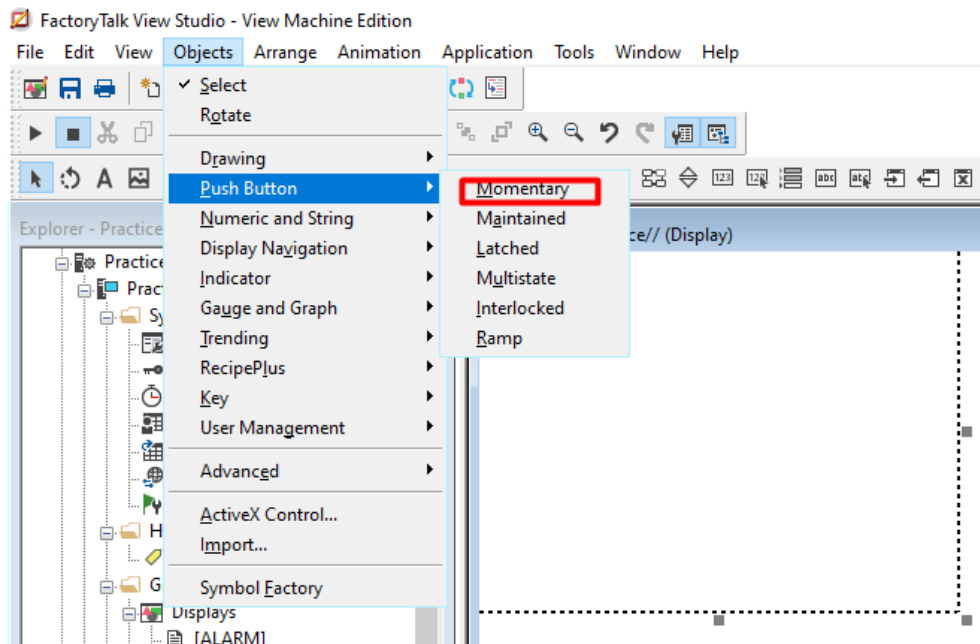


- En la ventana **Explorer**, dar doble clic en **Startup** y marcar la casilla **Initial graphic displays** para que la ventana seleccionada aparezca inicialmente al cargar nuestro proyecto.

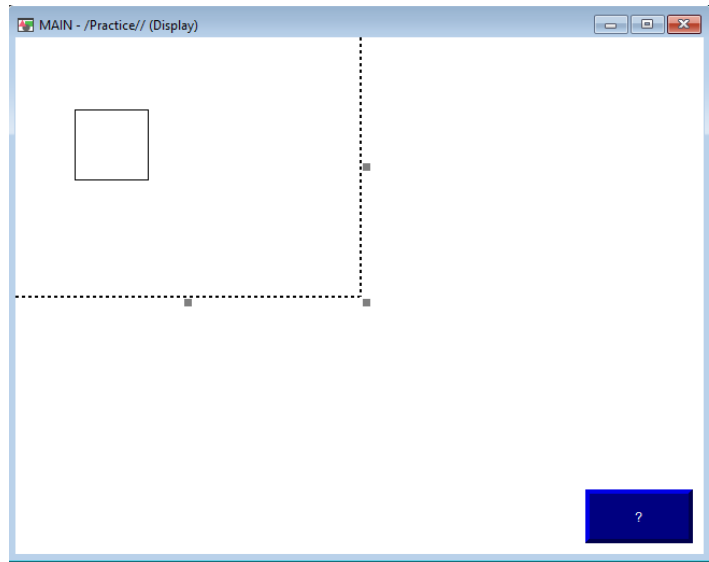


### Crear botones y asignaciones de tags en FactoryTalk Machine Edition

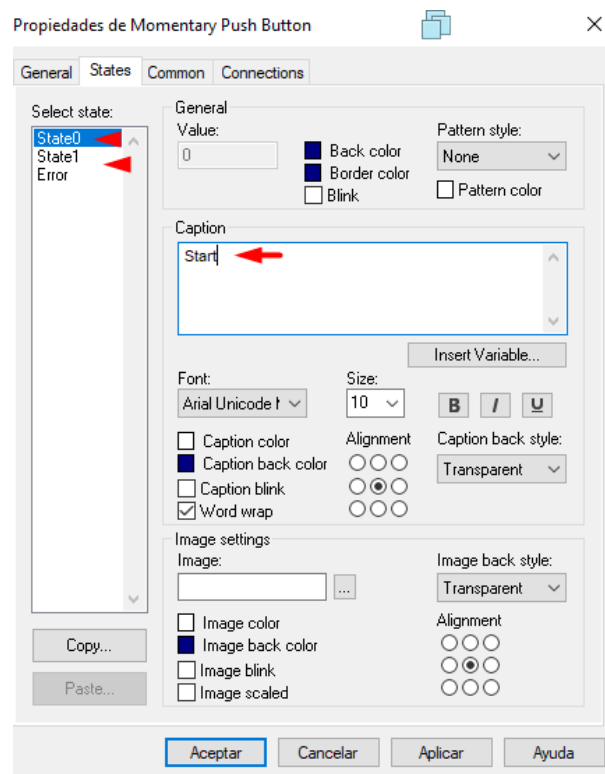
- En la barra de menú de FTV, dar clic en **Objects** y seleccionar **Push Button**, luego escoger **Momentary**. Como podemos notar existen más opciones para el accionamiento mecánico de botón, éstos variarán de acuerdo con la elección del usuario o aplicación.



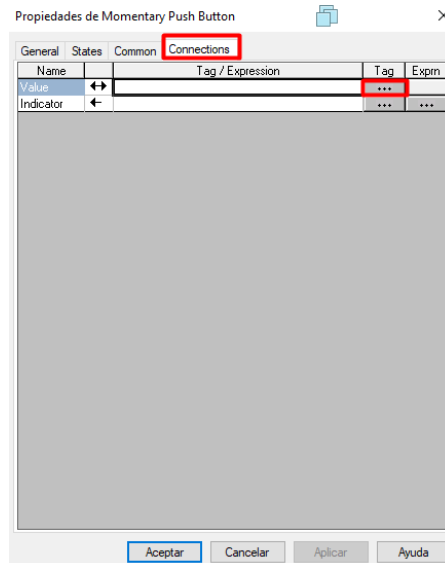
2. Se debe “dibujar” el botón y a continuación se puede observar la figura agregada en nuestra aplicación.



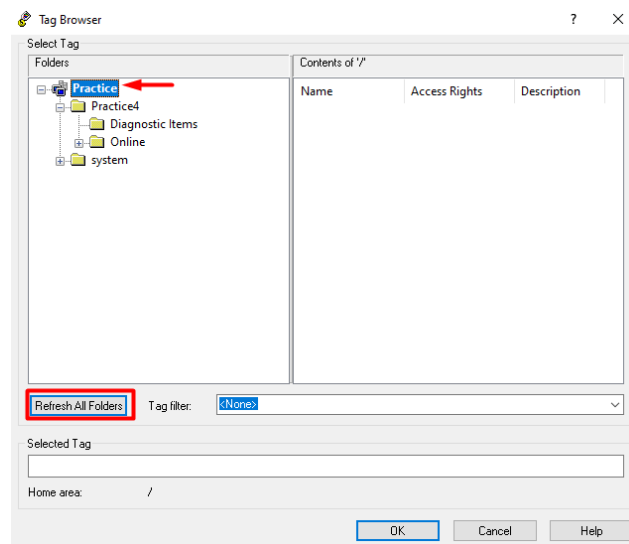
3. Dar clic derecho al botón agregado y seleccionar **Propiedades** para abrir las propiedades de la ventana de Display. En la pestaña **States** de la ventana de **Propiedades de Momentary Push Button**, se puede colocar el texto para el botón en los estados del botón de acuerdo con el valor lógico.



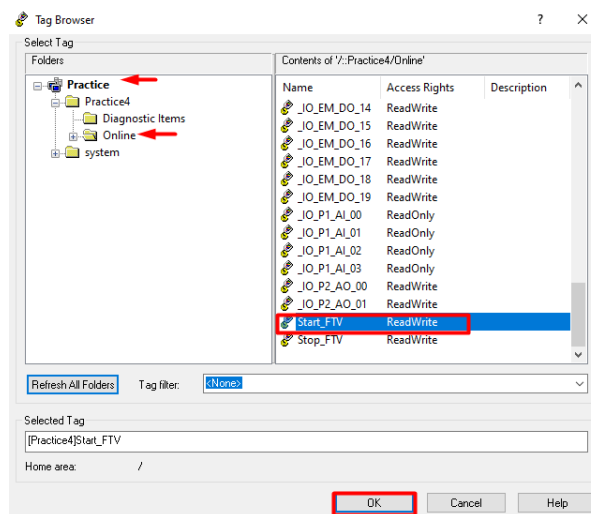
- En la pestaña **Connections**, dar clic en “...” de la fila **Value** y columna **Tag** para la asignación de una etiqueta al botón creado.



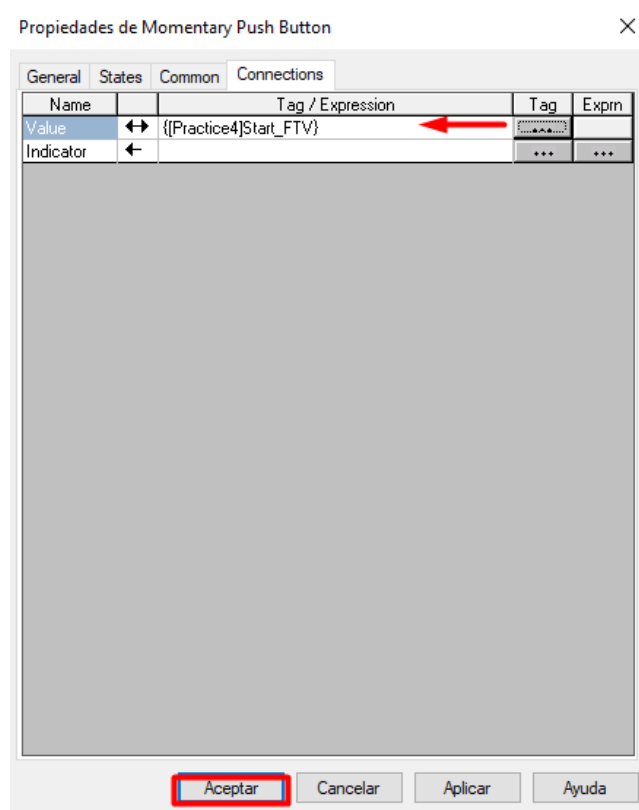
- La ventana **Tag Browser** aparecerá, donde se mostrará todos los tags creados tanto como en Factory Talk View Machine Edition y Studio 5000. Cabe mencionar que los tags de Studio 5000 aparecerán sí la comunicación de una aplicación en FactoryTalk View Machine Edition se lo ha realizado con éxito. 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**.

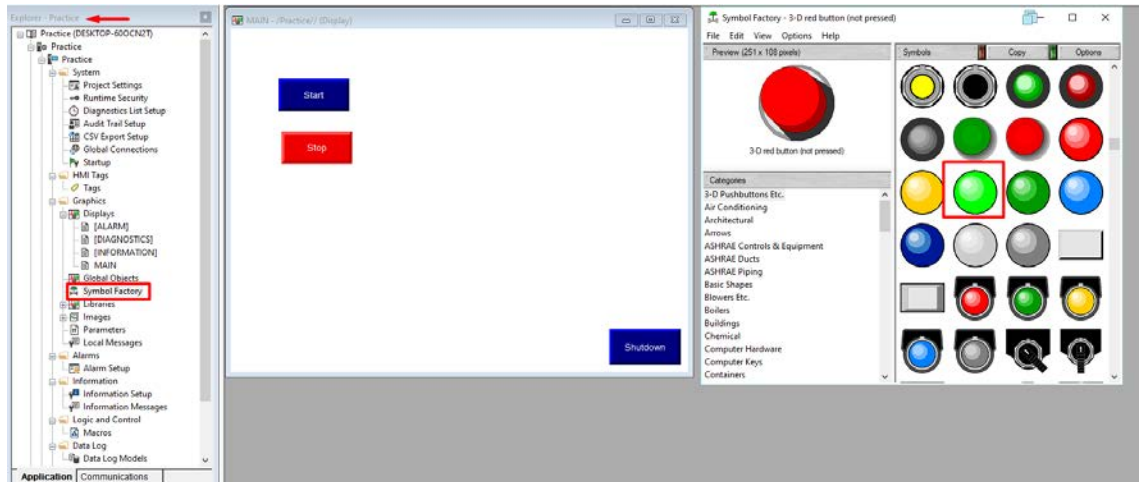


- A continuación, la ventana anterior de **Propiedades de Momentary Push Button** con la dirección del tag asignado, los mismos pasos lo realizaremos para los demás botones creados para nuestra aplicación.

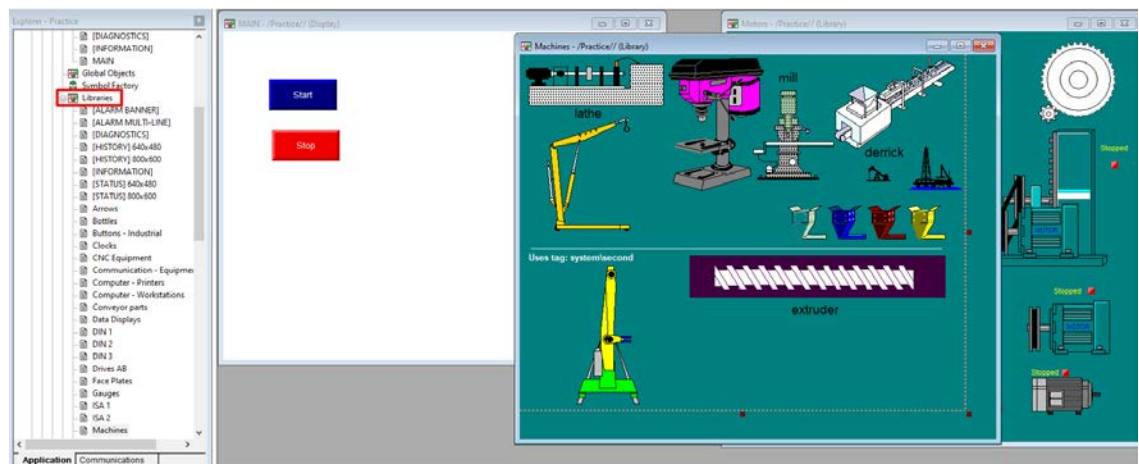


## Crear un símbolo u objeto en Factory Talk Machine Edition.

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.

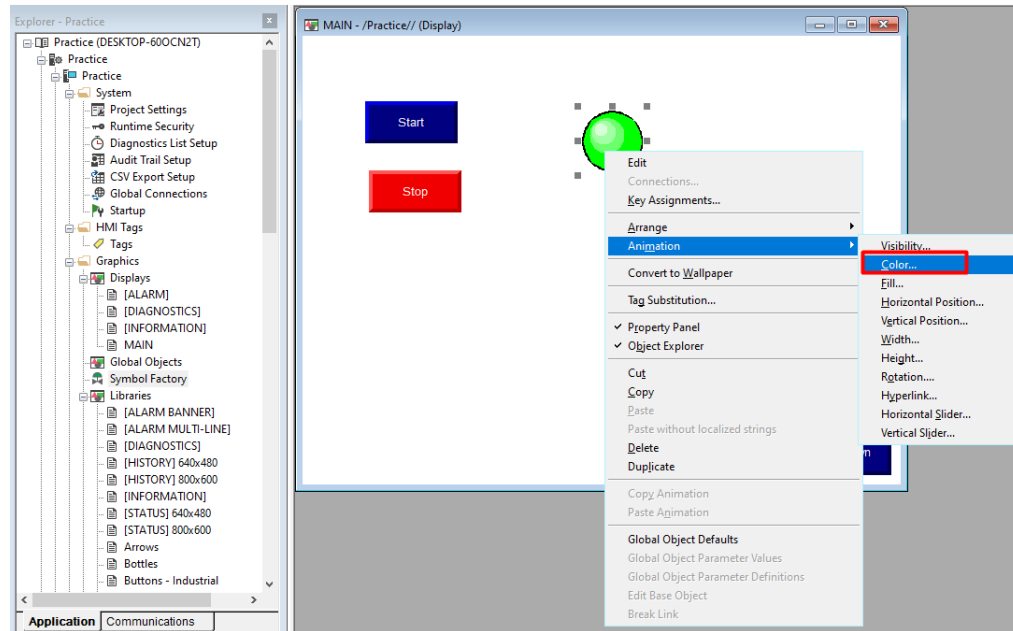


2. En la ventana **Explorer**, desplegar **Libraries** para obtener más elementos referentes a la aplicación del proceso que se está realizando.

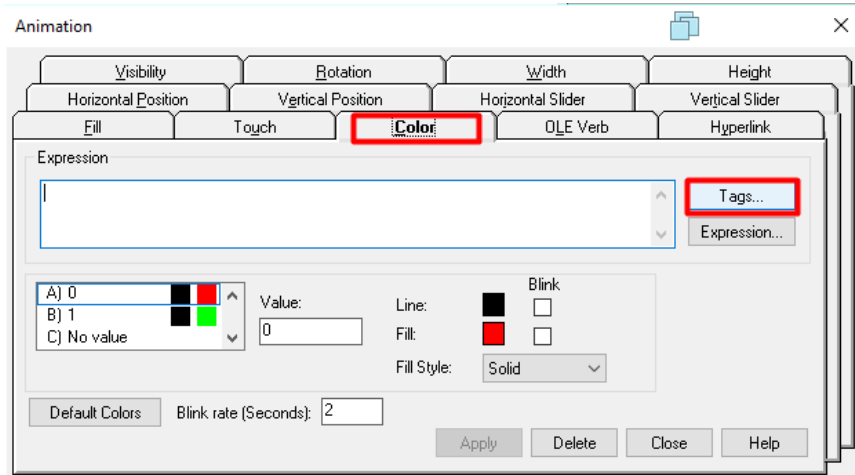


3. Arrastrar el objeto seleccionado a la ventana de la pantalla. Dar clic derecho y seleccionar **Animation**, aquí nos aparecerá una lista de animaciones como, por ejemplo, **visibility** donde el objeto será visible de acuerdo con el estado que anteriormente hallamos configurado ya sea falso o verdadero, **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**.

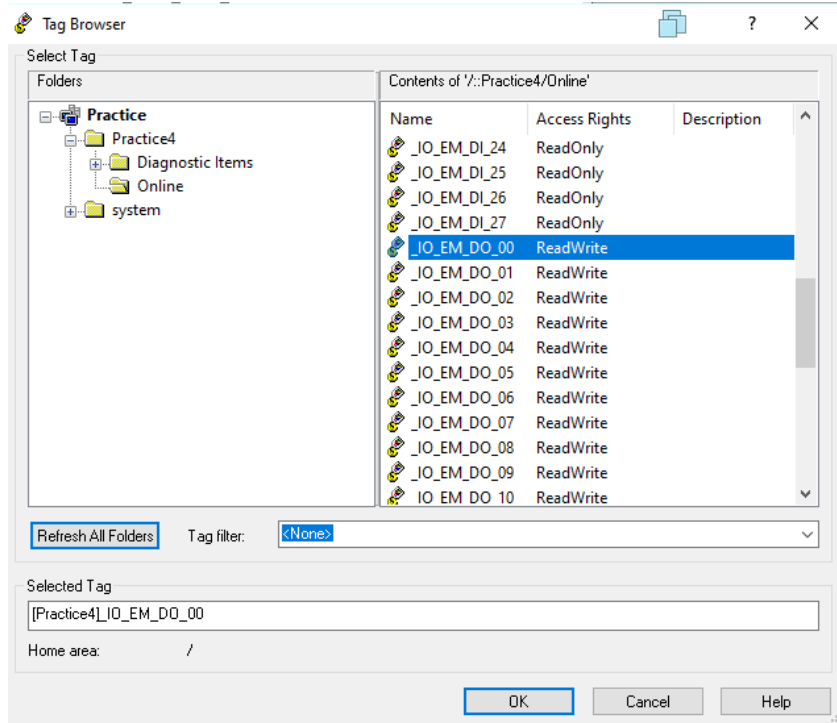




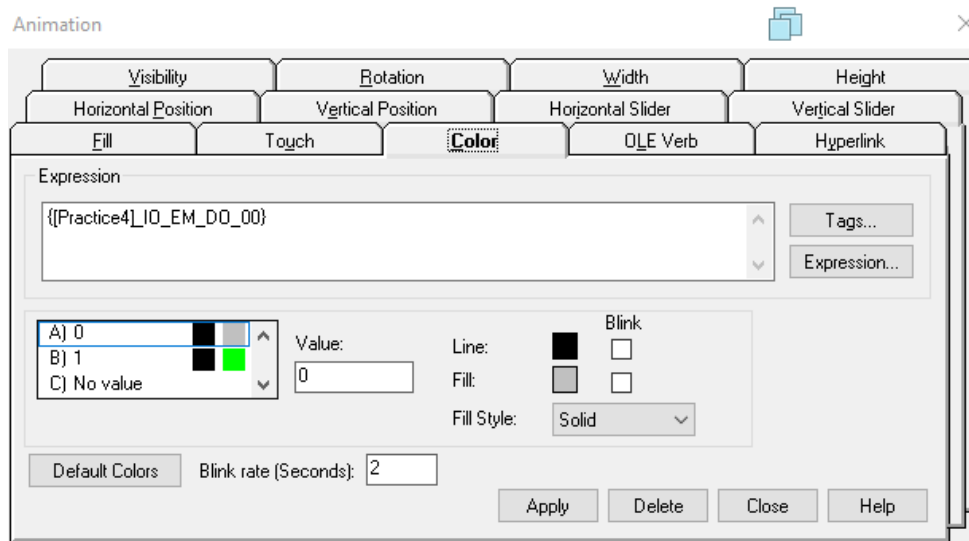
4. La ventana de **Animation** aparecera. Seleccionar la pestaña de **Color** para cambiar algunos parámetros como el color de acuerdo al estado que se encuentre el tag, etc. Dar clic en **Tags...** para asignar un tag al simbolo.



5. 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**.



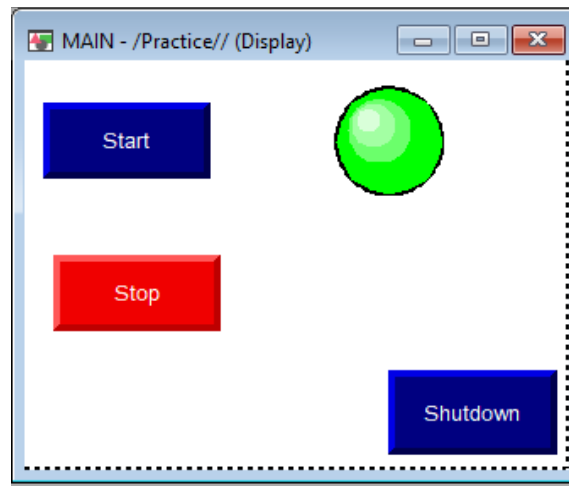
6. La ventana de **Animation** aparecerá con la dirección del tag asignado. Además, podemos cambiar el color de acuerdo con el estado del tag en la sección de **Value** y **Fill**. Luego, dar clic en **Apply** y en **Close**.



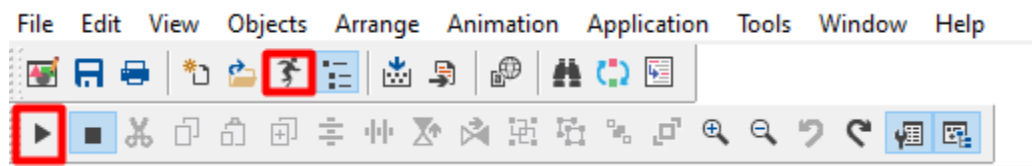
6. Finalmente, nuestro proyecto es como muestra la figura a continuación. El botón **Shutdown** está por defecto en el display main, cuya función permite salir de la aplicación en el PanelView y regresar al menú principal del mismo.







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



### Actividades por desarrollar

Detalle cual fue el procedimiento realizado para la sintonización fina, especifique que consideraciones se tuvieron en cuenta al momento de realizar el cambio en el setpoint.

Conclusiones

(Agregar mínimo tres conclusiones)

Recomendaciones

(Agregar mínimo tres recomendaciones respecto al desarrollo de la práctica, posibles mejoras)

### Bibliografía

Yang, Y. (2022, March 3). *Introduction to PID Control Loops*.

<https://info.erdosmiller.com/blog/introduction-to-pid-control-loops>

