Programación de pantallas táctiles

Descripción

Vamos a desarrollar la automatización de una máquina sencilla estampadora formada por 2 cilindros neumáticos. El primer cilindro se utiliza para posicionar y sujetar la pieza durante el estampado y es un cilindro neumático de doble efecto controlado por una electroválvula de 5/3 con una entrada para hacer avanzar el cilindro (A1) y otra para hacerlo retroceder (R1) también dispone de dos finales de carrera (F1) y (I1) para indicar su posición final e inicial respectivamente. El segundo cilindro se encarga de la estampación. Está formado por un cilindro de doble efecto controlado con una electroválvula 5/2 para hacer avanzar el cilindro se activa el solenoide del mismo (A2) cuando este está inactivo el cilindro retrocede.

Objetivo

El objetivo es automatización de la estampación de piezas mediante un PLC conectado a un HMI, controlando el número de piezas estampadas.

1 Funcionamiento

El operario introducirá en el HMI el número de piezas que quiere estampar y pulsará un botón de inicio en el HMI. Tras esto, el PLC moverá el cilindro 1 (A1) hasta la posición F1 y en esa posición bajará el cilindro 2 (A2) una vez realizada la estampación (F2) se recogen los cilindros y si el número de piezas a estampar es superior al realizado se repite el proceso. Visualizar en el HMI el número de piezas estampadas.
Objetivo: Repaso grafcet y Unity Pro. Introducción al Vijeo Designer y conexión al PLC

2 Opciones adicionales.

- Esquema de la instalación.
- Ver estado de los finales de carrera.
- Ver estado de las electroválvulas (movimiento del cilindro).
- Poder actuar sobre el cilindro (modo automático, paro y marcha).

Objetivo: Aprender a implementar las mismas funcionalidades que tendríamos con un sinóptico físico.

3 Opciones de animación.

Poder visualizar el movimiento de los cilindros neumáticos en el cuadro.
Objetivo: utilizar animaciones de posición, acciones y scripts.

4 Opciones avanzadas.

- Visualizar las piezas realizadas mediante un gráfico de barras.
- Visualizar la etapa en la que se encuentra el Grafcet.

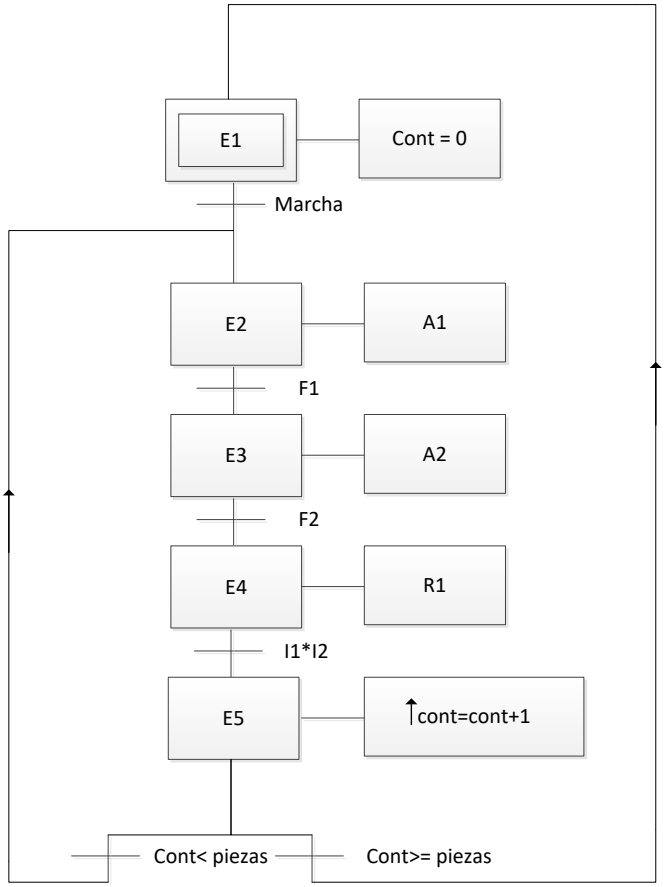
- Incluir una entrada de alarma en el sistema para detectar que cualquiera de los cilindros se ha atascado. La alarma debe de saltar si ha pasado cierto tiempo desde que se inició el movimiento del cilindro, pero no se ha llegado al sensor de destino.

Objetivo: utilizar alarmas, acceso a variables internas y gráficos de barras.

Solución guiada del problema:

Apartado 1:

Implementar el automatismo en el Unity Pro. Para ayudar se plantea siguiente grafcet



Una posible conexión de las entradas al autómata propuesta es:

I1	%I0.1.0
F1	%I0.1.1
I2	%I0.1.2
F2	%I0.1.3
A1	%Q0.1.16
R1	%Q0.1.17
A2	%Q0.1.18

El contador se puede implementar con el bloque de contador CTU de la librería de tipos o mediante una variable auxiliar definida previamente y sumando en la etapa correspondiente.

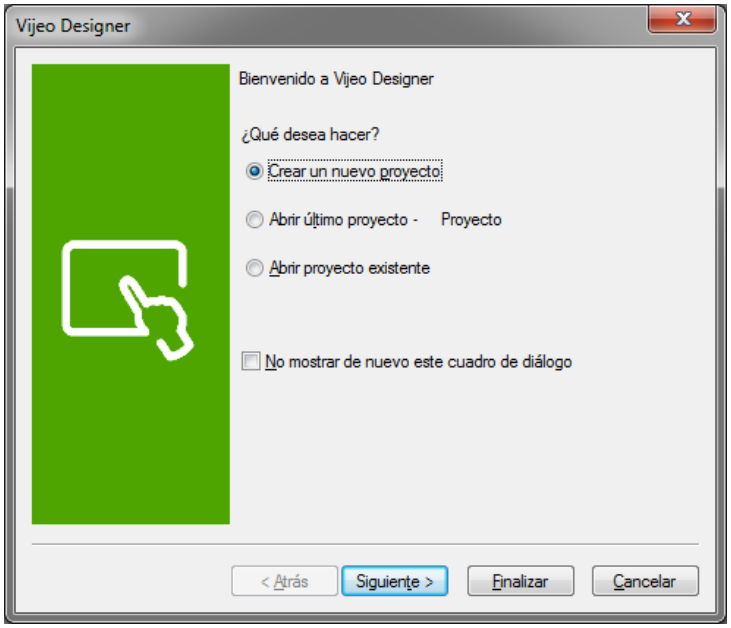
Se aconseja implementar el programa mediante una sección SFC

La variable **marcha** estará definida como una variable de tipo boobleano (BOOL) e interna. Y la variable **piezas** será una variable interna de tipo entero (INT).

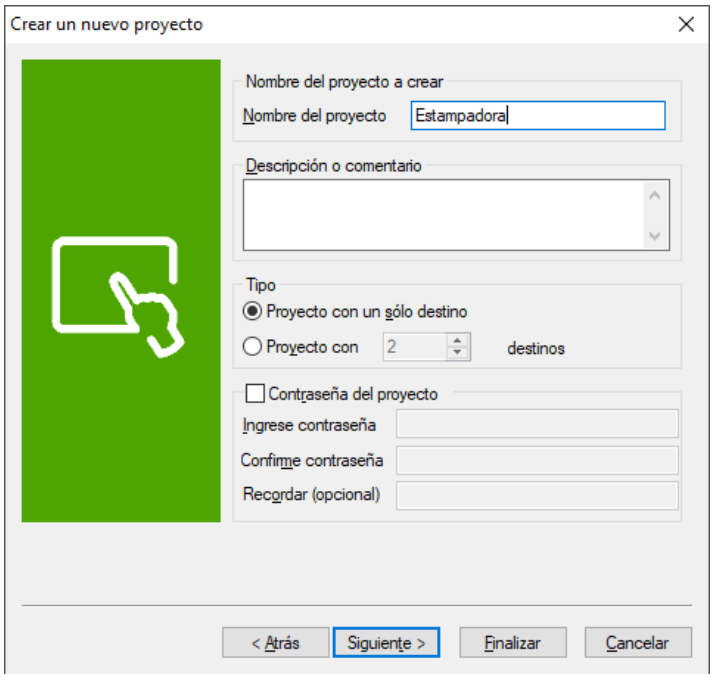
Estas variables se utilizarán para controlar el programa desde el terminal táctil.

En este paso se prueba el correcto funcionamiento del programa. Las variables marcha y piezas se introducen como valores en una tabla de animación del Unity Pro.

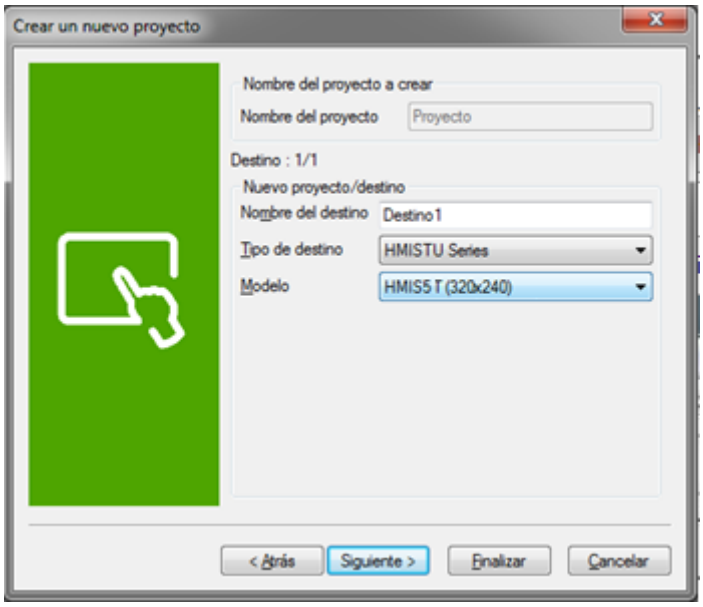
Cuando todo funcione correctamente pasamos a realizar el diseño del programa en el Terminal táctil con el vijeo-designer. Para ello hay que abrir la aplicación vijeo-designer y seguimos los pasos detallados a continuación.



Seleccionamos “crear nuevo proyecto”

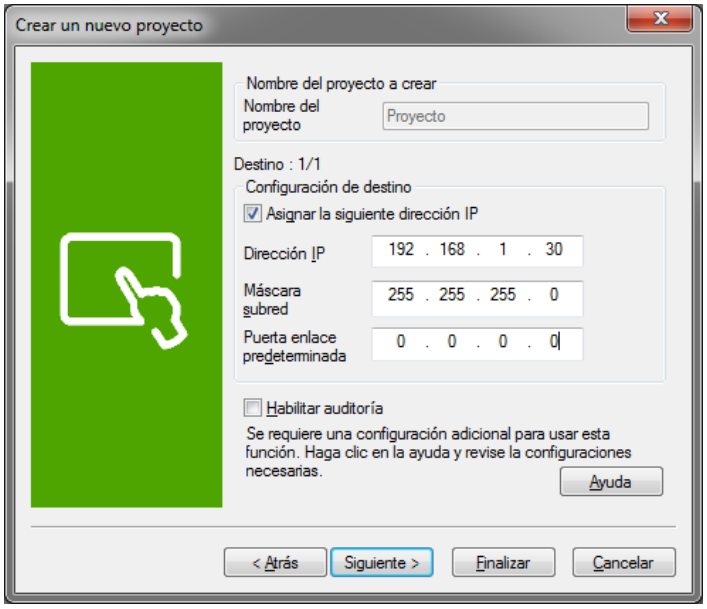


Nombre del proyecto: “Estampadora”

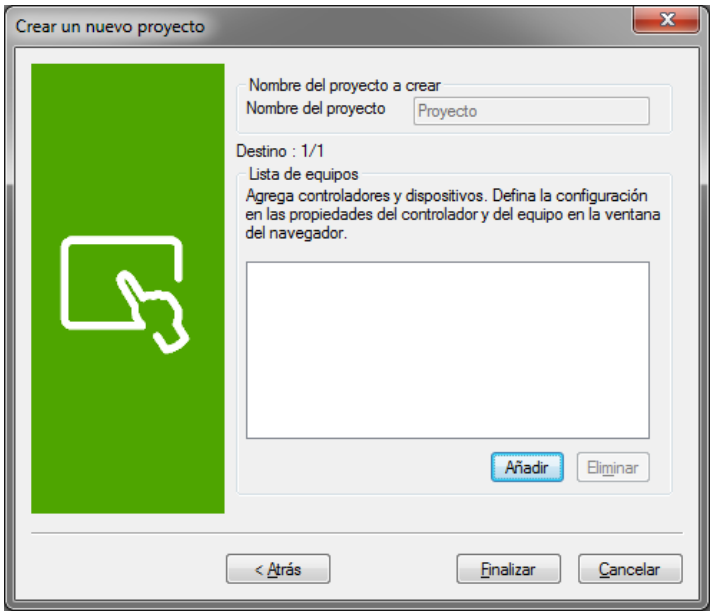


Seleccionamos el modelo del terminal táctil HMISTU -> HMIS5T

Comprobar el modelo del terminal en la placa del mismo.

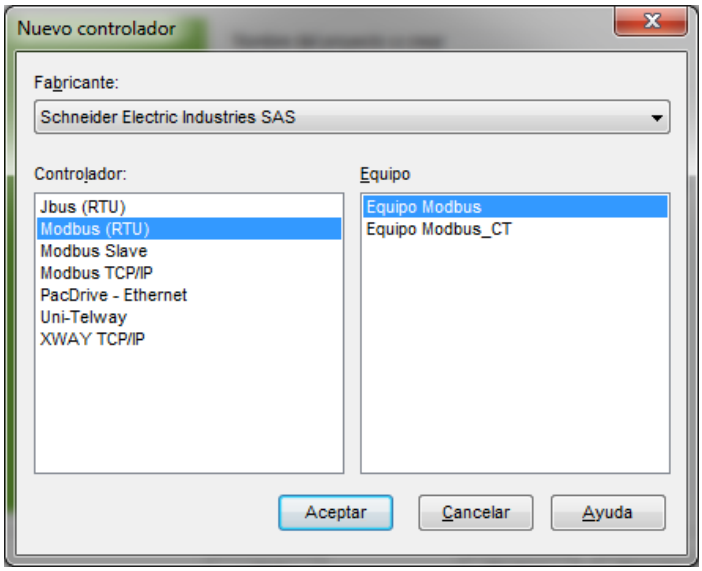


Introducimos la dirección IP que queremos que tenga el terminal táctil



Ahora nos pregunta si queremos añadir un destino: Esto es un PLC u otro dispositivo similar del que queramos obtener datos.

Seleccionamos Añadir para incluir nuestro PLC

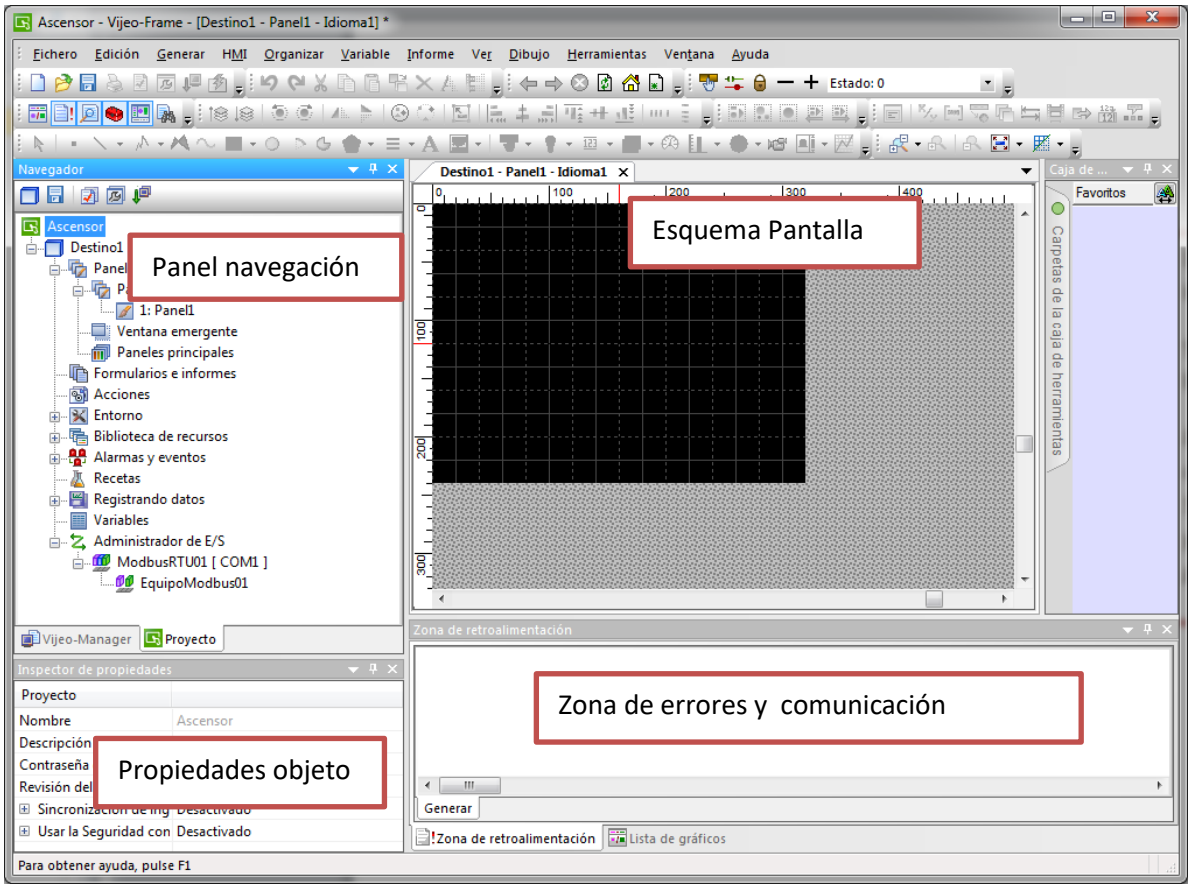


Del fabricante Schneider Electric Industries SAS

Seleccionamos Modbus(RTU) -> Equipo Modbus

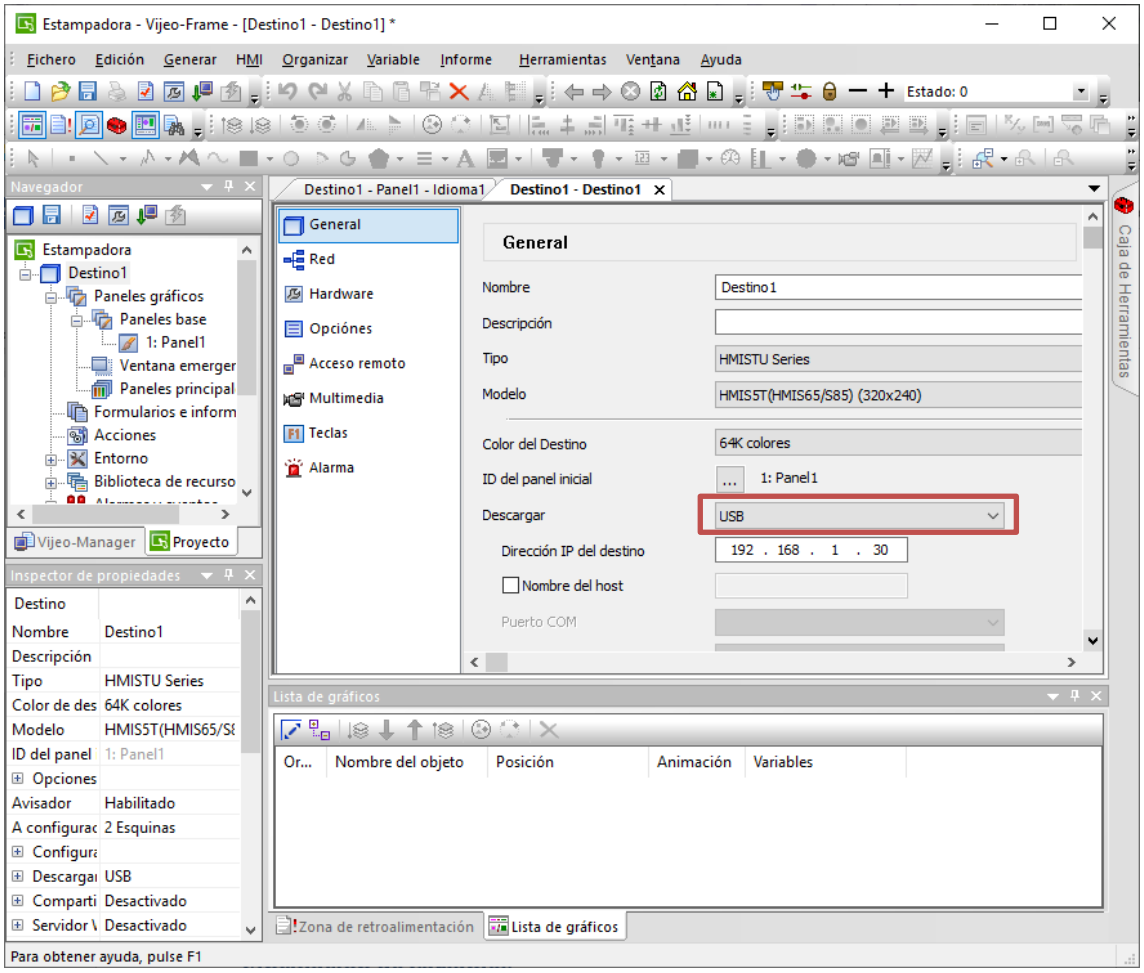
Esto indica que nos comunicaremos con un PLC de Schneider con protocolo Modbus y mediante comunicación serie puerto (COM)

Vista general del programa Vijeo designer



El panel de navegación es la parte más importante del programa y nos permite acceder a los diferentes apartados del proyecto:

Destino1 Mostrará todas las propiedades de la pantalla táctil: Su configuración, alarmas comunicación, red, acceso remoto…

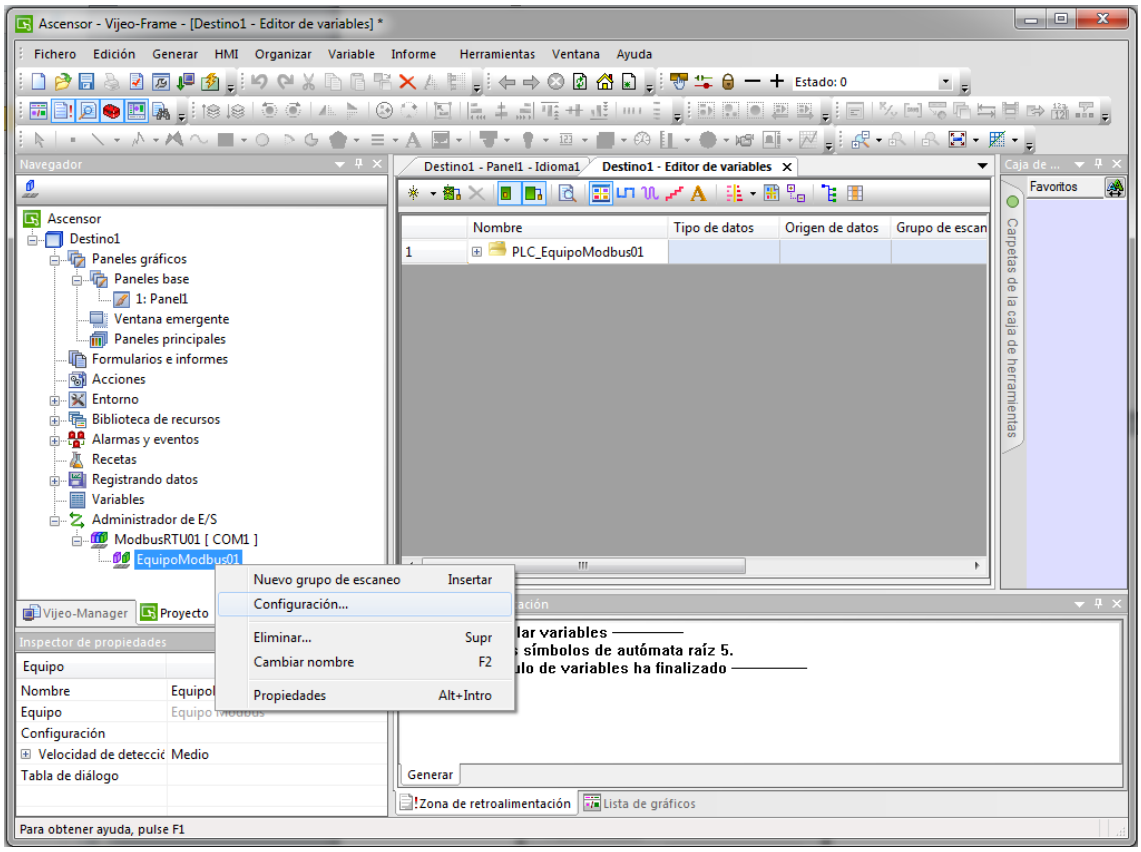


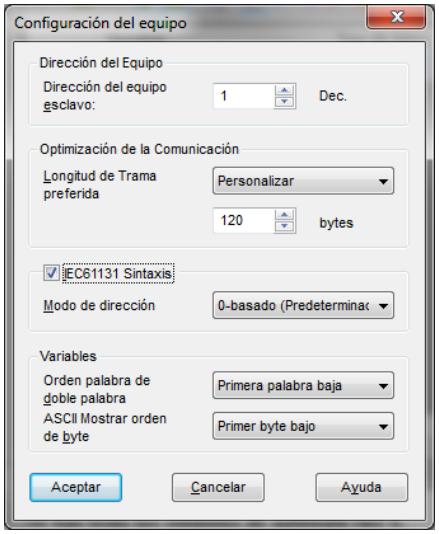
Tenemos que indicar que la forma de descargar el programa va a ser mediante USB.

También tenemos en el panel de navegación opciones para:

- dibujar los diferentes paneles gráficos que se van a mostrar en el programa.
- Impresión de informes.
- Automatizar acciones.
- Definir variables de entorno: Contraseñas e Idiomas.
- Definir recursos: un recurso es una variable que define una propiedad. Color, Imagen, Texto o tipo de letra. Podemos definir el recurso *ColorBotonMarcha* con las tonalidades que queramos y luego en el panel usar este nombre para asignar a un botón u otro objeto este valor. Recetas: Permite guardar una tabla con diferentes configuraciones de la máquina que se correspondan a distintos productos.
- Registro de dados: Permite guardar en los valores de las variables para visualizarlos en gráficos temporales y exportarlos para analizarlos posteriormente.
- Variables: En este apartado se definen las variables de la pantalla y si están o no enlazadas con un PLC.
- Administrador de E/S: Aquí se configuran los diferentes PLCs conectados a la pantalla. Su comunicación, dirección y protocolo.

En el panel de navegación vamos a configurar las comunicaciones para que coincidan con la forma en la que envía los datos el PLC. Pinchamos en EquipoModbus01 con el botón derecho y entramos en configuración.





Dirección 1

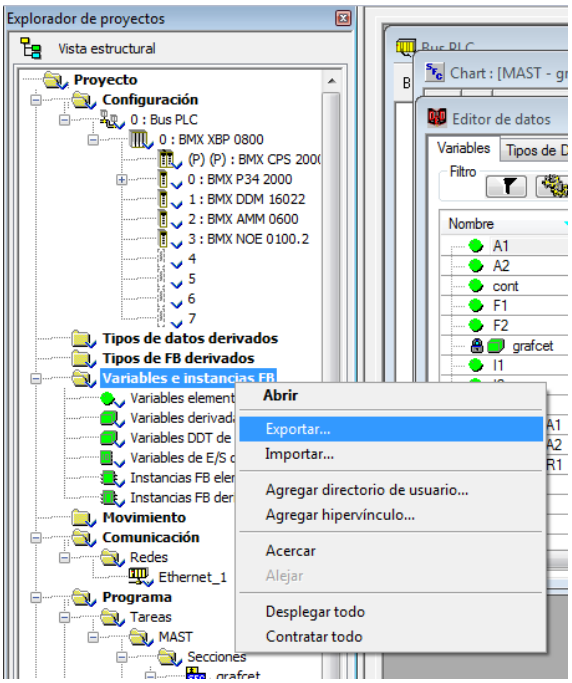
Marchamos IEC61131

Primera palabra baja

Vincular las variables del PLC al vijeo-designer.

Dado que muchos autómatas y en concreto los de Schneider suelen trabajar con variables definidas por el usuario sin fijar su dirección de memoria en el PLC, resulta complicado saber tras cada compilación del programa del PLC, en qué dirección de memoria se encuentran las variables importantes.

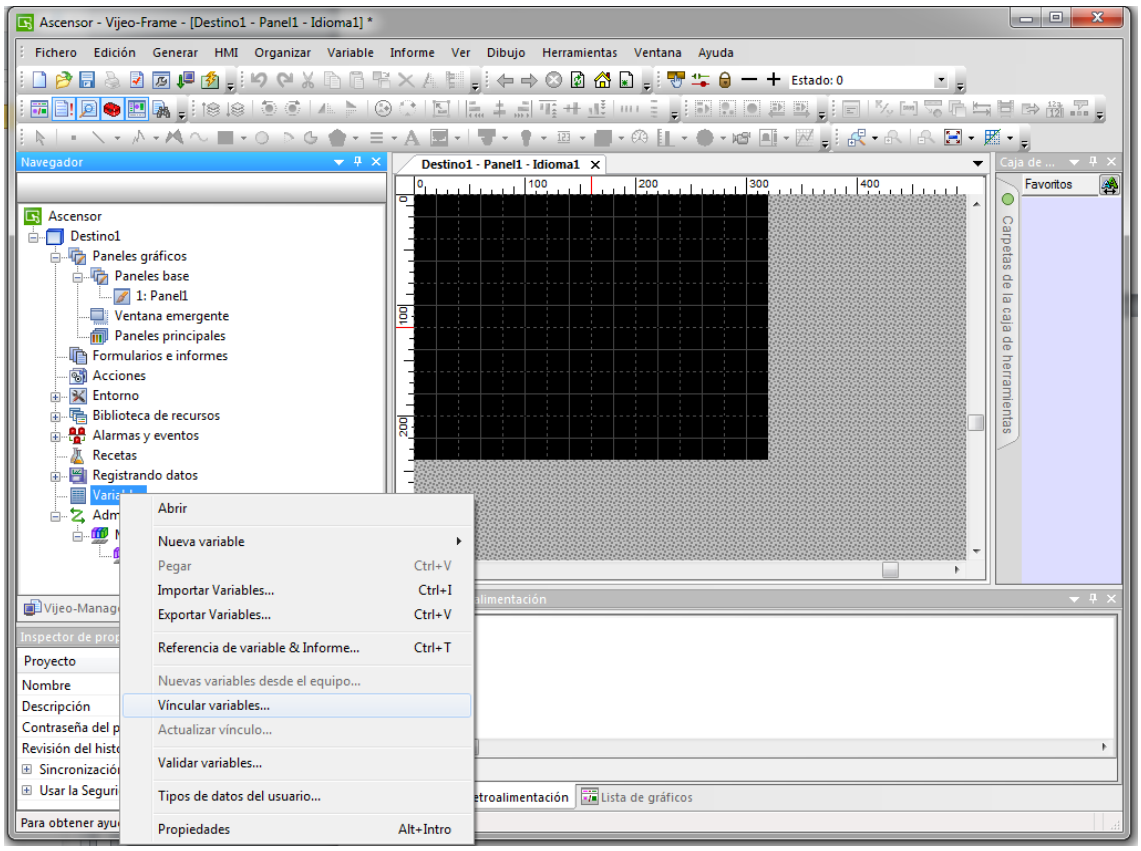
Desde el **Unity Pro** tras compilar y enviar el programa al PLC podemos guardar la información de los nombres y direcciones de las variables para utilizarlos en el vijeo-designer. Para hacer esto hacemos click con el botón derecho en Variables e instancias y seleccionamos Exportar…:

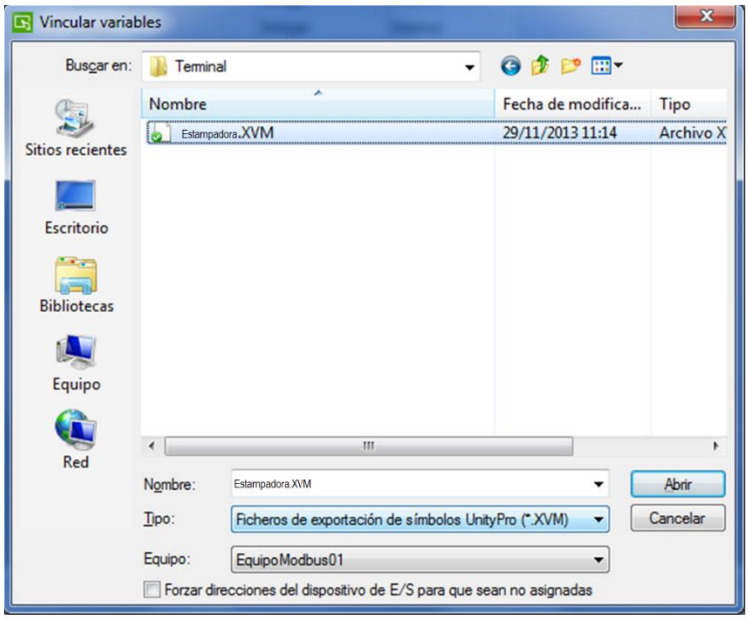


Guardamos las variables en un fichero con extensión *.XVM : Estampadora.xvm.

Este fichero guarda una relación entre los nombres de variables y la dirección de memoria en la que se almacenan dichas variables en el PLC. Hay que tener en cuenta que tras cada compilación – envío de modificaciones del programa del PLC estas direcciones cambian. Con lo que hay que volver a exportar las variables cada vez que se modifique el programa del PLC en el UNITY PRO.

El vijeo-designer permite leer estas variables mediante la opción vincular variables. Que se activa mediante el botón derecho sobre “variables” en el panel de navegación

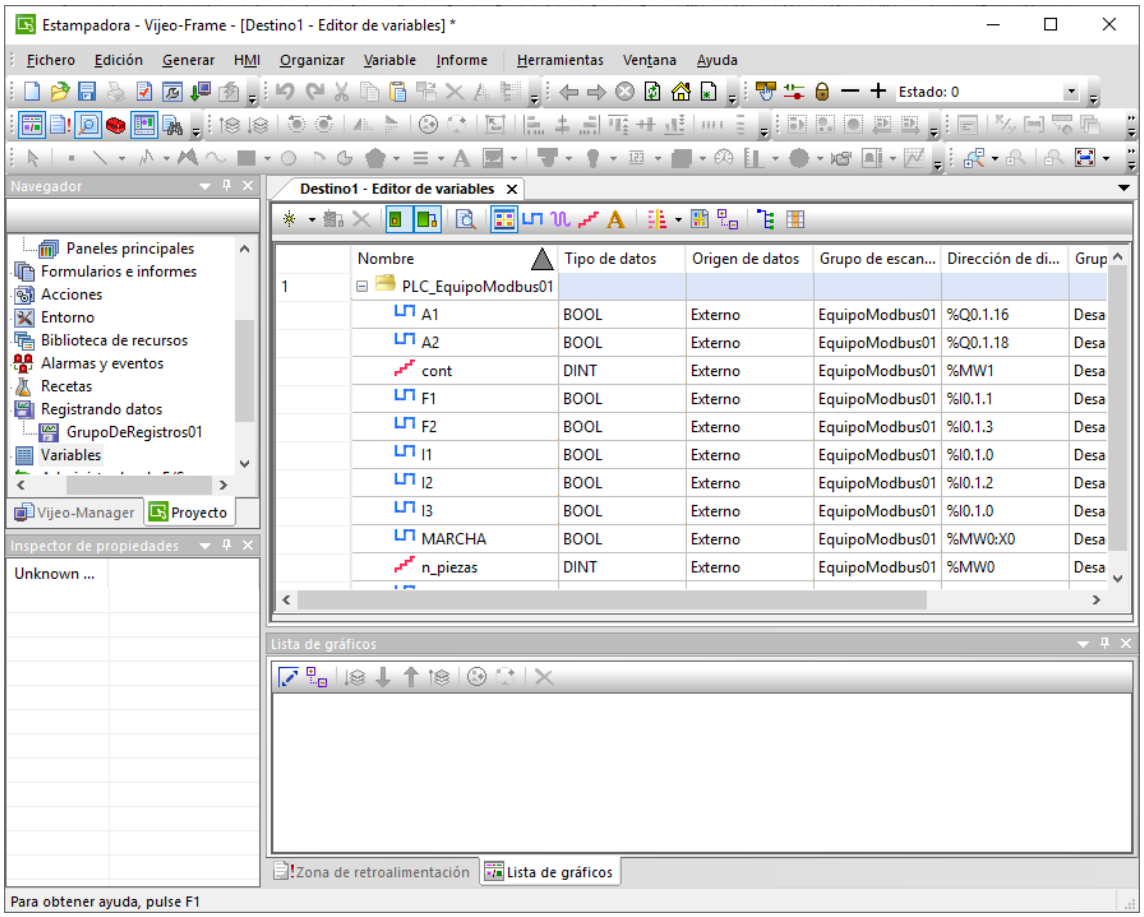




Seleccionamos el fichero Estampadora.xvm que hemos creado anteriormente (ya compilado)

Seleccionamos todas las variables y pulsamos en añadir.

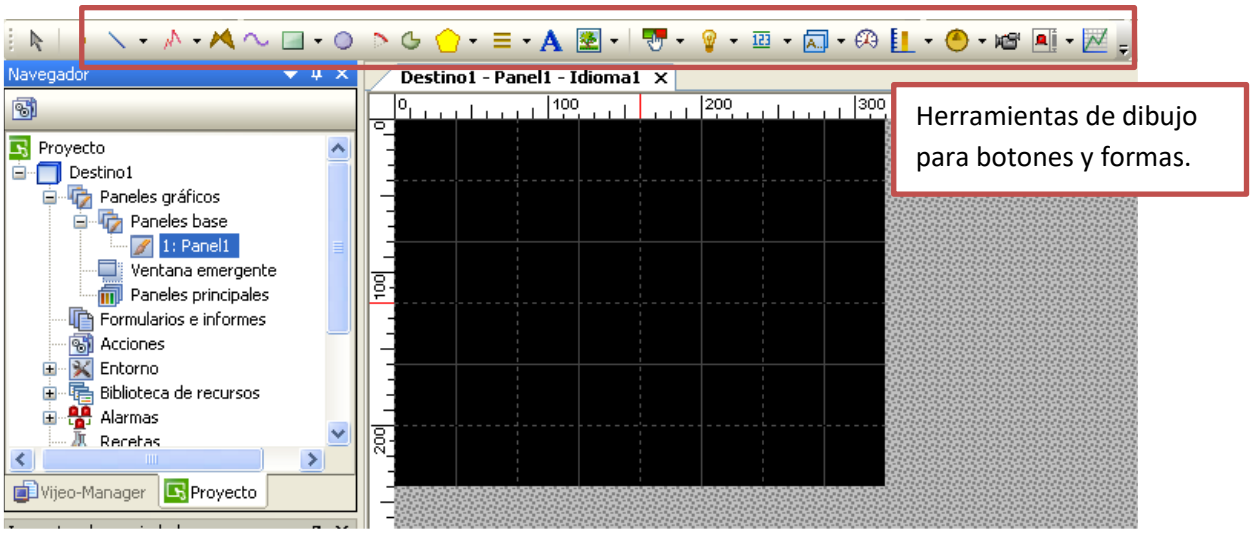
En el apartado “Variables” del panel de navegación podemos ver ahora todas las variables de nuestro programa del PLC. Además podemos crear las variables internas que queramos.



Creación del panel

Ahora vamos a dibujar el panel de la estampadora. Debe de quedar al finalizar el proceso de esta forma:

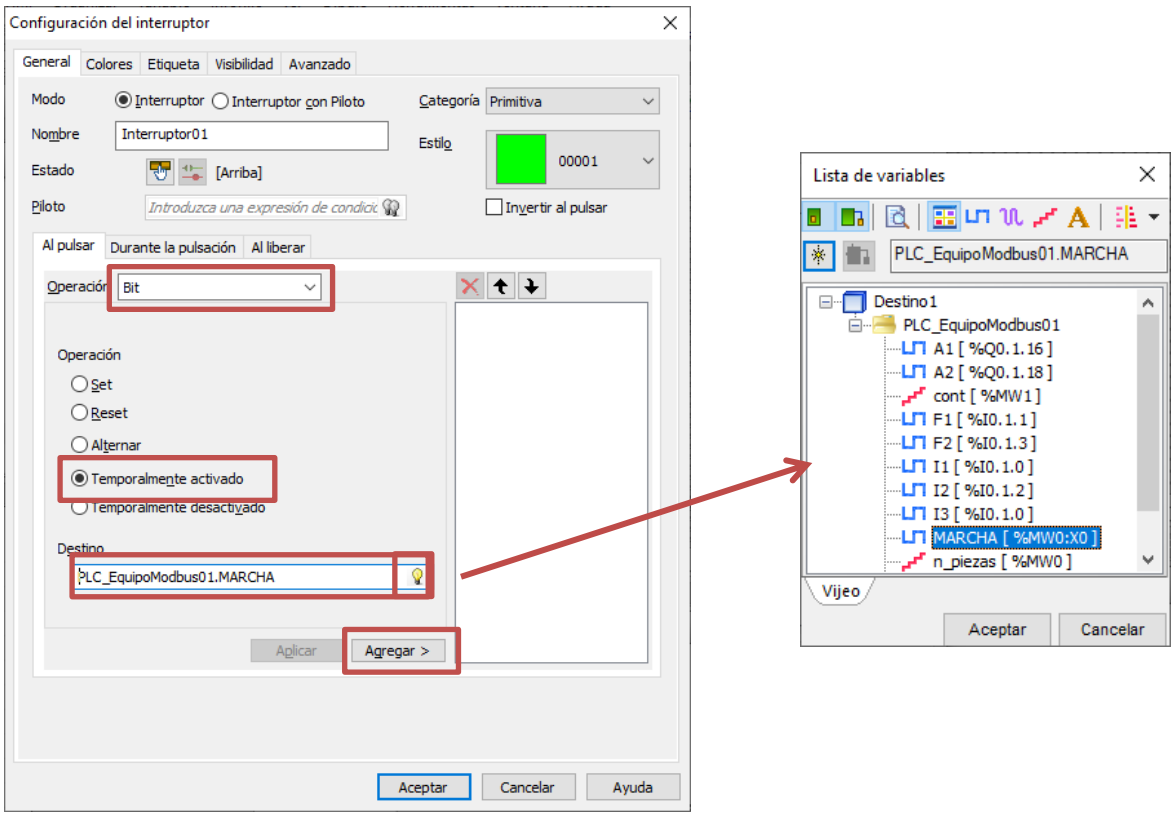
Por defecto, el programa nos crea un panel titulado “Panel1”. Podemos encontrarlo en “Paneles gráficos” → Paneles base → Panel 1.



Nos aparece el espacio disponible del panel, en él podemos añadir objetos como: Dibujos, textos, imágenes, botones, indicadores y gráficos.

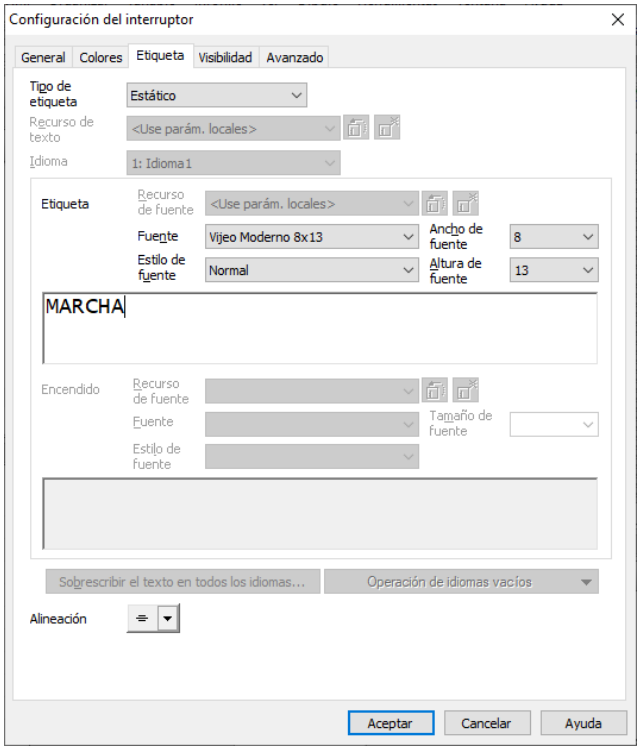
Añadimos el texto mediante el botón "ESTAMPADORA", "Número de piezas:" y "Realizadas:"

Para añadir el botón de marcha, hacemos clic en la herramienta y lo colocamos en el espacio disponible del panel. Se abre una ventana con las propiedades:

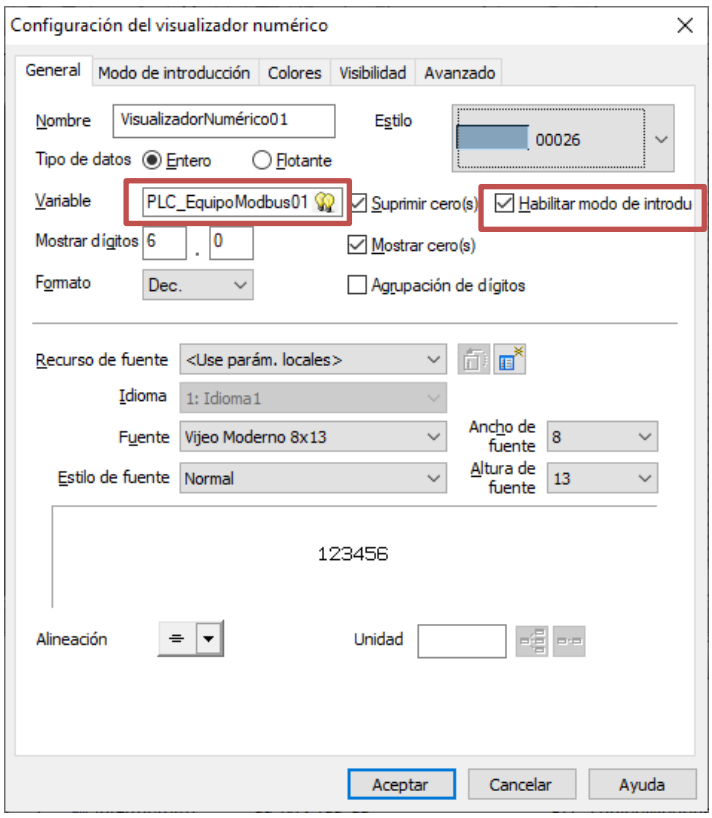


Seleccionamos la operación Bit. Temporalmente activado. Y en Destino pulsamos en la bombilla y seleccionamos la variable Marcha. Finalmente pulsamos a Agregar. La acción aparecerá en la lista lateral.

Para que en el Botón aparezca la palabra Marcha vamos a la pestaña etiqueta -> tipo de etiqueta Estático y escribimos MARCHA



Para introducir las piezas a realizar en el terminal seleccionamos visualizador numérico.



Pulsar en la bombilla y seleccionar la variable "piezas".

Para poder modificarlas *habilitar modo de introducción*

Repetir e mismo proceso pero sin habilitar el modo de introducción.

Para ajustar la posición de los difererentes objetos: Texto, botones y entradas muméricas. Se selecciona el objeto con el ratón y mediante las teclas de arriba y abajo podemos moverlos por el panel.

Compilar y transferir el proyecto al panel.

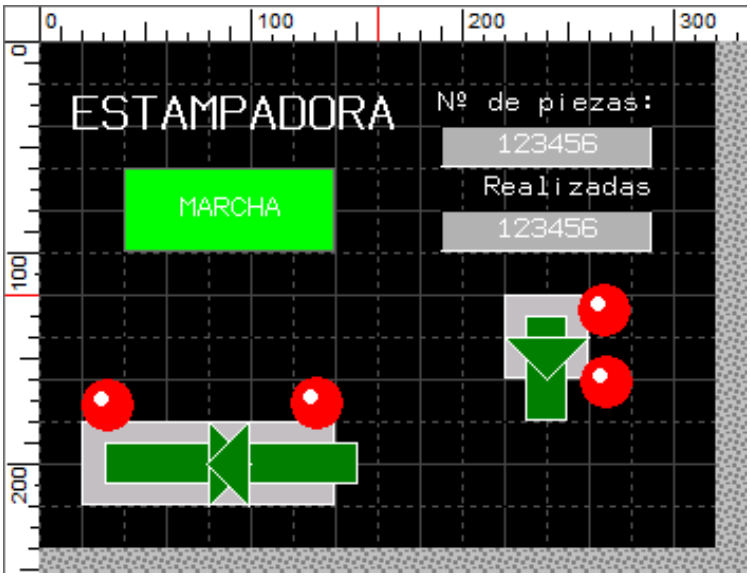
Conectamos el panel por USB al PC. Se puede descargar el proyecto directamente con el botón

Por otro lado, el programa ofrece opciones de testeo del programa y compilación previa que nos van a ayudar a encontrar errores.

Por último comentar que se puede simular el comportamiento del panel con lo que junto con la simulación del PLC permite simulaciones previas a la instalación del equipo.

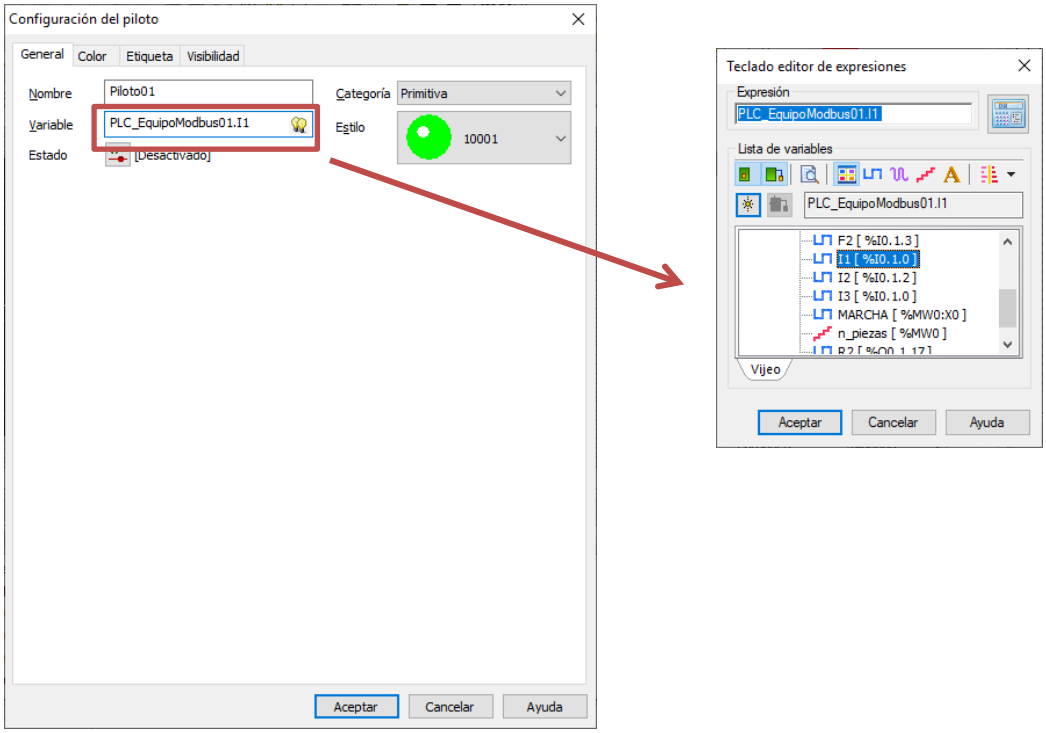
Apartado 2:

Modificamos el Panel hasta tener un esquema de la instalación, similar a:

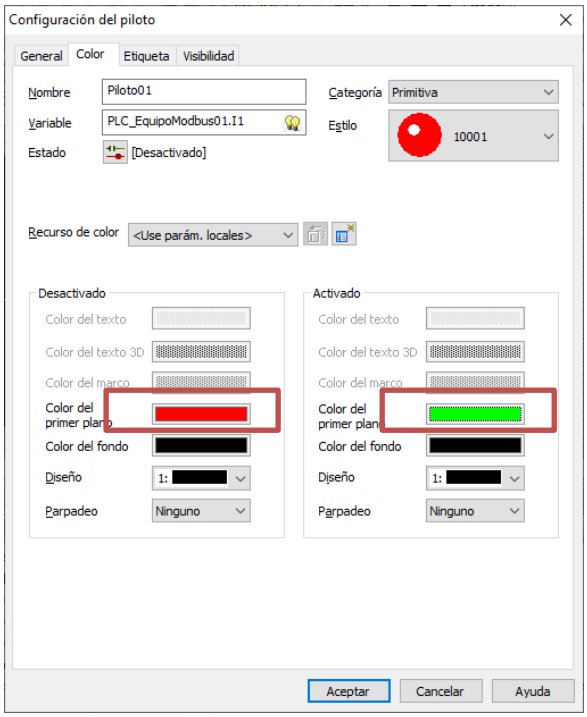


Los cilindros neumáticos se han dibujado con dos rectángulos uno gris y el otro verde.

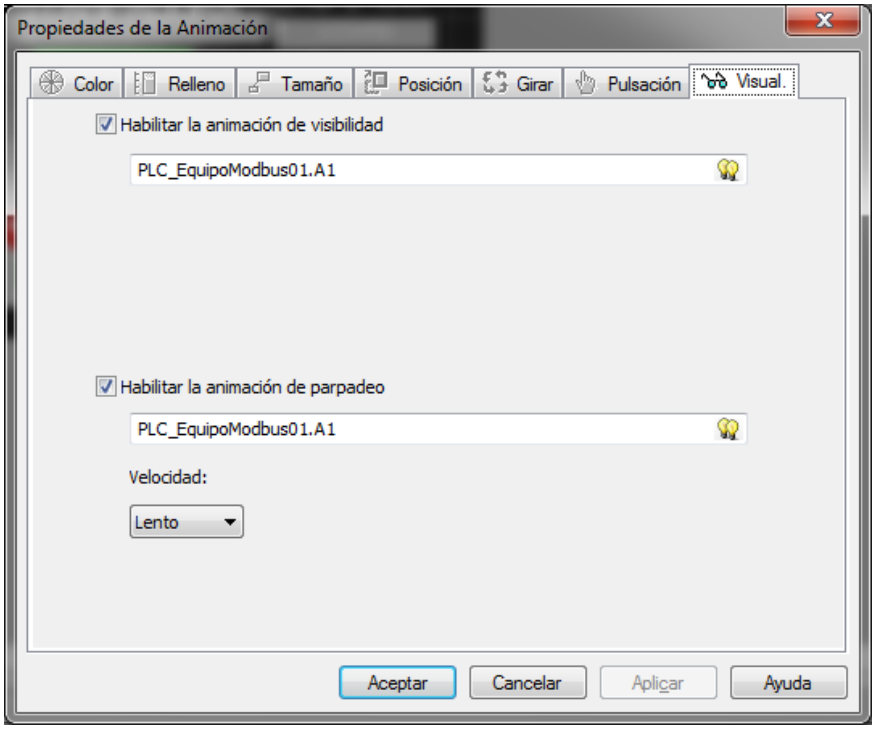
Los finales de carrera son pilotos a los que les asignamos las variables (I1,F1,I2,F2) y los colores: rojo desactivado y verde activado.



Seleccionamos la pestaña color para que el color rojo sea el valor desactivado y el verde el Activado.



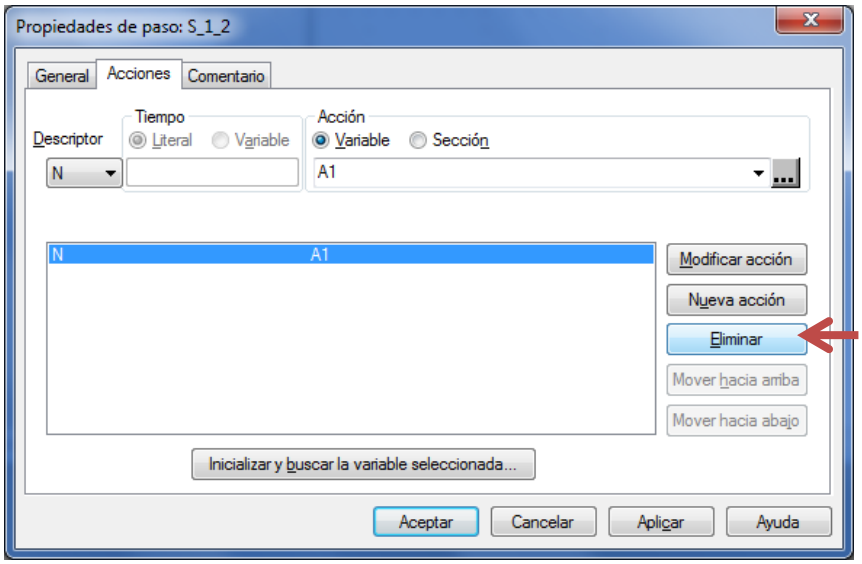
Los triángulos son polígonos a los que se les ha introducido una animación de visualización tanto de visibilidad como de parpadeo asociada a las variables de movimiento de los cilindros



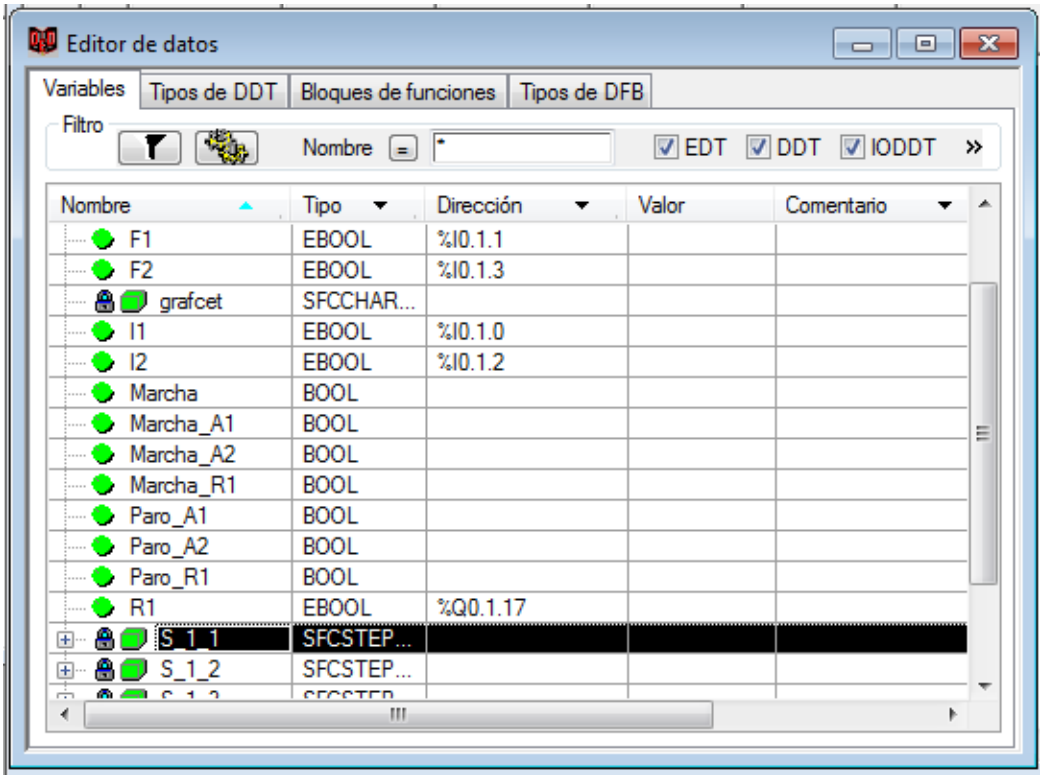
Probamos el funcionamiento de estos elementos en el Terminal táctil.

Control Manual de los cilindros

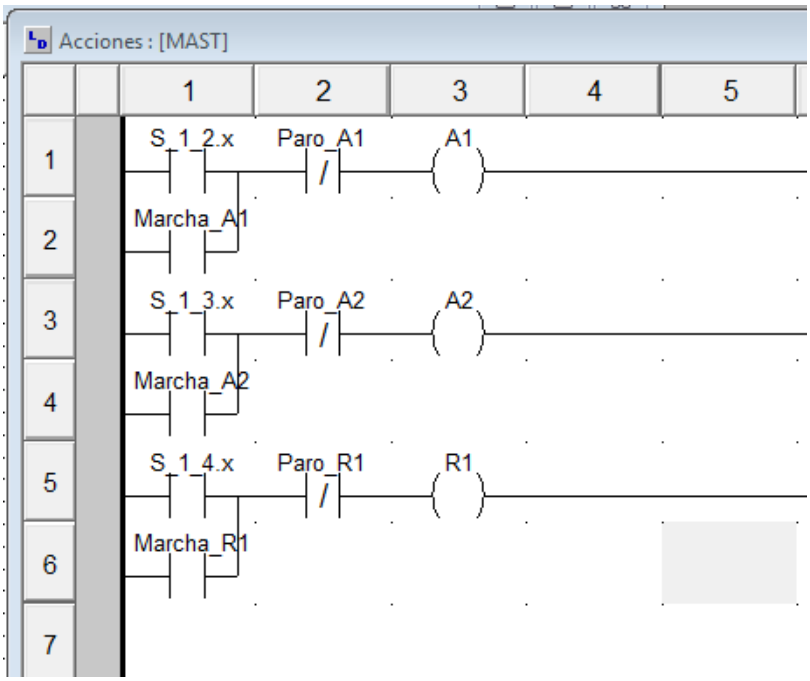
Tenemos que modificar el programa del PLC. Se aconseja eliminar de la sección SFC creada para realizar el programa de estampación todas las acciones asociadas a las etapas que activen (A1, R1 y A2). Y la implementación de estos accionamientos se realizará mediante diagrama de contactos (LD).



Una vez hemos eliminado todas las acciones, creamos las variables auxiliares para activar y parar los accionamientos Marcha_A1, Marcha_A2, Marcha_R1, Paro_A1 Paro_A2, Paro_R1



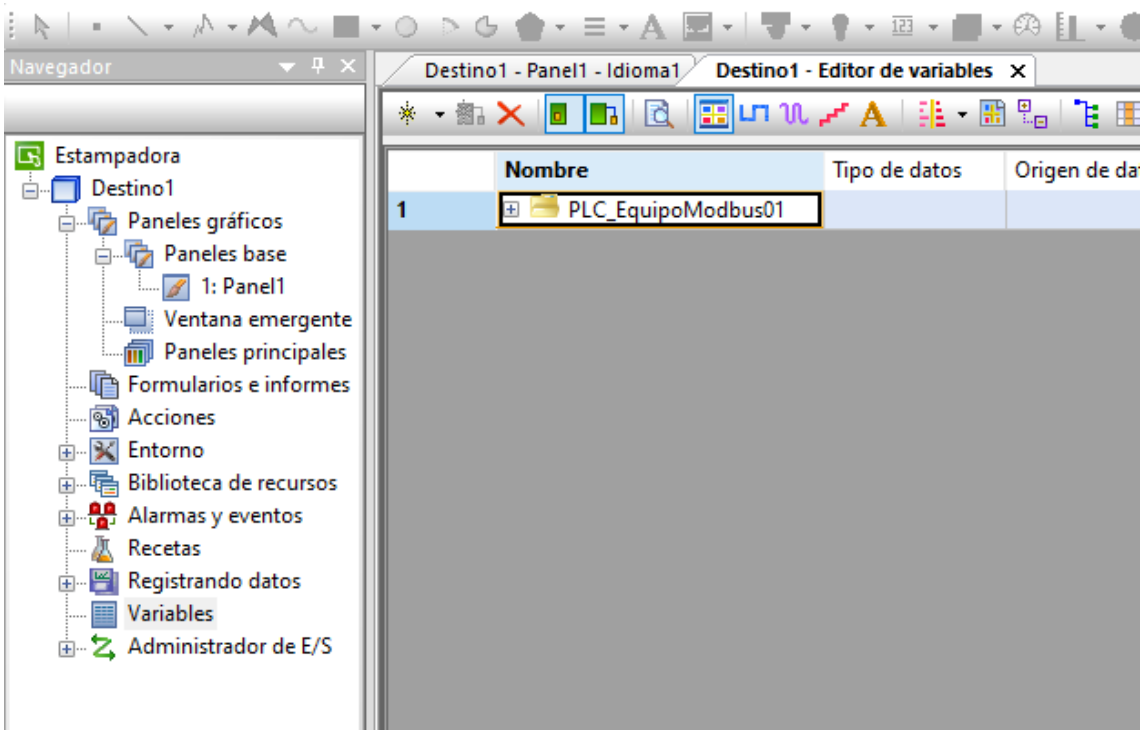
Creamos una nueva sección LD **Acciones** para implementar todas las activaciones utilizando el conocimiento de los estados del grafcet mediante las variables S_1_2.x que se activa cuando la etapa 2 del grafcet está activa



Como se puede observar con esta modificación podemos detener el cilindro (**A1**) si se activa la variable **Paro_A1** y si esta no está activada con **Marcha_A1** podemos activarlo. Si ninguna de ellas está activa entonces la salida se activa si **S_1_2.x** está activa, tal y como se indica en el grafcet.

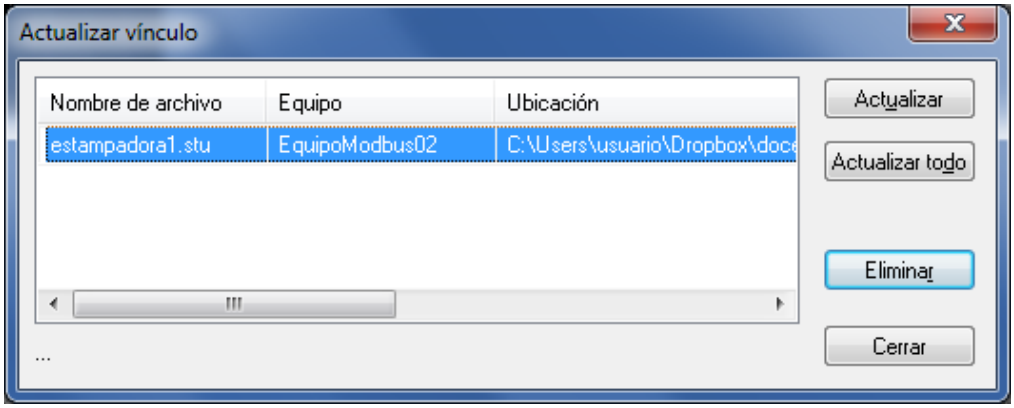
Una vez realizada esta modificación compilamos, grabamos y enviamos el nuevo programa al PLC. Y exportamos las variables al fichero “Estampadora.XVM” igual que antes.

Ahora hay que modificar el programa del Terminal táctil. Abrimos el Vijeo-Designer. Ahora lo que ha cambiado es la existencia de nuevas variables. Por lo tanto debemos de vincularlas al Vijeo-Designer. Primero en variables eliminamos las variables viejas



Clicando con el botón derecho en PLC_EquipoModbus01 y seleccionando eliminar.

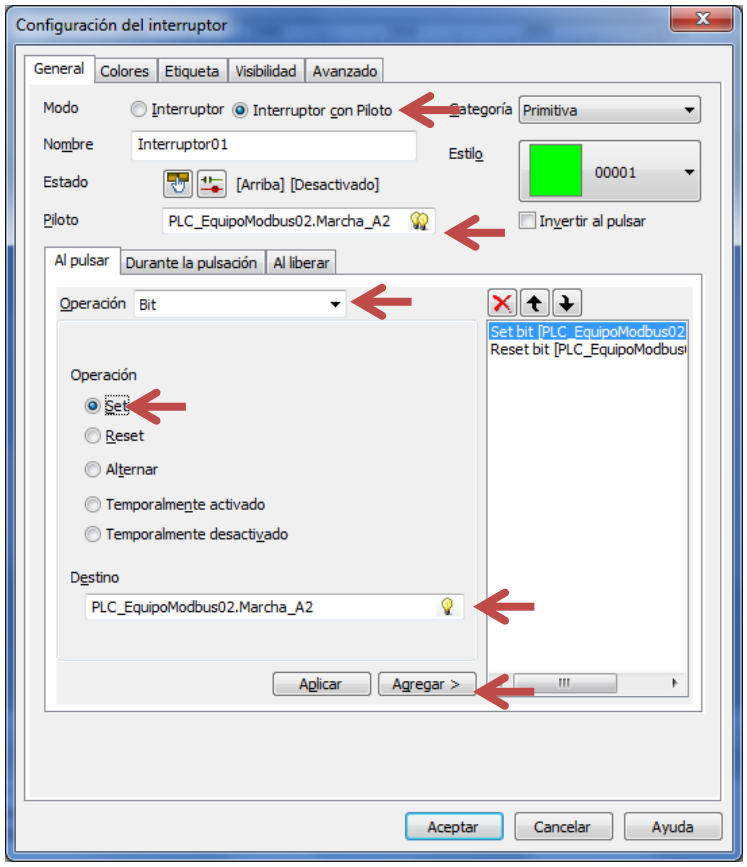
Posteriormente con el botón derecho sobre variables en el panel de navegación. Seleccionamos Actualizar vinculo y en la ventana emergente eliminar



Una vez hemos eliminado el vínculo lo volvemos a crear e importamos todas las variables del fichero “Estampadora.XVM”.

Ahora vamos a utilizar una posibilidad del vijeo-Designer que es la creación de ventanas emergentes. En el **navegador** seleccionamos ventana emergente y seleccionamos la opción nueva ventana emergente. Se crea un panel en blanco que vamos a modificar hasta que tenga un aspecto similar a:

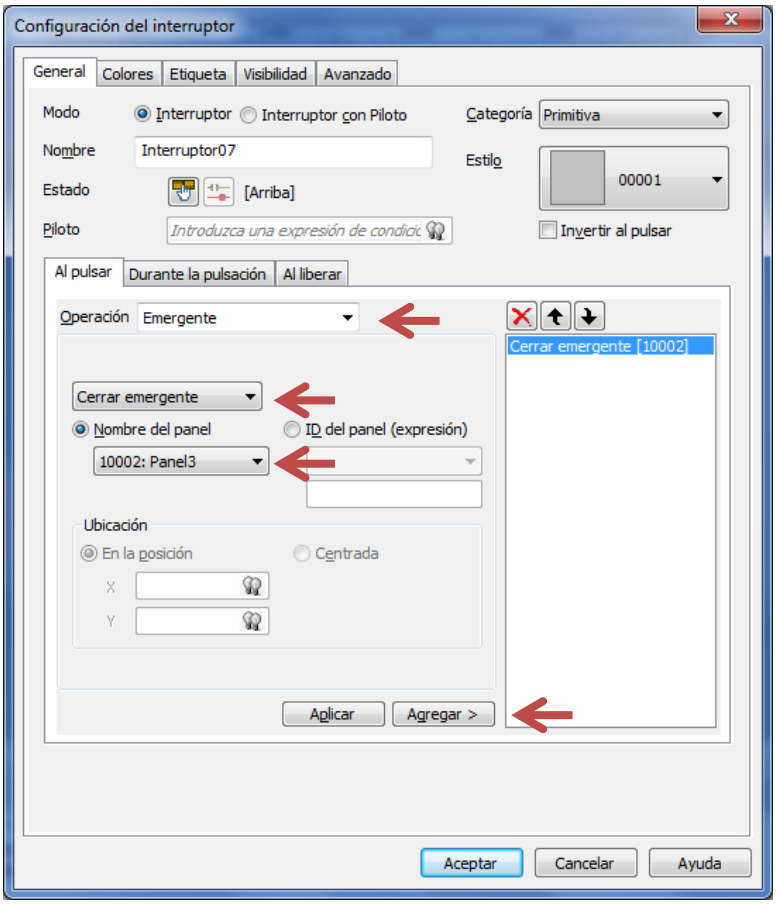
Se han introducido 3 botones para controlar completamente el cilindro 2. El botón de marcha hace un set de la variable Marcha_A2 y un reset de Paro_A2.



Faltaría hacer el Reset de Paro_A2. Para ver si Marcha_A2 está activado lo añadimos en piloto. Seleccionamos posteriormente los colores que queramos en la pestaña colores.

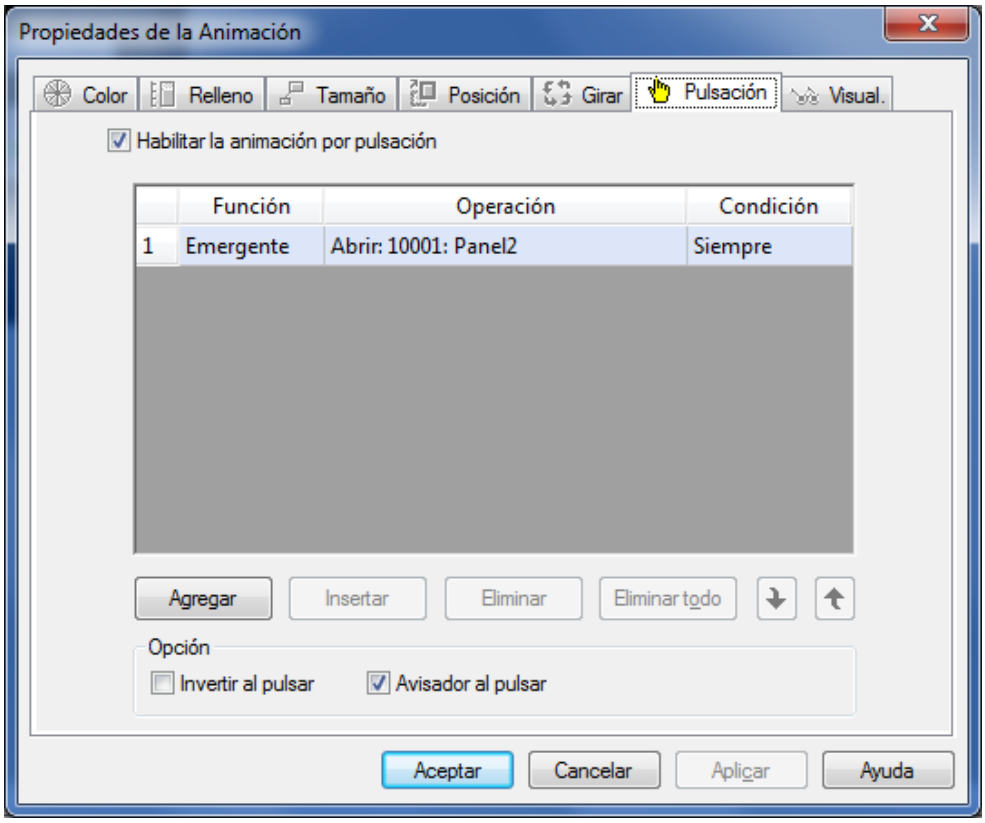
El botón Paro debe de hacerun set de la variable Paro_A2 y un reset de la variable Marcha_A2. Por último el botón Auto hace un reset de ambas variables Marcha_A2 y Paro_A2. De esta forma se consigue controlar el comportamiento del cilindro. Por ejemplo esta pantalla muestra la configuración del botón de marcha para hacer el set de Marcha_A2:

Por último añadimos un botón en la esquina superior izquierda para cerrar el panel con una x como etiqueta. Para indicarle que se cierre el panel debemos de seleccionar:



Hacemos un panel emergente similar para el cilindro 1, pero ahora con 6 botones. 3 para Avanzar y 3 para retroceder. Igualmente un botón para cerrar la ventana.

Ahora hay que añadir una animación que active la ventana emergente del cilindro. Para ello hacemos doble clic sobre el dibujo del cilindro y Habilitamos la animación por pulsación y añadimos la función Emergente del panel que deseemos abrir

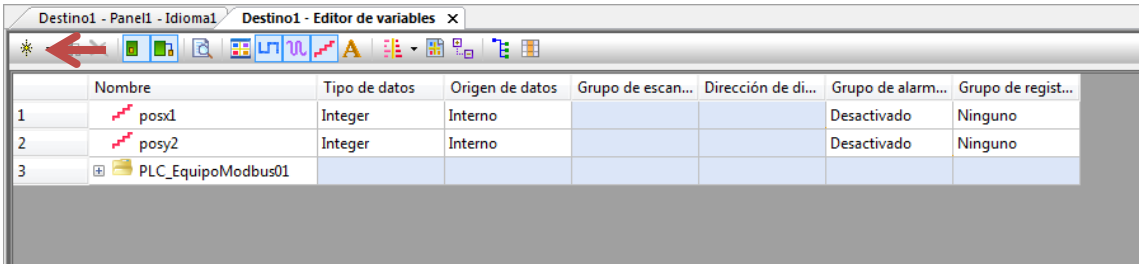


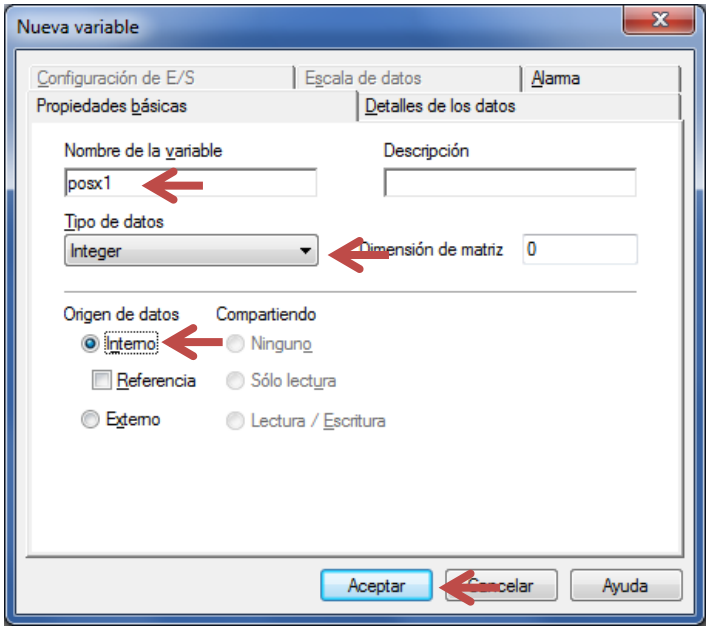
Así con los dos cilindros. Compilamos y enviamos el programa al Terminal. Testeamos el correcto funcionamiento de los accionamientos de Marcha y paro.

Apartado 3: Animaciones

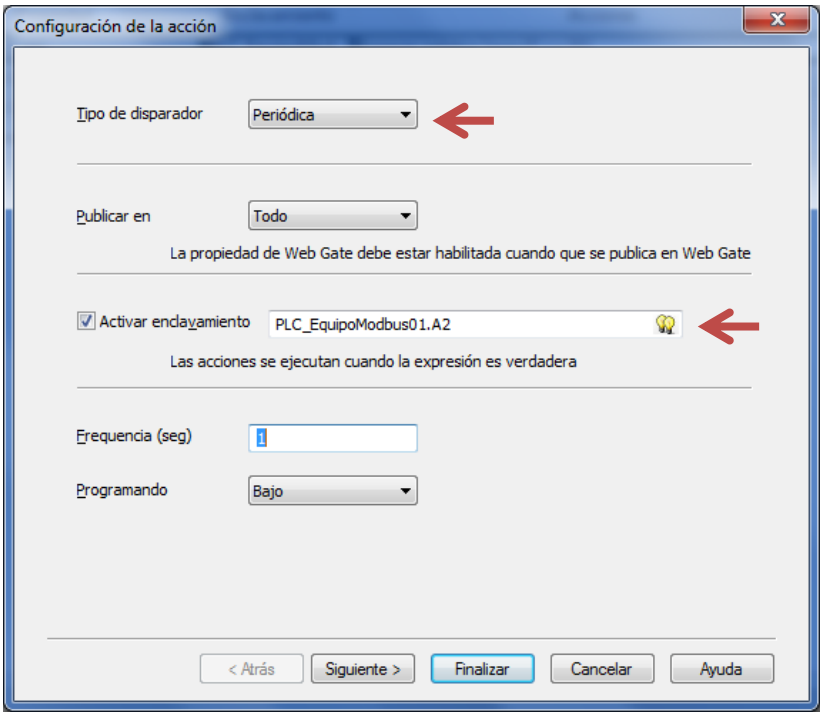
Ahora queremos visualizar el movimiento de los cilindros y ver como estos de desplazan en el panel.

Para ello vamos a utilizar lar variables internas del Terminal táctil. En concreto vamos a crear dos variables enteras posx1 y posy2 para indicar la posición horizontal del cilindro 1 y la vertical del cilindro 2. Hacemos doble clic sobre variables en el navegador y las creamos

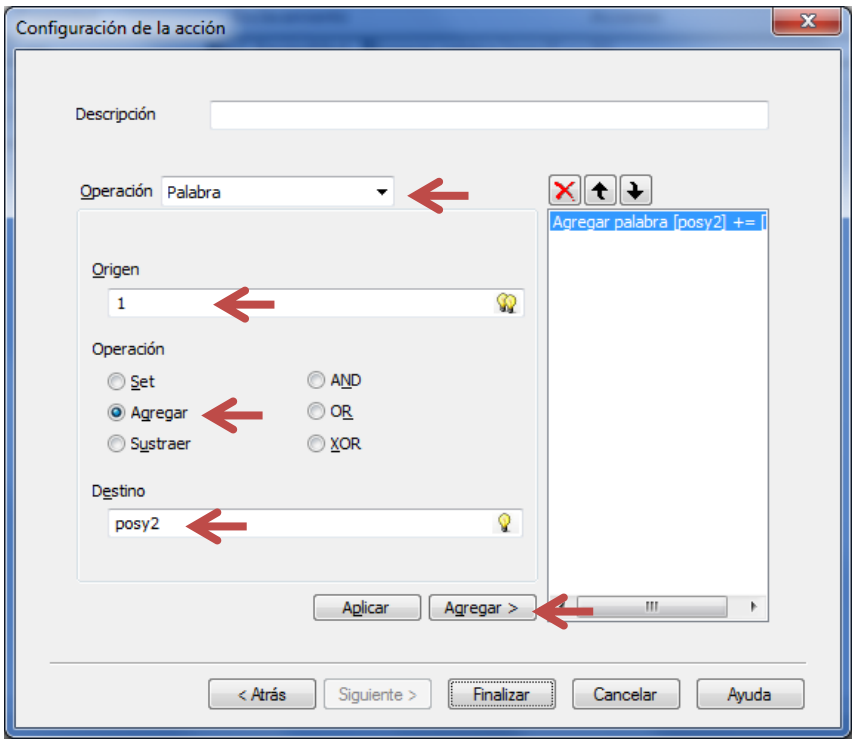




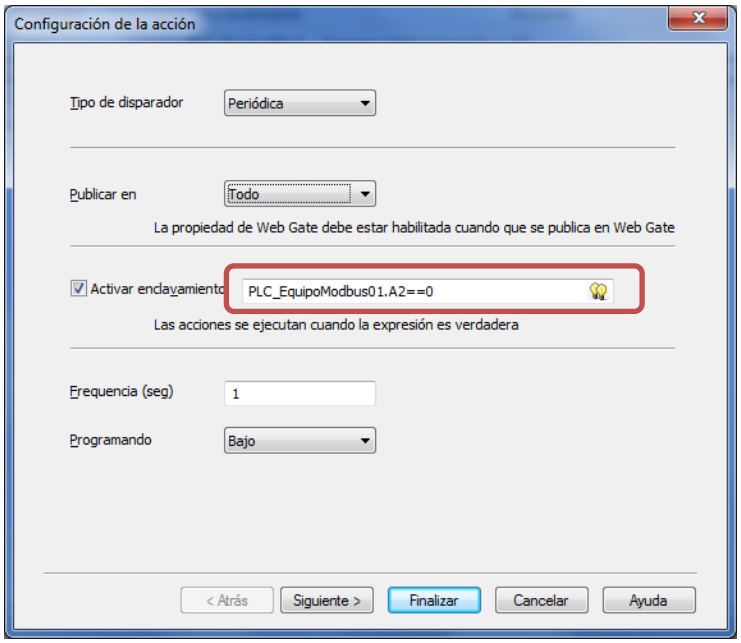
Ahora queremos que ambas variables almacenen la posición del cilindro. Para ello creamos una acción. En el navegador seleccionamos **Acciones** y con el botón derecho seleccionamos **nueva acciones** En este caso queremos que la variable posy2 se incremente cuando esté activa la variable **A2** del autómata.



Primero seleccionamos el tipo de disparador. Lo hacemos periódico cada 1 segundo y con la condición de tener activo A2 así sabemos que la acción se ejecuta cada segundo únicamente cuando tenemos activo A2. Ahora decimos que se incremente el valor de la variable posy2 en una unidad



Así cuando A2 está activo la variable posy2 se va incrementando cada segundo. Ahora como es un cilindro de simple efecto tenemos que sustraer 1 al valor de posy2 cuando A2 no esté activo. Para ello creamos una nueva acción periódica:

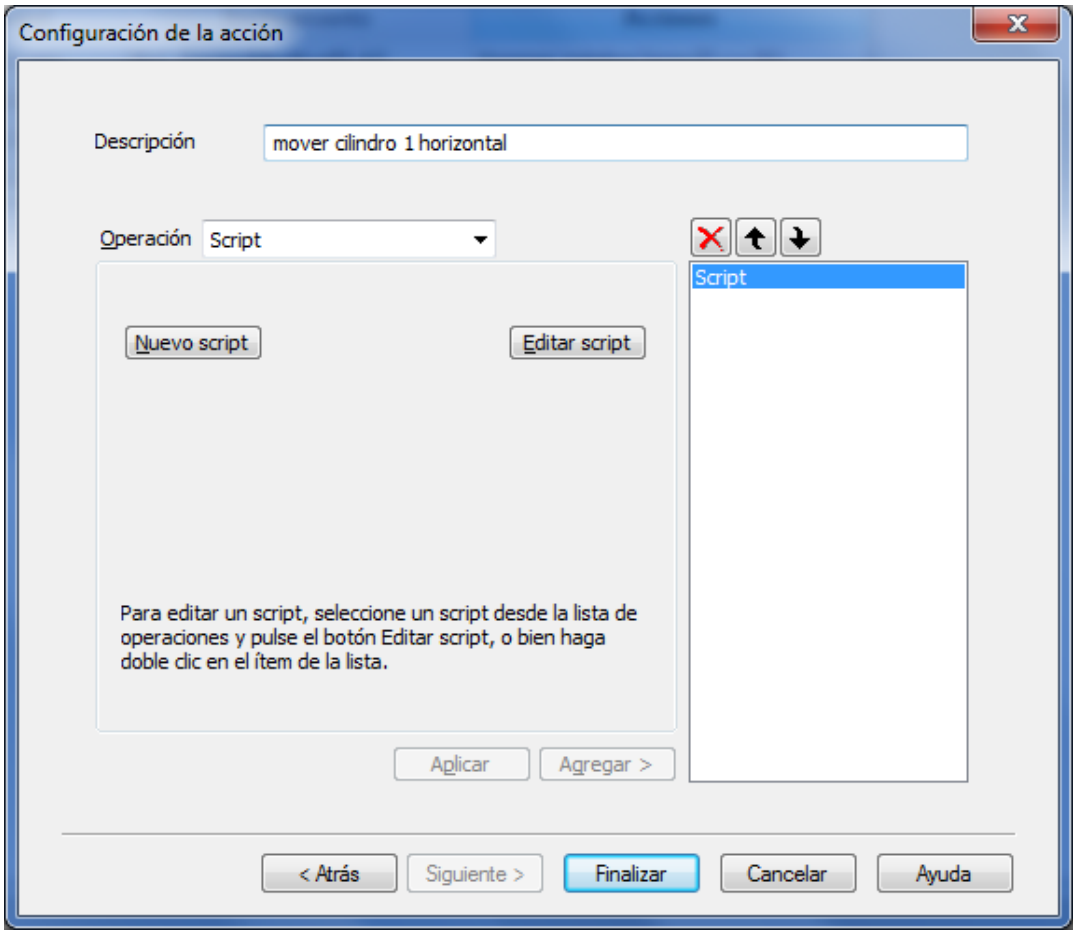


Como el cilindro se detiene cuando se activan los finales de carrera creamos otras dos acciones periódicas que se activen cuando estemos en I2 y en F2 y hagan que posy2 valga 0 en I2 y 10 en F2.

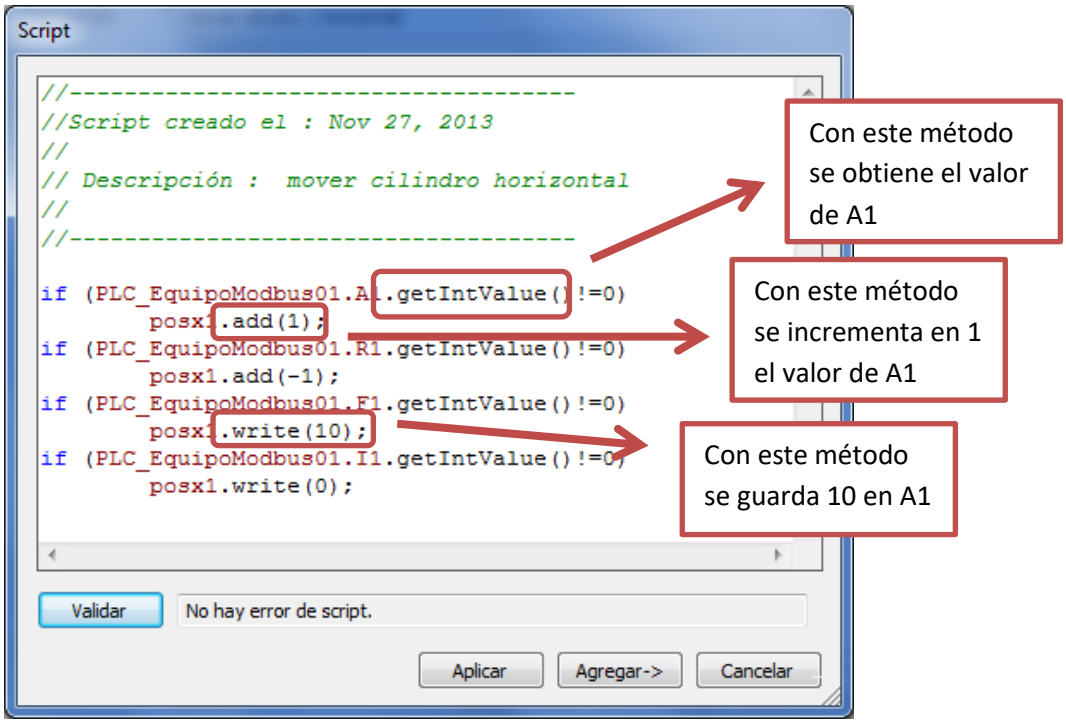
Finalmente las acciones deben de quedar como se indica en la siguiente tabla.

	Disparador	Propiedad	Enclavamiento	Acciones
1	Periódica	Repetir cada 1 seg.	PLC_EquipoModbus01.A2	Agregar palabra [posy2] += [1]
2	Periódica	Repetir cada 1 seg.	PLC_EquipoModbus01.A2==0	Sustraer palabra [posy2] -= [1]
3	Periódica	Repetir cada 1 seg.	PLC_EquipoModbus01.I2	Set palabra [posy2] = [0]
4	Periódica	Repetir cada 1 seg.	PLC_EquipoModbus01.F2	Set palabra [posy2] = [10]

Como se puede ver conseguir una pequeña automatización mediante las acciones simples requiere varias acciones diferentes. En el vijeo-designer existe la posibilidad de crear scripts que ejecutan pequeñas secuencias de código. Creamos una nueva acción periódica para ejecutar un script.



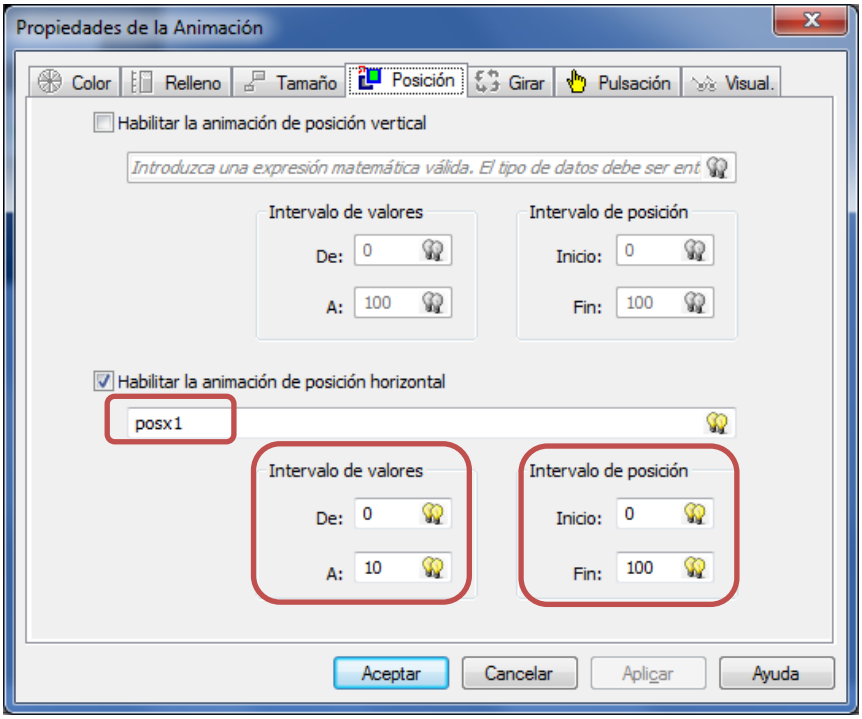
Pinchamos en nuevo script y escribimos el siguiente código



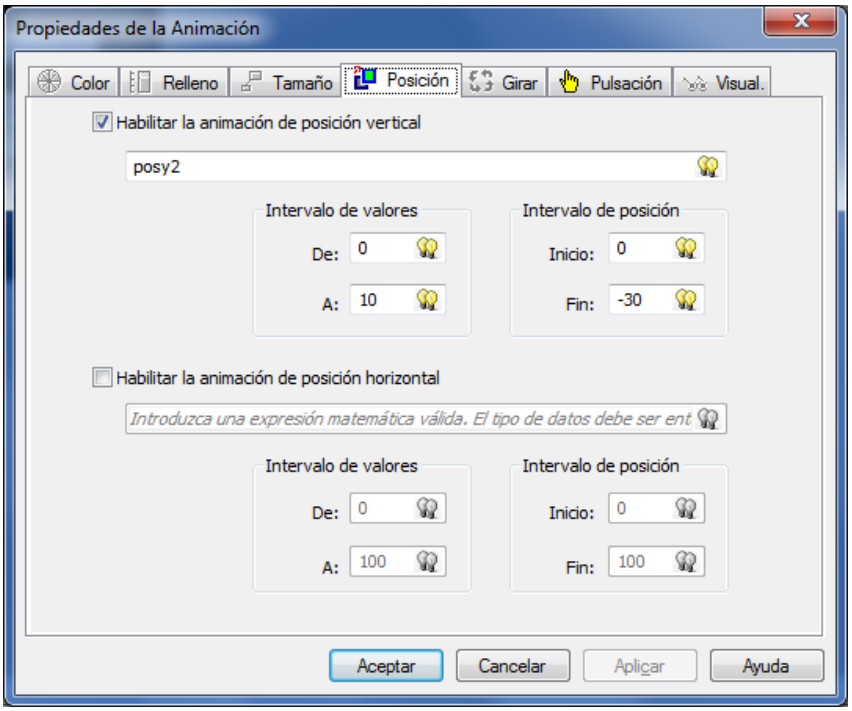
De esta forma se consigue lo mismo que con las cuatro acciones anteriores. Ahora ya tenemos las variables posx1 y posy2 que indican el movimiento del cilindro.

Ahora crearemos las animaciones en los rectángulos que simulan los vástagos de los cilindros.

Abrimos el panel1 y con doble clic sobre el cilindro activamos la animación de movimiento, tal como indica la figura:



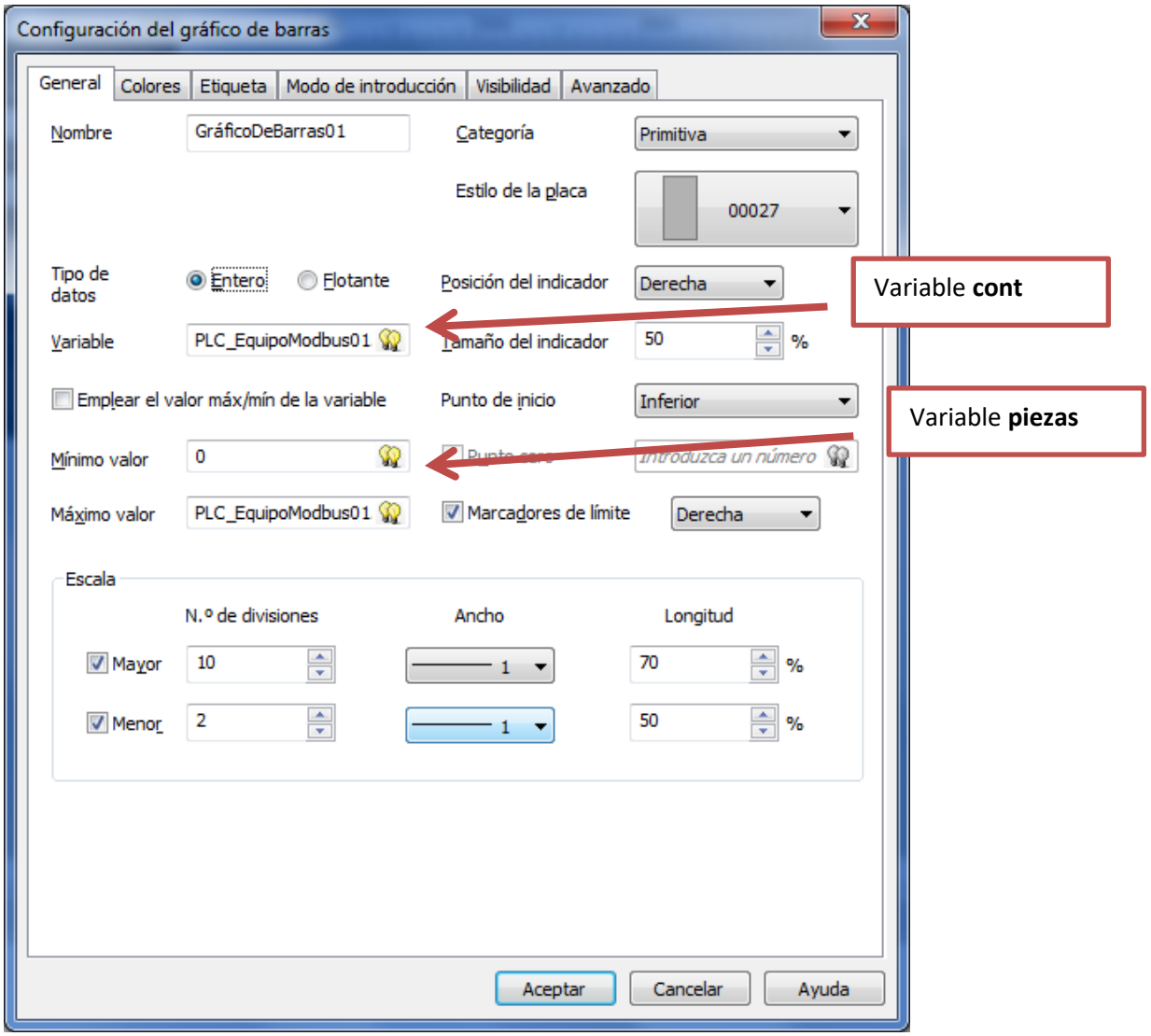
Hacemos lo mismo con el vástago del segundo cilindro:



Apartado 4:

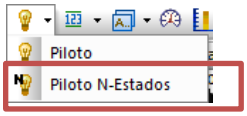
Gráfico de barras

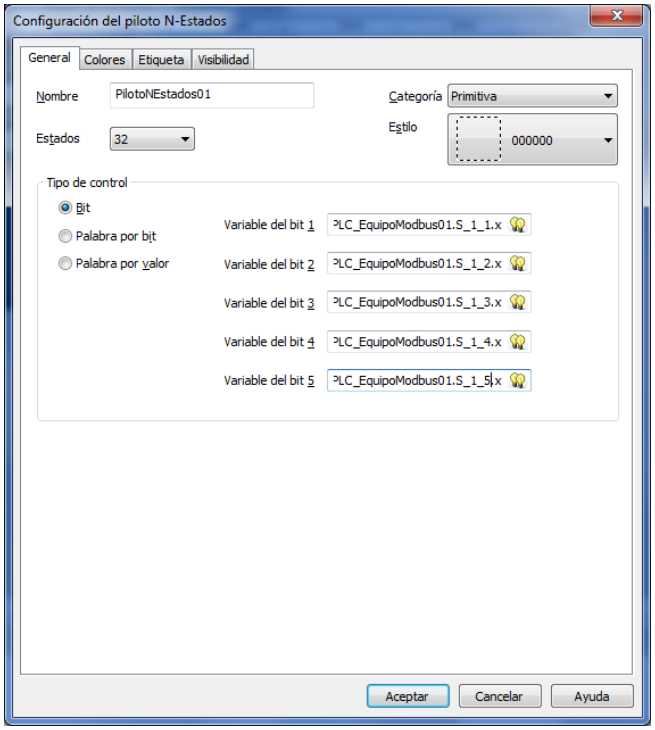
Empezamos con el gráfico de barras de piezas realizadas. Para añadirlo pulsamos y asignamos la variable del **cont** y como límite superior la variable **piezas**



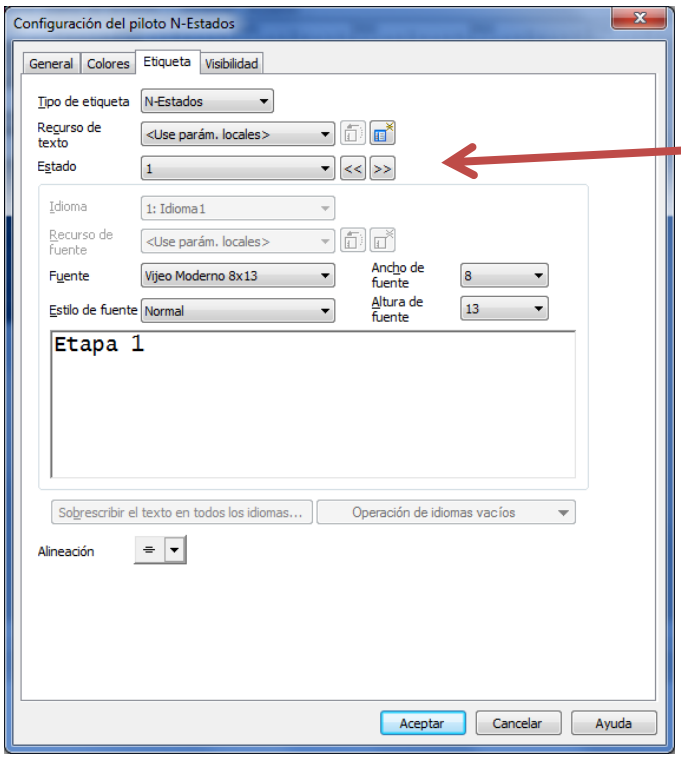
Piloto de varios estados

Ahora añadimos el estado en el que se encuentra el grafcet. Lo hacemos mediante





Con estos bits se forma un número binario cuando está activa la etapa 1 el nº es: 00001 → 1 para la etapa 2: 00010 → 2, el 00100 → 4, el 01000 → 8 y el 10000 → 16



Añadimos la etiqueta de N-Estados:

1 → Etapa 1

2 → Etapa 2

4 → Etapa 3

8→ Etapa 4

16 → Etapa 5

Otra opción más sencilla que se debe probar es la de escribir 5 textos con cada una de las etapas con posteriormente sobre el texto y botón derecho animación, y activar visualización cuando se active S_1_1.x o la que corresponda.

Alarmas

Vamos a crear una alarma que nos avise si el cilindro se ha atascado. Para ello vamos a ver si la variable posx1 supera el valor de 10 eso significa que lleva 10 segundos sin llegar al sensor de final de carrera

En la ventana de variables añadimos posx1 y posy2 al GrupoDeAlarmas1

	Nombre	Tipo de datos	Origen de datos	Grupo de escan...	Dirección de di...	Grupo de alarmas	Grupo de
1	posx1	Integer	Interno			GrupoDeAlarmas1	Ninguno
2	posy2	Integer	Interno			GrupoDeAlarmas1	Ninguno
3	PLC_EquipoModbus01						

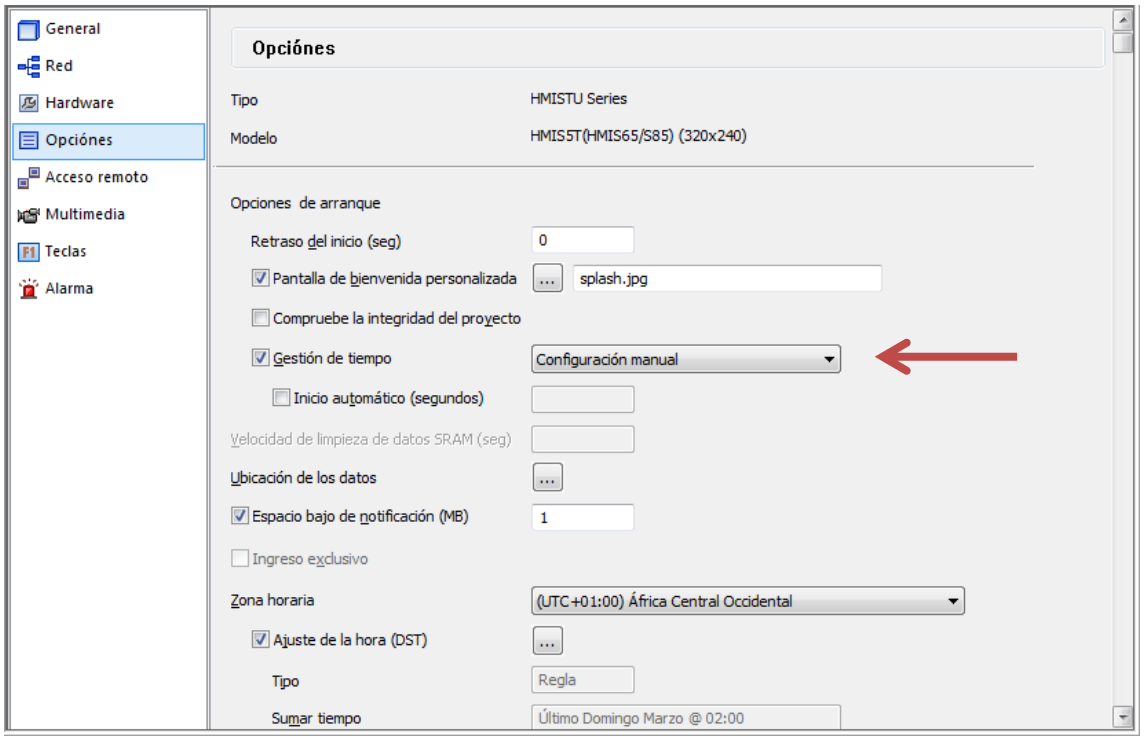
Modificamos sus propiedades para definir la alarma:

Inspector de propiedades

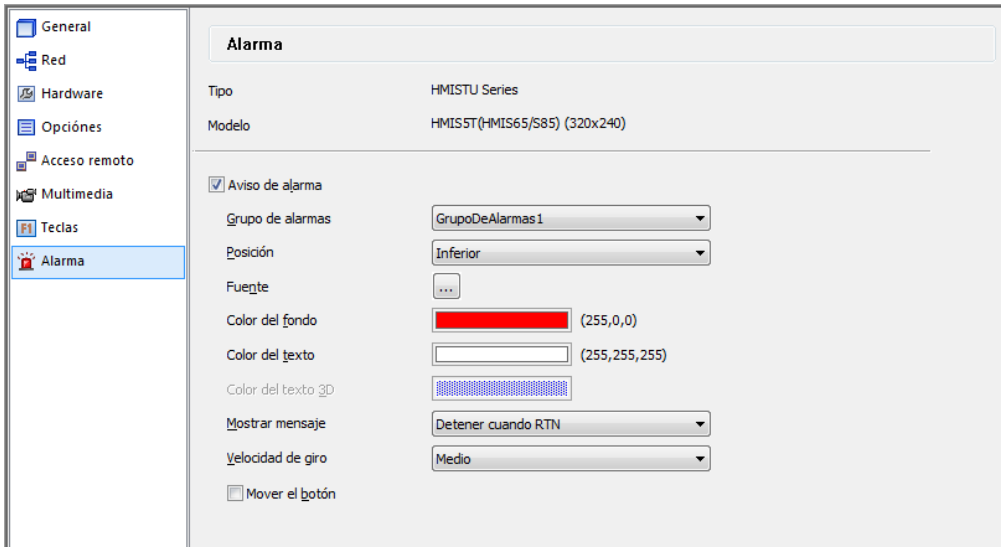
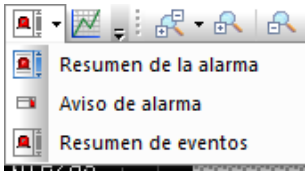
Variable	
Nombre	posx1
Descripción	
Dimensión de matriz	0
Tipo de datos	Integer
Origen	Interno
Referencia	Desactivado
Compartiendo	
Grupo de registros	Ninguno
Detalles de los datos	
Alarma	Habilitado
Mensaje de alarma	El cilindro 1 se ha atascado
Tipo de alarma	Alarmas de límite
Banda muerta	0
BajoBajo	-10
Bajo	-1
Alto	11
AltoAlto	20
Grupo de alarmas	GrupoDeAlarmas1
Acción de contacto	
Acción del disparador	
Gravedad	1
Sonido	

Repetimos para el cilindro 2. De esta forma podemos controlar si han pasado más de 10 segundos desde que se inició el movimiento pero no se ha llegado al final de carrera lo que indicaría al terminal táctil que se ha atascado el cilindro.

Ahora vamos a modificar la configuración del terminal para que pueda almacenar la hora de la alarma y que visualice la alarma. En el navegador abrimos Destino1 y configuramos Gestión del tiempo



Y en la sección Alarma, activar Aviso de alarmas.



Probar el funcionamiento. Después en un nuevo panel, explorar el uso de resumen de la alarma.