Manual de prácticas del Laboratorio de Automatización Avanzada

Trabajo realizado con el apoyo del Programa UNAM-DGAPA-PAPIME Proyecto PE113324

Elaborado por:
Ing. Pedro Luis Galindo Roblero M.F. Gabriel Hurtado Chong M.A. Luis Yair Bautista Blanco

Índice de prácticas

Práctica 1: Control de velocidad mediante una HMI …… 3

Práctica 2: Coordinación de Servomotores mediante HMI …… 19

Práctica 3: Proceso Industrial completo …… 38

Práctica 1: Control de velocidad mediante una HMI

Rúbrica de evaluación

CONCEPTOS, RUBROS O ASPECTOS A EVALUAR	BUENO (2 PUNTOS) Completo entendimiento del problema, realiza la actividad cumpliendo todos los requerimientos.	REGULAR (1 PUNTO) Bajo entendimiento del problema, realiza la actividad cumpliendo algunos de los requerimientos.	NO ACEPTABLE (0 PUNTOS) No demuestra entendimiento del Problema o de la actividad.
1. Seguridad en la ejecución de la actividad	Identifica correctamente los peligros y fuentes de energía, minimiza los riesgos aplicando las medidas de control, realiza la verificación y firma con su nombre.	Identifica parcialmente los peligros, sin aplicar todas las medidas de control.	No aplica ninguna medida de control, no verifica y no firma.
2. Ejecución de la práctica	Muestra un entendimiento completo durante el desarrollo de las actividades, la práctica cumple con todos los requerimientos	Muestra un entendimiento moderado durante el desarrollo de las actividades, la práctica no cumple con todos los requisitos	No demuestra entendimiento de las actividades, la práctica no cumple con los requisitos.
3. Tabla de ordenamiento de variables	Genera la tabla para las entradas y salidas e incluye dirección, símbolo y descripción de cada variable.	Genera una tabla incompleta o con información errónea.	No desarrolla la tabla de ordenamiento de variables.
4. Interfaz de control y supervisión	Diseña la interfaz de control y supervisión cumpliendo con todas las especificaciones solicitadas.	Incluye algunas de las especificaciones solicitadas con carencias u omisiones.	No reporta la interfaz de control y supervisión.
5. Repetibilidad y resistencia a fallos	El sistema opera cíclicamente sin ningún inconveniente, el sistema no falla ante perturbaciones (activaciones erróneas de usuarios).	El sistema se detiene en algún punto de operación, el sistema se detiene ante perturbaciones.	El sistema no es cíclico, se detiene el proceso ante cualquier perturbación.

I. Seguridad en la ejecución

	Peligro o fuente de energía	Riesgo asociado	Medidas de control	Verificación
1ro	Voltaje alterno	Electrocución	Identificar los puntos energizados antes de realizar la actividad y evitar contacto	☐
2do	Voltaje alterno	Electrocución	Identificar los puntos energizados antes de realizar la actividad y evitar contacto	☐
3ro	Voltaje continuo	Daño a equipo	Verificar polaridad y nivel antes de realizar la conexión del equipo o dispositivo	☐
		Apellidos y nombres:	_____	

II. Objetivos de aprendizaje

Objetivo general: Realizar el control de velocidad de un servomotor utilizando una HMI.

Objetivos específicos:

- El alumno será capaz de configurar y controlar la velocidad de un servomotor utilizando una HMI.
- Comprenderá la importancia de las HMI en la automatización industrial y cómo aplicarlas en un entorno real.
- El alumno será capaz de diseñar una interfaz HMI efectiva para el control de velocidad y ajustar los parámetros para optimizar el rendimiento del sistema.

III. Material y equipo

Ilustración 1 Computadora

Ilustración 2 Controlador Siemens S7-1500T

Ilustración 3 Servocontrolador Sinamics S210

Ilustración 4 Servomotor Simotics S-1FK2

Ilustración 5 HMI MTP 700 Unified Comfort

IV. Introducción

Desde las líneas de producción de las grandes fábricas automotrices hasta las máquinas de embalaje que preparan los productos que consumimos a diario, la automatización industrial es el motor que impulsa la eficiencia y la precisión en la manufactura moderna. Detrás de cada sistema automatizado, hay dos tecnologías clave que trabajan en conjunto para garantizar movimientos exactos y procesos optimizados: los servomotores (Motion Control) y las Interfaces Humano-Máquina (HMI).

Ilustración 6 SIMATIC HMI MTP700, Unified Comfort Panel

Los servomotores son esenciales en procesos donde se requiere control preciso de velocidad, posición y torque. Un claro ejemplo es su uso en embobinadoras de cables, donde la velocidad y la tensión deben ajustarse de acuerdo con el material y la aplicación. Aquí, el operador no puede permitirse errores: demasiada tensión y el cable se rompe; muy poca, y la bobina queda suelta y defectuosa. Es la HMI la que permite al operador supervisar y ajustar estos parámetros en tiempo real, logrando un proceso eficiente y de alta calidad.

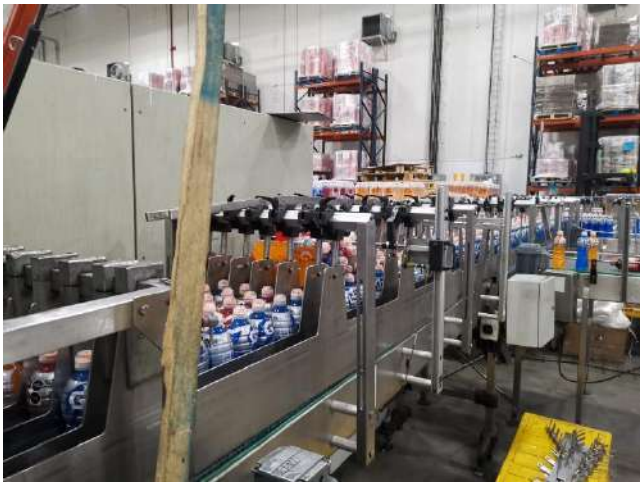


Ilustración 7 Paletizadora de Grupo PepsiCo

Otro ejemplo clave es el uso de servomotores en paletizadores industriales, que organizan cajas y productos en patrones específicos para facilitar su almacenamiento y distribución. Gracias a una HMI intuitiva, los operadores pueden modificar el diseño de los pallets según las necesidades del día, sin necesidad de reprogramar manualmente el sistema. Este nivel de flexibilidad y control es lo que hace a la automatización una herramienta tan poderosa en la industria moderna. Incluso en la manufactura electrónica, donde cada milímetro cuenta, los servomotores garantizan presición exacta de componentes en circuitos impresos, mientras que la HMI permite monitorear la producción y detectar posibles fallos en tiempo real. En un entorno donde la velocidad y la precisión definen la calidad del producto final, la capacidad de los operadores para interactuar de manera eficiente con los sistemas es crucial. La HMI MTP 700 se conecta al sistema a través de TIA Portal, teniendo acceso a comunicación con diversa gamas de controladores de la misma marca o incluso de otras.

V. Desarrollo de la actividad

Creación de nuevo proyecto en TIA Portal V19 y selección de equipos de trabajo

1. Abrir un nuevo proyecto con TIA Portal V19 y utilizar la herramienta “Detect” para agregar un controlador S7-1500
2. Colocar en el controlador las siguientes características:
   a. Nombre del PLC: PLC_Servos
   b. Verificación de IP: 192.168.105.10
   c. Desactivar la casilla “Protect confidential PLC configuration data”
   d. Seleccionar la casilla “Full Access (no protection)”
   e. Dentro de dar click a la casilla “Permit access with PUT/GET communication from remote partner”

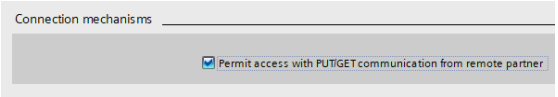


Ilustración 8 Protection & Security -> Connection mechanisms -> Activar casilla

3. Agregar un servocontrolador al proyecto:
   - Nombre: Servo_1
   - Asignación de IP: 192.168.105.11
   - Protection & security; desactivar User management & access control (UMAC) for the drive; desactivar: “Encrypt sensitive drive data”
   - Harwdare settings colocar alimentación a 220 V
   - Añadir un servomotor 1FK2104-4AK1 terminación x-xMxx (Utilizar el filtro de búsqueda para añadir el motor)

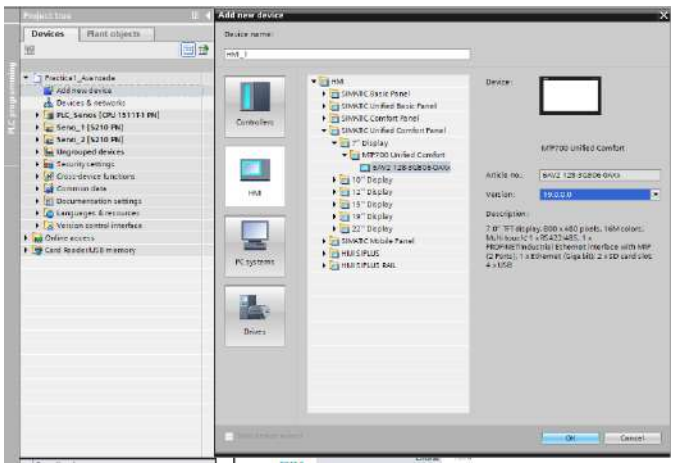


Ilustración 9 MTP700 Unified Comfort -> Versión 19.0.0.0

4. Añadir una HMI al proyecto con la ruta: Add new device -> HMI -> SIMATIC Unified Comfort Panel -> 7’’ Display -> MTP700 Unified Comfort -> 6AV2 128-3GB06-0AXx
   a. Desde la Network view, seleccionar la HMI, en las propiedades, en la sección de PROFINET Interface [X1] colocar la IP 192.168.105.9; el nombre por default de la HMI se puede conservar para el proyecto

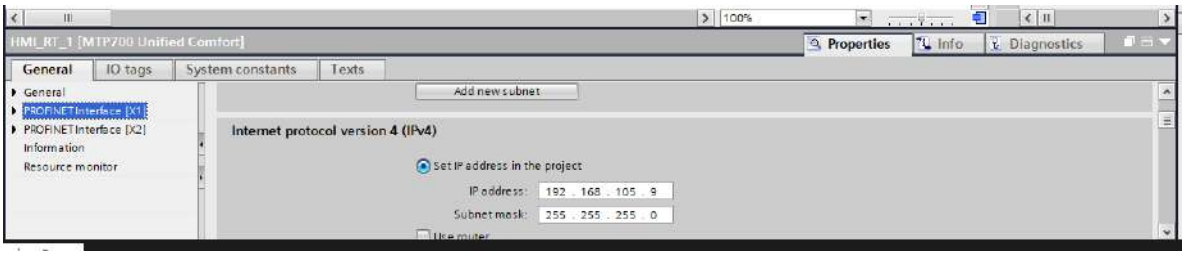


Ilustración 10 Network view -> PROFINET Interface [X1] -> Internet protocol version 4 (IPv4) -> IP: 192.168.105.9

b. En la carpeta de la HMI del proyecto, dar doble click sobre “Runtime settings”; a continuación, en general, desactivar la casilla de “Activate encrypted transfer”

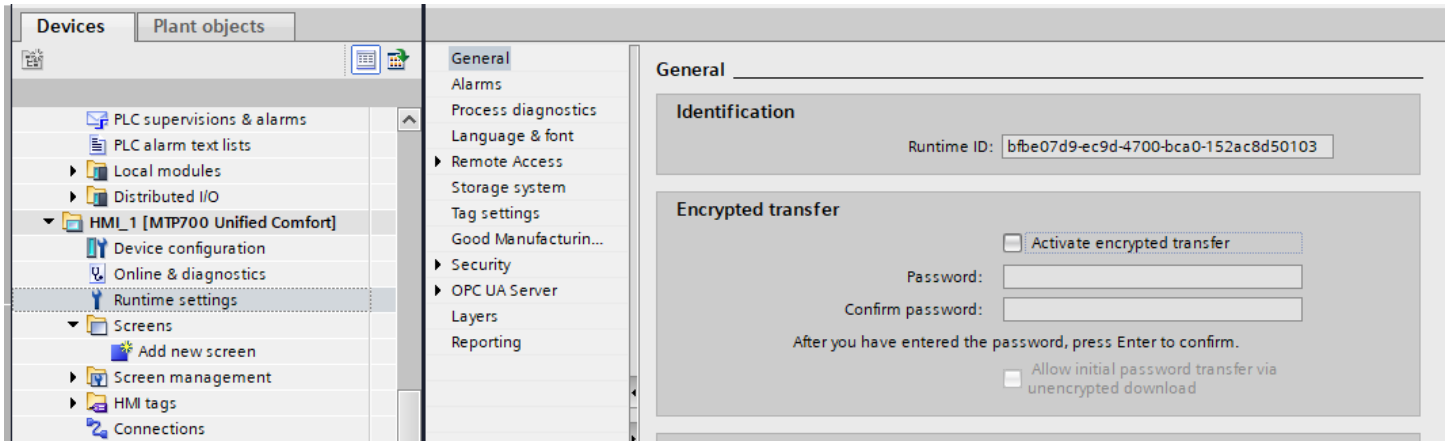


Ilustración 11 Project tree -> HMI_1 -> Runtime settings -> General -> Encrypted transfer -> (Desactivar) Activate encrypted transfer

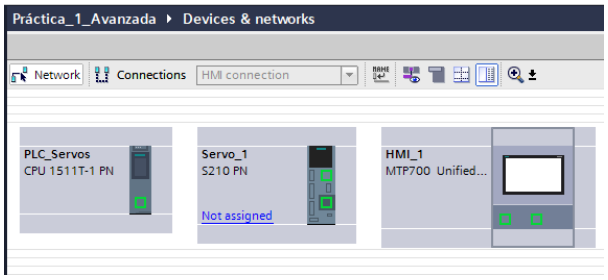


Ilustración 12 Dispositivos agregados al proyecto

Configuración de comunicación

En la Netwok view, deben aparecer tres elementos, el PLC, el servocontrolador y la HMI; conectar el PLC con el servocontrolador al dar click sobre el cuadro verde del PLC y arrastrándolo hacia el puerto inferior verde del servocontrolador, el lazo de autonombrará PN/E.

Conectar la HMI a este nuevo lazo generado (elegir el puerto derecho de la HMI).

Hasta el momento, la HMI se ha dado de alta como un dispositivo que estará dentro del nodo de comunicación, sin embargo, no se ha declarado de donde estará recibiendo o mandado los datos correspondientes. Por lo tanto, desde la ventana de dispositivos y redes, dar click en el ícono “Connections”; verificar que la casilla contigua diga “HMI connection”; conectar el puerto del HMI (el izquierdo) con el puerto del PLC para generar un lazo de comunicación, este nuevo lazo se renombrará como “HMI_Connection_1” en automático.

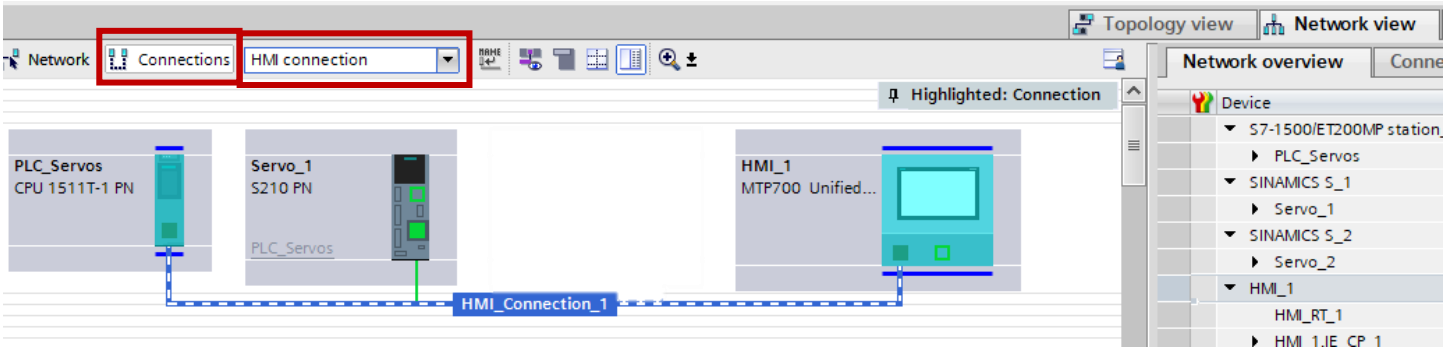
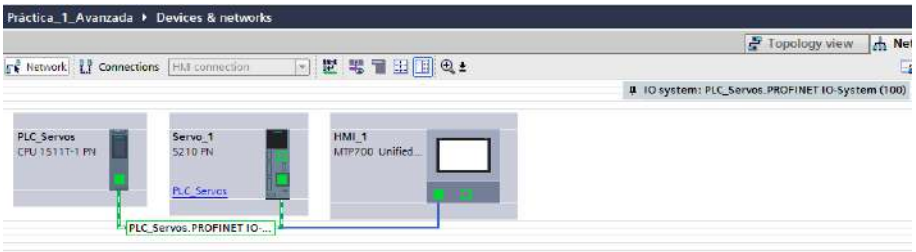


Ilustración 13 HMI_Connection_1



La configuración IRT solo será entre el PLC y el servocontrolador, ya que, la HMI solo será de monitoreo y al pertenecer a la red PROFINET, es más que suficiente.

Ilustración 14 Red PLC_Servos.PROFINET

Una manera de verificar que la HMI y el PLC esta conectados correctamente es dando doble click en la pestaña “Connections”, ubicada en la carpeta de la HMI; aquí se encontrará las propiedades de la comunicación generada.

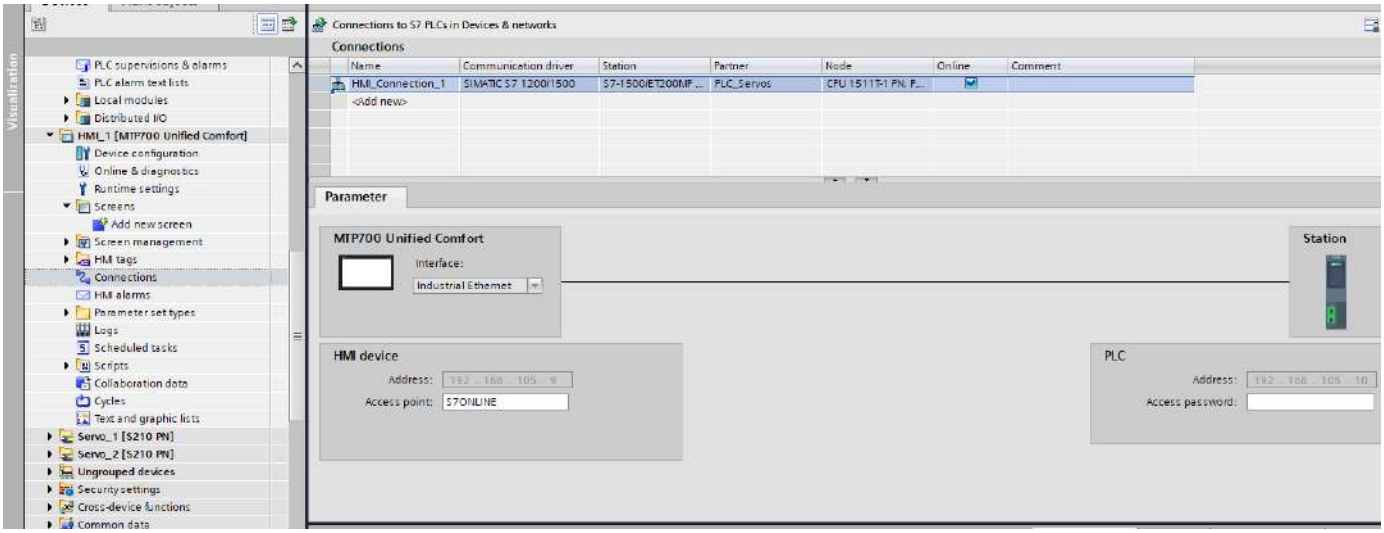


Ilustración 15 Project tree -> HMI_1 -> Connections

Para finalizar las configuraciones de comunicación, regresar a la vista de redes, dar click en el puerto de comunicación del PLC y cambiar los parámetros:

- Advanced options -> Synchronization -> Synchronization role: Sync master
- Advanced options -> IO communication -> Send clock: 2.000 [ms].

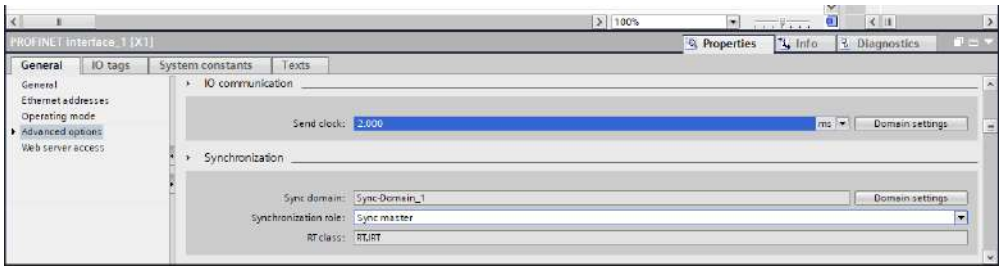


Ilustración 16 Configuración IRT del PLC

En el puerto de servomotor, verificar que en la configuración de telegrama se tenga un bloque organizacional de datos, de no estar, crearlo (como en este caso).

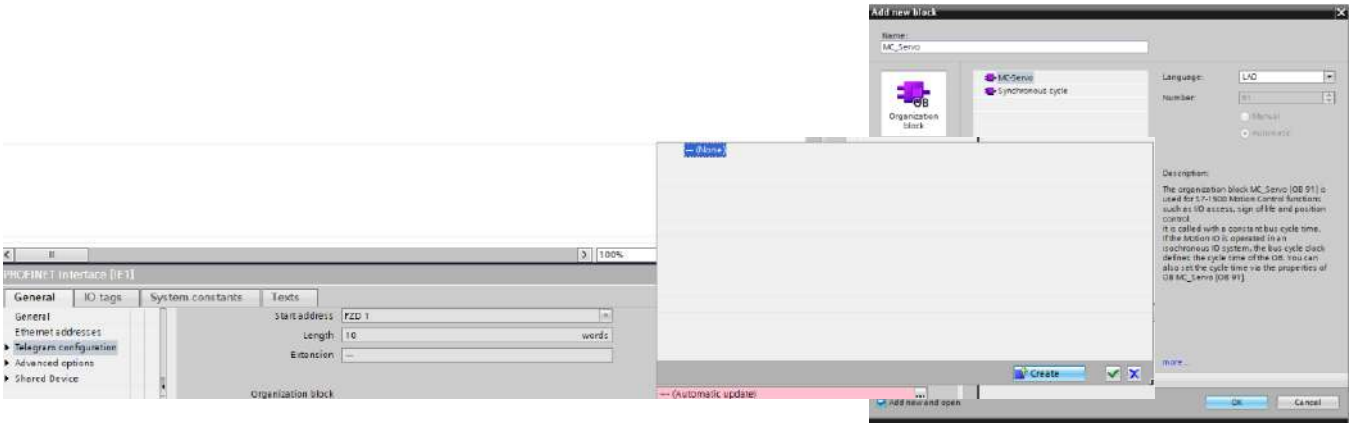


Ilustración 17 Servo_1 -> Properties -> Telegram configuration -> Organization block -> Create -> MC_Servo -> Ok

No olvidar colocar en Isochronous mode, colocar los Ti/To values en “Automatic minimum”.

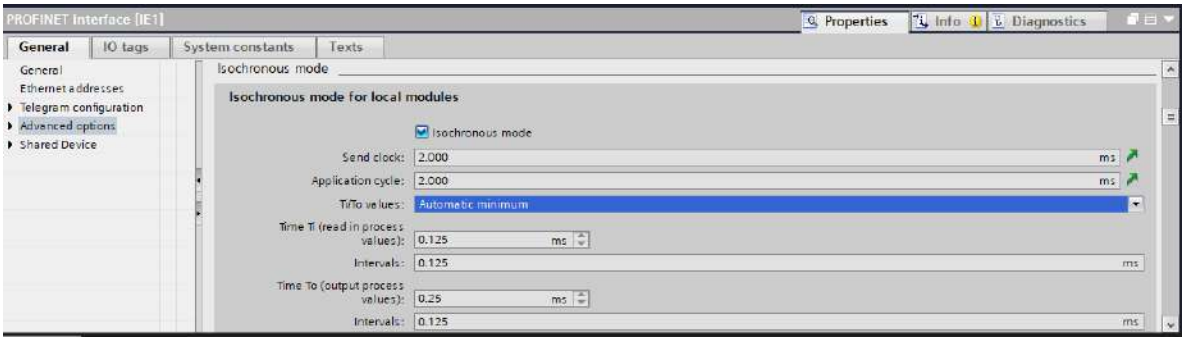


Ilustración 18 Advanced options -> Automatic minimum

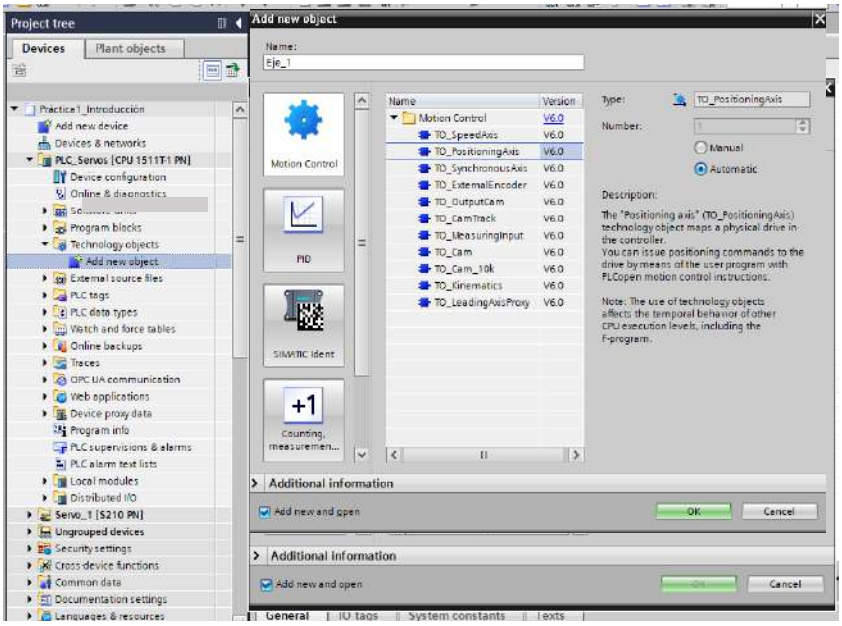


Ilustración 19 PLC_Servos -> Technology objects -> Add new object -> Motion Control -> TO_PositioningAxis: Eje_1

Bloques de control tecnológicos y objeto tecnológico

Dentro del proyecto, agregar un objeto tecnológico de posición, el cual tendrá características de ser un eje rotatorio y con un módulo de 0 a 360°. El nombre del eje será Eje_1. Ligar la comunicación de este objeto tecnológico al Servo_1.

Dentro del Main, del lado derecho de la ventana principal de TIA Portal, la pestaña de instrucciones contiene diversas carpetas que contienen bloques de funciones, dentro del apartado “Technology” ubicar las instrucciones tecnológicas.

Añadir un bloque de energización (MC_Power) para activar el servocontrolador en el Main, el cual se llamará “Energización”, este bloque permite activar el Eje_1.

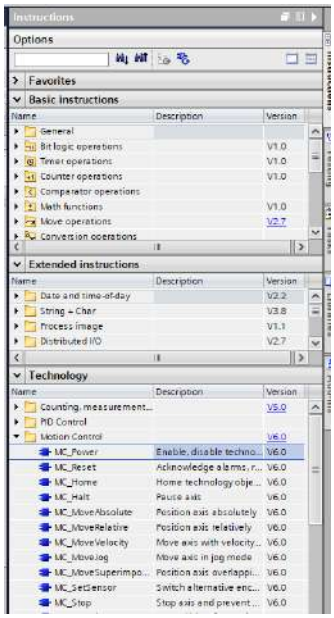


Ilustración 20 Instructions -> Technology -> Motion Control -> MC_Power

Es importante mencionar que para recibir datos desde un dispositivo como una HMI, el identificador del dato no puede ser del tipo entrada (%I), ya que no son activaciones físicas, por lo que es necesario crear tags de memoria (%M) en donde recaerán los valores.

En el tag table del PLC crear:

- Arranque_HMI: Valor proveniente de la HMI (Bool)
- Paro_HMI: Valor proveniente de la HMI (Bool)
- Enable: Bobina de salida (Bool)
- Actualizar_velocidad: Acepta el cambio de velocidad (Bool)
- Velocidad: Parámetro de velocidad obtenido de la HMI (LReal)

Ya que se necesita que los valores arranque y enable esten energizados durante el tiempo de ejecución del programa, generar una lógica de contactos para que las señales se enclaven y queden activas el tiempo necesario.

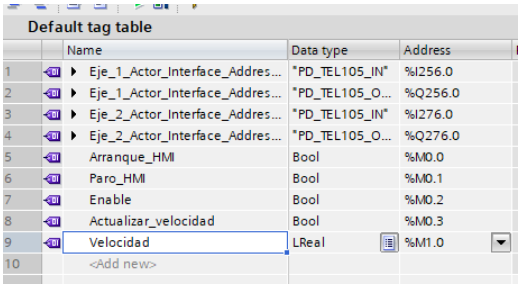


Ilustración 22 Default tag table del PLC

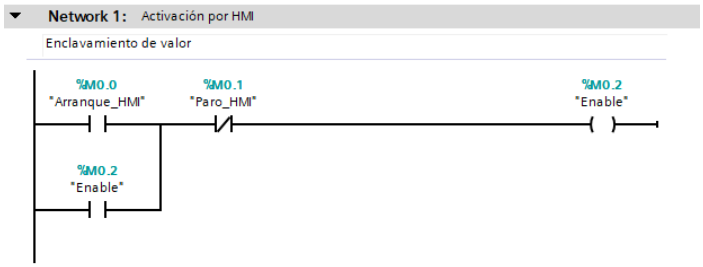


Ilustración 21 Enclavamiento de señal proveniente de la HMI.

Regresando al bloque MC_Power, colocar las variables siguientes:

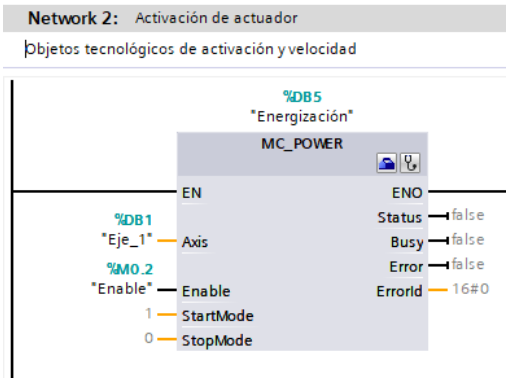


Ilustración 23 Energización (Bloque MC_POWER)

- Eje_1 -> Axis
- Enable -> Enable

Regresando nuevamente al apartado de instrucciones, añadir un bloque tecnológico de tipo MC_MOVEVELOCITY y añadirle los siguientes valores:

Agregar un bloque del tipo MC_MoveVelocity, añadirlo al main y nombrarlo EJE1_VELOCITY_DB. Colocar los siguientes valores:

- Eje_1 -> Axis
- Actualizar_velocidad -> Excute
- Velocidad -> Velocity

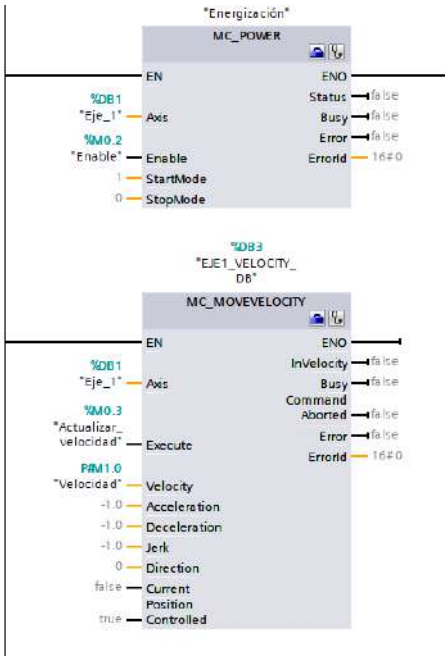


Ilustración 24 Bloques de Energización y Eje1_Velocity_DB

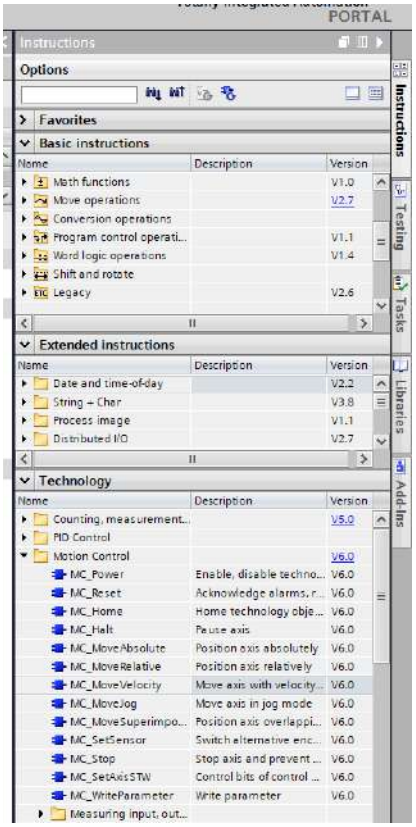


Ilustración 25 Instructions -> Technology -> Motion Control -> MC_MoveVelocity

Crear pantallas de monitoreo en una HMI

Desde el árbol del proyecto, encontrar la carpeta relacionada al HMI_1, a continuación, dar doble click en la opción “Add new screen”. La ventana principal de TIA Portal cambiará como en la ilustración 26.

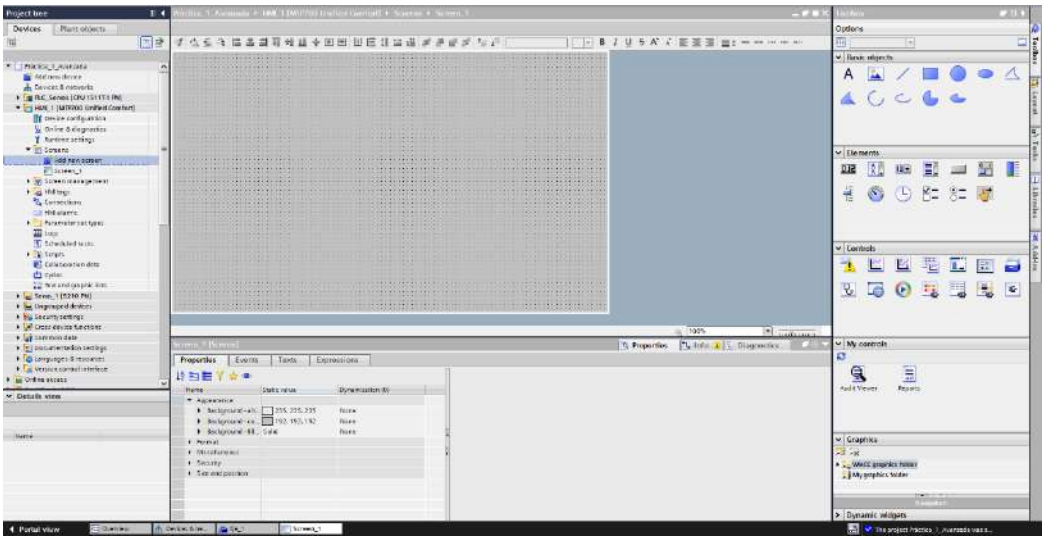


Ilustración 26 Nueva pantalla de HMI Unified Comfort Panel

Es recomendable tener una pantalla de inicio y una pantalla de trabajo, por lo que en la primera pantalla se desarrollará una portada para el proyecto. Renombrar la pantalla como Home.

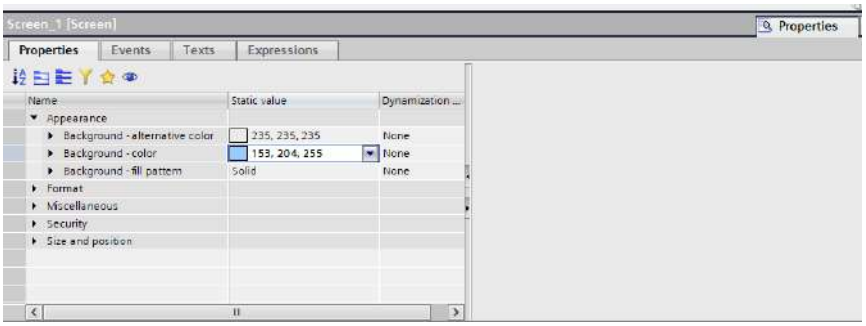


Ilustración 27 Properties -> Appearance -> Background - color

Para modificar la vista de la pantalla, se puede hacer uso de las propiedades presentes en el inspector y de las herramientas, por ejemplo, para la pantalla de “Home” se cambia el color del fondo por una tonalidad azul cielo. Del lado derecho, en la barra de herramientas, se encuentran diversas pestañas que contiene objetos de monitoreo, accionamiento de visualización, entre otros, que permiten el desarrollo más agradable y completo de la pantalla de supervisión.

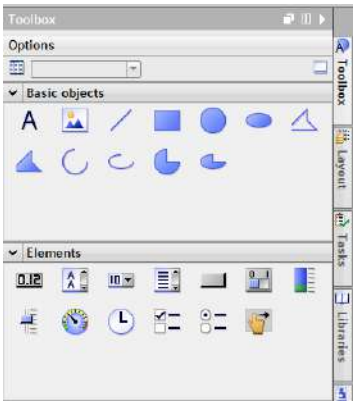


Ilustración 28 Basic objects

Para agregar una imagen a la pantalla, se selecciona el objeto básico de tipo Graphic view y se arrastra al espacio que se va a requerir, en este caso, se requiere colocar imágenes alusivas a los escudos de la universidad. Adicionalmente, se puede encontrar en este mismo espacio la herramienta de texto, con la cual se puede colocar enunciados o frases requeridas.

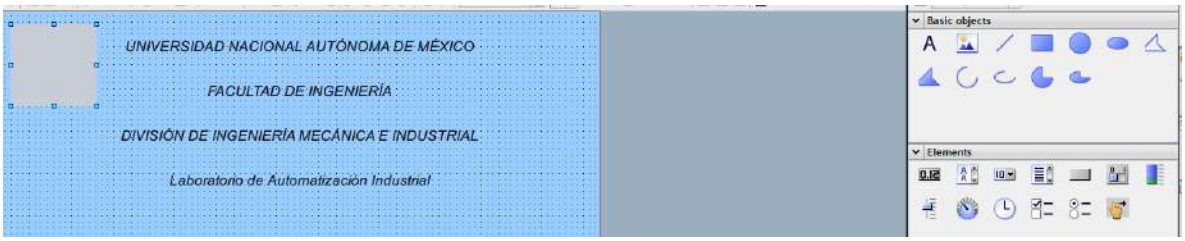


Ilustración 29 Uso de herramientas Graphic view y Text

En el inspector, al tener seleccionada la imagen (como en la ilustración 29), en la ruta, General -> Graphic; en el ícono de la flecha hacía abajo, darle click para que aparezca un menú.

Al aparecer el menú, dar click en el ícono de la parte inferior izquierda para agregar una imagen desde la biblioteca de la computadora.

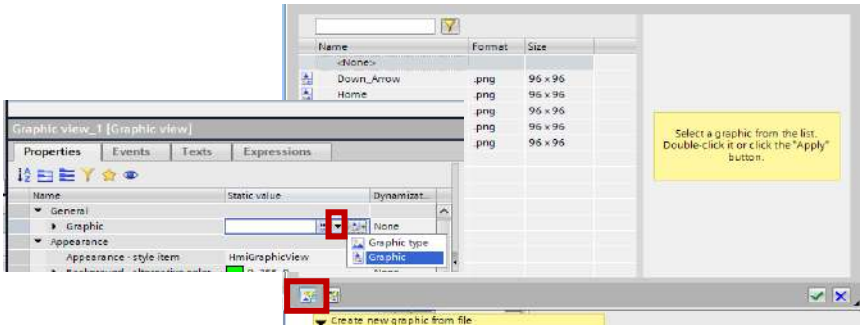


Ilustración 30 Graphic view -> General -> Graphic ->(click) fecha hacia abajo -> Graphic -> Create new graphic file -> seleccionar archivo

Al agregar la imagen, en algunas ocasiones, la imagen permite quitar el fondo.

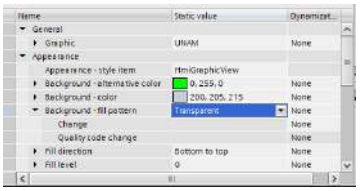


Ilustración 31 Properties -> Appearence -> Background - fill pattern > Transparent

Al final, después de configurar la pantalla de inicio, se puede continuar con la pantalla de monitoreo.

Ilustración 32 Pantalla Home

En todos los proyectos de que incluyan una supervisión o monitoreo desde una HMI, es necesario detener la ejecución del programa en cualquier momento, por lo tanto, se debe colocar un botón de “Stop Runtime”. En la ventana de herramientas, ir al apartado de “Elements”; seleccionar un botón y colocarlo en el proyecto. Al botón, se le colocará el texto “Stop Runtime” y en el inspector, cambiar a la pestaña de eventos. Al momento de presionar este botón (Press), la acción que se debe ejecutar es un StopRuntime.

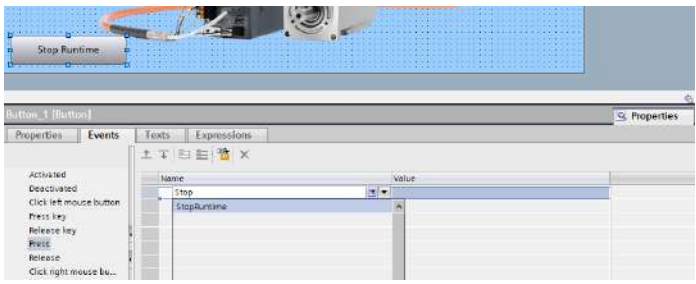


Ilustración 33 Botón -> "Stop Runtime" -> Events -> Press -> StopRuntime

Se agrega una nueva pantalla y se renombra como Monitoring, ya creada, en la pantalla Home, agregar un botón que permita el cambio de pantalla, se puede realizar de dos maneras, la primera es colocar un botón, añadir un evento al presionar, el cual se llamará “ChangeScreen” y se coloca la pantalla a la cual se requiere ir. La otra manera es directamente arrastrar la pantalla del árbol del proyecto y colocarla sobre la pantalla que actualmente se está trabajando (automáticamente generará un botón con el evento ya configurado).

Ilustración 34 Botón de cambio de pantalla

Tabla de variables en la HMI

En la carpeta de la HMI, en el árbol del proyecto, dar doble click en el default tag table para añadir los tags que permitirán que la HMI actualicen los valores de las variables que se ocupan en la rutina del PLC.

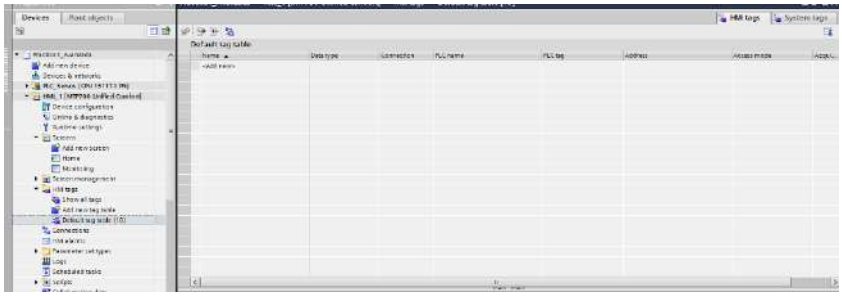


Ilustración 36 HMI_1 -> HMI tags -> Default tag table

Para añadir una variable, dar doble click sobre la opción <Add new> o directamente escribir sobre el espacio el nombre del tag; añadido el tag, dar click en los tres puntos que se encuentran en la columna “Connection” y seleccionar la red HMI_Connection_1, con ello, la HMI entenderá que el tag no tendrá una comunicación con el controlador al cual se está apuntando en la conexión, el tag puede ser escrito desde la HMI o ser de lectura desde el PLC.

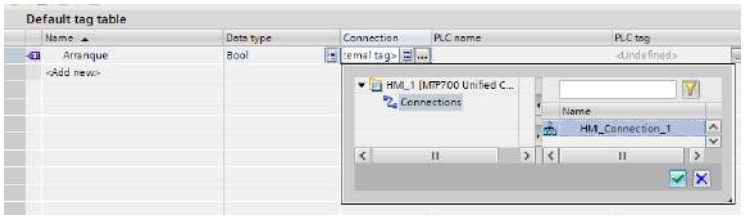


Ilustración 35 Direccionamiento de tag "Arranque" en la red “HMI_Connection_1”

Ya integrado el tag a la red, es necesario apuntar la dirección de origen al cual va estar respondiendo de acuerdo a la ejecución del programa.

Ilustración 37 Tag en espera de apuntador

Dar click en los tres puntos de la columna “PLC tag”, y buscar la carpeta de los PLC tags la dirección en donde se encuentra la variable de interés.

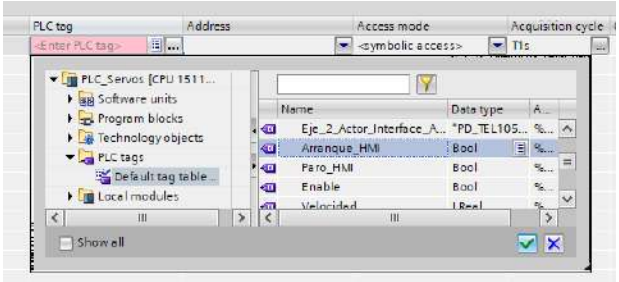


Ilustración 38 Para la variable arranque, PLC_Servos -> Default tag table -> Arranque_HMI

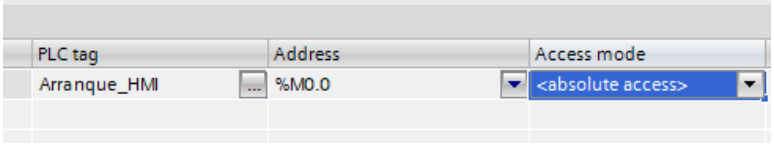


Ilustración 39 Access mode -> <absolute access>

En el apartado de Access mode, seleccionar <absolute access>, ahora se podrá observar la dirección (del PLC), en el apartado "Address".

Para que la transmisión de datos sea la más síncrona posible, modificar el tiempo de adquisición a 100 [ms].

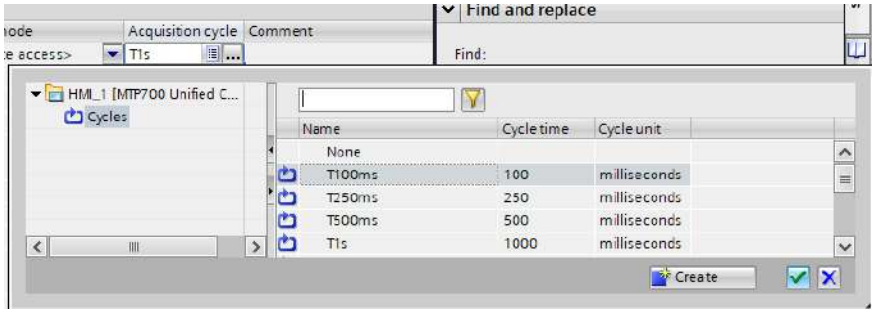


Ilustración 40 Acquisition cycle -> HMI_1 -> Cycles -> T100ms -> Ok

Realizar los pasos anteriores para los valores de:

- Paro
- Actualizar_V
- Velocidad: esta variable es de tipo Real largo (LReal)

Ilustración 41 Default tag table de HMI_1

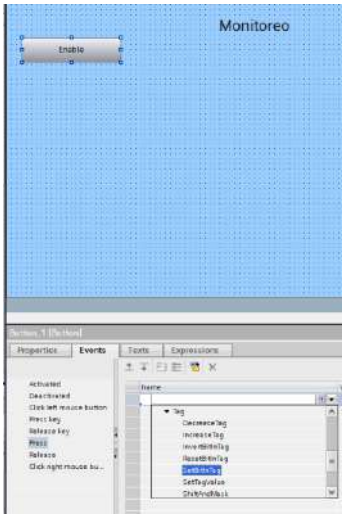


Ilustración 43 Botón Enable - > Events -> Press -> SetBitInTag

Cambiando a la pantalla de Monitoreo, agregar un encabezado que identifique a la nueva interfaz (Monitoreo). Agregar un botón, el cual tendrá la funcionalidad de activar el bloque tecnológico MC_Power. Renombrar el botón como "Enable", y añadirle el evento de fijar (Set) el valor de un bit al presionar el elemento.

Para referenciar el tag de interés, dar click en los tres puntos de la columna de "Value" y buscar el tag de "Arranque".

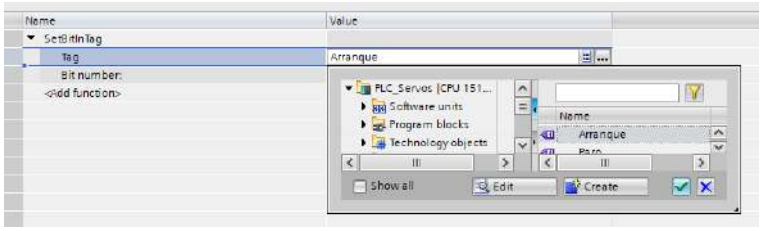


Ilustración 42 SetBitInTag -> Tag -> Arranque

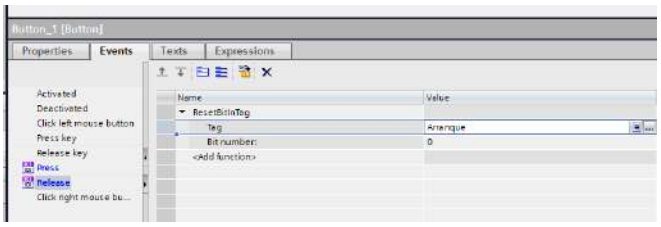


Ilustración 44 Botón Enable -> Events -> Release -> ResetBitInTag -> Tag: Arranque

Hasta el momento, el botón al ser presionado, mandará un uno lógico, sin embargo, se requiere que el funcionamiento sea el de un botón pulsador, por lo que es necesario tener un evento al dejar de pulsar el botón; sobre el mismo botón añadir un evento en la opción “Release”, función “ResetInBitTag” asociada a la misma variable.

En la pantalla de monitoreo, generar un botón para cambiar a la pantalla principal.

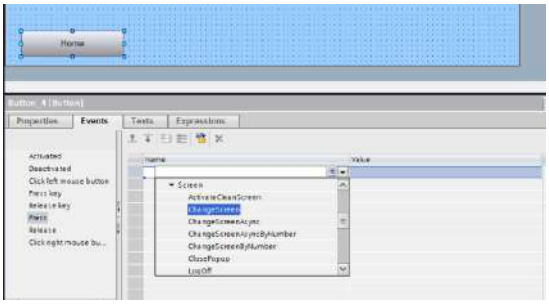


Ilustración 45 Botón de Home -> Events -> Press -> Screen -> ChangeScreen

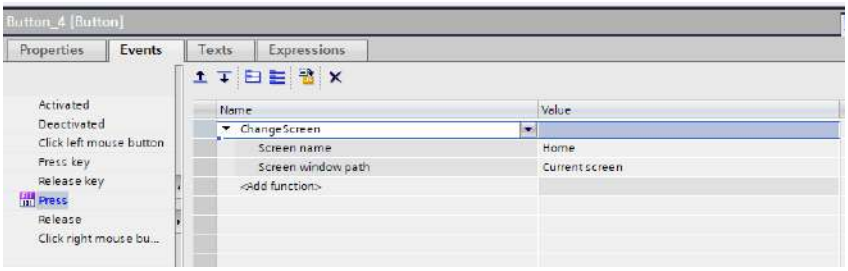


Ilustración 46 Botón Home -> Screen name: Home

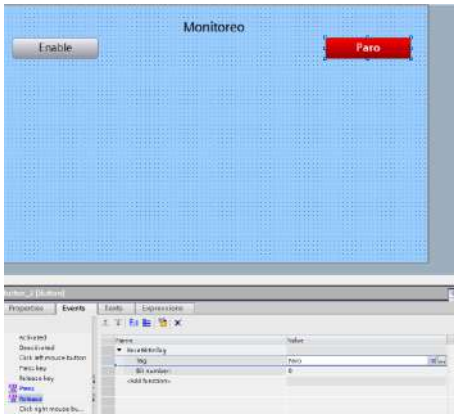


Ilustración 48 Botón de paro -> Eventos: Press & Release -> Variable: Paro

Para terminar de configurar la pantalla de monitoreo, crear un botón de paro, y un botón de actualizar velocidad.

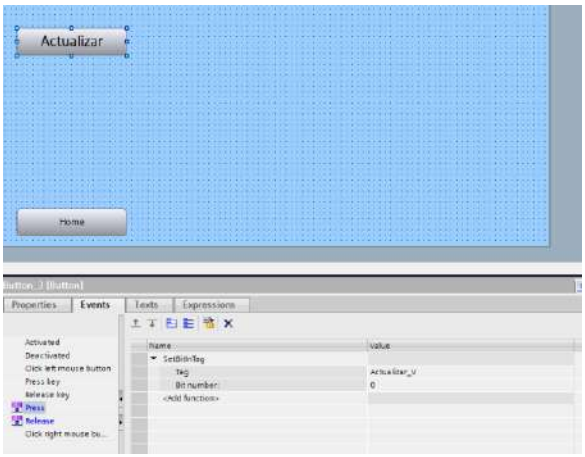


Ilustración 47 Botón "Acturalizar" -> Eventos: Press & Release -> Variable: Actualizar_V

Para generar una interfaz más completa, agregar un deslizador (slider) a la pantalla y caracterizarlo:

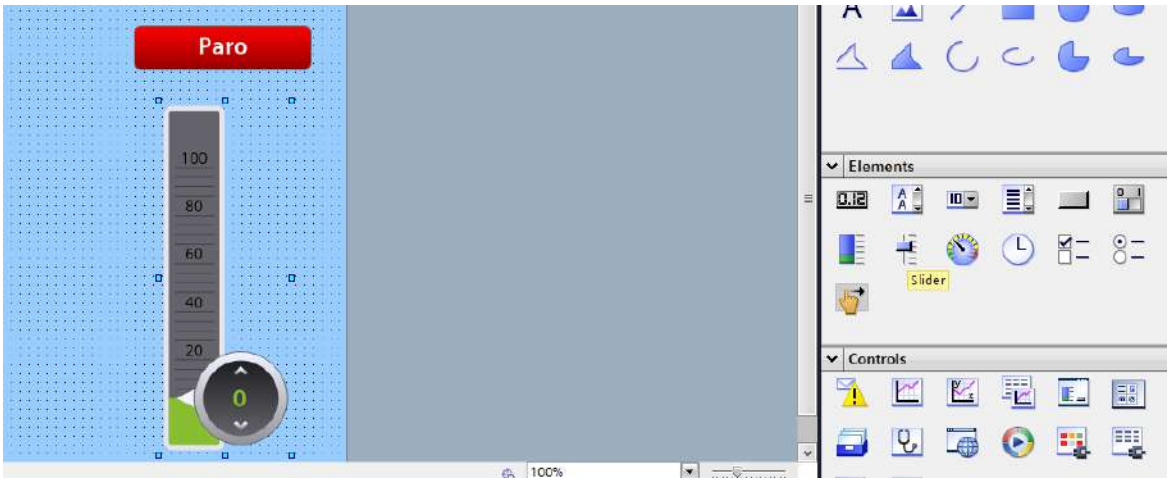


Ilustración 49 Elements -> Slider -> añadir a pantalla "Monitoring"

Colocar los valores de escala máximos y mínimos, 0 y 100, respectivamente

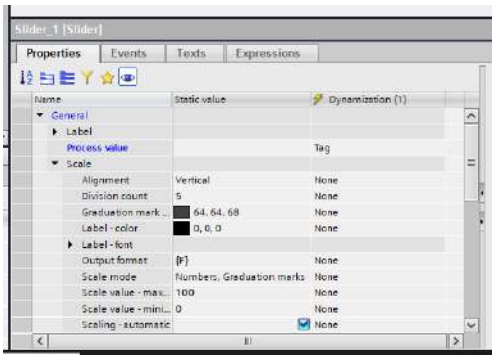


Ilustración 51 Slider -> Properties -> General -> Scale -> Scale value -max / min: 100/0

Colocar la variable de proceso:

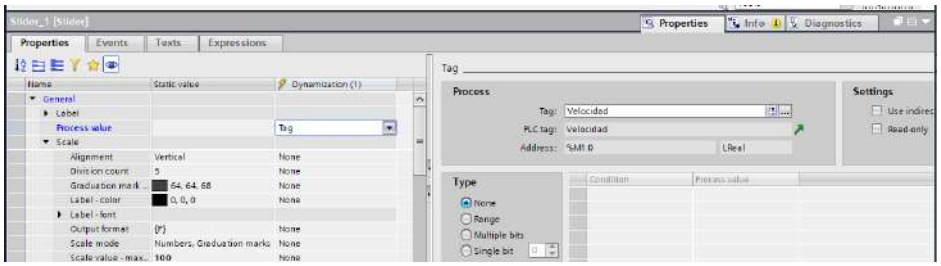


Ilustración 50 Slider -> Properties -> General -> Label -> Process value -> Dynamization: Tag -> Process -> Tag: Velocidad

Regresar a los tags del HMI y realizar un escalamiento de la variable velocidad

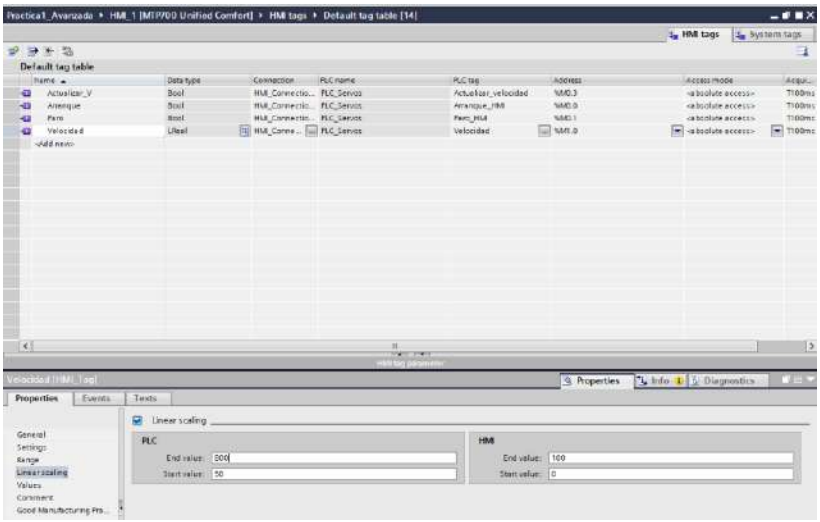


Ilustración 52 Default tag table (HMI) -> Seleccionar la variable velocidad, en el inspector -> Properties -> Linear scaling (activar casilla) -> PLC: Start value: 50 / End value; HMI: Start value: 0 / End value: 100

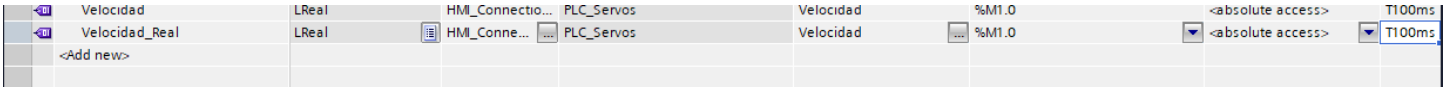


Ilustración 53 Variable Velocidad_Real -> Apunta a la misma variable que la variable Velocidad

Ya que el valor original se ha escalado, generar una nueva variable en el tag table de la HMI que sea referenciado a el mismo valor de velocidad para tener el valor de lectura real en el monitoreo.

Finalmente, agregar un elemento del tipo “Gauge” y configurarlo.

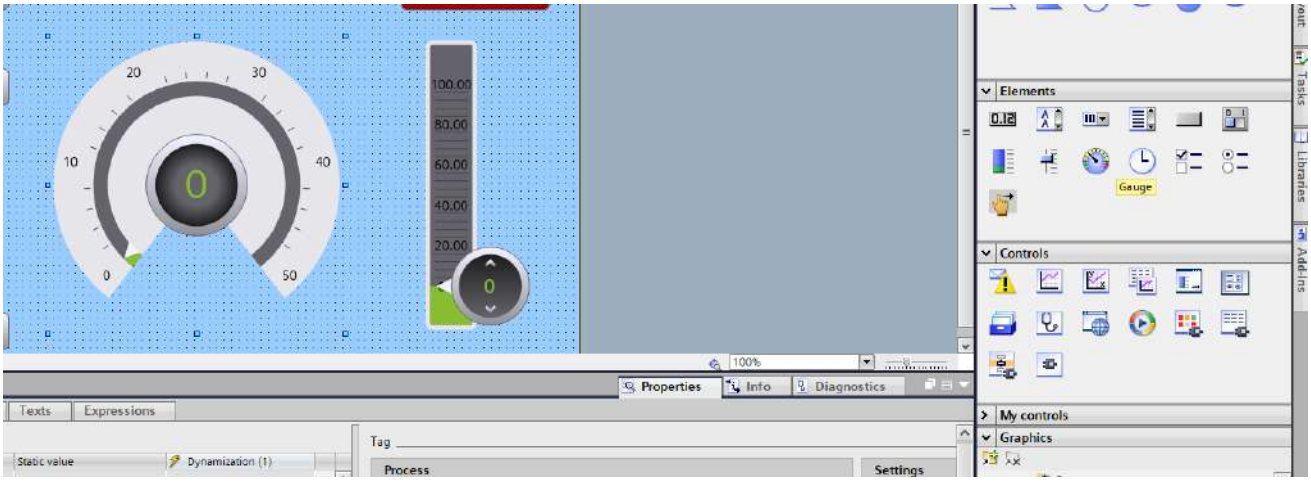


Ilustración 54 Elements -> Gauge -> Colocarlo en la pantalla de monitoreo

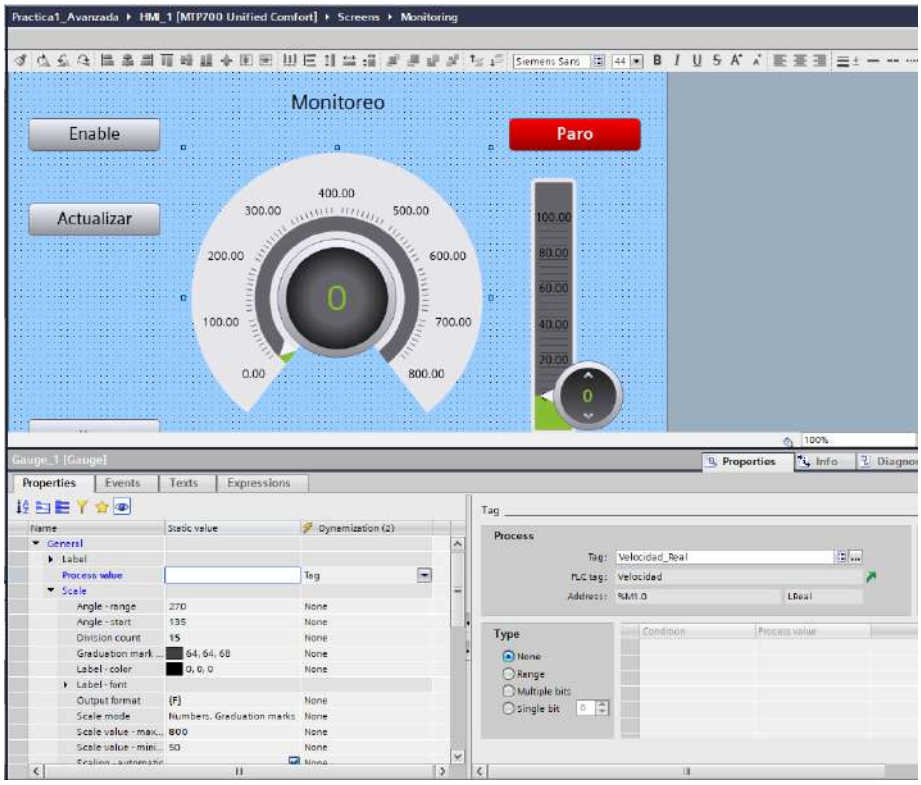


Ilustración 55 Gauge -> Scale value max: 800 / Scale value min: 50; Process value: Tag: Velocidad_Real

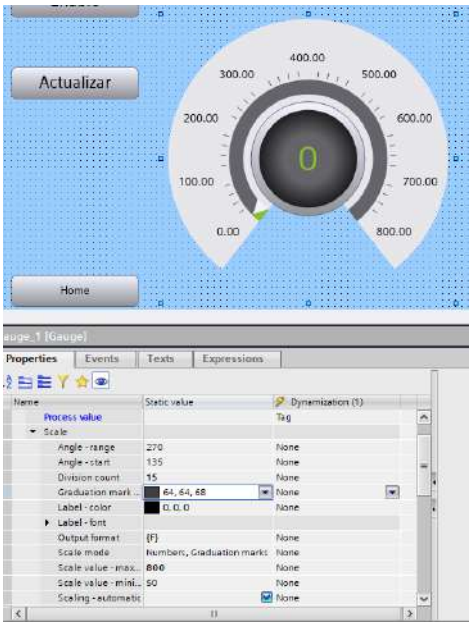


Ilustración 57 Gauge -> Properties -> Scale -> Division count: 15

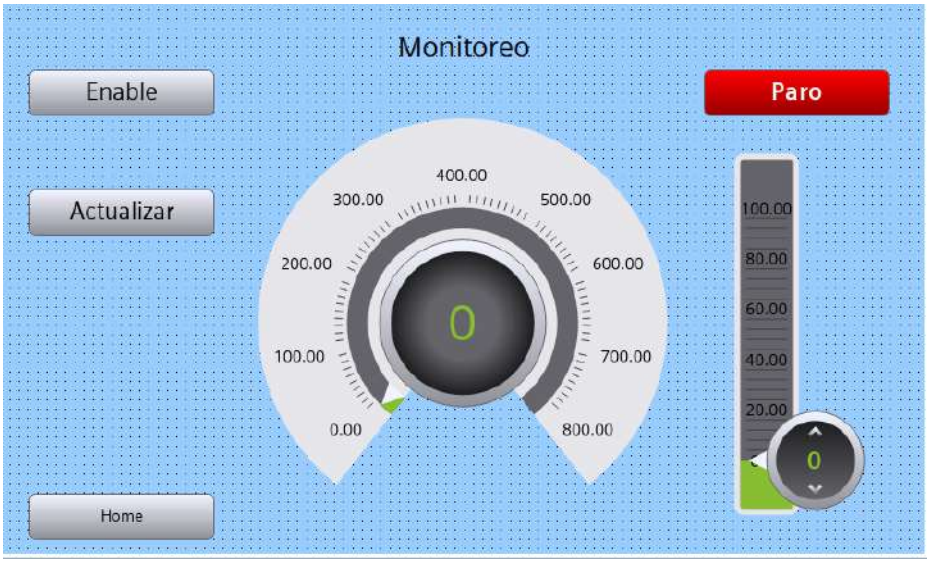


Ilustración 56 Pantalla de monitoreo

Entregables de la práctica

1. Código de Ladder comentado.
2. Pantallas realizadas en la HMI.
3. Imágenes del cambio de estado en el monitoreo de la HMI (fotografía de pantalla en ejecución).

VI. Conclusiones y observaciones.

_____

_____

_____

_____

Referencias

- Levine, W. S. (2010). *The control handbook: Control system applications.* CRC Press.
- Schröder, D. (2015). *Electrical drives and control systems*. Springer.
- Craig, J. J. (2020). *Introduction to robotics: Mechanics and control* (4th ed.). Pearson.

Práctica 2: Coordinación de Servomotores mediante HMI

Rúbrica de evaluación

CONCEPTOS, RUBROS O ASPECTOS A EVALUAR	BUENO (2 PUNTOS) Completo entendimiento del problema, realiza la actividad cumpliendo todos los requerimientos.	REGULAR (1 PUNTO) Bajo entendimiento del problema, realiza la actividad cumpliendo algunos de los requerimientos.	NO ACEPTABLE (0 PUNTOS) No demuestra entendimiento del Problema o de la actividad.
1. Seguridad en la ejecución de la actividad	Identifica correctamente los peligros y fuentes de energía, minimiza los riesgos aplicando las medidas de control, realiza la verificación y firma con su nombre.	Identifica parcialmente los peligros, sin aplicar todas las medidas de control.	No aplica ninguna medida de control, no verifica y no firma.
2. Ejecución de la práctica	Muestra un entendimiento completo durante el desarrollo de las actividades, la práctica cumple con todos los requerimientos	Muestra un entendimiento moderado durante el desarrollo de las actividades, la práctica no cumple con todos los requisitos	No demuestra entendimiento de las actividades, la práctica no cumple con los requisitos.
3. Tabla de ordenamiento de variables	Genera la tabla para las entradas y salidas e incluye dirección, símbolo y descripción de cada variable.	Genera una tabla incompleta o con información errónea.	No desarrolla la tabla de ordenamiento de variables.
4. Interfaz de control y supervisión	Diseña la interfaz de control y supervisión cumpliendo con todas las especificaciones solicitadas.	Incluye algunas de las especificaciones solicitadas con carencias u omisiones.	No reporta la interfaz de control y supervisión.
5. Repetibilidad y resistencia a fallos	El sistema opera cíclicamente sin ningún inconveniente, el sistema no falla ante perturbaciones (activaciones erróneas de usuarios).	El sistema se detiene en algún punto de operación, el sistema se detiene ante perturbaciones.	El sistema no es cíclico, se detiene el proceso ante cualquier perturbación.

I. Seguridad en la ejecución

	Peligro o fuente de energía	Riesgo asociado	Medidas de control	Verificación
1ro	Voltaje alterno 220 V	Electrocución	Identificar los puntos energizados antes de realizar la actividad y evitar contacto	☐
2do	Voltaje alterno 127 V	Electrocución	Identificar los puntos energizados antes de realizar la actividad y evitar contacto	☐
3ro	Voltaje continuo 24 V	Daño a equipo	Verificar polaridad y nivel antes de realizar la conexión del equipo o dispositivo	☐
		Apellidos y nombres:	_____	

II. Objetivos de aprendizaje

Objetivo general: Activar la coordinación dos servomotores para alcanzar posiciones definidas utilizando una HMI.

Objetivos específicos:

- El alumno será capaz de configurar y coordinar dos servomotores, haciendo uso de las librerías del software de programación, además, comprenderá la importancia de la coordinación de los mismos en la automatización industrial y aplicar estas técnicas en un entorno real
- El alumno será capaz de diseñar una interfaz de HMI efectiva para la activación y desactivación de ejes.
- El alumno realizará el monitoreo del estado de los actuadores para verificar que el sistema esté funcionando adecuadamente.

III. Material y equipo

Ilustración 1 Computadora

Ilustración 2 Controlador Siemens S7-1500T

Ilustración 3 Servocontrolador Sinamics S210

Ilustración 4 Servomotor Simotics S-1FK2

Ilustración 5 HMI MTP 700 Unified Comfort

IV. Introducción

Ilustración 6 Máquina CNC multieje

Hoy en día la precisión y la eficiencia son fundamentales para mantener la competitividad y la calidad en los procesos de producción de las grandes empresas manufactureras. Una tecnología esencial que contribuye a alcanzar estos objetivos es la coordinación de ejes, la cual permite que múltiples servomotores trabajen de manera sincronizada, garantizando movimientos precisos y armoniosos en maquinaria compleja. La coordinación de ejes es crucial en aplicaciones donde varios componentes deben moverse en perfecta sincronía. Por ejemplo, en la industria automotriz, los robots de ensamblaje deben trabajar de forma coordinada para colocar componentes con precisión milimétrica. Un pequeño desfase en los movimientos puede causar defectos o ralentizar la línea de producción, generando costos adicionales. En estos sistemas, las Interfaces Humano-Máquina (HMI) permiten a los operadores monitorear en tiempo real el desempeño de la maquinaria, ajustar velocidades o posiciones y garantizar un flujo de trabajo adecuado.

En la industria del embalaje, por ejemplo, las máquinas deben cortar, doblar y sellar materiales de manera simultánea para mantener una alta velocidad de producción sin comprometer la calidad. Sin una sincronización adecuada, comprometería al sistema a tener errores como desalineaciones o tiempos de producción más largos, que se traduce a un aumento en los costos de producción.

Otras aplicaciones Industriales destacadas son:

Ilustración 7 Sistema de enrollado de acero

- Robótica Industrial: Los robots que realizan tareas como soldadura, ensamblaje o pintura requieren una coordinación precisa de sus múltiples ejes para replicar movimientos humanos con exactitud y consistencia, reduciendo errores y optimizando la producción.
- Máquinas Herramienta CNC: En el mecanizado de piezas complejas, la coordinación de ejes permite que las herramientas de corte se muevan en múltiples direcciones simultáneamente, logrando formas precisas y reduciendo el tiempo de producción.
- Impresión y Conversión de Materiales: Equipos que manejan materiales continuos, como papel o láminas metálicas, dependen de la coordinación de ejes para cortar, enrollar o estampar sin interrupciones ni defectos, garantizando una producción uniforme y de alta calidad.

Empresas líderes en automatización como Siemens y Rockwell Automation han desarrollado tecnologías avanzadas para facilitar la coordinación de ejes. Siemens, con su plataforma StartDrive - TIA Portal y los controladores S7-1500, ofrece soluciones que permiten programar y sincronizar servomotores de manera sencilla e intuitiva. Por otro lado, Rockwell Automation, con su tecnología

Kinetix y Logix en Studio 5000, proporciona herramientas para controlar múltiples ejes con alta precisión, adaptándose a las necesidades específicas de la industria.

En el PLC S7-1500, los objetos tecnológicos representan dispositivos físicos como los servomotores, lo cual facilita su configuración y control. Estos objetos ayudan a programar movimientos, ajustar velocidades y sincronizar acciones. Al usar estos objetos en conjunto con la HMI, se puede coordinar el funcionamiento de varios dispositivos de manera sencilla y visual, permitiendo ajustes en tiempo real según las necesidades de la producción.

V. Desarrollo de la actividad

Creación de nuevo proyecto en TIA Portal V19 y selección de equipos de trabajo

1. Abrir un nuevo proyecto con TIA Portal V19 y utilizar la herramienta “Detect” para agregar un controlador S7-1500
2. Colocar en el controlador las siguientes características:
   a. Nombre del PLC: PLC_Servos
   b. Verificación de IP: 192.168.105.10
   c. Desactivar la casilla “Protect confidential PLC configuration data”
   d. Seleccionar la casilla “Full Access (no protection)”
   e. Verificar que en Protection & Security; Connection mechanims este activa la casilla “Permit Access with PUT/GET communication from remote partner”
   f. Verificar la que la casilla “Only allow secure PG/PC and HMI communication”

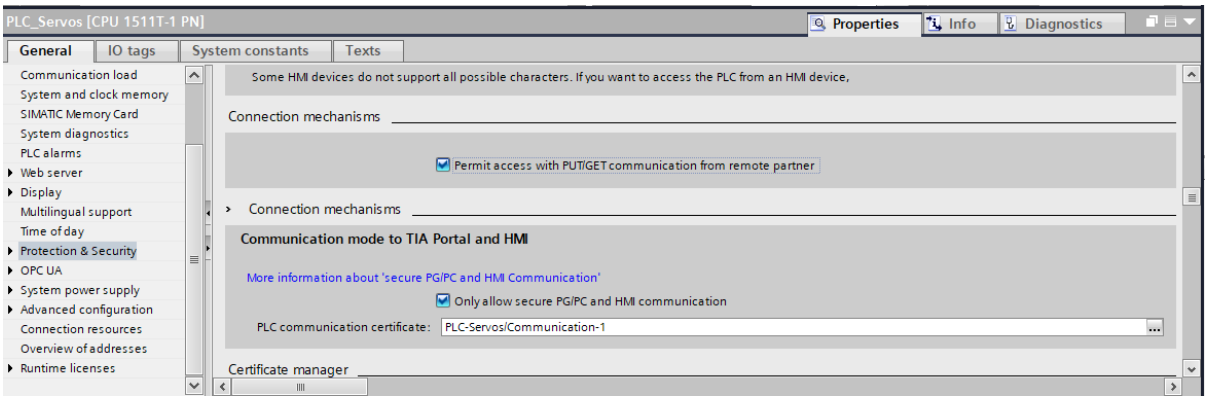


Ilustración 8 Casillas PUT/GET y PG/PC activas en el PLC

3. Agregar un servocontrolador al proyecto:
   a. Nombre: Servo_1
   b. Asignación de IP: 192.168.105.11
   c. Protection & security; desactivar User management & access control (UMAC) for the drive; desactivar: “Encrypt sensitive drive data”
   d. Harwdare settings colocar alimentación a 220 V

   e. Añadir un servomotor 1FK2104-4AK1 terminación x-xMxx (Utilizar el filtro de búsqueda para añadir el motor)

4. Agregar un segundo servocontrolador al proyecto:
   a. Nombre: Servo_2
   b. Asignación de IP: 192.168.105.12
   c. Protection & security; desactivar User management & access control (UMAC) for the drive; desactivar: “Encrypt sensitive drive data”
   d. Harwdare settings colocar alimentación a 220 V
   e. Añadir un servomotor 1FK2104-4AK1 terminación x-xMxx (Utilizar el filtro de búsqueda para añadir el motor)

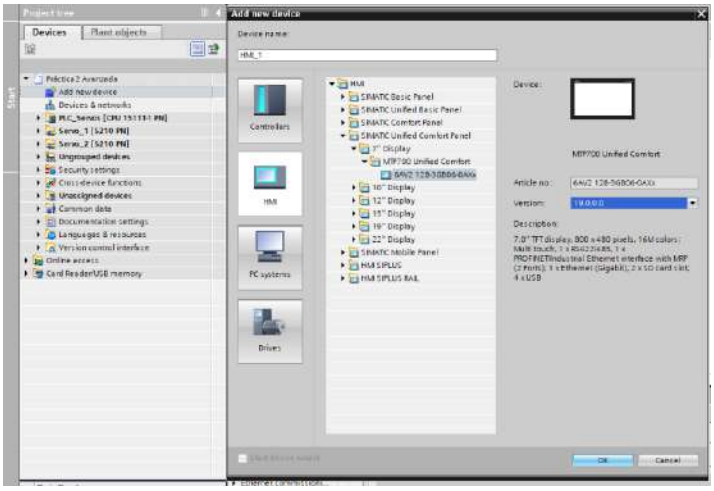


Ilustración 9 Verificar que la versión se 19.0.0.0

5. Añadir una HMI al proyecto con la ruta: Add new device -> HMI -> SIMATIC Unified Comfort Panel -> 7’’ Display -> MTP700 Unified Comfort -> 6AV2 128-3GB06-0AXx -> V 19.0.0.0
   a. Desde la Network view, seleccionar la HMI, en las propiedades, en la sección de PROFINET Interface [X1] colocar la IP 192.168.105.9; el nombre por default de la HMI se puede conservar para el proyecto
   b. En la carpeta de la HMI (ubicada en el árbol del proyecto), dar doble click en “Runtime settings”; en general, desactivar la casilla “Activate encrypted transfer”

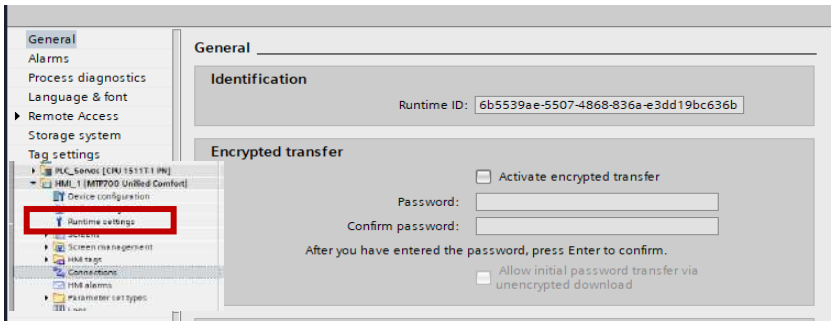


Ilustración 10 HMI ->Runtime settings -> General -> Encrypted transfer -> (Desactivar) Activate encrypted transfer

Comunicación de equipos

Generar el lazo de comunicación sólo entre el PLC y los servocontroladores, tanto en la vista de redes como en la vista topológica (comunicación IRT). A continuación, conectar la HMI en la vista de redes al lazo de comunicación PLC_Servos.PROFINET.

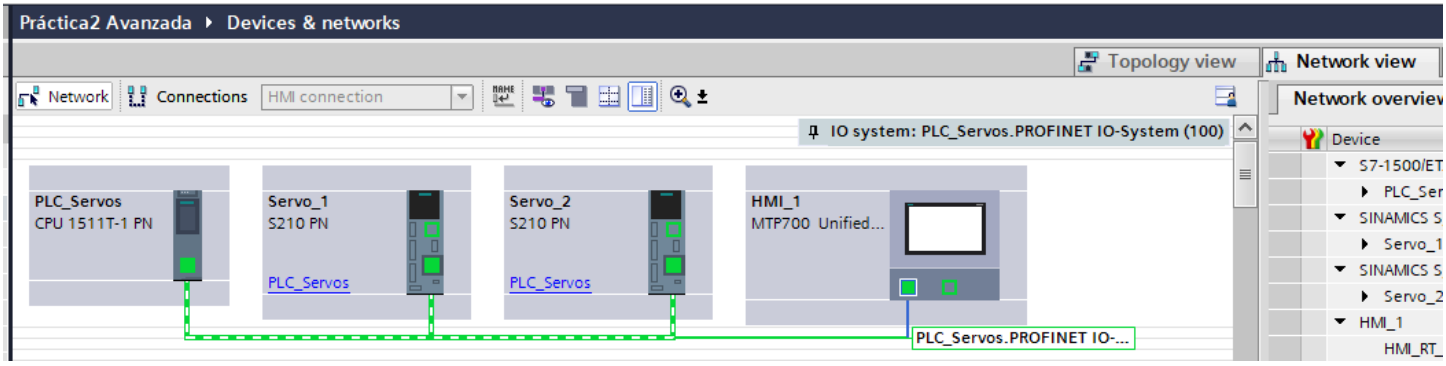


Ilustración 11 Conexión de equipos en Network view -> Puerto izquierdo de HMI conectado a PLC_Servos.PROFINET

Dar click en la pestaña “Conections”, verificar que la pestaña contigua diga “HMI connection”; y conectar el puerto del HMI (el izquierdo) con el puerto del PLC para generar un lazo de comunicación, este nuevo lazo se renombrará como “HMI_Connection_1” en automático.

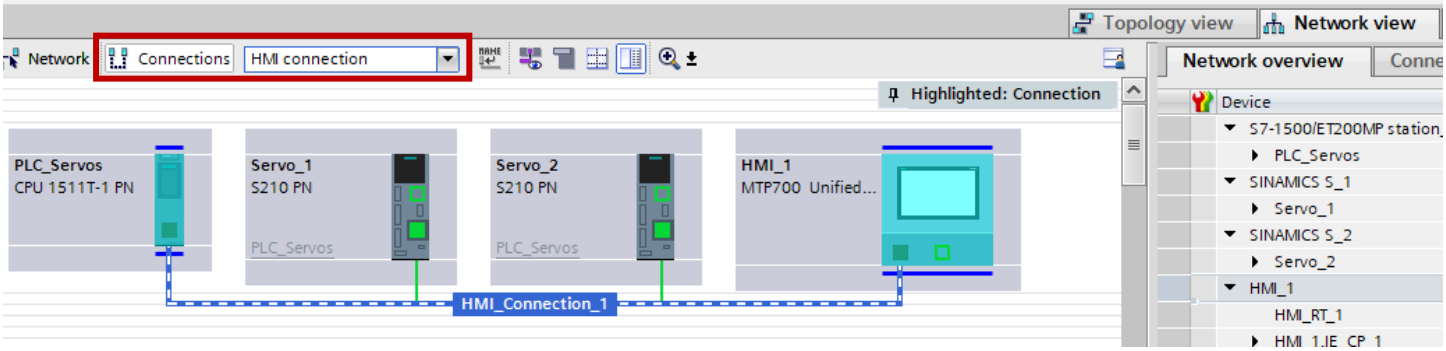


Ilustración 12 Network view -> Connetions -> (Verificar HMI connections) -> HMI_Conneciton_1

Al crearse este nodo de comunicación, desde el inspector, TIA Portal reconoce a los dispositivos como “Local [HMI]” y “Remoto [PLC]”, incluso es posible ver las direcciones IP’s de los equipos.

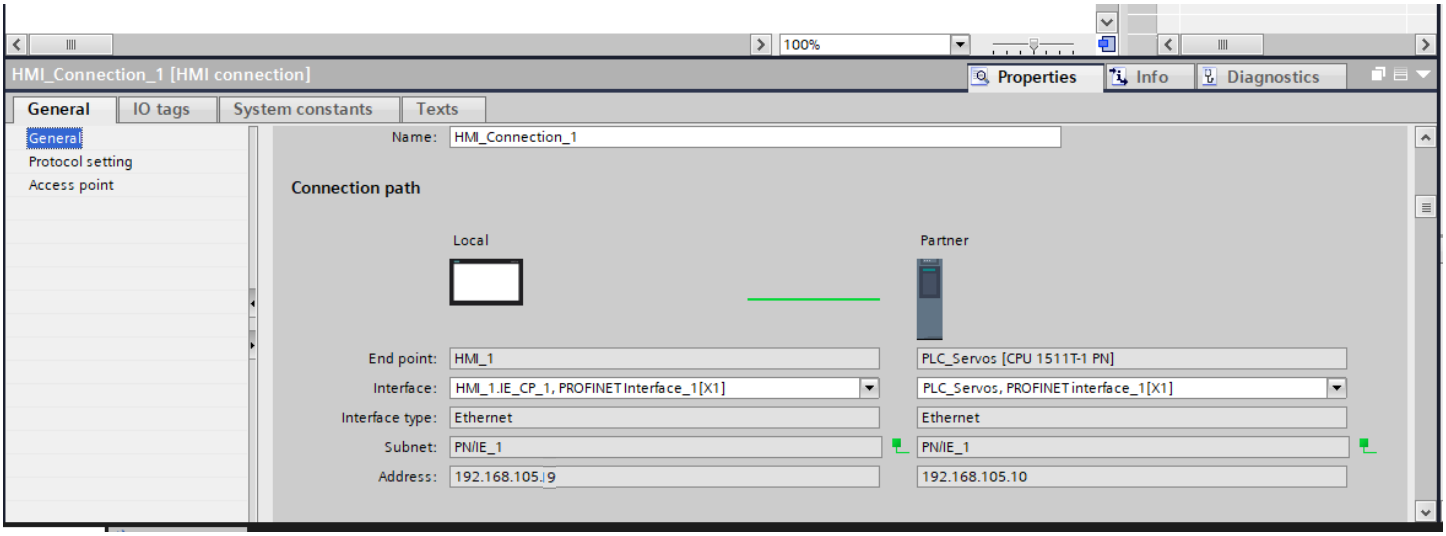


Ilustración 13Propiedades de la red HMI_Connection_1

La configuración IRT solo será entre el PLC y los servocontrolador, ya que, la HMI solo será de monitoreo y al pertenecer a la red PROFINET, es más que suficiente.

Nuevamente en la vista de redes, configurar los puertos del PLC y de los servocontroladores.

En el PLC:

En el rol de sincronización, asignarle maestro; en IO communication, send clock en 2.0000 ms

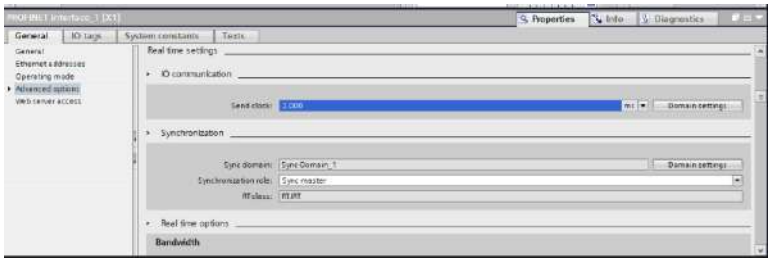


Ilustración 15 PLC_Servos -> Advance options -> IO communication: 2.000 ms / Synchronization: Sync master

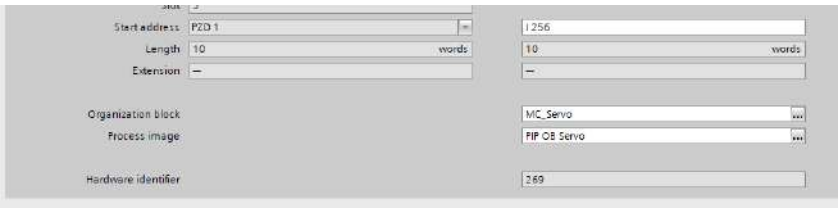


Ilustración 14 Configuración de bloque organizacional

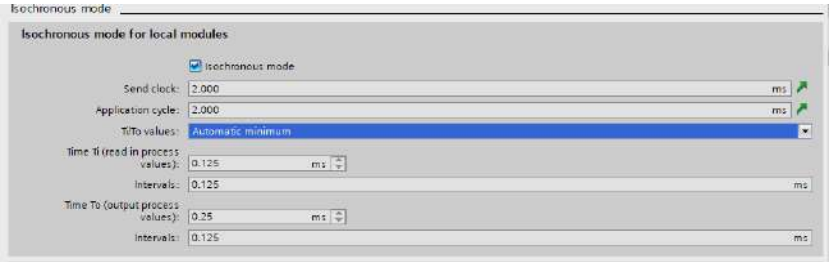


Ilustración 16 Propiedades en el modo isocrónico de cada servocontrolador

En el Servo_1 y Servo_2:

Crear el bloque organizacional y ligarlo al servocontrolador, solo es necesario crear un bloque, este mismo bloque puede ser compartido con los demás servocontroladores. En el modo de sincronización, en ambos servocontroladores, activar el modo automático en el Isochronous mode -> Ti/To values:

En el HMI no es necesaria una configuración adicional desde la vista de redes, sin embargo, para verificar que el lazo de comunicación se creó, desde el árbol del proyecto, dentro de la carpeta del HMI_1, dar doble click en la pestaña de “Connections” y ver la HMI_Connection_1

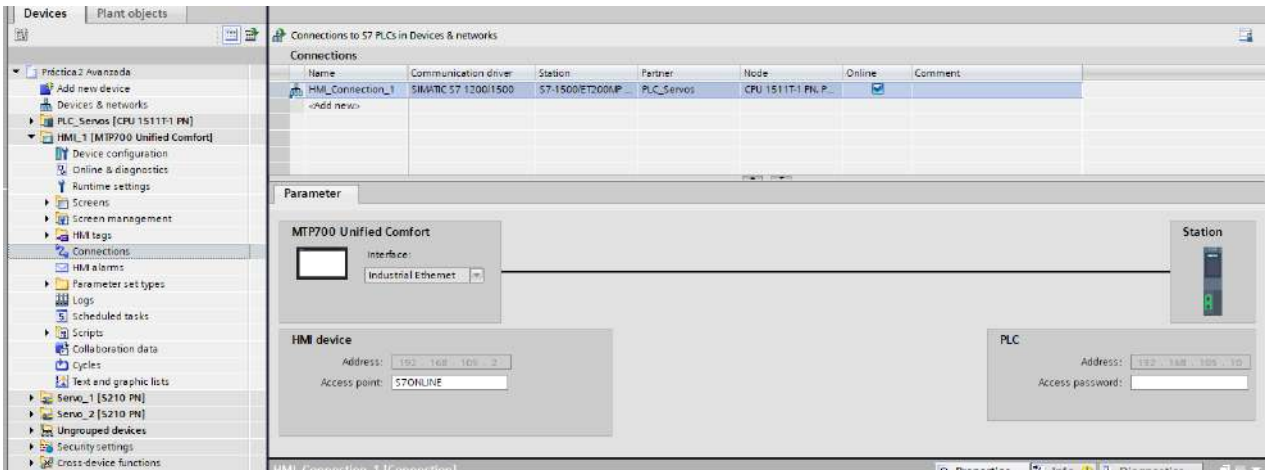


Ilustración 17 HMI -> Connections -> HMI_Connection_1

Objeto tecnológico de tipo maestro y seguidor

Los objetos tecnológicos van ligados a un servocontrolador, por lo tanto, para el primer objeto, se agrega uno de tipo Posición (PositioningAxis), se renombrará como “MasterAx”.

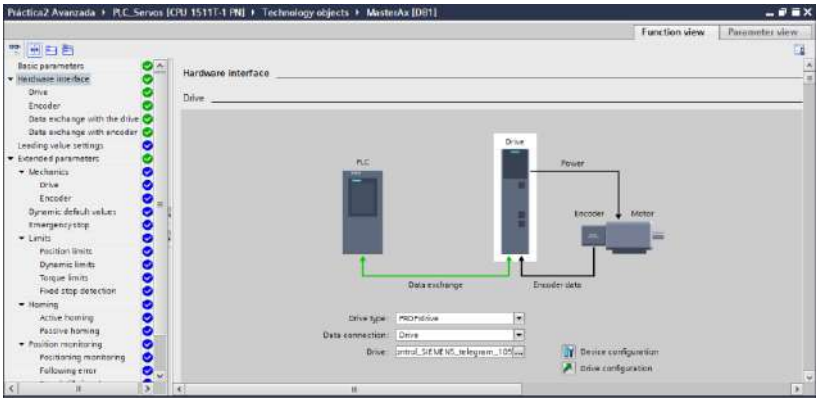


Ilustración 18 Configuración del MasterAx

En los parámetros básicos del objeto tecnológico;

- Axis type: rotary
- Activar casilla “Enable modulo” en Modulo

En Hardware interface, opción drive:

- Seleccionar el Servocontrolador 1

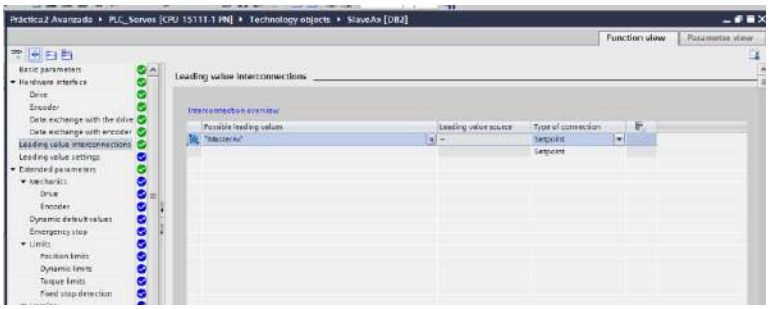


Ilustración 19 Declaración del eje maestro en el SlaveAx

Agregar un segundo objeto tecnológico, del tipo síncrono (Synchronous); nombrarlo SlaveAx. Este eje será del tipo “rotary” y activar la casilla “enable module”.

Para el Hardware interface, opción drive, será el Servo_2.

En la pestaña de “Leading value interconections”, seleccionar al eje maestro.

Descarga de proyecto a cada equipo

Asegurándose estar conectado a la red PLC_Servos (cualquier banda), compilar el proyecto del PLC, verificar que no existan errores de IP para la descarga y descarga los archivos mediante la interfaz de comunicación. Colocar el CPU en RUN.

De manera similar, cargar la configuración a cada uno de los servomotores (Servo_1 y Servo_2).

No olvidar hacer el one button Tuning para cada servomotor, con ello se verifica la correcta comunicación entre servocontroladores – servomotores. Posteriormente a guardar los datos, hacer un Jog en cada servomotor desde los objetos tecnológicos, dentro de la carpeta de objetos tecnológicos del controlador, activar el “Axis control panel” de cada servo para verificar comunicación Profinet.

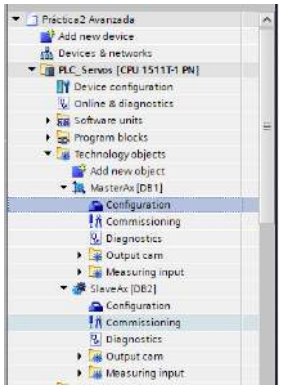


Ilustración 20 Commissionig -> Axis control panel

Librería CtrlAxis

Dentro de las herramientas de TIA Portal (ventana derecha), seleccionar la pestaña de Librerías, buscar el ícono “Open global library”; buscar la ubicación de la librería LAxisCtrl_DBAnyBased V19 añadirla al proyecto. Si es una versión anterior, TIA Portal se encargará de hacer el Upgrade correspondiente para poder ser compatible con el proyecto.

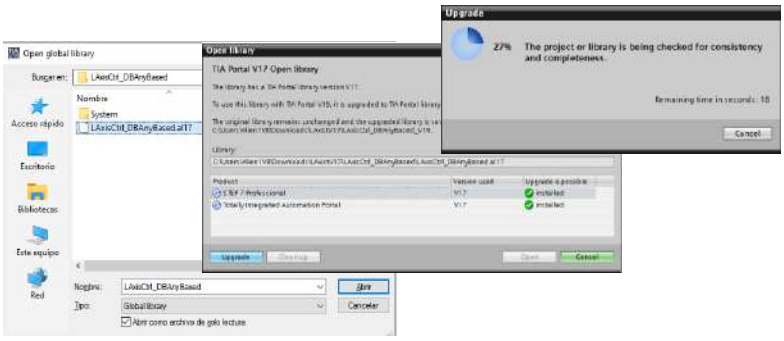


Ilustración 21 Upgrade de librería, de V17 a V19

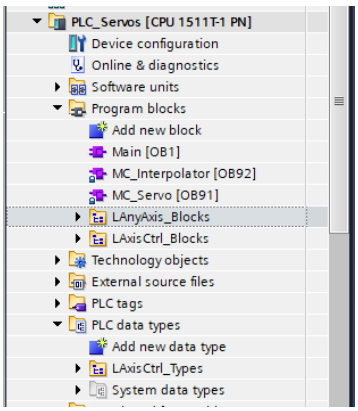


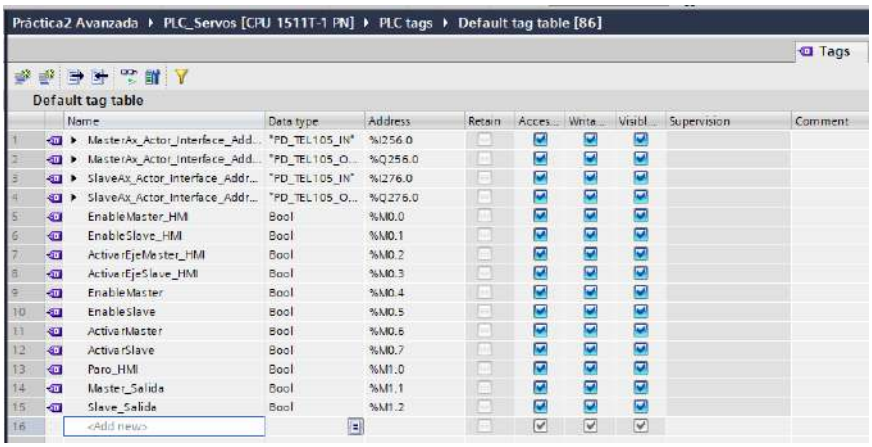
Ilustración 22 Program blocks y data types del PLC

Agregar los archivos de la librería en el árbol del proyecto:

LAxisCtrl	*Project tree*
LAxis Ctrl_Types ->	Programa blocks
LAxisCtrl_Blocks->	
S7-1500T_LAnyAxis_Blocks->	PLC data types

Rutina de control en Ladder y tags

En la tabla de entradas y salidas colocar variables de memoria correspondientes a la activación del sistema de monitoreo, es importante recalcar que la HMI no escribirá en identificadores de entrada, únicamente trabajará sobre variables de memoria, por ello se deben generar la lógica de programación en el Main mediante contactos que cambiarán de estado al recibir las señales de la HMI.



	Name	Data type	Address	Retain	Acces...	Write...	Visibl...	Supervision	Comment
1	MasterAx_Actor_Interface_Add...	"PD_TEL105_IN"	%I256.0		✓	✓	✓		
2	MasterAx_Actor_Interface_Add...	"PD_TEL105_O...	%Q256.0		✓	✓	✓		
3	SlaveAx_Actor_Interface_Addr...	"PD_TEL105_IN"	%I276.0		✓	✓	✓		
4	SlaveAx_Actor_Interface_Addr...	"PD_TEL105_O...	%Q276.0		✓	✓	✓		
5	EnableMaster_HMI	Bool	%M0.0		✓	✓	✓		
6	EnableSlave_HMI	Bool	%M0.1		✓	✓	✓		
7	ActivarEjeMaster_HMI	Bool	%M0.2		✓	✓	✓		
8	ActivarEjeSlave_HMI	Bool	%M0.3		✓	✓	✓		
9	EnableMaster	Bool	%M0.4		✓	✓	✓		
10	EnableSlave	Bool	%M0.5		✓	✓	✓		
11	ActivarMaster	Bool	%M0.6		✓	✓	✓		
12	ActivarSlave	Bool	%M0.7		✓	✓	✓		
13	Paro_HMI	Bool	%M1.0		✓	✓	✓		
14	Master_Salida	Bool	%M1.1		✓	✓	✓		
15	Slave_Salida	Bool	%M1.2		✓	✓	✓		
16	<Add new>				✓	✓	✓		

Ilustración 23 Tags de lectura y escritura

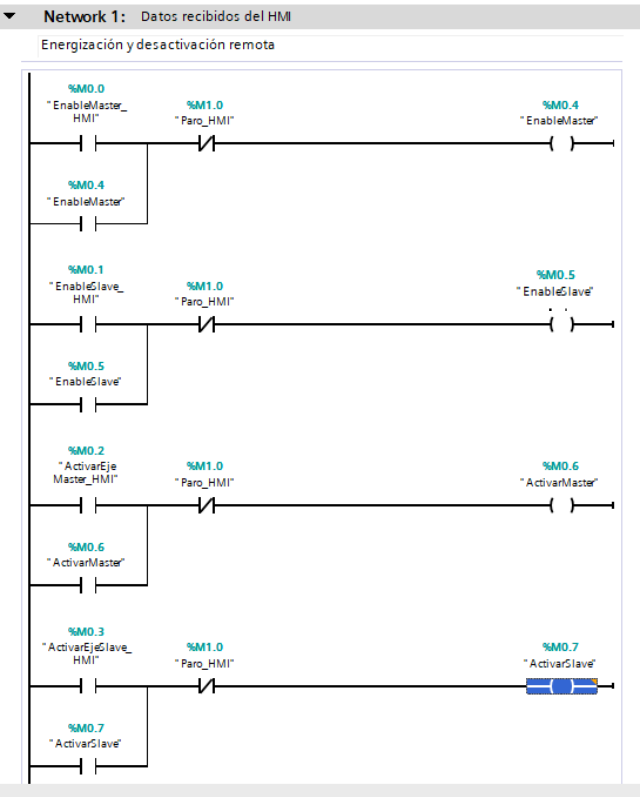


Ilustración 24 Datos recibidos del HMI

En la rutina principal del PLC, generar un enclavamiento de señales que serán generadas directamente desde la HMI mediante botones en configuración Push Botton.

Cada una de las señales de salida son referentes a la activación del eje, tanto maestro como seguidor, y posteriormente, la activación de la instrucción “moveVelocity” dentro del DB de configuración que a continuación se añade. (Ilustración 24)

En un nuevo Network de la rutina principal, añadir el DB correspondiente de la librería LAxisCtrl al peldaño y colocar el nombre “MasterAxis_DB”. Sobre el mismo peldaño, agregar un control para el seguidor (SlaveAxis_DB). (Ilustración 25)

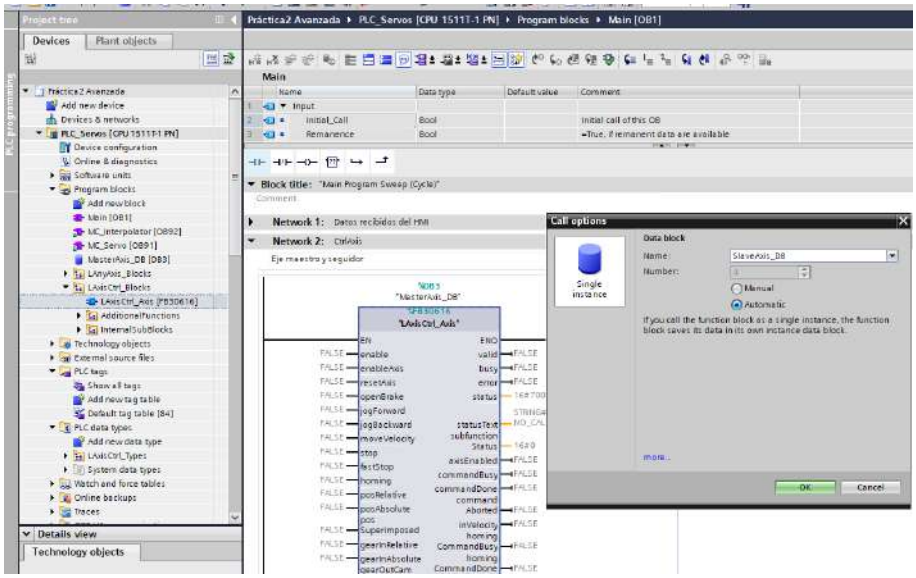


Ilustración 25 Bloque de control para eje maestro y para eje seguidor

Agrega un nuevo DB (AxesConfig_DB) para almacenar los parámetros de interés de cada uno de los ejes. Dentro del DB AxesConfig_DB, se añaden variables del tipo LAxisCtrl_typeAxisConfig, tanto para el maestro (MasterAx_Config), como para el seguidor (SlaveAx_Config).

Al terminar de añadir los archivos, es importante compilar el programa, iniciando desde el AxesConfig_DB, y luego el Main, con ello evitar errores al momento de cargar el proyecto al controlador.

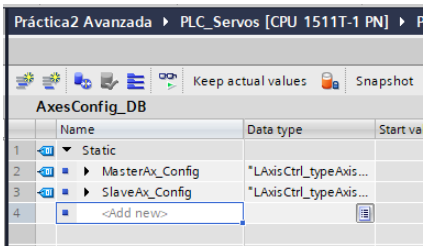


Ilustración 26 AxesConfig_DB -> Master / Slave

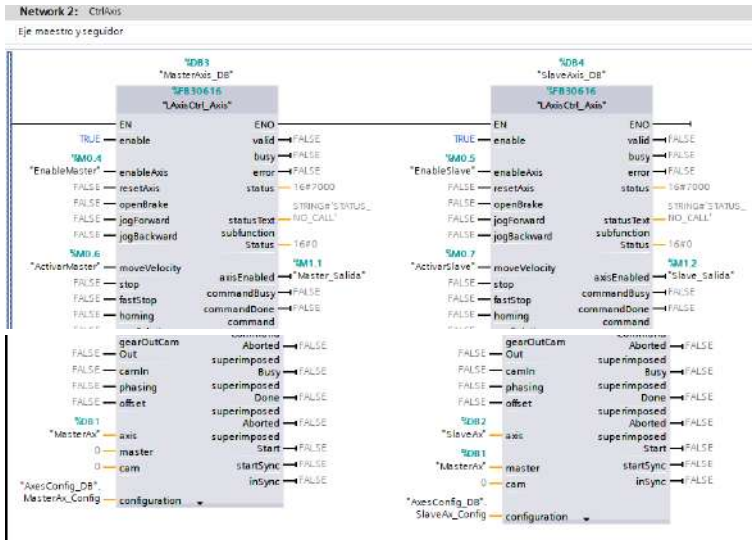


Ilustración 27 Variables en DB de maestro y seguidor

De nuevo en el Main, se relacionan las variables y los DB a los bloques de funciones como se muestra en las ilustraciones:

Master
TRUE -> enable
EnableMaster -> enableAxis
ActivarMaster -> moveVelocity
axisEnabled (salida) -> Master_salida
MasterAx -> axis

AxesConfig_DB.MasterAx_Config -> configuration

Slave
TRUE -> enable
EnableSlave -> enableAxis
ActivarSlave -> moveVelocity
axisEnabled (salida) -> Slave_salida
SlaveAx -> axis
MasterAx -> master
AxesConfig_DB.SlaveAx_Config -> configuration

Una vez realizados los cambios, compilar el Main. Regresar al AxesConfig_DB; dentro del él, se encuentra MasterAx_Config y SlaveAx_Config; desplegar la pestaña de MasterAX_Config para encontrar los parámetros de:

Jog

Velocity: 100.0

moveVelocity
Velocity: 200.0
velocityChanceOnTheFly: TRUE

A continuación, desplegar la pestaña de SlaveAX_Config para encontrar los parámetros de:

Jog	*moveVelocity*
Velocity: 100.0	Velocity: 200.0 velocityChanceOnTheFly: TRUE

No olvidar colocar los valores de velocidad en formato de flotantes, ya que, de no ser así, marcará un error TIA Portal.

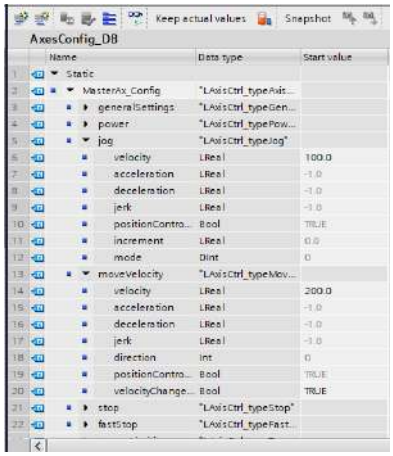


Ilustración 28 Parámetros de configuración MasterAx

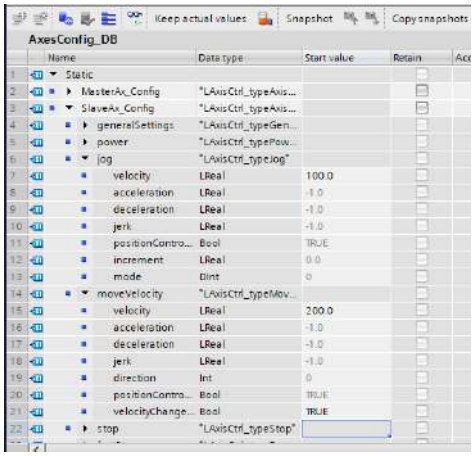


Ilustración 29 Parámetros de configuración SlaveAx

Pantallas de monitoreo

En el árbol del proyecto, en la carpeta de HMI_1, generar una nueva pantalla, la cual será designada como “Main”, además, generar otra pantalla donde se realizará el monitoreo del sistema.

En la panta principal, elaborar un diseño similar al de la práctica 1 (Control de velocidad mediante una HMI).

No olvidar colocar el botón de “Stop Runtime” que permite detener la ejecución de la aplicación en cualquier momento.

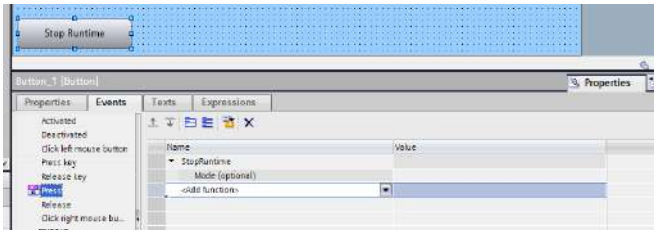


Ilustración 31 Botón -> "Stop Runtime" -> Events -> Press -> StopRuntime

Ilustración 30 Pantalla principal

Generar una segunda pantalla, la cual será de monitoreo; dentro la pantalla principal colocar un botón que permita el cambio a la segunda pantalla (Monitoring).

Dentro del monitoreo no olvidar colocar un botón que direccione a la pantalla principal

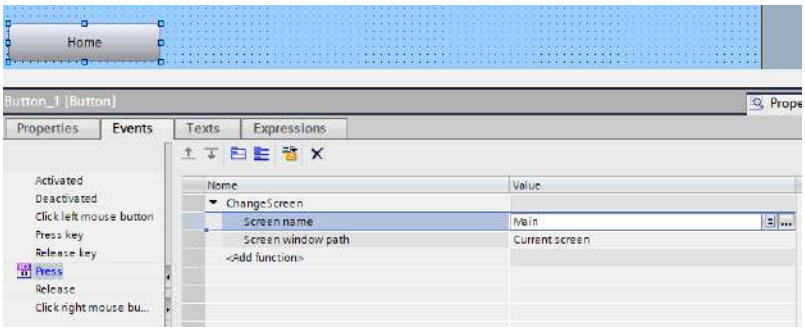


Ilustración 32 Botón -> "Home" -> Events -> Press -> ChangeScreen -> Screen name: Main

Al igual que la tabla de variables del PLC, la HMI requerirá de una tabla de variables para registrar los eventos sucedidos mediante los botones y para el caso del monitoreo, variables donde se refleje el estado de los actuadores (servomotores).

Dentro de la carpeta de la HMI_1 en el árbol del proyecto, desplegar la subcarpeta “HMI tags”; dar doble click en la opción “Default tag table”.

Dentro de la tabla colocar las variables de interés, en la ilustración 33, se muestran algunas variables, ya que es de suma importancia mostrar la manera de relacionar las variables de la HMI con el PLC.

Al seleccionar una variable de interés, existen dos maneras de direccionarlas; en la práctica 1, se utilizó las opciones de la tabla, a continuación, se muestra cómo realizarlo mediante el inspector.

Se selecciona una variable, en este caso es “EnableMaster”, en las propiedades de la variable, en la pestaña “General”, dar click en los tres puntos de la opción “Connection” y seleccionar la red “HMI_Connection_1”, de inmediato, identificará que el PLC_Servos es la fuente de comunicación, por lo que espera el tag de referencia de la tabla de tags (del PLC), sin embargo, puede ser en general cualquier dato contenido en el PLC (que sea del identificador memoria).

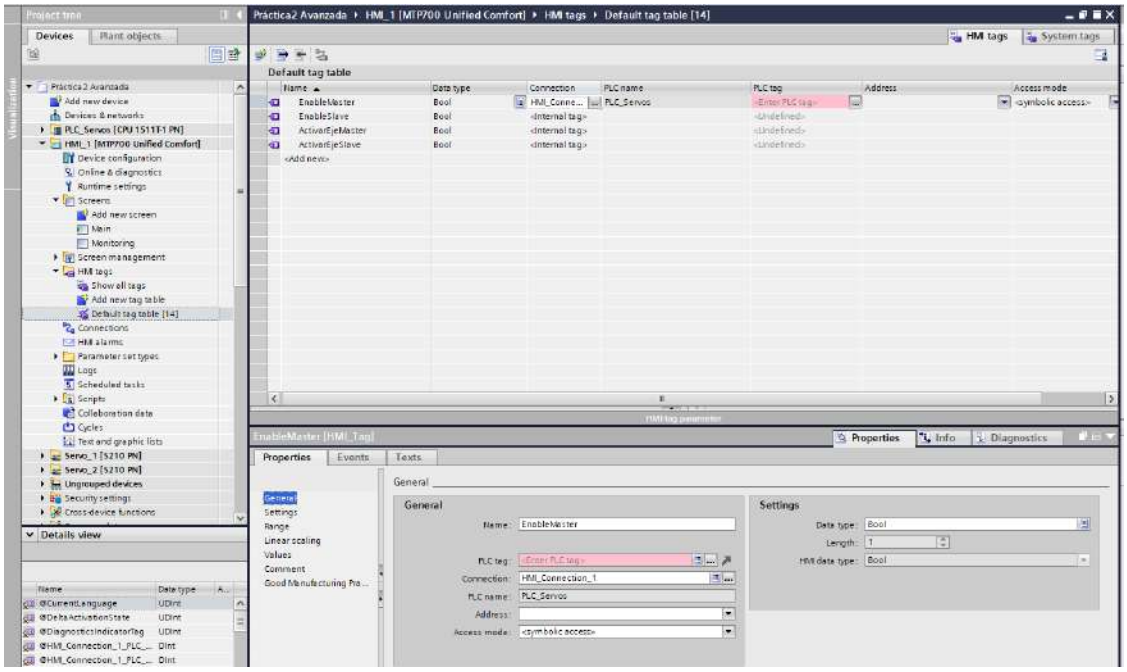


Ilustración 33 Selección de tag en HMI -> Properties -> General -> Connection -> HMI_Connection_1 -> Seleccionar tag del PLC

En la opción “PLC tag”, dar click en los tres puntos, buscar el tag de referencia. Ya que es un tag de escritura, se debe cambiar el “Access mode” por <absolute access>.

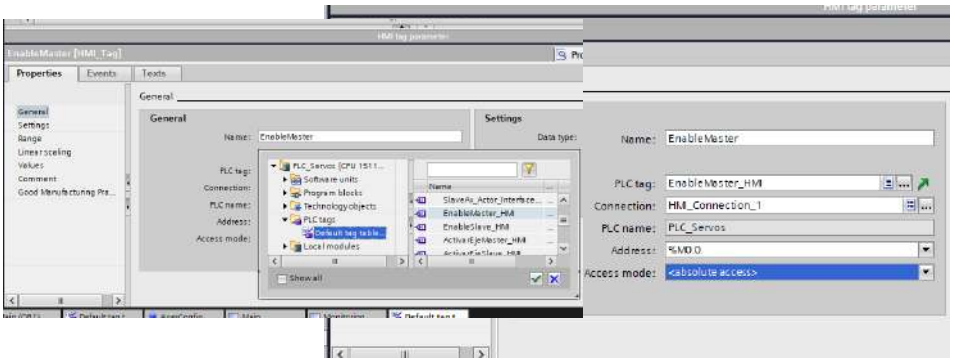


Ilustración 34 Referencia de tag del PLC -> Access mode: <absolute access> (aparecera la dirección del elemento en la que se almaena en el PLC)

Una vez referenciado el tag, es importante configurar la velocidad de actualización del dato en la HMI; por default, el valor es de 1 [s], no es un valor desfavorable, sin embargo, en procesos que requieren que la recepción de datos sea los más ágil posible, se disminuirá el tiempo a 100 [ms]

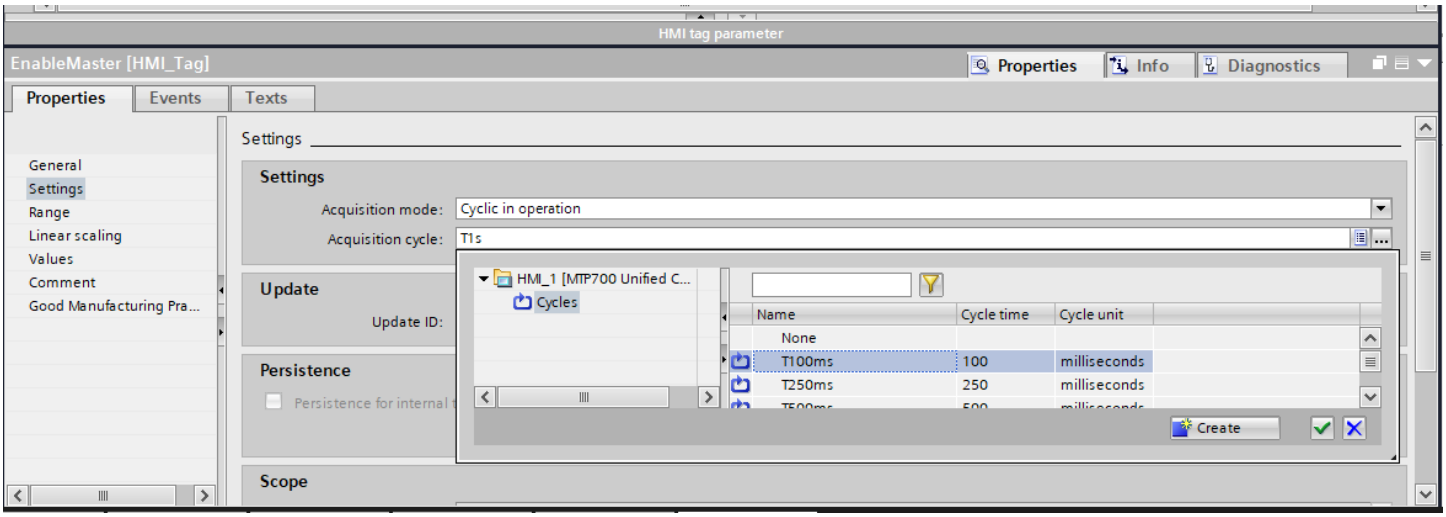
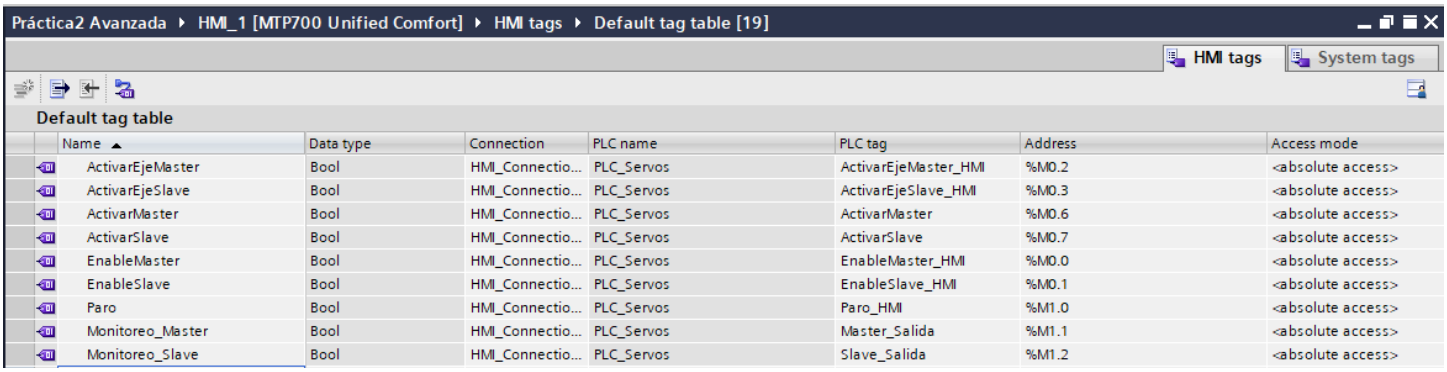


Ilustración 35 Se mantiene seleccionado el tag de interés -> Properties -> Settings -> Acquisition cycle: T100ms (colocar en todas la variables)

La tabla de tags que la HMI requerirá se muestra en la ilustración 36:



Name	Data type	Connection	PLC name	PLC tag	Address	Access mode
ActivarEjeMaster	Bool	HMI_Connectio...	PLC_Servos	ActivarEjeMaster_HMI	%M0.2	<absolute access>
ActivarEjeSlave	Bool	HMI_Connectio...	PLC_Servos	ActivarEjeSlave_HMI	%M0.3	<absolute access>
ActivarMaster	Bool	HMI_Connectio...	PLC_Servos	ActivarMaster	%M0.6	<absolute access>
ActivarSlave	Bool	HMI_Connectio...	PLC_Servos	ActivarSlave	%M0.7	<absolute access>
EnableMaster	Bool	HMI_Connectio...	PLC_Servos	EnableMaster_HMI	%M0.0	<absolute access>
EnableSlave	Bool	HMI_Connectio...	PLC_Servos	EnableSlave_HMI	%M0.1	<absolute access>
Paro	Bool	HMI_Connectio...	PLC_Servos	Paro_HMI	%M1.0	<absolute access>
Monitoreo_Master	Bool	HMI_Connectio...	PLC_Servos	Master_Salida	%M1.1	<absolute access>
Monitoreo_Slave	Bool	HMI_Connectio...	PLC_Servos	Slave_Salida	%M1.2	<absolute access>

Ilustración 36 Valores de monitoreo

Donde:

- EnableMaster/Slave -> habilitarán de eje (enableAxis del DB).
- ActivarEjeMaster/Slave -> habilitarán función de movimiento en verdadera (moveVelocity del DB).

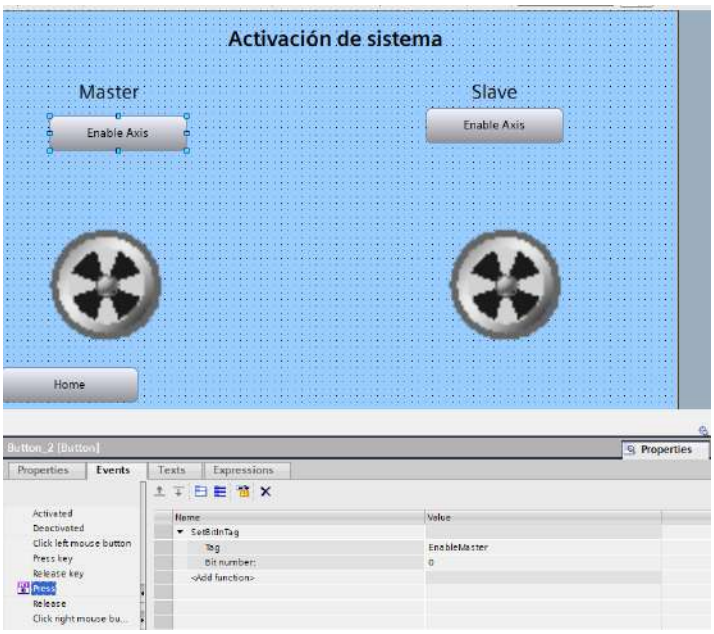


Ilustración 37 Botón "Enable Axis" -> Properties -> Events -> Press -> SetBitInTag -> Tag: EnableMaster

En la pantalla de monitoreo, se puede colocar un título referente a la activación del sistema de monitoreo. Además de añadir algunos elementos como botones y elementos gráficos que permiten atribuir su cambio visual a un evento físico.

En la pantalla de monitoreo se generan cinco botones, en configuración de pulsadores, que permitirán activar y desactivar el sistema de servomotores.

Para obtener la configuración de botón pulsador, botón de activación del eje maestro, se arrastra un botón, se renombra el botón (Enable Axis) y en el inspector, ventana de eventos, colocarle una instrucción al presionar el botón, la cual mandará un bit booleano a la variable “EnableMaster” (previamente configurado en la HMI tags).

Hasta el momento, solo se ha configurado la mitad del evento sobre el botón; al dejar de presionar el botón deberá de reiniciar su estado, por lo que en el comando de Release, colocar un reseteo de esta variable asociada o bit de trabajo.

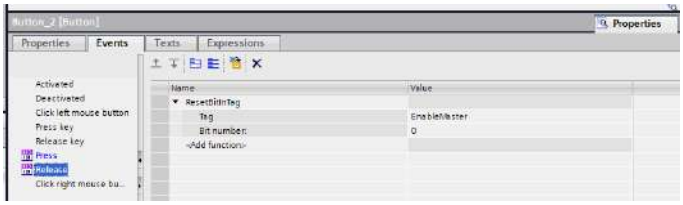


Ilustración 38 Release -> ResetBitInTag -> Tag: EnableMaster

Realizar la configuración con el botón del Slave, pero con el tag “EnableSlave”.

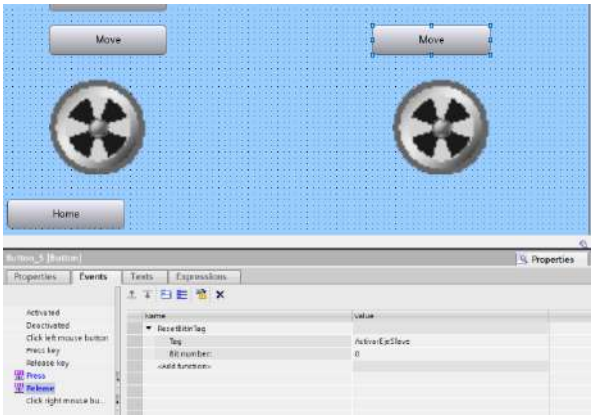


Ilustración 39 Botón "Move" -> Properties -> Events -> Press: SetBitInTag: ActivarEjeSlave / Release: ResetBitInTag: ActivarEjeSlave

De manera similar, los botones de “Move”, serán pulsadores, pero el Tag referenciado será “ActivarEjeMaster” y “ActivarEjeSlave”

Configurar el botón de cambio de ventana, para regresar a la pantalla principal (Home).

El agregar un elemento gráfico, como el que se ha mostrado en las capturas de pantalla, se realiza al desplegar “Graphics” del toolbox de la derecha; arrastrar el ícono a la pantalla de interés.

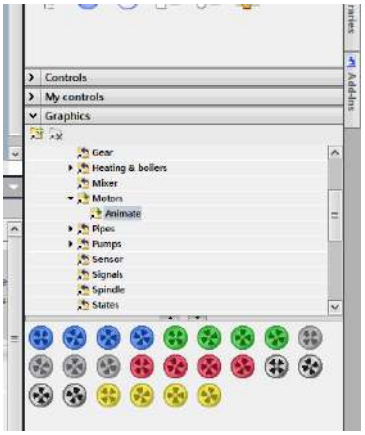
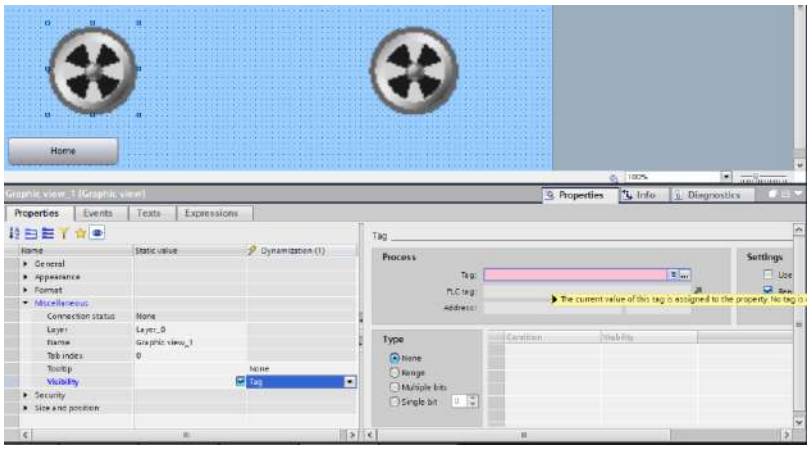


Ilustración 40 Graphics -> Motors -> Animate -> Arrastrar la animación de conveniencia

TIA Portal también ofrece la opción de importar más elementos gráficos (verificar documentación correspondiente).

Animación de visualización



Para generar una representación visual del estado del actuador, se puede hacer uso de las propiedades de visualización del objeto, lo cual es, hacer visible o invisible una imagen de acuerdo al valor que se lee de un tag.

Al seleccionar un objeto, como el que representa al actuador maestro, en la pestaña de propiedades del inspector, se puede desplegar la opción “Miscellaneous”, aquí se encuentra una herramienta que se llama “Visibility”, por default, tiene activada la casilla “Staticvalue”, en la casilla contigua, “Dynamization”, dar click en el menú y seleccionar “Tag”, el inspector se segmentará en dos interfaces, la primera es donde se acaba de activar la casilla y la segunda, es donde se activan las configuraciones adicionales. En esta segunda interface, seleccionar el tag de interés (seleccionar de HMI tag table).

Una vez seleccionado el tag de interés, en la pestaña “Type”, se configura el rango de trabajo, en este caso, se requiere que cuando el valor booleano este en cero (apagado), se vea el ícono, y cuando el valor pase a uno (encendido), no sea visible (se desmarca la casilla).

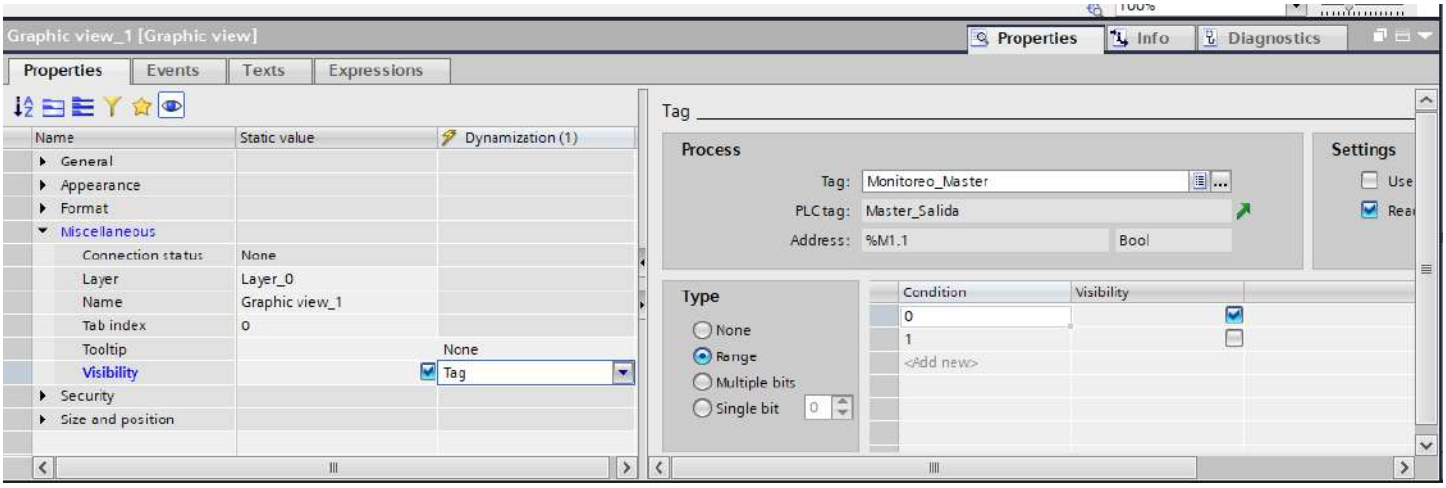


Ilustración 41 Configuración de visibilidad de un objeto gráfico

Realizar la misma configuración para el elemento gráfico del motor seguidor.

A continuación, agregar otro ícono gráfico, pero de propiedades diferentes al anterior, con el objetivo de apreciar una apariencia diferente (un cambio de color), que represente que el actuador ha cambiado de estado, en funcionamiento.

Ahora en las propiedades de visibilidad, cuando sea cero, no se verá y cuando sea uno, será visible.

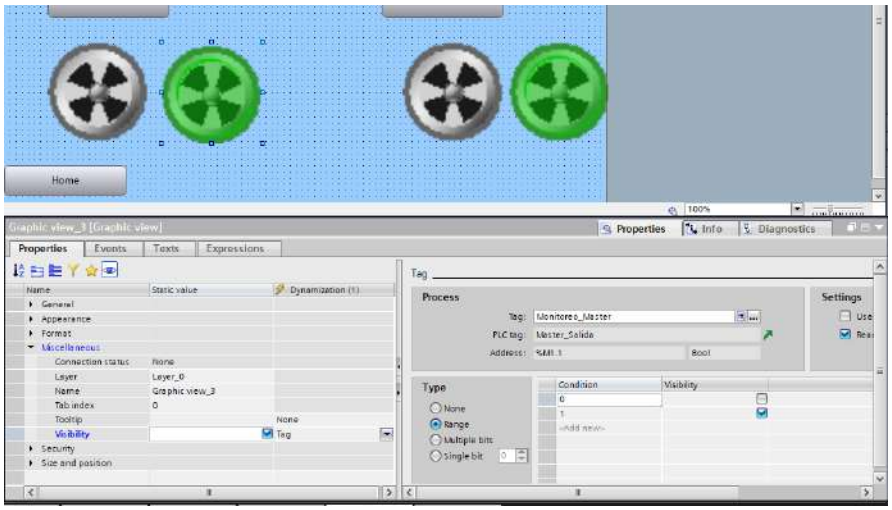


Ilustración 42 Animación de visibilidad para estado de encendido del actuador

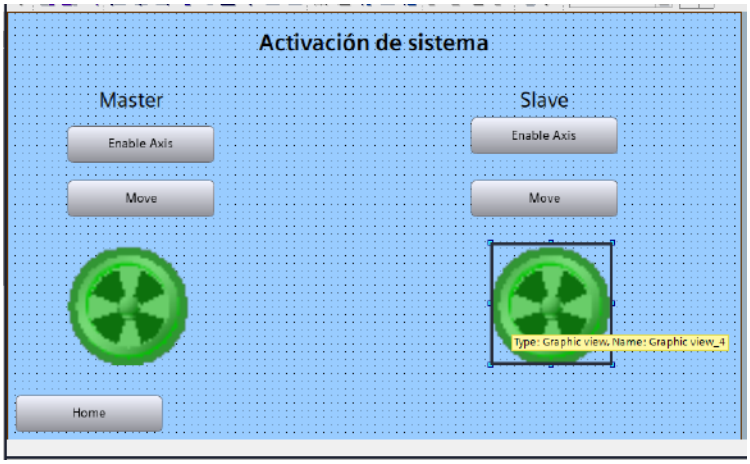


Ilustración 43 Centrar y sobreponer los elementos gráficos

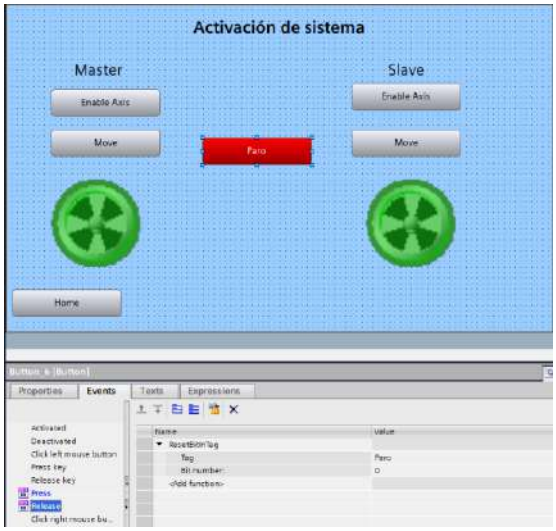


Ilustración 44 Colocar un botón de paro para detener el sistema

Entregables de la práctica

1. Código de Ladder comentado.
2. Pantallas realizadas en la HMI.
3. Imágenes del cambio de estado en el monitoreo de la HMI (fotografía de pantalla en ejecución).

VI. Conclusiones y observaciones.

_____

_____

_____

_____

Referencias

- McGowan, M., & Phillips, L. (2017). Implementing HMI in printing processes: Enhancing quality control. *Journal of Manufacturing Processes*, 29, 246-254. https://doi.org/10.1016/j.jmapro.2017.06.023
- Yang, J., & Zhang, W. (2018). Human-machine interface design for automated assembly systems. *International Journal of Advanced Manufacturing Technology*, 97 (1-4), 241-254. https://doi.org/10.1007/s00170-018-1963-5
- Carrillo, J., & González, A. (2015). *Sistemas de control industrial*: Teoría y práctica. McGraw-Hill.
- López, M. J., & Rodríguez, S. (2017). Automatización y control en la industria de la impresión. *Revista de Ingeniería y Tecnología*, 8 (2), 45-52.

Práctica 3: Proceso Industrial completo

Rúbrica de evaluación

CONCEPTOS, RUBROS O ASPECTOS A EVALUAR	BUENO (2 PUNTOS) Completo entendimiento del problema, realiza la actividad cumpliendo todos los requerimientos.	REGULAR (1 PUNTO) Bajo entendimiento del problema, realiza la actividad cumpliendo algunos de los requerimientos.	NO ACEPTABLE (0 PUNTOS) No demuestra entendimiento del Problema o de la actividad.
1. Seguridad en la ejecución de la actividad	Identifica correctamente los peligros y fuentes de energía, minimiza los riesgos aplicando las medidas de control, realiza la verificación y firma con su nombre.	Identifica parcialmente los peligros, sin aplicar todas las medidas de control.	No aplica ninguna medida de control, no verifica y no firma.
2. Ejecución de la práctica	Muestra un entendimiento completo durante el desarrollo de las actividades, la práctica cumple con todos los requerimientos	Muestra un entendimiento moderado durante el desarrollo de las actividades, la práctica no cumple con todos los requisitos	No demuestra entendimiento de las actividades, la práctica no cumple con los requisitos.
3. Tabla de ordenamiento de variables	Genera la tabla para las entradas y salidas e incluye dirección, símbolo y descripción de cada variable.	Genera una tabla incompleta o con información errónea.	No desarrolla la tabla de ordenamiento de variables.
4. Interfaz de control y supervisión	Diseña la interfaz de control y supervisión cumpliendo con todas las especificaciones solicitadas.	Incluye algunas de las especificaciones solicitadas con carencias u omisiones.	No reporta la interfaz de control y supervisión.
5. Repetibilidad y resistencia a fallos	El sistema opera cíclicamente sin ningún inconveniente, el sistema no falla ante perturbaciones (activaciones erróneas de usuarios).	El sistema se detiene en algún punto de operación, el sistema se detiene ante perturbaciones.	El sistema no es cíclico, se detiene el proceso ante cualquier perturbación.

I. Seguridad en la ejecución

	Peligro o fuente de energía	Riesgo asociado	Medidas de control	Verificación
1ro	Voltaje alterno	Electrocución	Identificar los puntos energizados antes de realizar la actividad y evitar contacto	☐
2do	Voltaje alterno	Electrocución	Identificar los puntos energizados antes de realizar la actividad y evitar contacto	☐
3ro	Voltaje continuo	Daño a equipo	Verificar polaridad y nivel antes de realizar la conexión del equipo o dispositivo	☐
4to	Neumática	Lesiones por partes en movimiento y/o por mangueras o tapones sueltos	Verificar la firmeza de las conexiones neumáticas antes de energizar el proceso. Identificar la trayectoria de movimiento y evitar contacto durante el funcionamiento	☐
5to	Exposición a ruido superior a 75dB	Lesiones auditivas	Verificar la firmeza de las conexiones neumáticas antes de energizar el proceso. Utilizar canceladores de ruido	☐
		Apellidos y nombres:	_____	

II. Objetivos de aprendizaje

Objetivo general: Realizar la simulación de un proceso industrial que integre controladores industriales (PLC), actuadores neumáticos, servomotores, interfaces humano máquina (HMI), botones eléctricos y un indicador visual mediante el uso de control en red.

Objetivos específicos:

- El alumno deberá ser capaz de simular un proceso industrial que integre actuadores neumáticos y servomotores, utilizando un control basado en red.
- El alumno entenderá el envío y recepción de datos mediante las instrucciones PUT/GET
- El alumno comprenderá la importancia de la integración de diferentes tipos de actuadores en un proceso industrial y cómo aplicar estas técnicas en un entorno real.
- El alumno será capaz de diseñar una interfaz HMI efectiva para el control y monitoreo de un proceso industrial simulado.

III. Material y equipo

Ilustración 1 Computadora

Ilustración 2 Controlador Siemens S7-1500T

Ilustración 3 Servocontrolador Sinamics S210

Ilustración 4 Servomotor Simotics S-1FK2

Ilustración 5 PLC S7-1200

Ilustración 6 Mangueras neumáticas

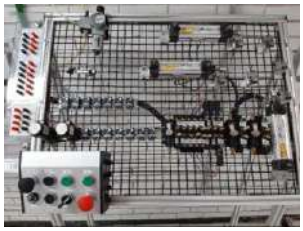


Ilustración 7 Mesa neumática

Ilustración 8 HMI MTP 700 Unified Comfort

Ilustración 9 Torreta industrial de tres colores

IV. Introducción

En la automatización industrial moderna, la integración de distintos tipos de actuadores, como los neumáticos, los motores de inducción, electroválvulas, servomotores y los indicadores luminosos como lámparas o torretas industriales, son esenciales para lograr que la manufatura de algún producto se realice de manera rápida y con la menor intervención de personal para abaratar el proceso. Sin embargo, para que estos sistemas trabajen de manera coordinada, es fundamental que se comuniquen a través de una red industrial bien estructurada. Las redes permiten no solo el control y monitoreo de los dispositivos, sino también la recopilación y análisis de datos en tiempo real, lo que mejora la toma de decisiones y la optimización de los procesos.

Ilustración 10 Sistema de mezclado para refrescos

Ilustración 11 SCADA PepsiCo

Un ejemplo claro de esta integración se encuentra en el embalaje automatizado, donde servomotores regulan la velocidad y posición de los productos en una banda transportadora, mientras que los cilindros neumáticos se encargan de empujar, prensar o sellar los paquetes. Sin una correcta integración de estos sistemas, podrían producirse errores como productos mal sellados o alineaciones incorrectas en la línea de producción, por lo que una HMI permite a los operadores visualizar y ajustar parámetros clave en tiempo real, garantizando que el proceso se mantenga estable.

Para que todos estos dispositivos se comuniquen de manera efectiva, es necesario el uso de redes industriales organizadas en distintos niveles:

- Red Local de Control (Nivel de Célula o Máquina): Se encarga de la comunicación entre sensores, actuadores, servomotores y PLCs dentro de un sistema o máquina específica. Tecnologías como PROFINET, EtherNet/IP o Modbus son comúnmente utilizadas en esta capa, ya que permiten tiempos de respuesta rápidos y una integración flexible.
- Red a Nivel de Proceso (Supervisión y Gestión): Permite la conexión entre varias estaciones de trabajo, líneas de producción y sistemas de monitoreo. En este nivel, se integran herramientas como SCADA y bases de datos que almacenan información sobre la producción para su análisis y optimización.

V. Desarrollo de la actividad

Automatización de un sistema de fabricación de envases de plástico

En una fábrica de producción de envases plásticos, donde se requiere una combinación de servomotores, actuadores neumáticos y sistemas HMI para coordinar el proceso de inyección, y transporte:

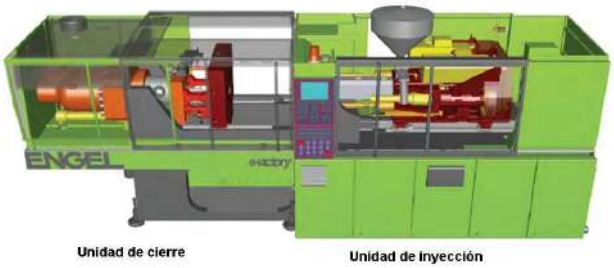


Ilustración 12 Sistema de inyección

El sistema se arranca de forma física o remota, a continuación, mediante un segundo accionamiento (físico o remoto), el sistema comienza a funcionar.

Dos cilindros neumáticos posicionan el molde bajo un inyector; a continuación, un tercer cilindro inyecta el material durante 3 segundos, posteriormente el cilindro de inyeccción regresa a su posición inicial, los cilindros encargados del molde se desactivan, haciendo que la pieza caiga sobre una banda transportadora y de inmediato, la banda arranca y saca el producto hasta la siguiente estación de trabajo, este proceso lo realiza durante 30 segundos. Toda esta operación es ciclica hasta que se pulse el botón de paro (remoto o físco). Si el sistema esta encendido pero no se activa el segundo accionamiento, un indicador visual estará presente (indicador ambar), si el sistema esta siendo ejecutado, el indicador verde se activará, en caso de un paro, el sistema activará una luz roja.

Todo este proceso es supervisado por un PLC S7-1500, que se encarga de coordinar cada dispositivo a través de una red local de control, sin embargo, el accionamiento de los cilindros es realizado por un PLC S7-1200; la HMI proporciona a los operadores una interfaz intuitiva para monitorear y activar secuencias en tiempo real.

Solución

Conexión de entre un PLC 1500 y un PLC 1200 de siemenes.

Generar un proyecto en el cual se tenga un CPU S7-1500T, dos servomotores S210 y una pantalla MTP 700 Unified Comfort Panel. En el PLC, activar la configuración PUT/GET.

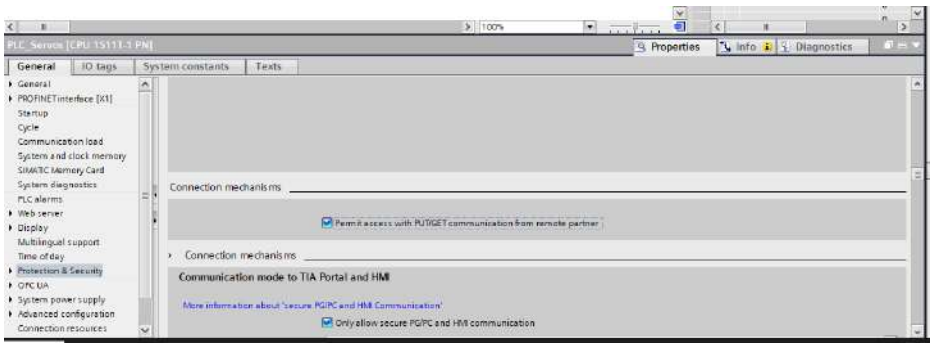


Ilustración 13 S7-1500T -> Inspector -> General -> Protection & Security -> Connection mechanisms -> (habilitar casilla) Permit access with PUT/GET communication from remote partner

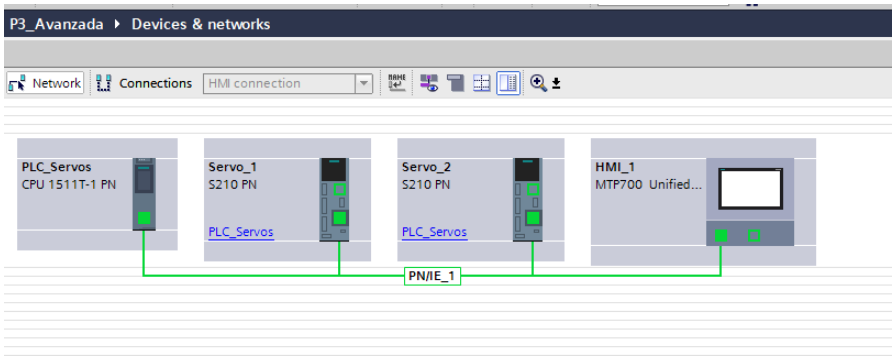


Ilustración 15 Network view; conexión PROFINET / Red HMI_Connection_1

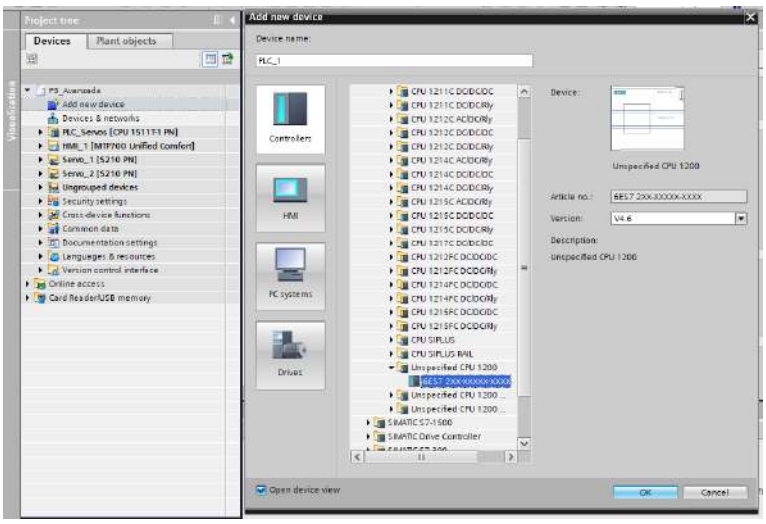


Ilustración 14 Add new device -> S7-1200 Unspecified -> Ok

Al proyecto, agregar un nuevo dispositivo, en este caso se tratará de un S7-1200; debido a que el PLC S7-1200 trabaja con versiones de software anterior a la versión 19; utilizar la autodetección de TIA Portal para cargar estos parámetros contenidos en el equipo.

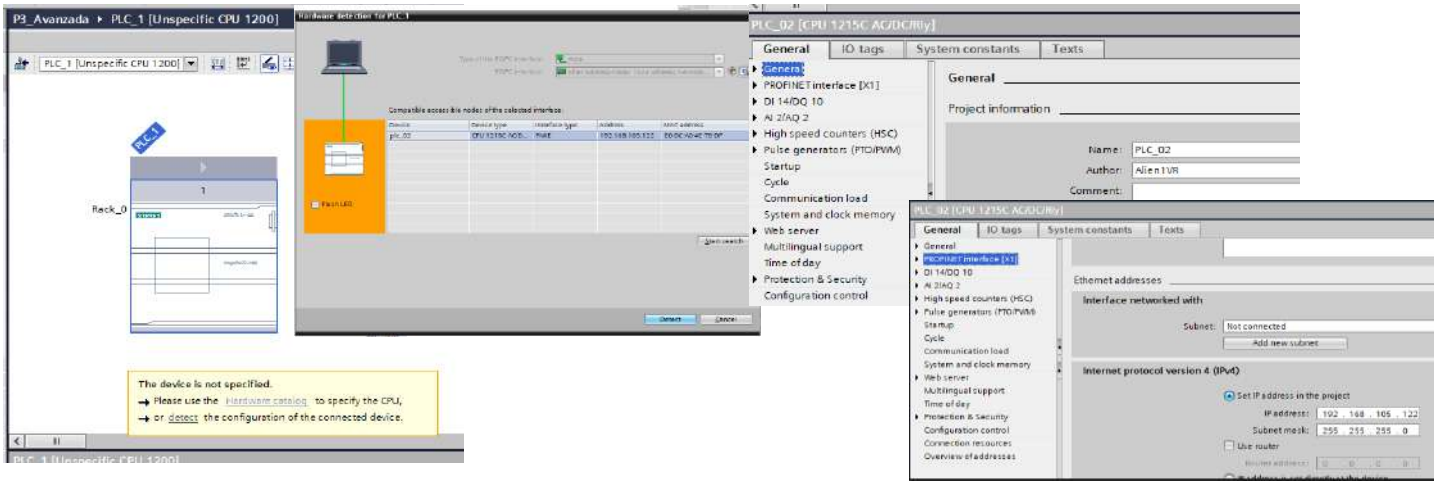


Ilustración 17 Detect -> Start search (elegir el equipo disponible) -> Nombrar al equipo: PLC_02 -> Verificar que la IP sea 192.168.105.122

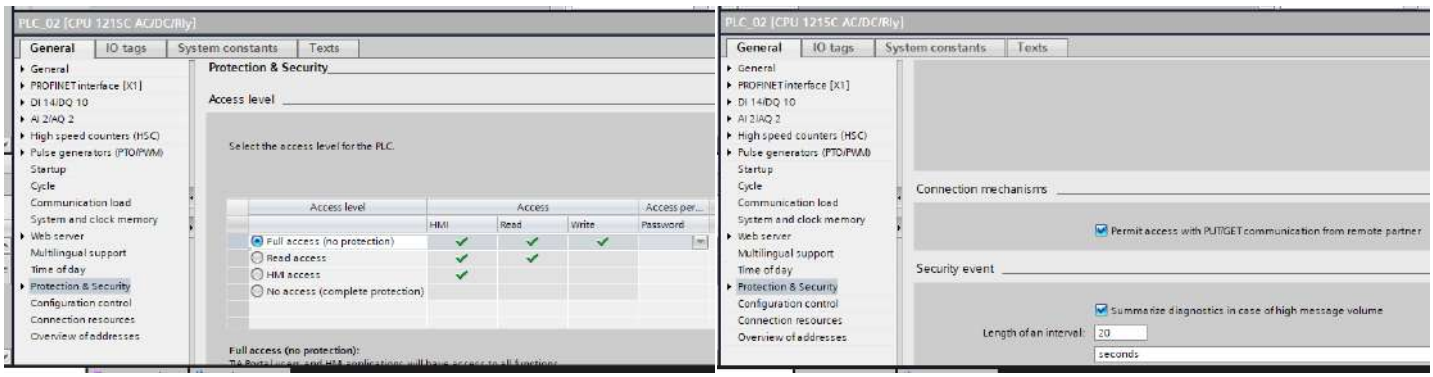


Ilustración 16 PLC_02 -> Protection & Security -> Access level: Full access / Connection mechanisms -> (actvar casilla) Permit access with PUT/GET communication from remote partner

Desde la vista de redes, conectar el PLC_02 a la red PROFINET

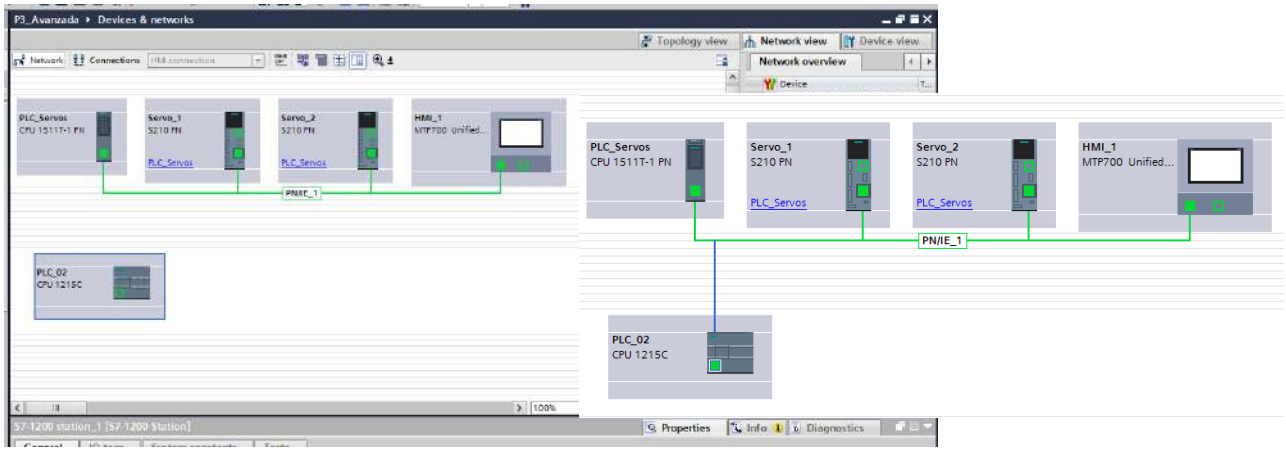
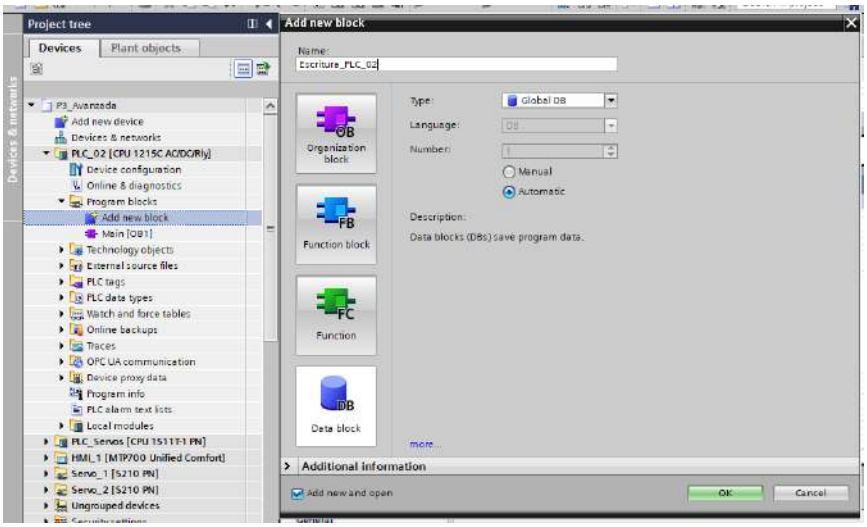


Ilustración 18 Vista general de los 5 elementos de trabajo

Integración de funciones PUT/GET

Ahora que ambos controladores se encuentran en el lazo de comunicación, crear dos DB de comunicación, en ellos se hará el envio y la recepción de datos del PLC_Servos al PLC_02.



De acuerdo al controlador en el que se esté, se nombrará al DB, por ejemplo, para el PLC_02, el DB de escritura se llamará: Escritura_PL_02".

Cuando el DB este creado, abrir el menú de propiedades, deshabilitar la casilla de "Optimized block access"

Repetir los pasos para el bloque de "Lectura" en el PLC_02.

Generer los bloques de lectura y escritura en el PLC_Servos.

Ilustración 19 PLC_02 -> Program blocks -> Add new block -> DB-> "Escritura_PLC_02"

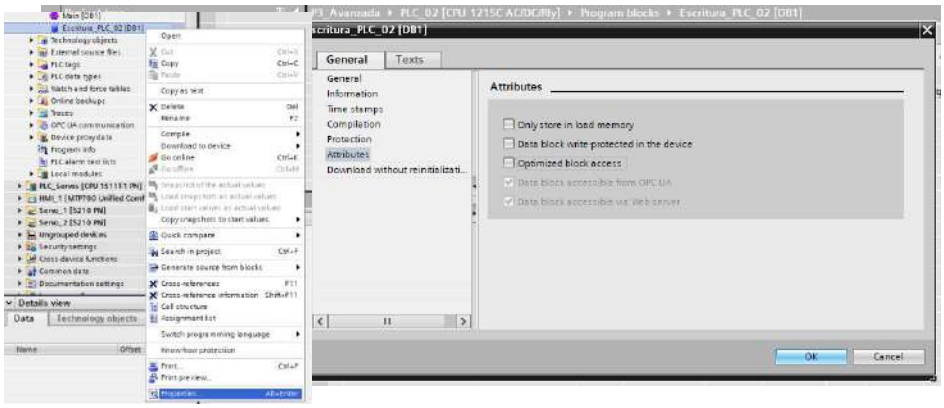


Ilustración 20 Escritura_PLC_02 -> (Click derecho) Properties -> General -> Attributes -> (Desactivar) Optimized block access

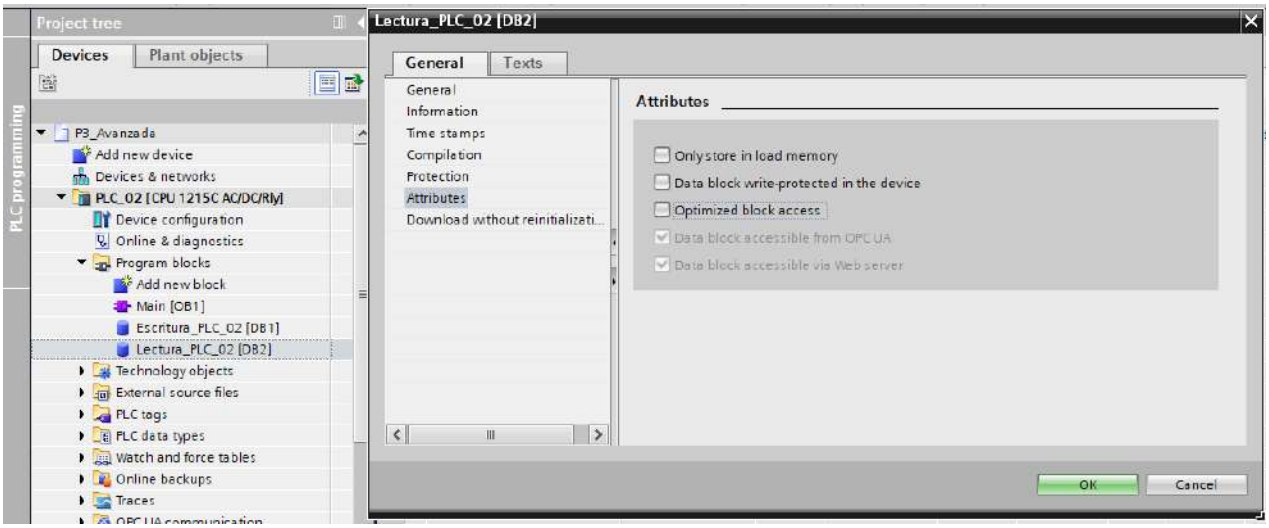


Ilustración 21 PLC_02 -> Program blocks -> Lectura_PLC_02 -> Properties -> General -> Attributes -> (Desmarcar) Optimized block access

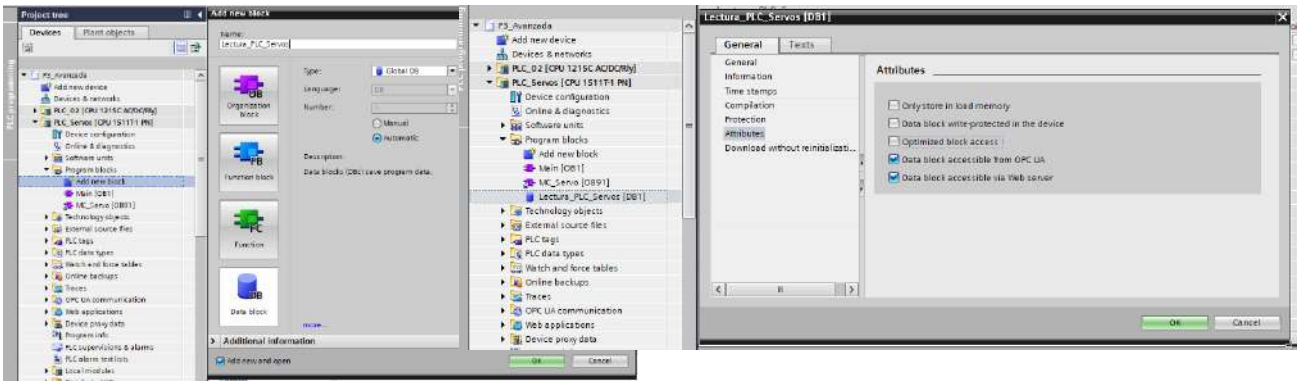


Ilustración 22 PLC_Servos -> Program blocks -> Add new blocks -> DB -> Lectura_PLC_Servos -> General -> Attributes -> (Desmarcar) Optimized block access

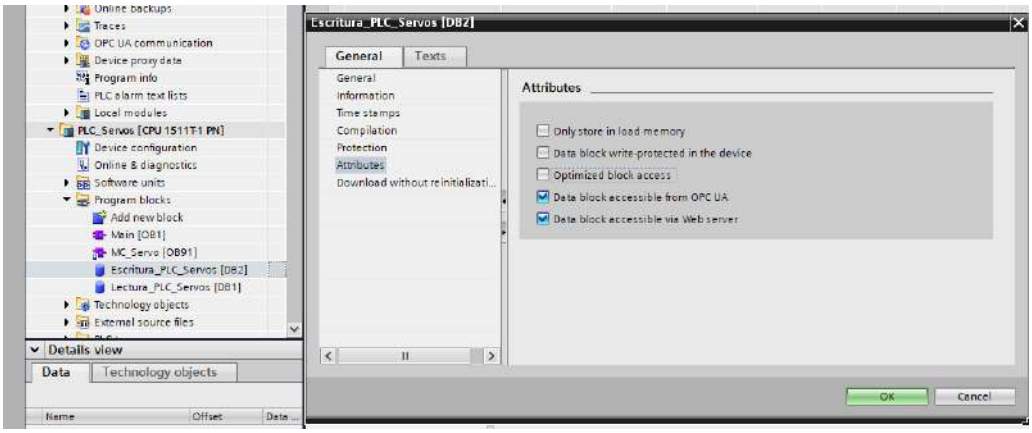


Ilustración 23 Escritura_PLC_Servos -> Properties -> Attributes -> (Desmarcar) Optimized block access

Para el uso de las funciones PUT/GET, los bloques requieren de un ciclo pedimento o de actualización de datos, ya que con ello, se harán peticiones para la lectura o escritura de los datos. Para ello, se requiere hacer uso de las marcas del reloj del PLC, las cuales se deben activar en las configuraciones:

Marcas de reloj del PLC_02 (S7-1200):

Seleccionar el CPU; en las pestaña "System and clock memory", habilitar la casilla "Enable the use of clock memory byte". Cambiar la dirección de almacenamiento del byte de memoria (que inicie en el 50)

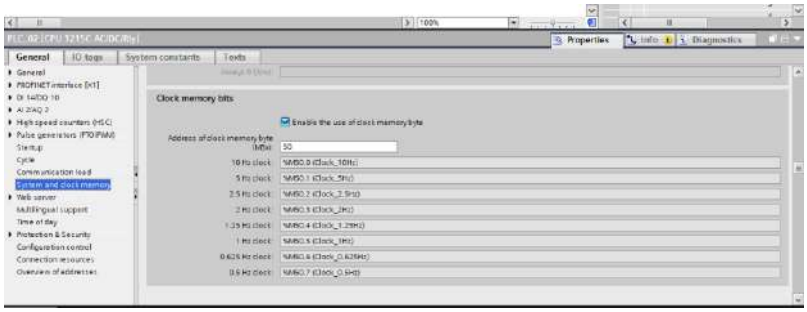


Ilustración 24 PLC_02 -> System and clock memory -> Clock memory bits -> (Activar la casilla) Enable the use of clock memory byte

Marcas de reloj del PLC_Servos (S7-1500):

Procedimiento similar, seleccionar el CPU; en las pestaña "System and clock memory", habilitar la casilla "Enable the use of clock memory byte". Cambiar la dirección de almacenamiento del byte de memoria (que inicie en el 51).

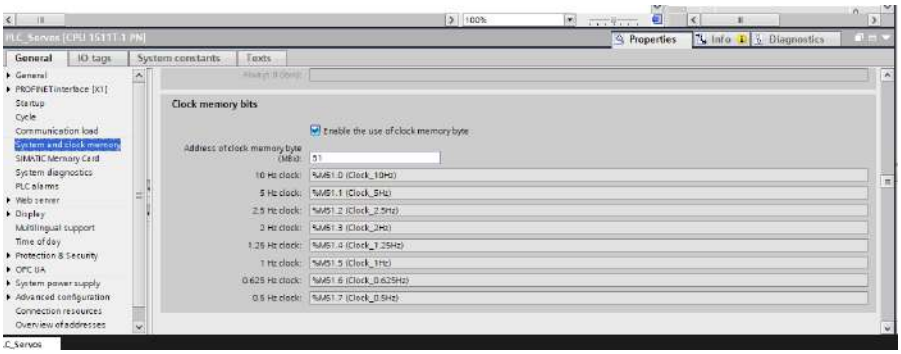


Ilustración 25 PLC_Servos -> System and clock memory -> Clock memory bits -> (Activar) Enable the use of clock memory byte

Integración de librería AxisCtrl

Integrar dos objetos tecnológicos del tipo Master y Slave, relacionado el Servo_1 y el Servo_2

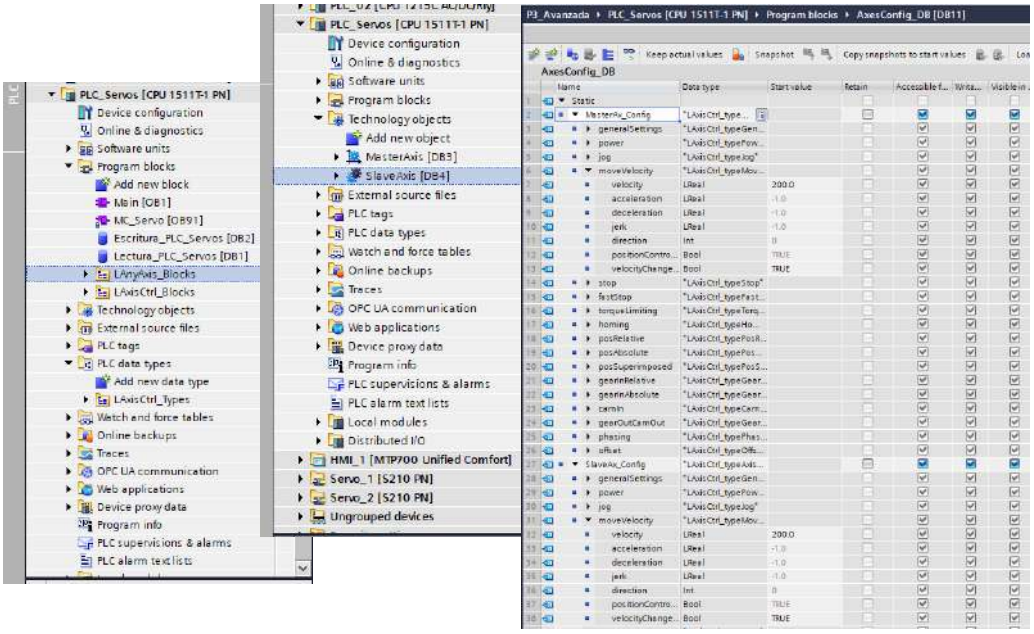


Ilustración 26 Intgración de libreria LAxisCtrl -> Integrar dos objetos tecnológicos; PositioningAxis (Master) / Synchronous (Slave) -> Crear un DB para almacenar las variables MasterAx_Config & SlaveAx_Config del tipo " LAxisCtrl_typeAxisConfig"

Para MasterAx_Config & SlaveAx_Config

- moveVelocity -> 200.0
- velocityChangeOnTheFly -> TRUE

Declaración de variables de trabajo

A continuación, es necesario establecer las variables de trabajo, es importante mencionar que las variables de trabajo van a esta ubicadas en dos lugares, primero, leer y escribir datos; segundo, operar las salidas de los equipos o monitorear las entradas. Para lectura y escitura entre los PLC se utilizarán los bloques previamente trabajados, tanto del PLC_02, como del PLC_Servos. Para lectura de sensores y activación de actuadores, se utilizará el defaul tag table de cada controlador.

Tags en el DB Lectura PLC_02

P3_Avanzada ▸ PLC_02 [CPU 1215C AC/DC/Rly] ▸ Program blocks ▸ Lectura_PLC_02 [DB2]

Lectura_PLC_02

	Name	Data type	Offset	Start value	Retain	Accessible
1	Static					
2	Sensor_a0	Bool	0.0	false		
3	Sensor_a1	Bool	0.1	false		
4	Sensor_b0	Bool	0.2	false		
5	Sensor_b1	Bool	0.3	false		
6	Sensor_c0	Bool	0.4	false		
7	Sensor_c1	Bool	0.5	false		

Ilustración 27 Sensores de posición para los pistones

Tags en el DB Escritura PLC_02

P3_Avanzada ▸ PLC_02 [CPU 1215C AC/DC/Rly] ▸ Program blocks ▸ Escritura_PLC_02 [DB1]

Escritura_PLC_02

	Name	Data type	Offset	Start value	Retain	Accessible
1	Static					
2	Salida_A+	Bool	0.0	false		
3	Salida_A-	Bool	0.1	false		
4	Salida_B+	Bool	0.2	false		
5	Salida_C+	Bool	0.3	false		

Ilustración 28 Electroválvulas

Tags en el DB de lectura en el PLC_Servos

P3_Avanzada ▸ PLC_Servos [CPU 1511T-1 PN] ▸ Program blocks ▸ Lectura_PLC_Servos [DB1]

Lectura_PLC_Servos

	Name	Data type	Offset	Start value	Retain	Accessible
1	Static					
2	Sensor_a0	Bool	0.0	false		
3	Sensor_a1	Bool	0.1	false		
4	Sensor_b0	Bool	0.2	false		
5	Sensor_b1	Bool	0.3	false		
6	Sensor_c0	Bool	0.4	false		
7	Sensor_c1	Bool	0.5	false		
8	<Add new>					

Ilustración 30 Valores de lectura de sensores

Tags en el DB de escritura en el PLC_Servos

P3_Avanzada ▸ PLC_Servos [CPU 1511T-1 PN] ▸ Program blocks ▸ Escritura_PLC_Servos [DB2]

Escritura_PLC_Servos

	Name	Data type	Offset	Start value	Retain	Accessible f...
1	Static					
2	SalidaA+	Bool	0.0	false		
3	SalidaA-	Bool	0.1	false		
4	SalidaB+	Bool	0.2	false		
5	SalidaC+	Bool	0.3	false		
6	<Add new>					

Ilustración 29 Valores de escritura

Tag default table del PLC_02

P3_Avanzada ▸ PLC_02 [CPU 1215C AC/DC/Rly] ▸ PLC tags ▸ Default tag table [49]

Default tag table

	Name	Data type	Address	Retain	Acces...	Wr
1	Clock_Byte	Byte	%MB50			
2	Clock_10Hz	Bool	%M50.0			
3	Clock_5Hz	Bool	%M50.1			
4	Clock_2.5Hz	Bool	%M50.2			
5	Clock_2Hz	Bool	%M50.3			
6	Clock_1.25Hz	Bool	%M50.4			
7	Clock_1Hz	Bool	%M50.5			
8	Clock_0.625Hz	Bool	%M50.6			
9	Clock_0.5Hz	Bool	%M50.7			
10	S_A0	Bool	%I0.0			
11	S_A1	Bool	%I0.1			
12	S_B0	Bool	%I0.2			
13	S_B1	Bool	%I0.3			
14	S_C0	Bool	%I0.4			
15	S_C1	Bool	%I0.5			
16	A+	Bool	%Q0.0			
17	A-	Bool	%Q0.1			
18	B+	Bool	%Q0.2			
19	C+	Bool	%Q0.3			
20	<Add new>					

Ilustración 31 Tabla de entradas y salidas PLC_02 (Sistema neumático)

Tag default table del PLC_Servos

P3_Avanzada ▸ PLC_Servos [CPU 1511T-1 PN] ▸ PLC tags ▸ Default tag table [97]

Default tag table

	Name	Data type	Address	Retain	Acces...
1	MasterAxis_Actor_Interface_Ad...	"PD_TEL105_IN"	%I256.0		
2	MasterAxis_Actor_Interface_Ad...	"PD_TEL105_O...	%Q256.0		
3	SlaveAxis_Actor_Interface_Add...	"PD_TEL105_IN"	%I276.0		
4	SlaveAxis_Actor_Interface_Add...	"PD_TEL105_O...	%Q276.0		
5	Clock_Byte	Byte	%MB51		
6	Clock_10Hz	Bool	%M51.0		
7	Clock_5Hz	Bool	%M51.1		
8	Clock_2.5Hz	Bool	%M51.2		
9	Clock_2Hz	Bool	%M51.3		
10	Clock_1.25Hz	Bool	%M51.4		
11	Clock_1Hz	Bool	%M51.5		
12	Clock_0.625Hz	Bool	%M51.6		
13	Clock_0.5Hz	Bool	%M51.7		
14	Start	Bool	%I0.0		
15	Stop	Bool	%I0.1		
16	Activacion	Bool	%I0.2		
17	L_Roja	Bool	%Q0.0		
18	L_Ambar	Bool	%Q0.1		
19	L_Verde	Bool	%Q0.2		
20	On	Bool	%M0.0		
21	Proceso	Bool	%M0.1		
22	Pilotaje_A+	Bool	%M0.2		
23	Pilotaje_A-	Bool	%M0.3		
24	Pilotaje_B+	Bool	%M0.4		
25	Pilotaje_C+	Bool	%M0.5		
26	Secuencia_A	Bool	%M0.6		
27	Molde_Open	Bool	%M0.7		
28	Banda	Bool	%M1.0		
29	Fin_Ciclo	Bool	%M1.1		
30	Arranque_HMI	Bool	%M1.2		
31	Paro_HMI	Bool	%M1.3		
32	Inicio_HMI	Bool	%M1.4		
33	<Add new>				

Ilustración 32 Tabla de entradas y salidas PLC_Servos (Control)

Compilar en cada PLC los datos y cambios generados.

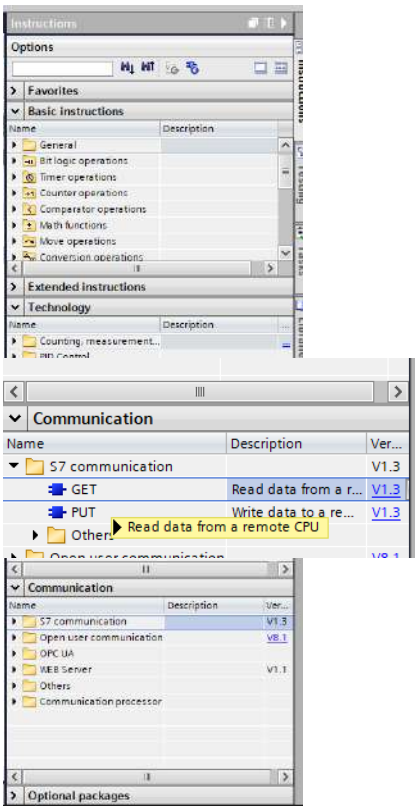


Ilustración 33 Main -> Instructions -> Communication -> S7 communication -> Get /Put

Programación de rutina de control

Al momento de utilizar una función GET, esta se comportará como un pedimiento de lectura de datos, por lo tanto habrá que cargarle los datos de donde serán extraidos (dispositivo remoto). En el PLC_Servos, al ser el equipo local, añadir los bloques de comunicación en la rutina principal (Main); los bloques se encuentran ubicados en la pestaña de instrucciones.

Desplegar la carpeta S7_communication y agregar la instrucción GET, en otro peldaño, la instrucción PUT.

Al get nombrarlo como “Obtener datos”

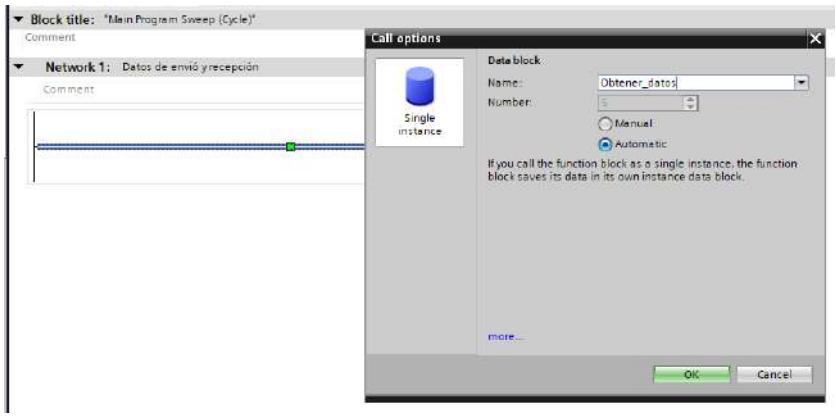


Ilustración 34 Get -> Network 1 -> "Obtener_datos"

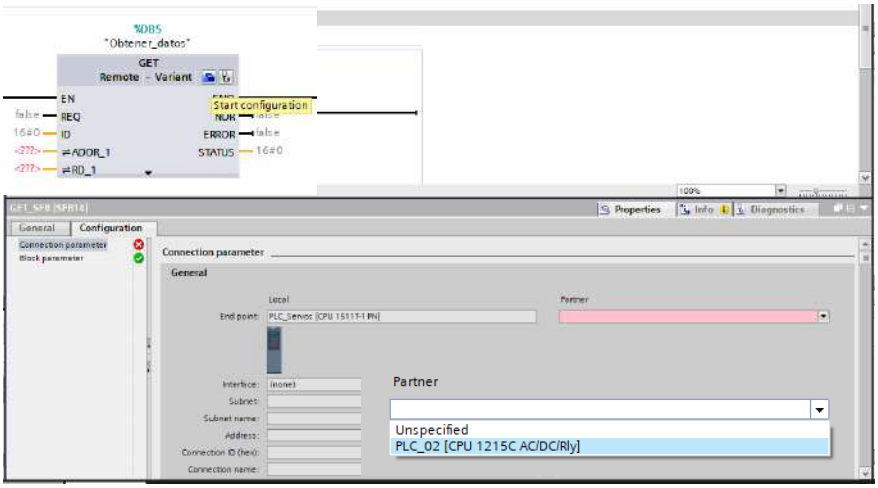


Ilustración 35 Start configuration -> Inspector -> Connection parameter -> General -> Partner -> PLC_02

Configuración del get:

Para configurarlo, dar click en el ícono “Start configuration” (ícono azul).

En el inspector aparecerá una menú de configuración

Colocar al PLC_02 como el Partner

En la configuración de redes, se había añadido al PLC_02 como un dispositivo dentro de la red Profinet, por tanto, TIA Portal autoconfigura los parámetros.

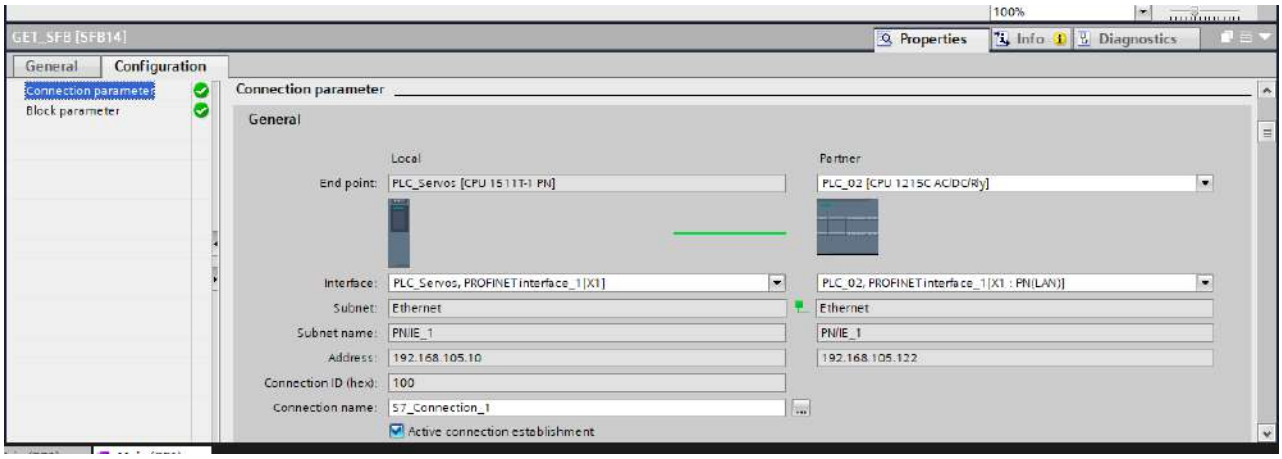


Ilustración 36 Configuración Local - Partner en bloque de comunicación GET

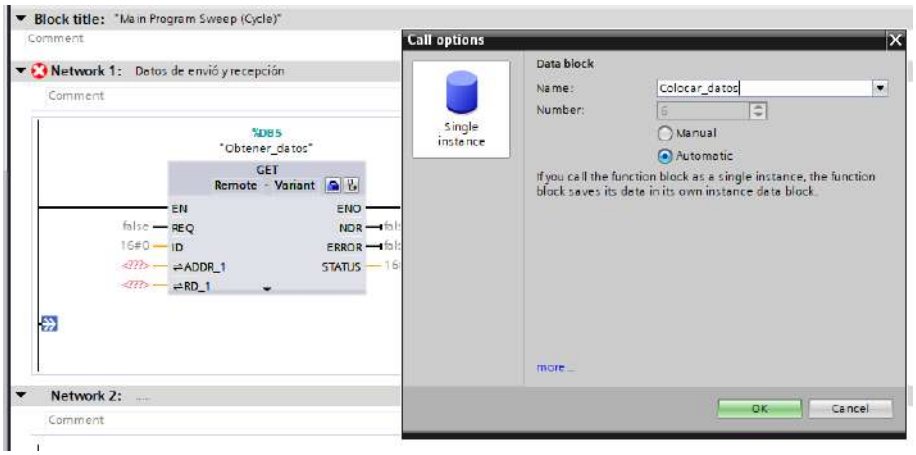


Ilustración 37 PUT -> Network 1 (new branch) -> "Colocar_datos"

Configuración de PUT

De manera similar a Get; configurar el bloque de datos presionando el ícono de “Start configuration”.

Colocar al PLC_02 como partner.

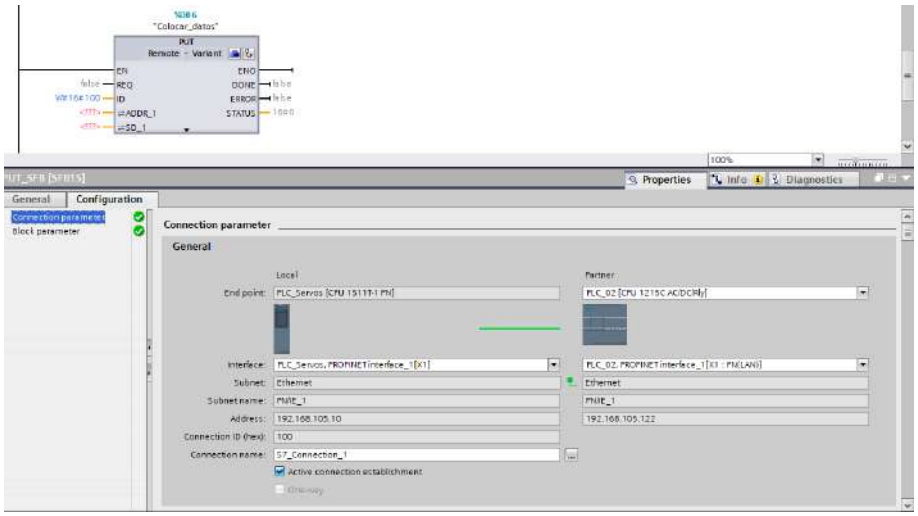


Ilustración 38 Configuración Local - Partner en bloque de comunicación PUT

La primer variable que se requiere colocar, es la velocidad de requimiento o actualización para activar el Get o el PUT (Request); al necesitar actualizar los datos en todo momento, se utilizará una variable que encenderá y apagará con una velocidad de 10 [HZ] “Clock_10Hz” (marcas de reloj).

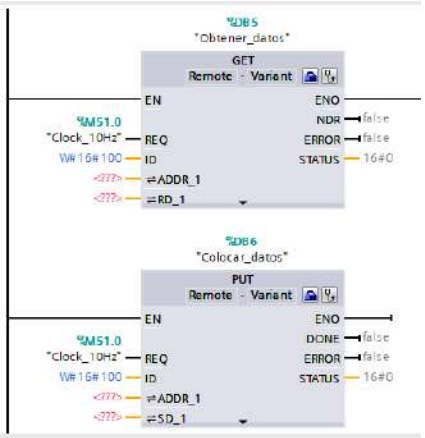


Ilustración 39 Marcas de petición (referenciadas a las marcas de reloj)

Ahora, lo que los bloques de comunicación requieren es un bloque de datos en los cuales leer (GET) o escribir (PUT) los datos. Get obtendrán los datos de lectura del bloque de instrucciones (DB) Lectura_PLC_02; Put es dónde colocar los datos, DB Escribir_PLC_02.

Para colocar la referencia de un bloque de datos, externo,la sintaxis es: “P#DB1.DBX0.0 BYTE 1”, donde:

- P#DB1 -> apuntador al bloque de datos (verificar el número de DB exacto1)
- DBX0.0 -> dirección de inicio o desde dónde comenzará a enviar o recibir los datos
- BYTE 1 -> rango de lectura o escritura para trabajar

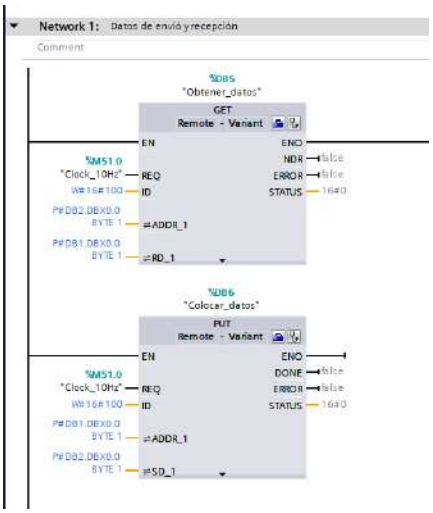


Ilustración 40 Bloque de datos para pedimento o recepción de señales

Una vez configurada el envío y recepción de datos, generar la lógica de programación para el problema

En el main del PLC_02 colocar los estados de las variables que se enviarán a los DB.

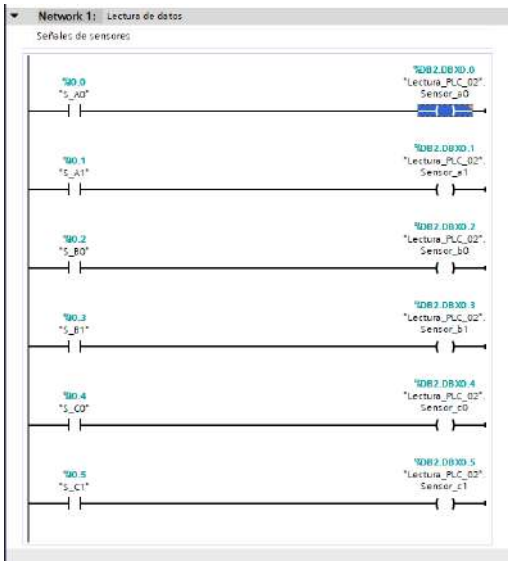


Ilustración 41 PLC_02 -> Información de sensores

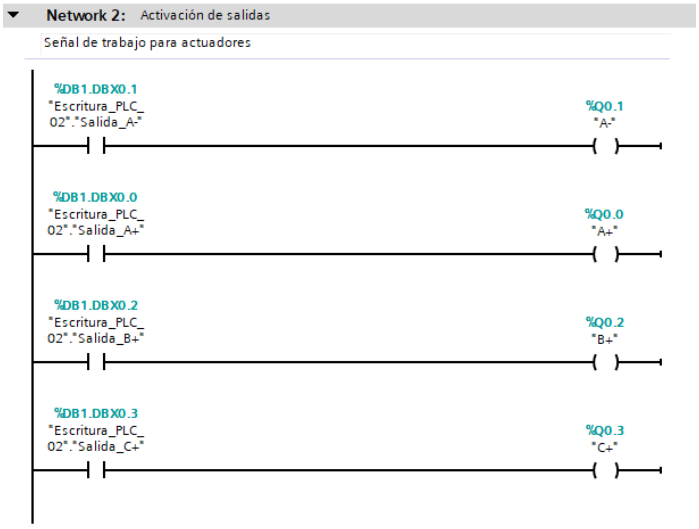


Ilustración 42 PLC_02 -> Activación de electroválvulas

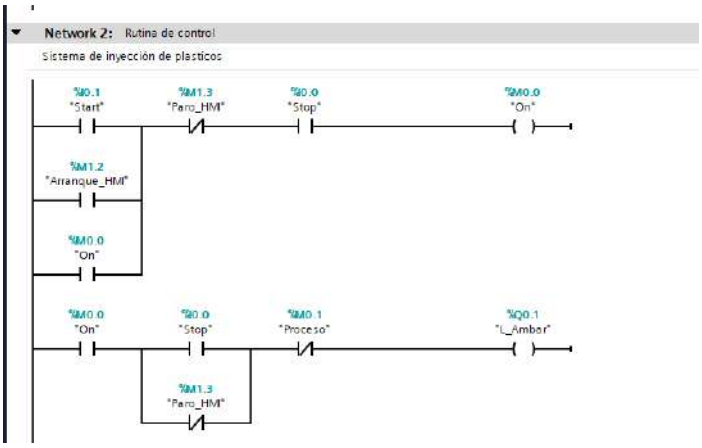


Ilustración 45 Código ladder, Parte 1

Lógica de programación en PLC_Servos

La programación de la rutina de control, es basada en las especificaciones del ejercicio:

El sistema entra en energización una vez pulsado el botón físico o el remoto (HMI).

De no estar la el botón de paro, remoto o físico, el indicador ambar enciende.

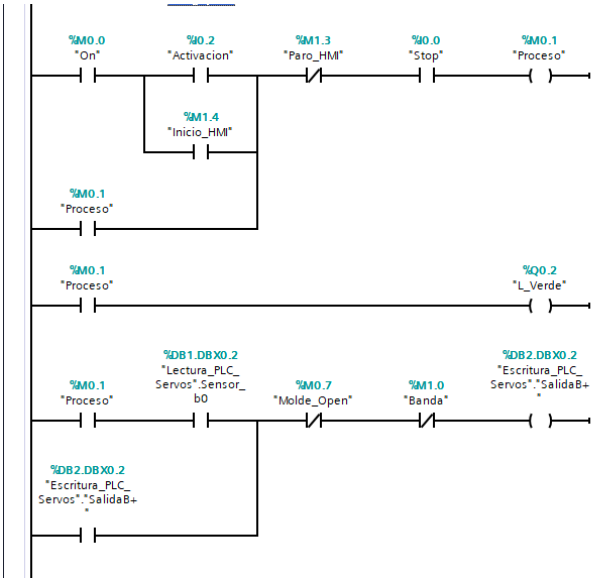


Ilustración 44 Código ladder; Parte 2

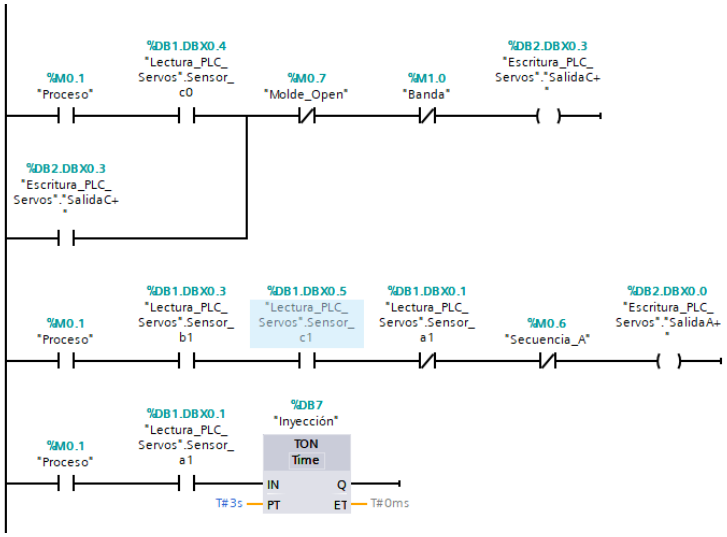


Ilustración 43 Código ladder, Parte 4

Al accionar el segundo pulsador, remoto o físico, el sistema inicia el proceso, por lo tanto, el indicador verde estará activo. Se activa secuencia de sujeción e inyección de material.

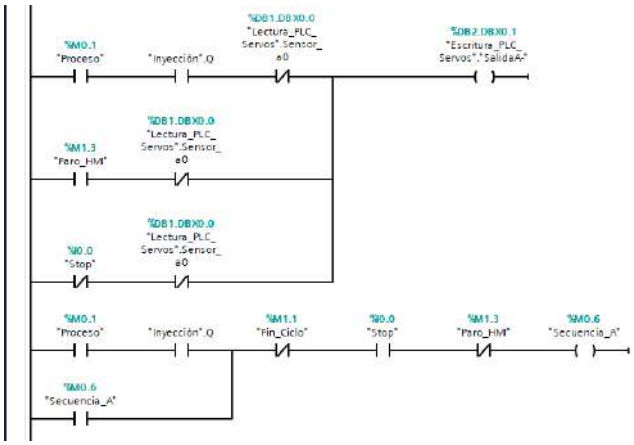


Ilustración 47 Código ladder, Parte 5

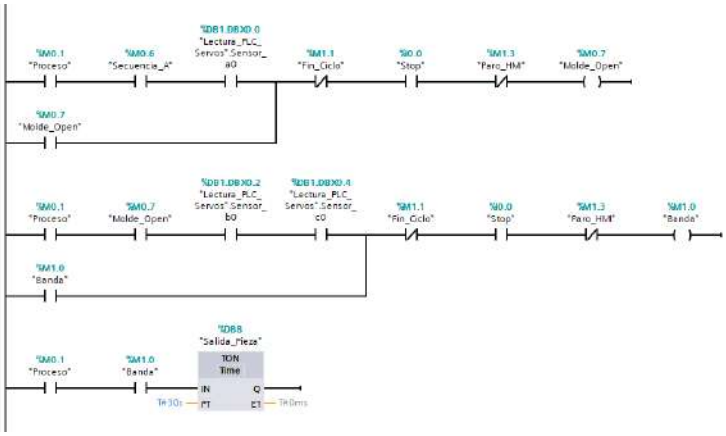


Ilustración 46 Código ladder, Parte 6

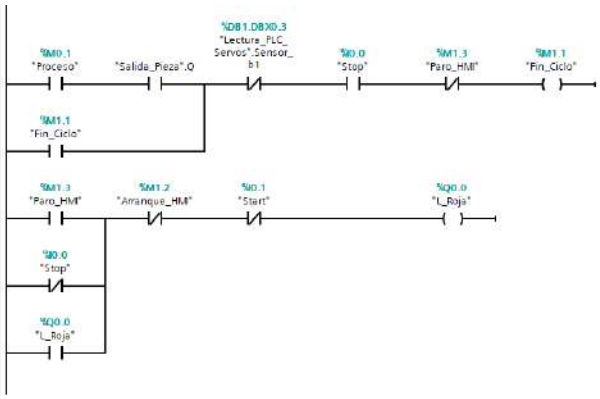


Ilustración 49 Código ladder, Parte 7

Ya inyectado el material, retorna el elemento, y activa una bandera se secuencia. Se debe abrir el molde para dejar caer la pieza. De inmediato la banda comienza a trabajar por cerca de 30 segundos para retirar la pieza a otra estación de trabajo.

La activación de la banda dependerá de la ejecución de los sermotores, por tanto los bloques de control para el eje Maestro y el eje Seguidor se colocarán en un Network independiente y en peldaños separados.

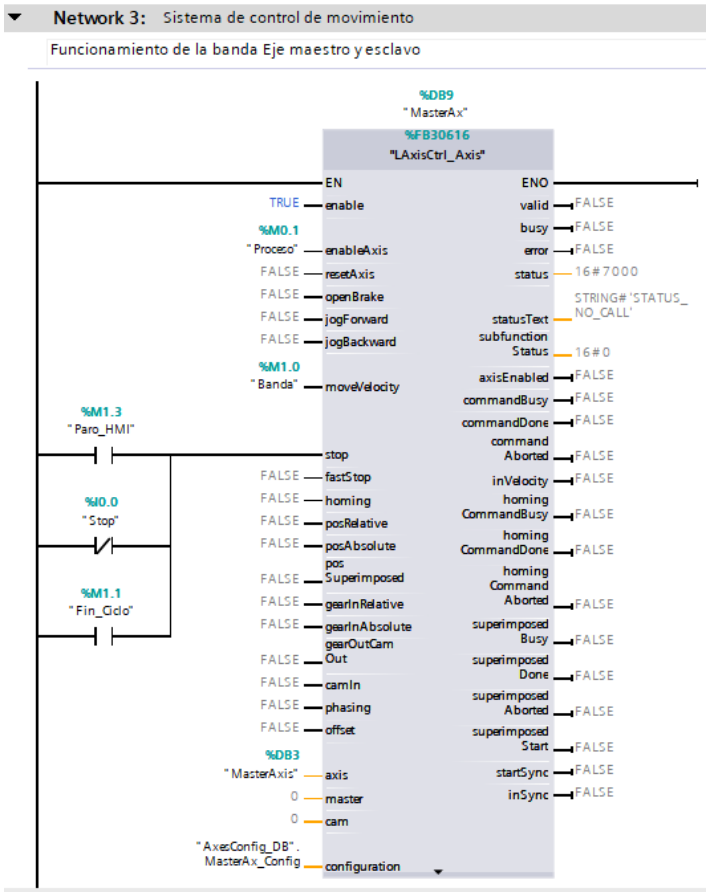


Ilustración 50 MasterAx configuration

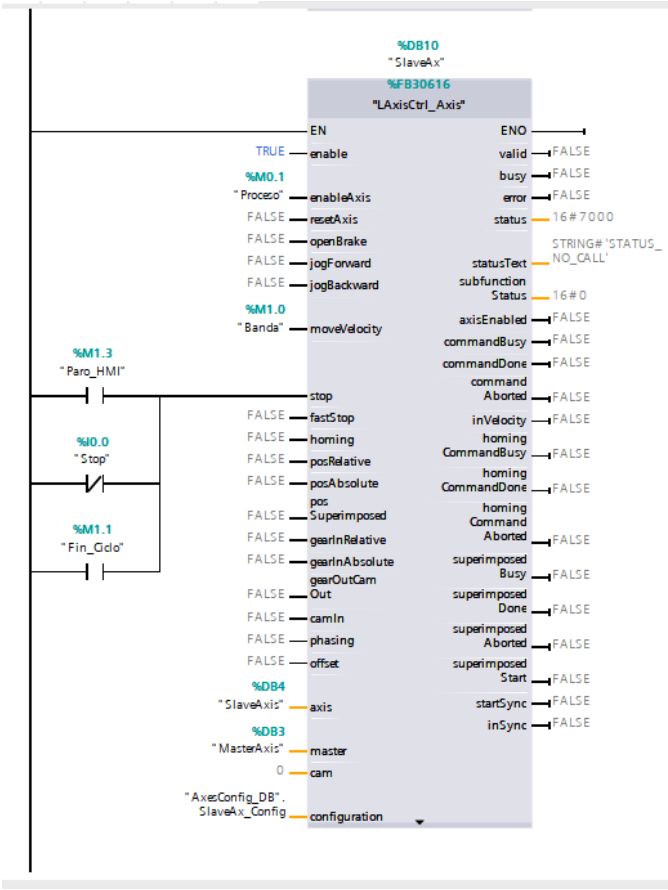
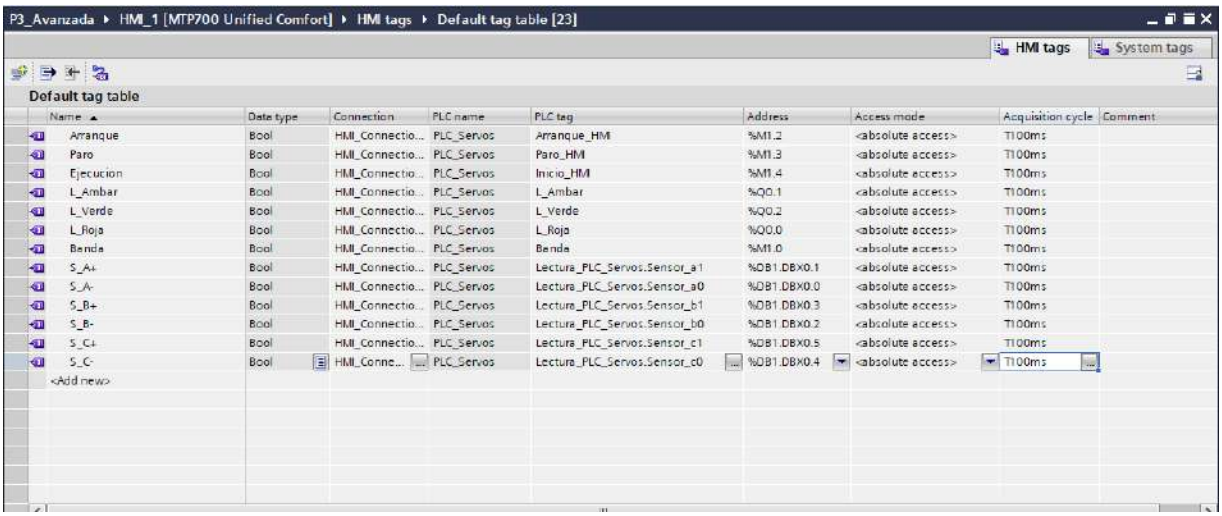


Ilustración 48 SlaveAx configuration

Configuración de HMI

Valores utlizados en Default tag table:



Name	Data type	Connection	PLC name	PLC tag	Address	Access mode	Acquisition cycle	Comment
Arranque	Bool	HMI_Connectio...	PLC_Servos	Arranque_HMI	%M1.2	<absolute access>	T100ms	
Paro	Bool	HMI_Connectio...	PLC_Servos	Paro_HMI	%M1.3	<absolute access>	T100ms	
Ejecucion	Bool	HMI_Connectio...	PLC_Servos	Inicio_HMI	%M1.4	<absolute access>	T100ms	
L_Ambar	Bool	HMI_Connectio...	PLC_Servos	L_Ambar	%Q0.1	<absolute access>	T100ms	
L_Verde	Bool	HMI_Connectio...	PLC_Servos	L_Verde	%Q0.2	<absolute access>	T100ms	
L_Roja	Bool	HMI_Connectio...	PLC_Servos	L_Roja	%Q0.0	<absolute access>	T100ms	
Banda	Bool	HMI_Connectio...	PLC_Servos	Banda	%M1.0	<absolute access>	T100ms	
S_A+	Bool	HMI_Connectio...	PLC_Servos	Lectura_PLC_Servos.Sensor_a1	%DB1.DBX0.1	<absolute access>	T100ms	
S_A-	Bool	HMI_Connectio...	PLC_Servos	Lectura_PLC_Servos.Sensor_a0	%DB1.DBX0.0	<absolute access>	T100ms	
S_B+	Bool	HMI_Connectio...	PLC_Servos	Lectura_PLC_Servos.Sensor_b1	%DB1.DBX0.3	<absolute access>	T100ms	
S_B-	Bool	HMI_Connectio...	PLC_Servos	Lectura_PLC_Servos.Sensor_b0	%DB1.DBX0.2	<absolute access>	T100ms	
S_C+	Bool	HMI_Connectio...	PLC_Servos	Lectura_PLC_Servos.Sensor_c1	%DB1.DBX0.5	<absolute access>	T100ms	
S_C-	Bool	HMI_Conne...	PLC_Servos	Lectura_PLC_Servos.Sensor_c0	%DB1.DBX0.4	<absolute access>	T100ms	
<Add new>								

Ilustración 51 Se utilizan parámetros contenidos en el PLC_Servos, tanto de tag default como de los DB de escritura y lectura

Pantalla principal

Ilustración 53 Home

Pantalla de monitoreo

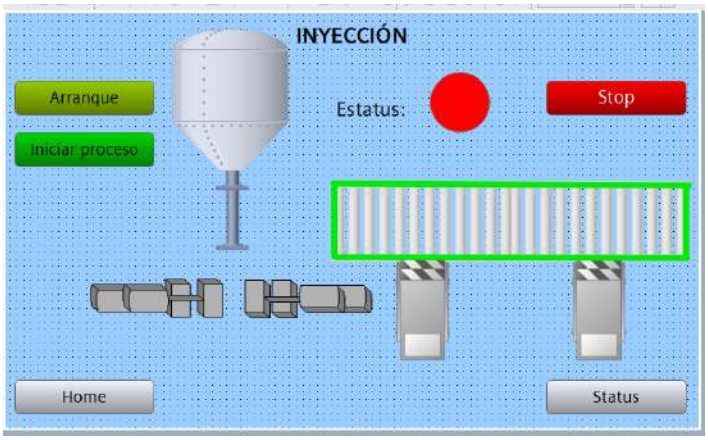


Ilustración 52 Processe

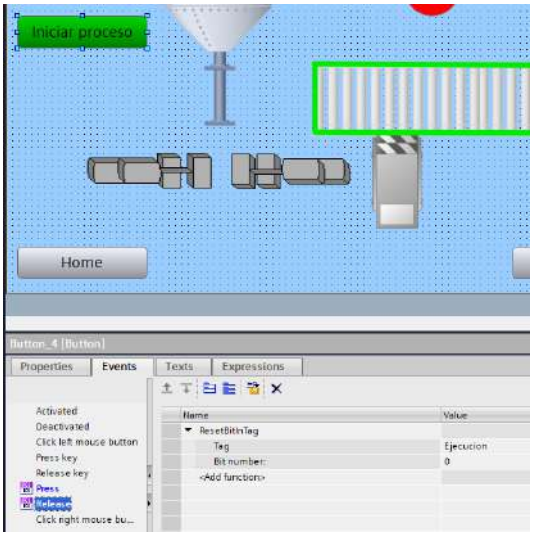


Ilustración 54 Configuración de pantalla de Proccesse

Para la pantalla de monitoreo se hizo uso de los componentes gráficos, como tanques, bandas y accesorios, a los cuales se les agregaba eventos de visualización o de aparaciencia (referenciado a un tag).

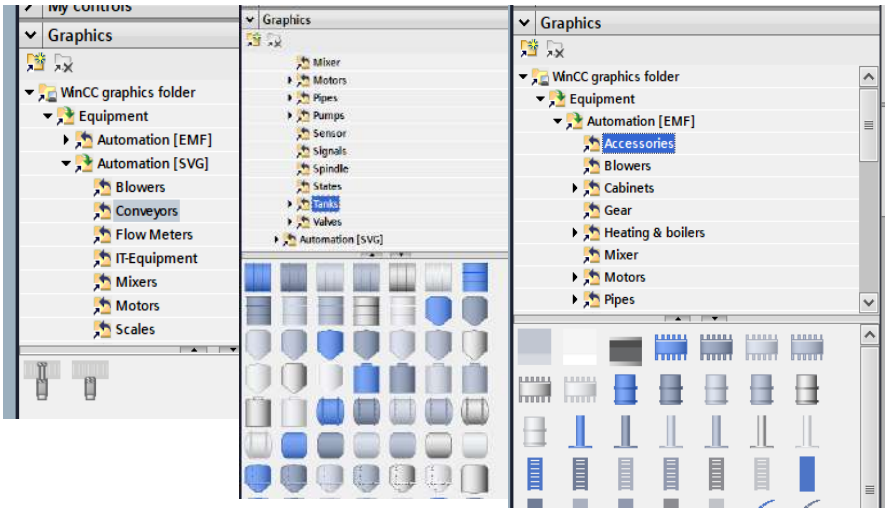


Ilustración 55 Graphics -> WinCC graphics folder -> Equipment -> Automation [SVG] -> Conveyors / Tanks -> Accessories

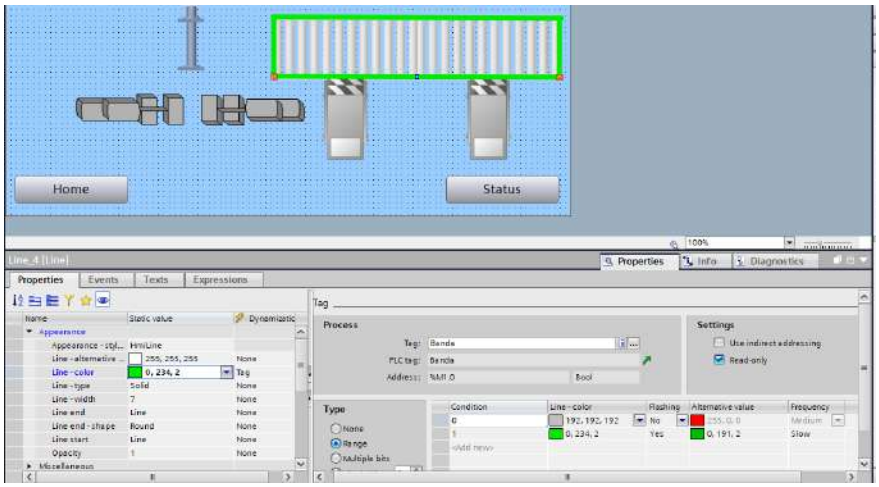


Ilustración 57 Función "Flashing" en un objeto del tipo línea

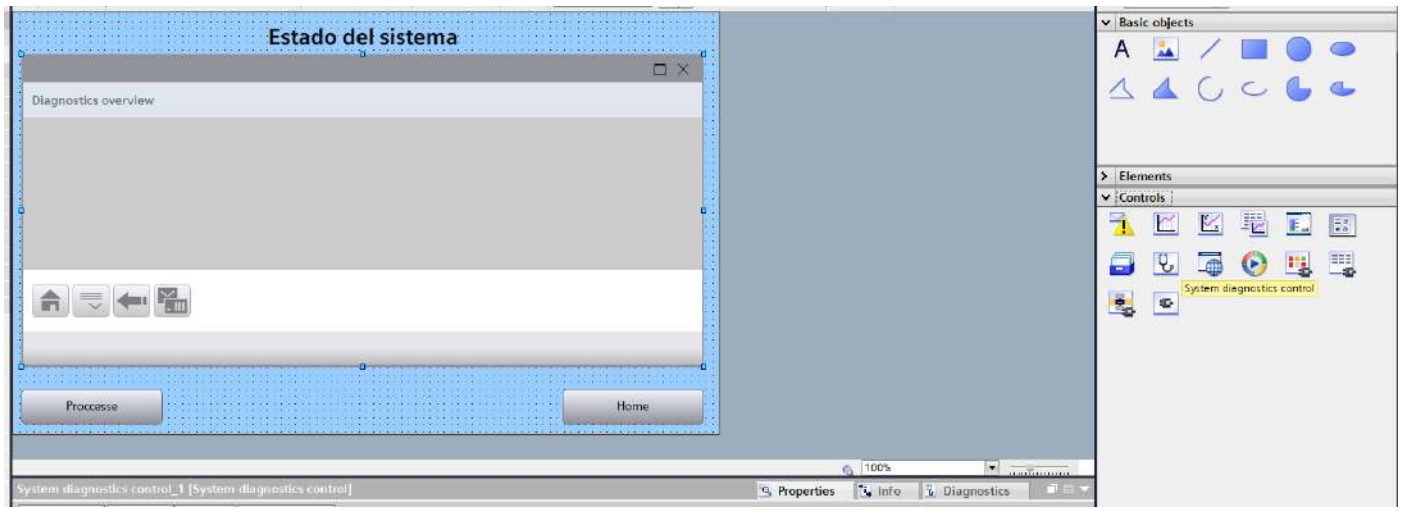


Ilustración 56 Pantalla de diagnóstico, permite verificar la comunicación entre los equipos

Entregables de la práctica

1. Código de Ladder comentado.
2. Pantallas realizadas en la HMI.
3. Imágenes del cambio de estado en el monitoreo de la HMI (fotografía de pantalla en ejecución).
4. Tablas de entradas y salidas (comentadas).

VI. Conclusiones y observaciones.

_____

_____

_____

_____

Referencias

- McGowan, M., & Phillips, L. (2017). Implementing HMI in printing processes: Enhancing quality control. *Journal of Manufacturing Processes*, 29, 246-254. https://doi.org/10.1016/j.jmapro.2017.06.023
- Yang, J., & Zhang, W. (2018). Human-machine interface design for automated assembly systems. *International Journal of Advanced Manufacturing Technology*, 97 (1-4), 241-254. https://doi.org/10.1007/s00170-018-1963-5
- Carrillo, J., & González, A. (2015). *Sistemas de control industrial*: Teoría y práctica. McGraw-Hill.
- López, M. J., & Rodríguez, S. (2017). Automatización y control en la industria de la impresión. *Revista de Ingeniería y Tecnología*, 8 (2), 45-52.