 UNIVERSIDAD SANTO TOMÁS PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA BUCARAMANGA	GESTIÓN DEL CONOCIMIENTO, APRENDIZAJE Y DOCENCIA	VERSIÓN 01
	GUÍA DE LABORATORIO No. 7	CODIGO -
		Página 1 de 12


PROGRAMA ACADÉMICO: Ingeniería Mecatrónica	PLAN DE ESTUDIOS: V	ACTA CONSEJO DE FACULTAD: 70
FECHA DE ELABORACIÓN: 10 de octubre de 2024	VERSIÓN: 1	FECHA DE ACTUALIZACIÓN:

1. INFORMACIÓN GENERAL	
ASIGNATURA:	Automatización Industrial y Laboratorio
PRÁCTICA DE:	Comunicación de siemens y allen bradley por medio profibus y modbus
ELABORADA POR:	Duvan Jose Valenzuela Ardila.

2. DESCRIPCIÓN GENERAL DE LA PRÁCTICA DE LABORATORIO
<p>Esta práctica pretende dar a conocer a los estudiantes las herramientas para la comunicación de plcs de diferentes marcas, se plantea como una guía con las configuraciones básicas para la comunicación siemens y Allen Bradley por medio de los protocolos Modbus RTU Y PROFIBUS. Los equipos propuestos en este documento son compatibles con los dispositivos de la familia control logix, y toda la información aquí encontrada es tomada de los manuales desarrollados por los fabricantes de los equipos de la marca PROSOFT TECHNOLOGY.</p>

3. COMPETENCIAS A FORMAR MEDIANTE LA PRÁCTICA
<ol style="list-style-type: none"> 1. Configurar los equipos que le permiten a los plcs de la marca Rockwell comunicarse con distintos plcs de diferentes marcas por medio de modbus y profibus. 2. Añadir dentro de la interfaz Studio 5000 Logix Designer los equipos necesarios para la comunicación. 3. Studio 5000 Logix Designer las configuraciones necesarias para la comunicación.

4. TEORÍAS Y CONCEPTOS QUE SE ABORDARÁN EN LA PRÁCTICA
<p>Modbus</p> <p>Es un protocolo de comunicación de datos que se utiliza en redes de automatización industrial para conectar dispositivos electrónicos. Originalmente desarrollado por Modicon (ahora Schneider Electric) en 1979, Modbus permite la comunicación entre muchos dispositivos conectados a la misma red, como un sistema que mide y transfiere datos entre sensores y computadoras. Su simplicidad y facilidad de implementación lo han convertido en uno de los protocolos más populares en la industria. [1]</p>

 UNIVERSIDAD SANTO TOMÁS <small>PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA</small> <small>BUCARAMANGA</small>	GESTIÓN DEL CONOCIMIENTO, APRENDIZAJE Y DOCENCIA	VERSIÓN 01
	GUÍA DE LABORATORIO No. 7	CODIGO -
		Página 2 de 12

Modbus es un protocolo de comunicación que opera en varios modos, siendo los más comunes el Modbus RTU, Modbus ASCII y Modbus TCP/IP. El Modbus RTU utiliza la transmisión de datos en formato binario, lo que lo hace eficiente en el uso del ancho de banda y robusto ante entornos industriales ruidosos, reduciendo el riesgo de errores en la comunicación. Por otro lado, el Modbus ASCII transmite datos en formato ASCII, lo que facilita su lectura y depuración, aunque es menos eficiente en términos de ancho de banda que el Modbus RTU. Finalmente, el Modbus TCP/IP extiende la funcionalidad de Modbus a redes Ethernet, permitiendo la comunicación a través de redes IP. Este modo es especialmente útil para aplicaciones que requieren integración con sistemas de TI y acceso remoto, brindando una mayor flexibilidad y versatilidad en la comunicación. [2]

Algunas características del protocolo modbus son:

- **Simplicidad y Facilidad de Implementación:** La simplicidad del protocolo facilita su implementación en dispositivos con recursos limitados. [2]
- **Compatibilidad y Estándar Abierto:** Modbus es un estándar abierto, lo que permite su implementación en dispositivos de diferentes fabricantes sin problemas de licencia. [2]
- **Flexibilidad:** Puede operar en diferentes medios de transmisión, como cables RS-485, RS-232 y Ethernet. [2]
- **Capacidad de Direccionamiento:** Modbus RTU y ASCII están limitados a 247 dispositivos esclavos por maestro, lo que puede ser una restricción en redes muy grandes. [2]
- **Seguridad:** La versión básica de Modbus no incluye características de seguridad como encriptación o autenticación, lo que lo hace vulnerable a ataques si no se implementan medidas de seguridad adicionales. [2]


Modbus sigue siendo una opción muy común en la industria debido a su robustez, simplicidad y amplia aceptación, haciendo de este protocolo una herramienta fundamental para la comunicación en redes de automatización industrial. [2]

Topologías utilizadas por el protocolo Modbus: Bus, Estrella, Anillo, Árbol, Malla.

c) *Process Field Bus (PROFIBUS)*

Es un estándar de comunicación utilizado en redes de automatización de procesos y manufactura. Desarrollado por Siemens y estandarizado por la IEC, PROFIBUS permite la transmisión de datos y control entre dispositivos de campo y sistemas de control. Es conocido por su alta velocidad de comunicación y su capacidad para integrar dispositivos de diferentes fabricantes, lo que facilita la interoperabilidad y la flexibilidad en el diseño de sistemas industriales. [3]

La arquitectura de PROFIBUS se basa en el modelo Maestro-Eslavo, donde el maestro (o controlador) se comunica con múltiples dispositivos esclavos (o periféricos) para monitorear y controlar procesos. Esta arquitectura permite la configuración flexible de redes con diferentes tipos de dispositivos y ofrece un alto grado de control y supervisión, lo que lo hace adecuado para entornos industriales complejos y dinámicos. [3]

 UNIVERSIDAD SANTO TOMÁS PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA BUCARAMANGA	GESTIÓN DEL CONOCIMIENTO, APRENDIZAJE Y DOCENCIA	VERSIÓN 01
	GUÍA DE LABORATORIO No. 7	CODIGO -
		Página 3 de 12

Tipos de PROFIBUS:

- **PROFIBUS DP (Decentralized Periphery):** Utilizado principalmente para la comunicación entre dispositivos de campo y sistemas de control en tiempo real. Es ideal para aplicaciones que requieren una alta velocidad de transmisión de datos, como en la industria de procesos.[19]
- **PROFIBUS PA (Process Automation):** Diseñado específicamente para aplicaciones en la industria de procesos, como la medición y el control de variables como presión, temperatura y caudal. PROFIBUS PA utiliza técnicas de transmisión robustas para garantizar la fiabilidad en entornos industriales exigentes. [3]

Características Principales de PROFIBUS:


- **Velocidad de Comunicación:** PROFIBUS ofrece velocidades de transmisión de datos que van desde 9.6 kbps hasta 12 Mbps en su versión DP. Esto permite una comunicación rápida y eficiente entre dispositivos en tiempo real. [3]
- **Diagnósticos Avanzados:** El protocolo PROFIBUS incluye capacidades avanzadas de diagnóstico que permiten detectar fallos, identificar problemas y realizar mantenimiento predictivo en dispositivos conectados a la red. Esto contribuye a una mayor disponibilidad y fiabilidad del sistema. [3]
- **Flexibilidad y Escalabilidad:** PROFIBUS es altamente flexible y escalable, lo que significa que puede adaptarse a diferentes requisitos de aplicaciones y crecer conforme a las necesidades de la planta industrial. Además, es compatible con una amplia variedad de dispositivos y equipos de diferentes fabricantes. [3]
- En resumen, PROFIBUS es un protocolo de comunicación robusto, rápido y versátil que ofrece una amplia gama de funcionalidades para la automatización industrial, desde el control en tiempo real hasta el monitoreo de procesos y la seguridad funcional. Su uso extendido en diversas industrias es testimonio de su eficacia y confiabilidad en entornos industriales exigentes. [3]

5. PROBLEMAS DE INVESTIGACIÓN QUE DEBEN RESOLVER LOS ESTUDIANTES

- Controladores lógicos programables.
- Comunicaciones industriales
- Configuración de registros en redes modbus.
- Funcionamiento general de los módulos MVI56E y ILX56-PBM.
- Conexión de software y hardware de los módulos MVI56E y ILX56-PBM.

6. EQUIPOS Y MATERIALES NECESARIOS

- **Módulo MVI56E-MCM/MCMXT para redes Modbus Maestro/Esclavo para ControlLogix**

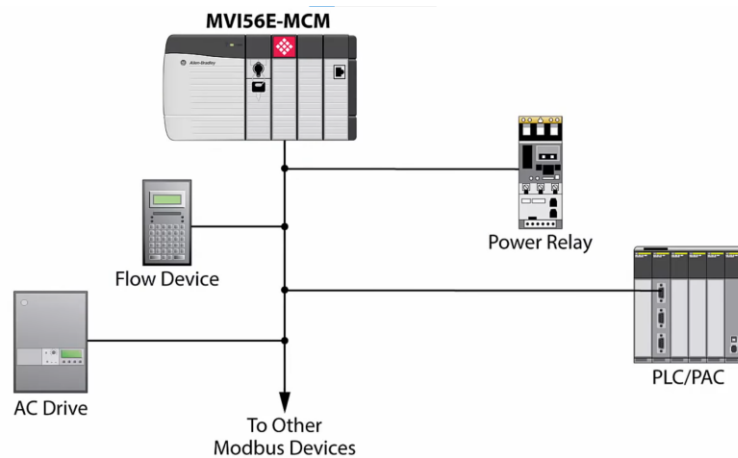
 UNIVERSIDAD SANTO TOMÁS PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA BUCARAMANGA	GESTIÓN DEL CONOCIMIENTO, APRENDIZAJE Y DOCENCIA	VERSIÓN 01
	GUÍA DE LABORATORIO No. 7	CODIGO -
		Página 4 de 12



- **ILX56-PBM ProSoft:** PROFIBUS DPV1 MAESTRO / ESCLAVO, 5000 BYTES MÁXIMO, 450MA @ VDC, 2MA @ 24VDC, IP20



La experiencia de ProSoft Technology a nivel mundial en el campo de la automatización nos brinda la confianza y la certeza de que los equipos que ofrecen son altamente eficientes, esta es una de las pocas compañías concentradas exclusivamente en las comunicaciones industriales. Por estas razones se sugiere adquirir los módulos de comunicación ya presentamos los cuales están fabricados para adaptarse a la gran variedad PLCs de ROCKWELL AUTOMATION de la familia CONTROLLOGIX, esto nos permite tener rango amplio en la adquisición de un nuevo PLC para el laboratorio de automatización de la universidad Santo Tomás seccional Bucaramanga, debido a que el equipo CompactLogix L23E no permite la adición de módulos en su hardware. [4]




7. PROCEDIMIENTO

En el siguiente espacio se da una inducción enfocada en las configuraciones básicas para que el modulo MVI56E modbus RTU maestro/esclavo y el modulo PROFIBUS DPV1 Master/Slave, las cuales permiten la correcta comunicación por medio de los diferentes protocolos.

Configuración del Módulo MVI56E Maestro/Esclavo:

- PASO 1: Ir a la página de prosoft y descargar el siguiente archivo (<https://mx.prosoft-technology.com/Productos/Rockwell-Automation/Platform/ControlLogix/Modulo-para-rede-Modbus-Maestro-Esclavo-mejorado-para-ControlLogix>)
Link de la pagina para descargar la librería que usaremos en la configuración del dispositivo

 UNIVERSIDAD SANTO TOMÁS PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA BUCARAMANGA	GESTIÓN DEL CONOCIMIENTO, APRENDIZAJE Y DOCENCIA	VERSIÓN 01
	GUÍA DE LABORATORIO No. 7	CODIGO -
		Página 6 de 12

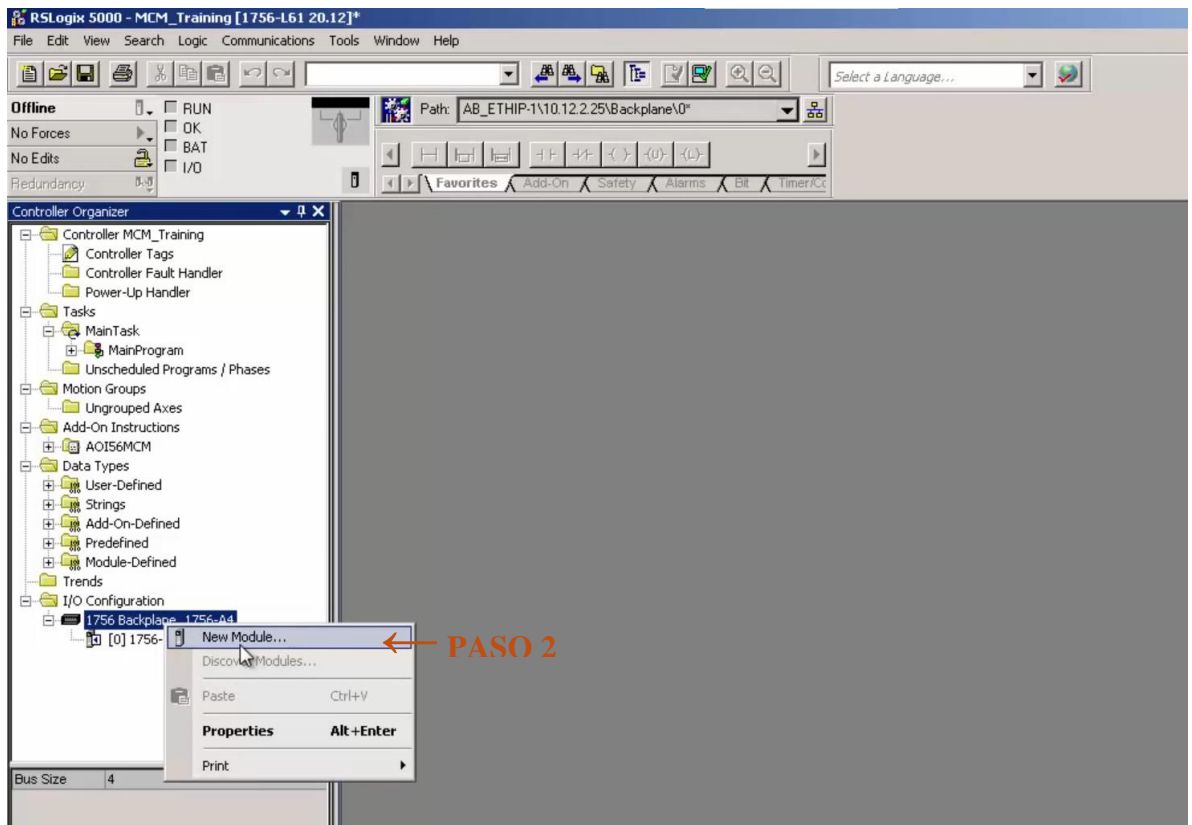
- Imágenes
- Características y Beneficios
- Especificaciones
- Hardware
- Descargas
- Video**
- Certificaciones

Descargas

MVI56E-MCM/MCMXT Datasheet	431,46 kB	Última Modificación 09.12.2022
MVI56E-MCM/MCMXT User Manual	4,61 MB	Última Modificación 23.11.2023
MVI56E-MCM FAQ	138,50 kB	Última Modificación 29.05.2013
MVI56(E) Add-On Profile v1.16	107,49 kB	Última Modificación 07.07.2023
MVI56E-MCM EDS File	1,51 kB	Última Modificación 13.08.2013
MVI56E-MCM Sample Ladder & Add-On Instruction	1,48 MB	Última Modificación 07.05.2013
MVI56E-MCM Faceplate for Factory Talk	267,55 kB	Última Modificación 18.09.2012
MVI Modbus Memory Tool	33,09 kB	Última Modificación 20.07.2021
Conformal Coating Option	2,14 MB	Última Modificación 18.05.2020
LifeCycle Letter MVI56E Series	108,16 kB	Última Modificación 14.06.2024
ProSoft Configuration Builder		
MVI56E CAD Drawings (STEP)	11,16 MB	Última Modificación 15.09.2023

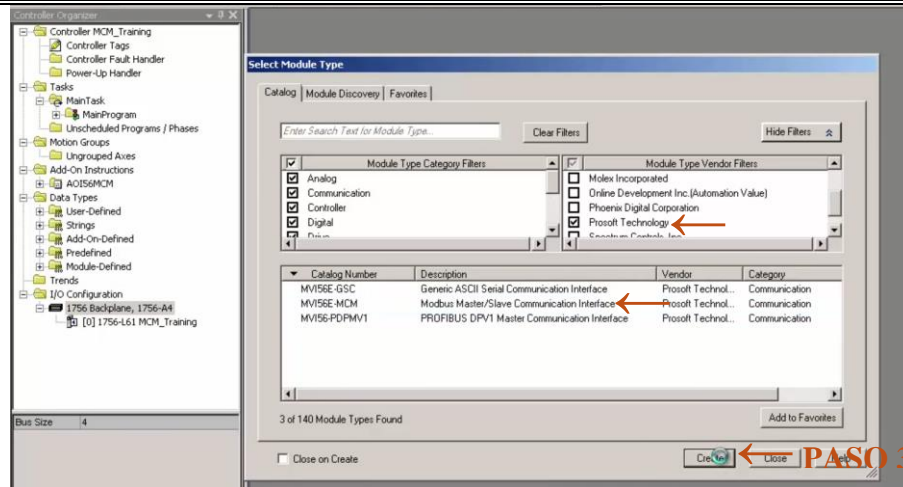
← PASO 1

- PASO 2: En la interfaz de RS Logix 5000 Designer añadimos el modulo con todas sus configuraciones.

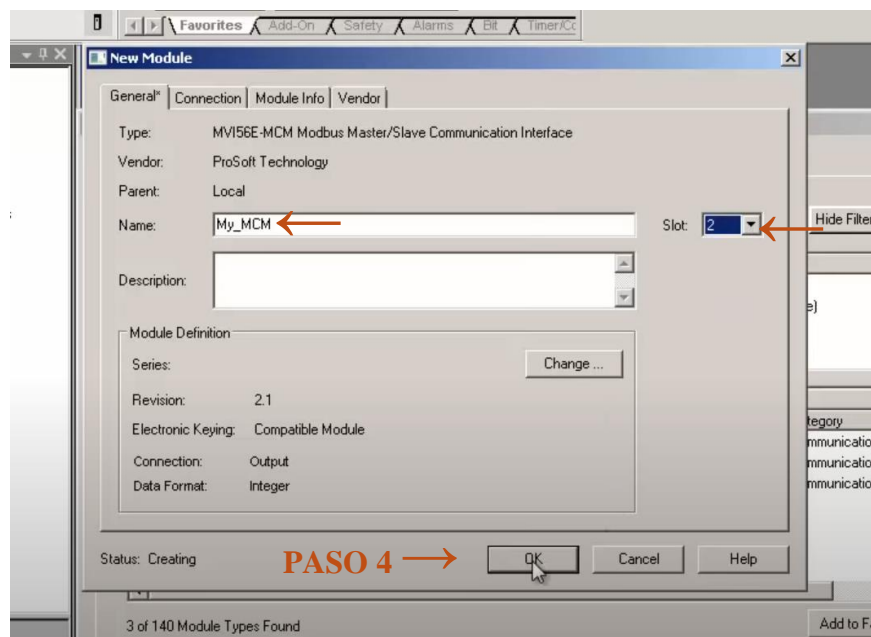


← PASO 2

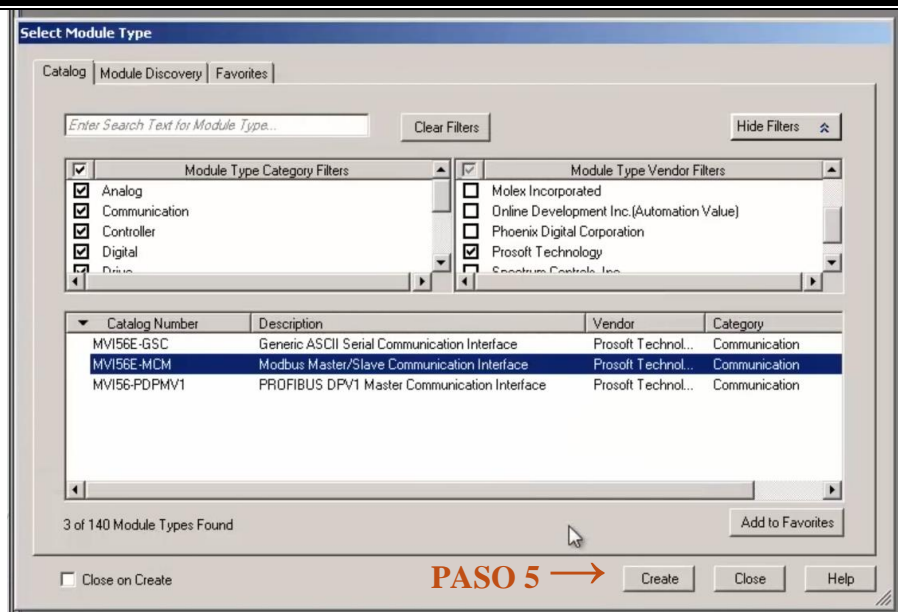
- PASO 3: En la ventana emergente buscamos en la parte de “Module Type Vendor Filters” la marca Prosoft Tecnology y el modulo a instalar el cual es el: MVI56E



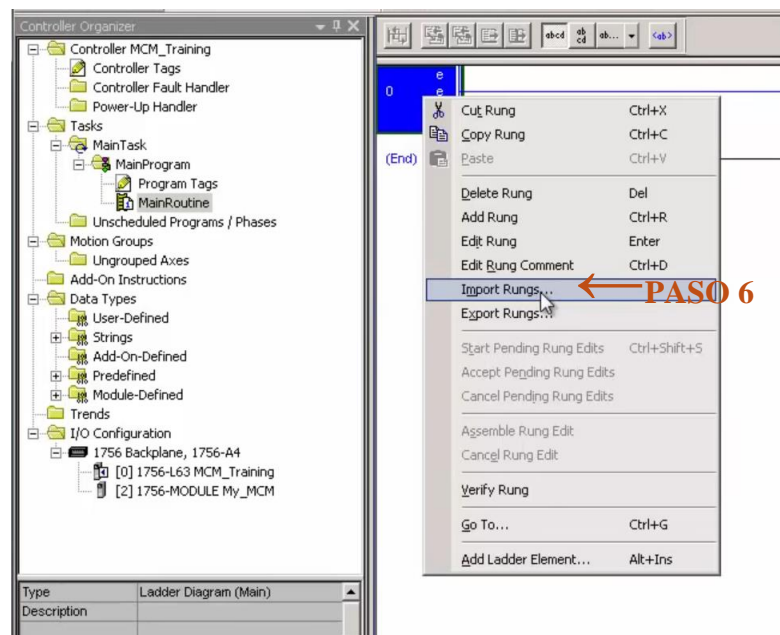
- PASO 4: En la ventana emergente le damos un nombre a el módulo creado y definimos el slot donde quedara configurado.



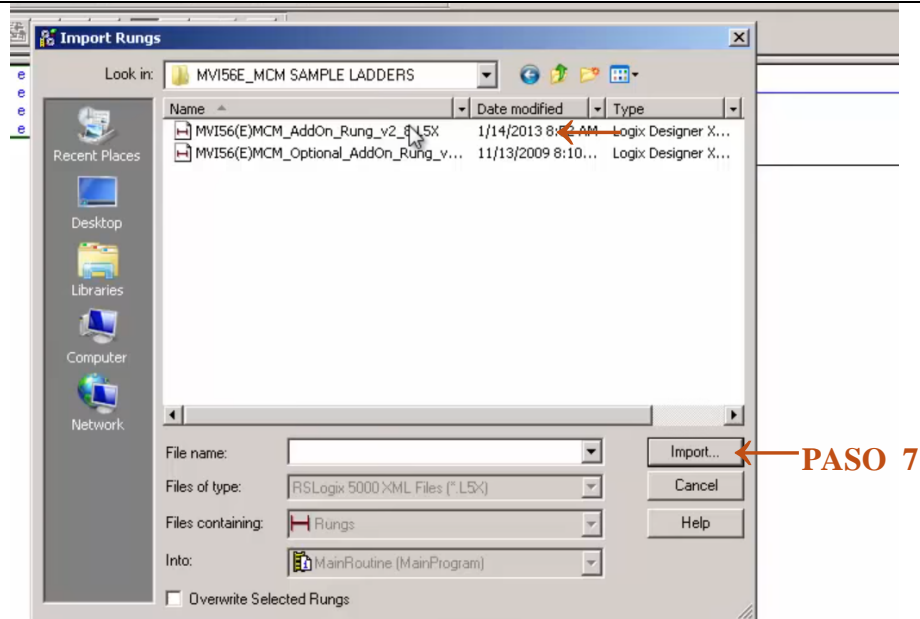
- PASO 5: CLICK EN CLOSE



- PASO 6: Ir al “MainRoutine” click derecho sobre la línea de código y click en “import rungs”



- PASO 7: Buscamos el archivo descargado en el paso 1 e importamos el siguiente elemento.



- PASO 8: En la ventana emergente ir a tags y configurar las ubicaciones de memoria con el mismo numero de slot configurado en el dispositivo

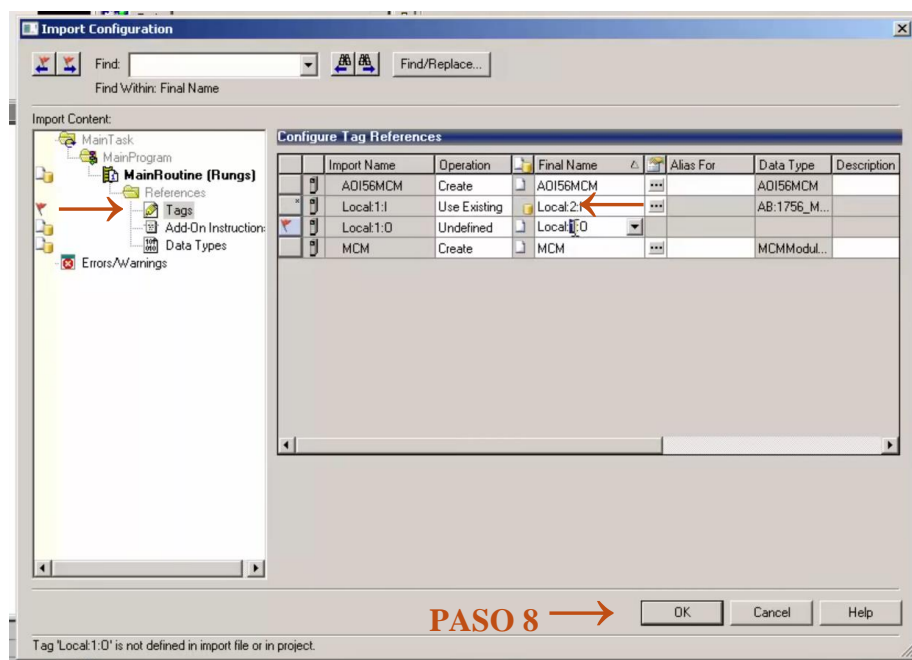
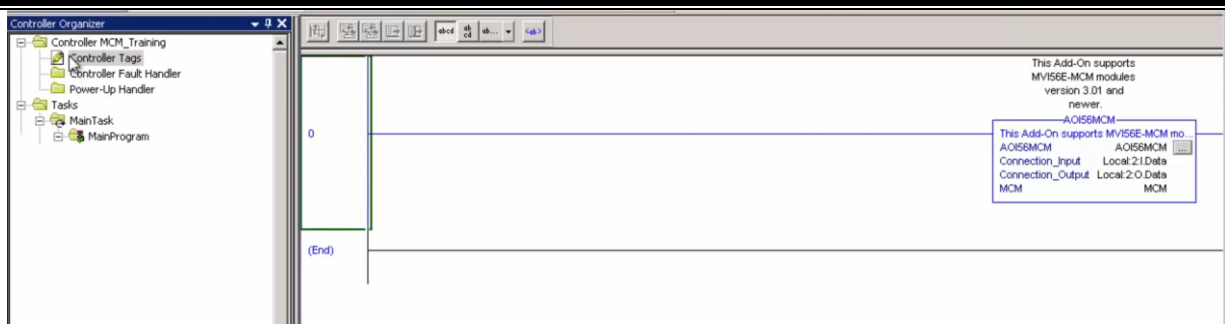
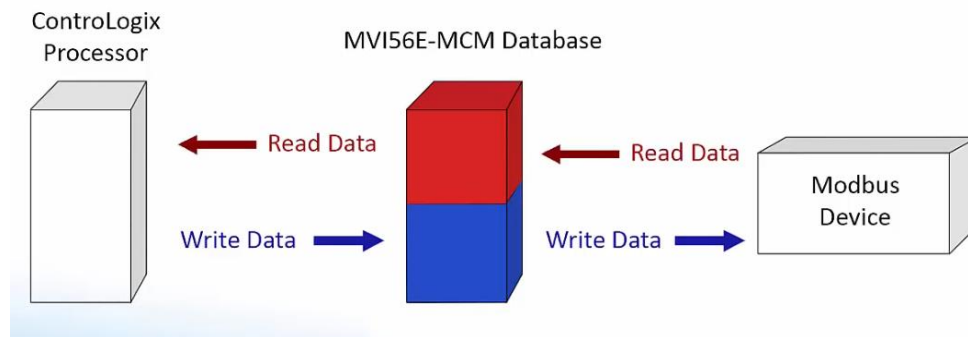


Imagen de referencia de cómo debe aparecer la función añadida



- PASO 9: Ir a controller tags en MCM.CONFIG.ModDef configurar la base de datos del modulo de comunicación.



La base de datos del modulo esta dividida en dos partes una para traer los datos de la red al controlador y otra para llevar los datos desde el contorlador hacia la red.

WriteStartReg: Especifica en qué lugar de la memoria del módulo de 10.000 registros se colocarán los datos enviado desde las etiquetas WriteData en el procesador ControLogix.

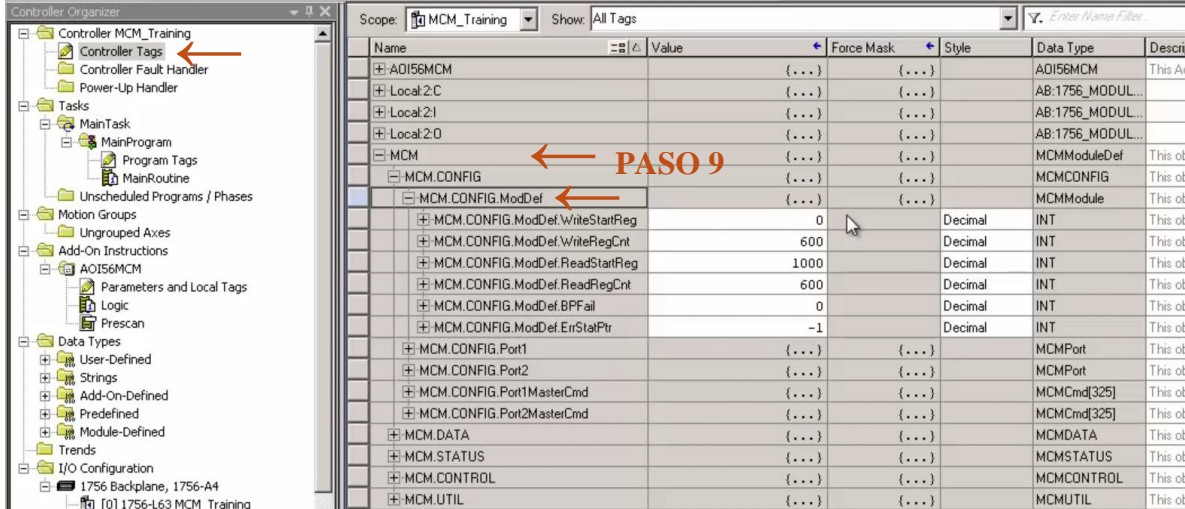
WriteRegCnt: Especifica cuántos registros de datos almacenará el módulo MVI56E-MCM. solicitud del procesador ControLogix.

ReadStartReg: Especifica qué registros en el área de datos de lectura del módulo se enviarán al Etiquetas ReadData en el procesador ControLogix.

ReadRegCnt: Establece cuántos registros de datos enviará el módulo MVI56E-MCM el procesador ControLogix.

BPFail: Establece el número consecutivo de fallas del backplane que causarán la Módulo para detener las comunicaciones en la red Modbus. Normalmente utilizado cuando el módulo está configurado como Esclavo.

ErrStatPtr: También se utiliza principalmente cuando el módulo está configurado como Esclavo. Este El parámetro coloca los datos de ESTADO en la base de datos del módulo.

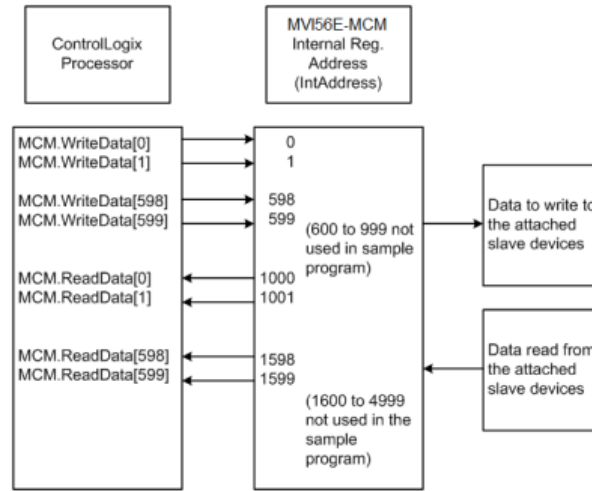


Name	Value	Force Mask	Style	Data Type	Descri
AOI56MCM	{...}	{...}		AOI56MCM	This Ac
Local2:C	{...}	{...}		AB:1756_MODUL...	
Local2:I	{...}	{...}		AB:1756_MODUL...	
Local2:O	{...}	{...}		AB:1756_MODUL...	
MCM	{...}	{...}		MCMModuleDef	This ob
MCM.CONFIG	{...}	{...}		MCMCONFIG	This ob
MCM.CONFIG.ModDef	{...}	{...}		MCMModule	This ob
MCM.CONFIG.ModDef.WriteStartReg	0		Decimal	INT	This ob
MCM.CONFIG.ModDef.WriteRegCnt	600		Decimal	INT	This ob
MCM.CONFIG.ModDef.ReadStartReg	1000		Decimal	INT	This ob
MCM.CONFIG.ModDef.ReadRegCnt	600		Decimal	INT	This ob
MCM.CONFIG.ModDef.BPFail	0		Decimal	INT	This ob
MCM.CONFIG.ModDef.ErrStatPtr	-1		Decimal	INT	This ob
MCM.CONFIG.Port1	{...}	{...}		MCMPort	This ob
MCM.CONFIG.Port2	{...}	{...}		MCMPort	This ob
MCM.CONFIG.Port1MasterCmd	{...}	{...}		MCMCmd[325]	This ob
MCM.CONFIG.Port2MasterCmd	{...}	{...}		MCMCmd[325]	This ob
MCM.DATA	{...}	{...}		MCMDATA	This ob
MCM.STATUS	{...}	{...}		MCMSTATUS	This ob
MCM.CONTROL	{...}	{...}		MCMCONTROL	This ob
MCM.UTIL	{...}	{...}		MCMUTIL	This ob

Los valores que estan en las casilla como ejemplo corresponden a la siguiente ubicación dentro de la base de datos:

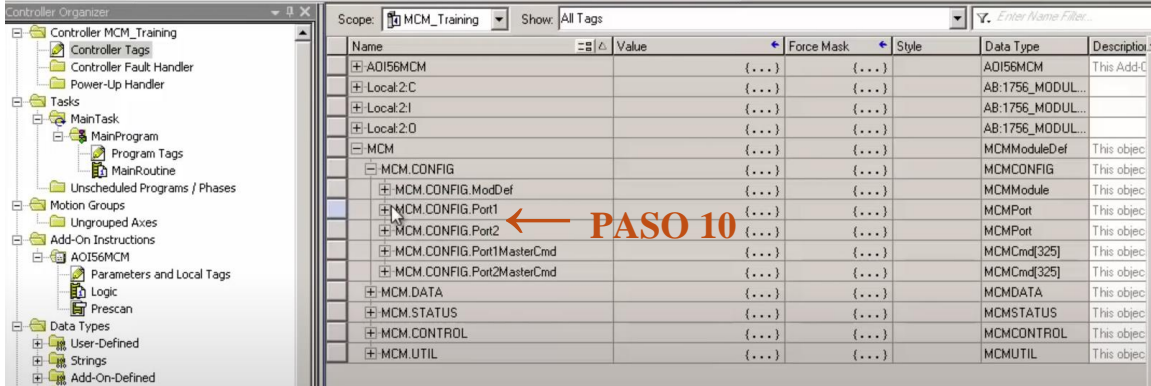


Name	Value	Force Mask	Style	Data Type	Descri
MCM.CONFIG.ModDef	{...}	{...}		MCM...	This object contain
MCM.CONFIG.ModDef.WriteStartReg	0		Decimal	INT	This object contain
MCM.CONFIG.ModDef.WriteStartReg.0	0		Decimal	BOOL	This object contain
MCM.CONFIG.ModDef.WriteStartReg.1	0		Decimal	BOOL	This object contain
MCM.CONFIG.ModDef.WriteStartReg.2	0		Decimal	BOOL	This object contain
MCM.CONFIG.ModDef.WriteStartReg.3	0		Decimal	BOOL	This object contain
MCM.CONFIG.ModDef.WriteStartReg.4	0		Decimal	BOOL	This object contain
MCM.CONFIG.ModDef.WriteStartReg.5	0		Decimal	BOOL	This object contain
MCM.CONFIG.ModDef.WriteStartReg.6	0		Decimal	BOOL	This object contain
MCM.CONFIG.ModDef.WriteStartReg.7	0		Decimal	BOOL	This object contain
MCM.CONFIG.ModDef.WriteStartReg.8	0		Decimal	BOOL	This object contain
MCM.CONFIG.ModDef.WriteStartReg.9	0		Decimal	BOOL	This object contain
MCM.CONFIG.ModDef.WriteStartReg.10	0		Decimal	BOOL	This object contain
MCM.CONFIG.ModDef.WriteStartReg.11	0		Decimal	BOOL	This object contain
MCM.CONFIG.ModDef.WriteStartReg.12	0		Decimal	BOOL	This object contain
MCM.CONFIG.ModDef.WriteStartReg.13	0		Decimal	BOOL	This object contain
MCM.CONFIG.ModDef.WriteStartReg.14	0		Decimal	BOOL	This object contain
MCM.CONFIG.ModDef.WriteStartReg.15	0		Decimal	BOOL	This object contain
MCM.CONFIG.ModDef.WriteRegCnt	600		Decimal	INT	This object contain



Después de configurar la base de datos del módulo se configuran los puertos en la etiquetas MCM.CONFIG.PORT(1 o 2)

• PASO 10: Configuración de los parámetros del puerto:



Name	Value	Force Mask	Style	Data Type	Description
ADI56MCM	(...)	(...)	(...)	ADI56MCM	This Add-On...
Local2C	(...)	(...)	(...)	AB:1756_MODUL...	
Local2I	(...)	(...)	(...)	AB:1756_MODUL...	
Local2O	(...)	(...)	(...)	AB:1756_MODUL...	
MCM	(...)	(...)	(...)	MCMModuleDef	This objec...
MCM.CONFIG	(...)	(...)	(...)	MCMCONFIG	This objec...
MCM.CONFIG.ModDef	(...)	(...)	(...)	MCMModule	This objec...
MCM.CONFIG.Port1	(...)	(...)	(...)	MCMPort	This objec...
MCM.CONFIG.Port2	(...)	(...)	(...)	MCMPort	This objec...
MCM.CONFIG.Port1MasterCmd	(...)	(...)	(...)	MCMCmd[325]	This objec...
MCM.CONFIG.Port2MasterCmd	(...)	(...)	(...)	MCMCmd[325]	This objec...
MCM.DATA	(...)	(...)	(...)	MCMData	This objec...
MCM.STATUS	(...)	(...)	(...)	MCMStatus	This objec...
MCM.CONTROL	(...)	(...)	(...)	MCMControl	This objec...
MCM.UTIL	(...)	(...)	(...)	MCMUtil	This objec...

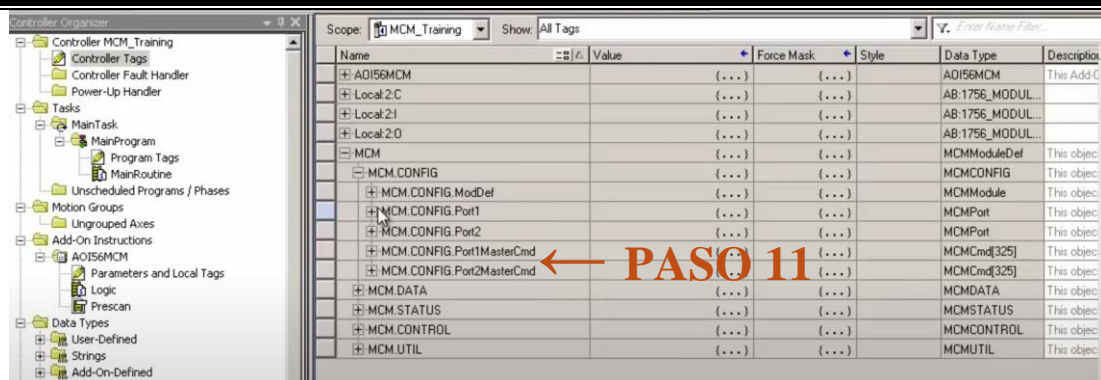
- 1) Este parámetro habilita o deshabilita el puerto, para habilitar se utiliza el valor de "1" y para deshabilitar el valor de "0"
- 2) Este parámetro indica que tipo de puerto será. Si el valor es "0" será un maestro y si el valor es "1" será un esclavo.
- 3) Los parámetros ubicados deben coincidir con los dispositivos a los que se va conectar en la red de modbus, la velocidad en baudios, el bit de paridad, en este caso "0" es sin paridad y "1" es el bit de paridad. Los bits de datos y los bits de parada.

[-] MCM.CONFIG.Port1	{...}	
[+] MCM.CONFIG.Port1.Enabled	1	← 1
[+] MCM.CONFIG.Port1.Type	0	← 2
[+] MCM.CONFIG.Port1.FloatFlag	0	
[+] MCM.CONFIG.Port1.FloatStart	0	
[+] MCM.CONFIG.Port1.FloatOffset	0	
[+] MCM.CONFIG.Port1.Protocol	0	
[+] MCM.CONFIG.Port1.Baudrate	19200	← 3
[+] MCM.CONFIG.Port1.Parity	0	←
[+] MCM.CONFIG.Port1.DataBits	8	←
[+] MCM.CONFIG.Port1.StopBits	1	←
[+] MCM.CONFIG.Port1.RTSOn	0	
[+] MCM.CONFIG.Port1.RTSOff	0	
[+] MCM.CONFIG.Port1.MinResp	0	
[+] MCM.CONFIG.Port1.UseCTS	0	
[+] MCM.CONFIG.Port1.SlaveID	0	
[+] MCM.CONFIG.Port1.BitInOffset	0	
[+] MCM.CONFIG.Port1.WordInOffset	0	

Complementando la información esta imagen saca de el manual del dispositivo, deja mas claro las funciones principales que se deben configurar en el puerto dependiente de la comunicación.

Parameter	Description
Enabled	1 = Enable port, 0 = Disable port
Type	0 = Master 1 = Slave 2 = Slave with unformatted pass-through 3 = Slave with formatted pass-through, with data swapping 4 = Slave with formatted pass-through, with no data swapping
Protocol	0 = Modbus RTU mode 1 = Modbus ASCII mode
Baud rate	Sets the baud rate for the port. Valid values for this field are 110, 150, 300, 600, 1200, 2400, 4800, 9600, 19200, 384 (for 38,400 baud), 576 (for 57,600 baud) and 115 or 1152 (for 115,200 baud)
Parity	0 = None 1 = Odd 2 = Even
DataBits	8 = Modbus RTU mode 7 or 8 = Modbus ASCII mode
StopBits	Valid values are 1 or 2.
RTS On	0 to 65535 milliseconds to delay after RTS line is asserted on the port before data message transmission begins. This delay can be used to allow for radio keying or modem dialing before data transmission begins.
RTS Off	0 to 65535 milliseconds to delay after data message is complete before RTS line is dropped on the port.

- PASO 11: Configuración de los comandos maestros de cada puerto.



Este modulo tienen la posibilidad de crear 325 comandos para el puerto 1, los cuales funcionaran apartir de la configuracion que ya se tiene creada en la base de datos del modulo. Se debe consultar el manual de usuario de los diferentes dispositivos que estaran conectados en la red modbus para saber que direccion se desean leer o escribir dentro de la red.

1) ENABLE: 0 = Deshabilitado el comando no se ejecutará, pero se puede habilitar usando el comando opción de control en lógica de escalera. 1 = habilitado, el comando está habilitado y se enviará al dispositivo de destino. 2 = Escritura condicional solo para Func 5, 15, 6 o 16 los datos se enviarán solo al dispositivo de destino cuando los datos a escribir han cambiado.


2) IntAddress: Especifica el registro de la base de datos interna del módulo que se asociará con el comando. Si el comando es una función de lectura, los datos leídos desde el dispositivo servidor se almacena comenzando en el valor de registro de la base de datos interna del módulo ingresado en este campo. Este valor de registro debe estar en el área de ReadData de la memoria del módulo, definido por el inicio del registro de lectura y el inicio de lectura. Registre los parámetros de recuento en la sección Módulo.

Si el comando es una función de escritura, los datos que se escribirán en el servidor del dispositivo se obtiene a partir de la base de datos interna del módulo del registro especificado, este valor de registro debe provenir de Write Data área de la memoria del módulo, definida por el inicio del registro de escritura y escriba los parámetros de recuento de registros en la sección Módulo.

Para los códigos de función Modbus 3, 4, 6 o 16, el rango permitido es de 0 a 9999 (enteros de 16 bits).

Para los códigos de función Modbus 1, 2, 5 o 15, el rango permitido es de 0 a 159.999 (bits). Nota: Este rango de direcciones de bits está disponible con el firmware MVI56EMCM v3.07.0

3) COUNT: Establece cuántas palabras continuas (códigos de función 3, 4 y 16) o bits (Códigos de función 1, 2 y 15) para solicitar desde el dispositivo esclavo. Los valores válidos son de 1 a 125 palabras para los códigos de función 3, 4 y 16, mientras que puede especificar un rango de 1 a 2000 para los códigos de función 1, 2 y 15. Nota: Estos valores son los máximos permitidos en el protocolo Modbus. Algunos dispositivos pueden admitir menos palabras o bits que el máximo permitido.

 UNIVERSIDAD SANTO TOMÁS PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA BU CAR A M A N G A	GESTIÓN DEL CONOCIMIENTO, APRENDIZAJE Y DOCENCIA	VERSIÓN 01
	GUÍA DE LABORATORIO No. 7	CODIGO -
		Página 15 de 12

4) **NODE:** Dirección de nodo del dispositivo en la red para leer o escribir datos a. Las direcciones válidas son del 1 al 247. La dirección 0 está reservada para transmitir comandos de escritura (transmitirá un comando de escritura a todos dispositivos en la red).

5) **FUNC:**

Especifica la función Modbus que ejecutará el comando. tesis

Los códigos de función se definen en el protocolo Modbus.

1 = Leer estado de la bobina (0xxxx)

2 = Leer estado de entrada (1xxxx)

3 = Leer registros de retención (4xxxx)

4 = Leer registros de entrada (3xxxx)

5 = Bobina forzada (escritura única) (0xxxx)

6 = Forzar (escritura única) Registro de retención (4xxxx)

15 = Preestablecido (Escritura) Múltiples Bobinas (0xxxx)

16 = Preestablecido (Escritura) Registros Múltiples (4xxxx)

6) **DevAddress:** Especifica la dirección Modbus Slave para los registros asociados con ese comando. Esta es la dirección de compensación para el dispositivo esclavo Modbus.

Con Modbus, para leer una dirección de 40001, ¿cuál será realmente?

El puerto de salida transmitido es el código de función 03 (un byte) con una dirección de

00 00 (dos bytes). Esto significa que para leer una dirección de 40501, use

Func 3 con una DevAddress de 500.

Esto se aplica a las direcciones Modbus 10001 a 47999.

A continuación se muestra una definición que le ayudará con la configuración de DevAddress:

Códigos de función 1, 5 o 15

- DevAddress = dirección Modbus en el dispositivo - 0001

Ejemplo: dirección Modbus 0001 = DevAddress 0

- Dirección Modbus 1378 = Dirección Dev 1377

Código de función 2

- DevAddress = dirección Modbus en el dispositivo - 10001

Ejemplo: dirección Modbus 10001 = DevAddress 0

- Dirección Modbus 10345 = Dirección Dev 344

Códigos de función 3, 6 o 16

- DevAddress = dirección Modbus en el dispositivo - 40001

Ejemplo: dirección Modbus 40001 = DevAddress 0

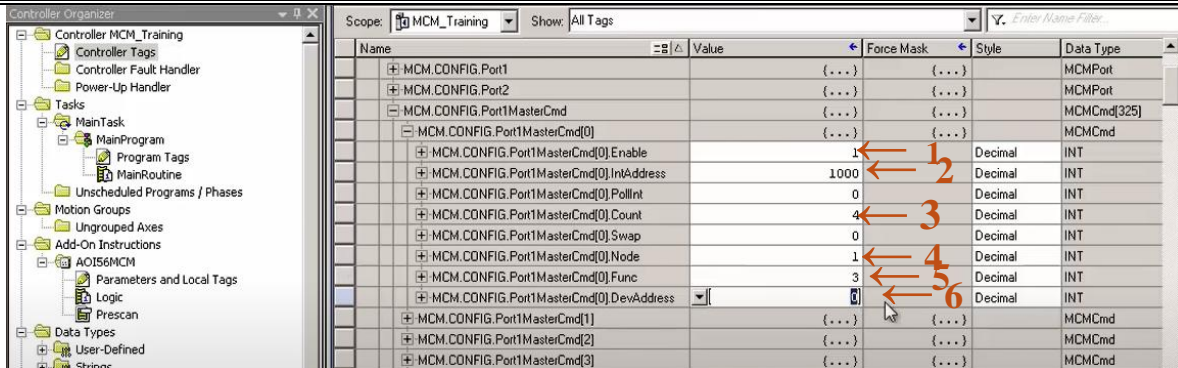
- Dirección Modbus 40591 = Dirección Dev 590

Código de función 4

- DevAddress = dirección Modbus en el dispositivo - 30001

Ejemplo: dirección Modbus 30001 = DevAddress 0

- Dirección Modbus 34290 = Dirección Dev 4289



Ejemplo de configuración para leer "Holding register"

MCM.CONFIG.Port1MasterCmd[0]	{...}
MCM.CONFIG.Port1MasterCmd[0].Enable	1
MCM.CONFIG.Port1MasterCmd[0].IntAddress	1000
MCM.CONFIG.Port1MasterCmd[0].PollInt	0
MCM.CONFIG.Port1MasterCmd[0].Count	10
MCM.CONFIG.Port1MasterCmd[0].Swap	0
MCM.CONFIG.Port1MasterCmd[0].Node	1
MCM.CONFIG.Port1MasterCmd[0].Func	3
MCM.CONFIG.Port1MasterCmd[0].DevAddress	0

Label	Description
Enable = 1	The module will send the command every time it goes through the command list.
IntAddress = 1000	Places the data read from the Slave device into the module at address 1000. IntAddress 1000 of the module memory will be copied into the tag MCM.DATA.READDATA[0] .
Count = 10	Reads 10 consecutive registers from the Slave device.
Node = 1	Issues the Modbus command to node 1 on the network.
Func = 3	Issues Modbus Function Code 3 to Read Holding Registers.
DevAddress = 0	Function Code 3, DevAddress of 0 will read address 40001 Along with a count of 10, this command reads 40001 to 40010.

Mapa de la memoria modbus en el dispositivo:

se muestra la dirección Modbus predeterminada para el módulo se puede acceder a cada registro dentro del módulo como una dirección de bit 0xxx, dirección de bit 1xxx, dirección de registro 3xxx o dirección de registro 4xxx.

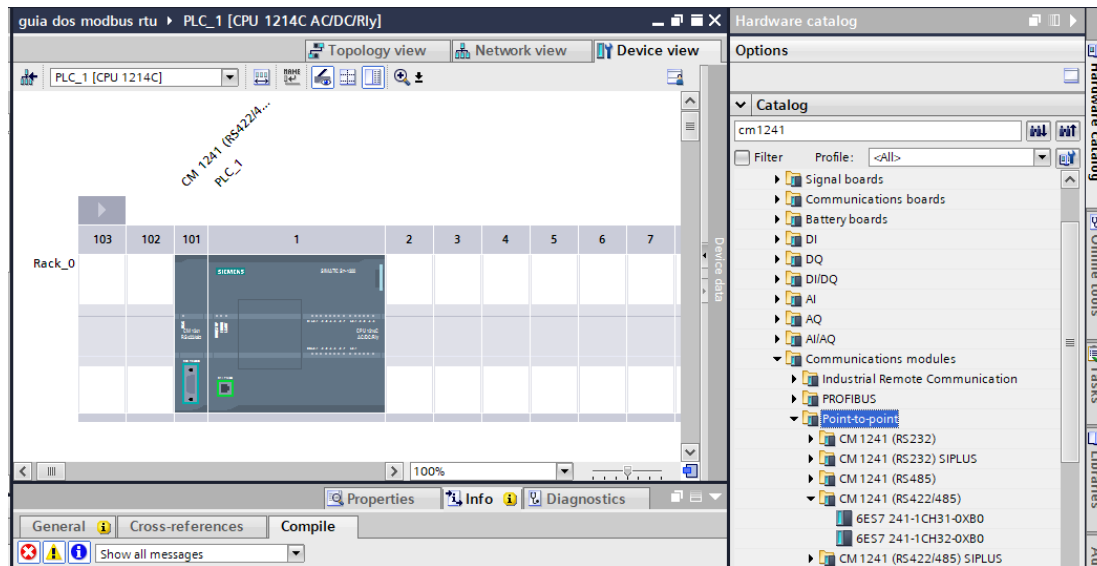


MVI Address	0xxx	1xxxx	3xxxx	4xxxx	Tag Address
0	0001 to 0016	10001 to 10016	30001	40001	WriteData[0]
1	0017 to 0032	10017 to 10032	30002	40002	WriteData[1]
2	0033 to 0048	10033 to 10048	30003	40003	WriteData[2]
3	0049 to 0064	10049 to 10064	30004	40004	WriteData[3]
4	0065 to 0080	10065 to 10080	30005	40005	WriteData[4]
5	0081 to 0096	10081 to 10096	30006	40006	WriteData[5]
6	0097 to 0112	10097 to 10112	30007	40007	WriteData[6]
7	0113 to 0128	10113 to 10128	30008	40008	WriteData[7]
8	0129 to 0144	10129 to 10144	30009	40009	WriteData[8]
9	0145 to 0160	10145 to 10160	30010	40010	WriteData[9]
10	0161 to 0176	10161 to 10176	30011	40011	WriteData[10]
50	0801 to 0816	10801 to 10816	30051	40051	WriteData[50]
100	1601 to 1616	11601 to 11616	30101	40101	WriteData[100]
200	3201 to 3216	13201 to 13216	30201	40201	WriteData[200]
500	8001 to 8016	18001 to 18016	30501	40501	WriteData[500]
598	9569 to 9584	19569 to 19584	30599	40599	WriteData[598]
599	9585 to 9600	19585 to 19600	30600	40600	WriteData[599]
600 to 999	N/A	N/A	N/A	N/A	Reserved
1000			31001*	41001	ReadData[0]
1001			31002*	41002	ReadData[1]
1002			31003*	41003	ReadData[2]
1003			31004*	41004	ReadData[3]
1004			31005*	41005	ReadData[4]
1005			31006*	41006	ReadData[5]
1006			31007*	41007	ReadData[6]
1007			31008*	41008	ReadData[7]
1008			31009*	41009	ReadData[8]
1009			31010*	41010	ReadData[9]
1010			31011*	41011	ReadData[10]
1050			31051*	41051	ReadData[50]
1100			31101*	41101	ReadData[100]
1200			31201*	41201	ReadData[200]
1500			31501*	41501	ReadData[500]
1598			31599*	41599	ReadData[598]
1599			31600*	41600	ReadData[599]

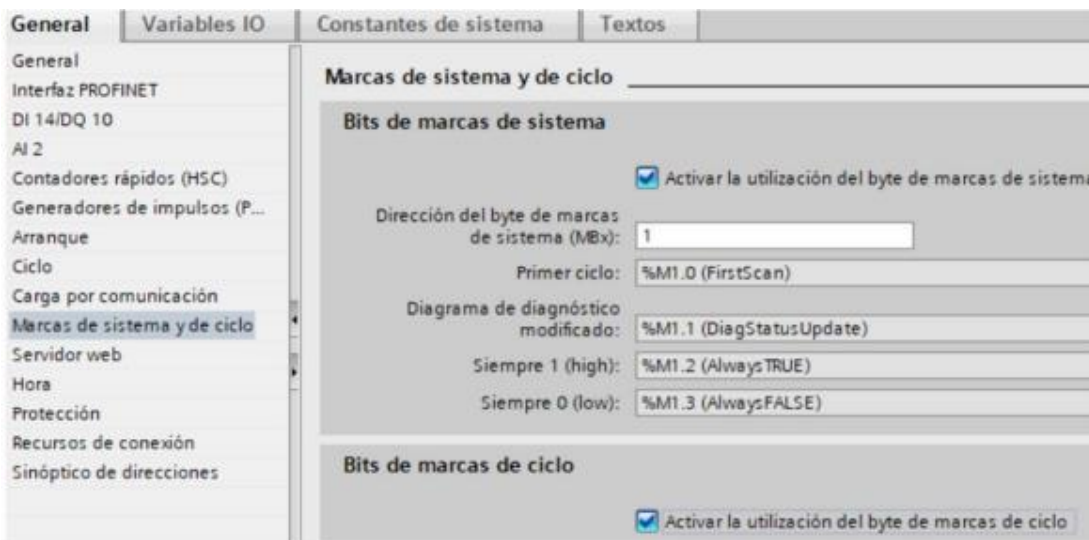
Configuración de una comunicación MODBUS RTU entre PLCs Siemens y Allen Bradley.

En esta sección del documento se mostrarán las configuraciones básicas para comunicar el Módulo MVI56E Maestro/Esclavo diseñado para PLCs de la familia RS-Logix y el módulo CM 1241 para PLCs SIEMENS. Esto desde la interfaz de TIA Portal y Studio 5000 Logix Designer.

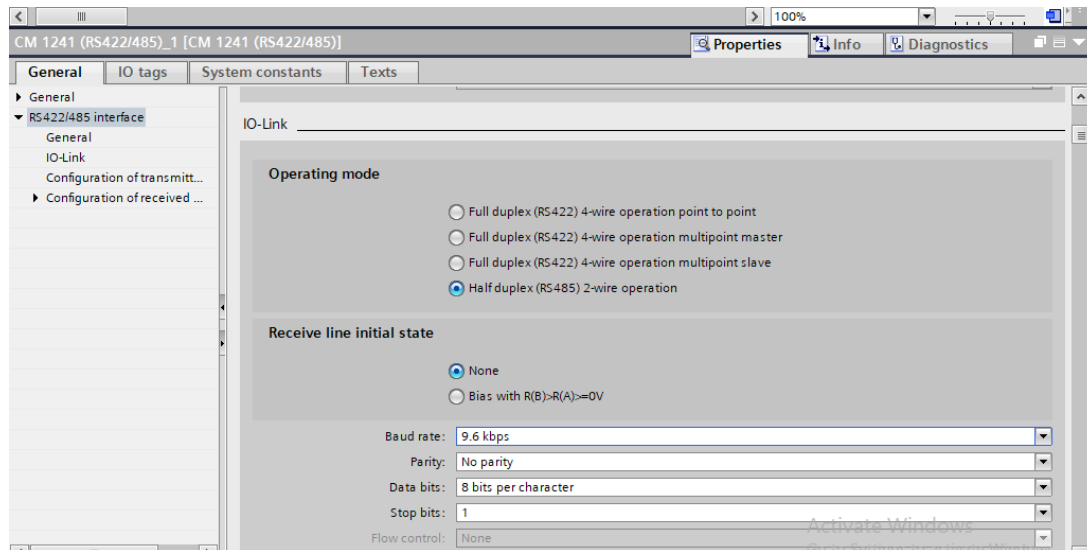
Añadir el plc y el modulo correspondiente en la interfaz de tia portal.



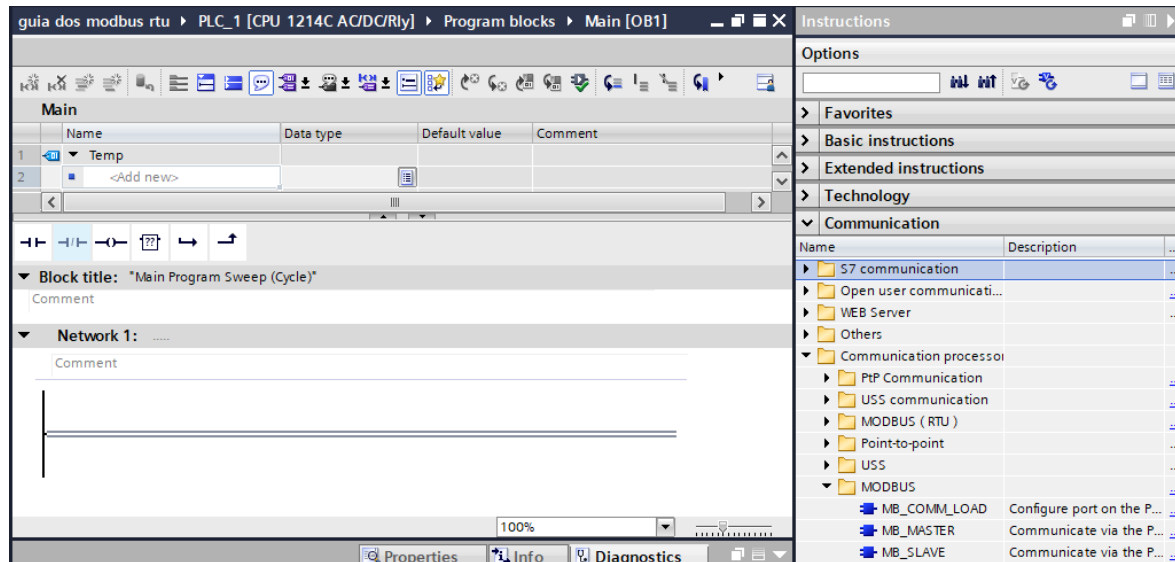
Activar las marcas de sistema y de ciclo.

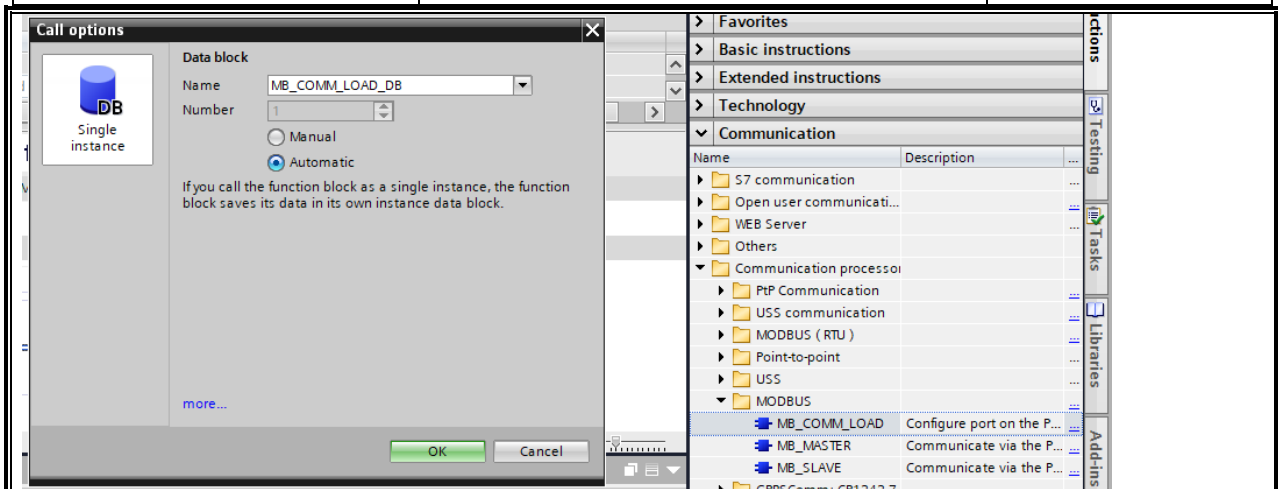


Configuraciones generales del modulo las cuales deben coincidir con el modulo configurado para plics controllogix.



En la biblioteca de instrucciones añadir las funciones para la comunicacion modbus desde siemens.



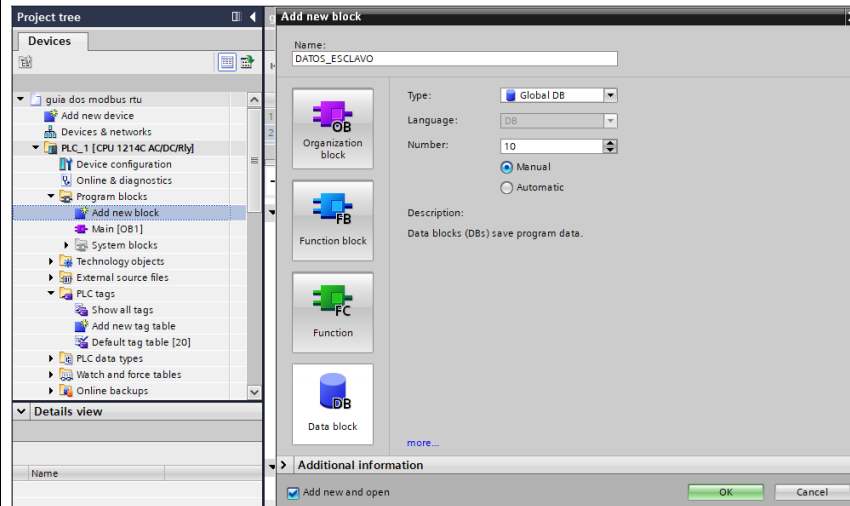


Como queda cada funcion añadida.

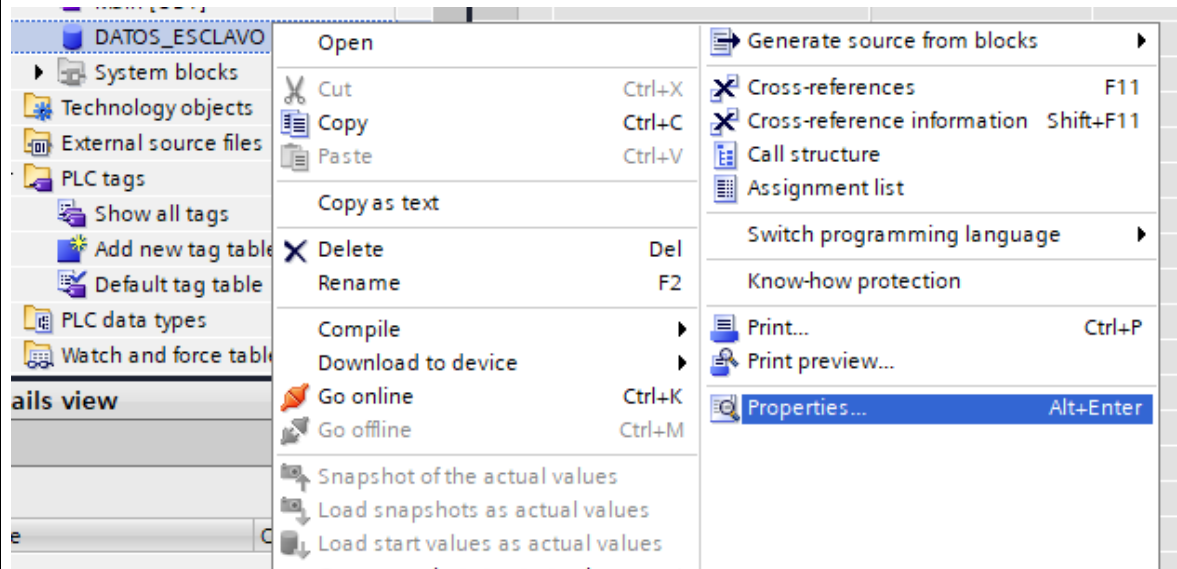


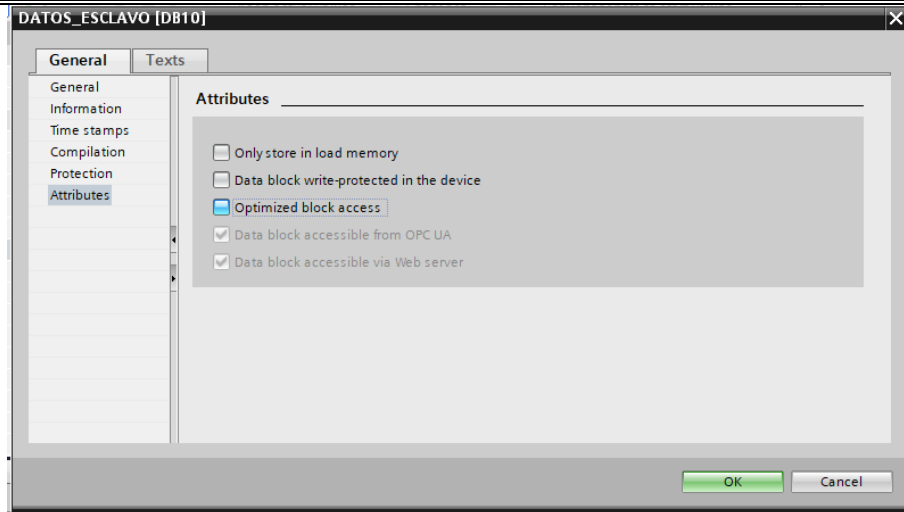
PLC SIEMENS S7-1200 SLAVE Y RS LOGIX MASTER
Lectura y escritura de HOLDING REGISTER.

Se añade un DB de datos don se almacenaran los datos de la comunicación.



En las propiedades del Db de datos quitamos el acceso optimizado al bloque.

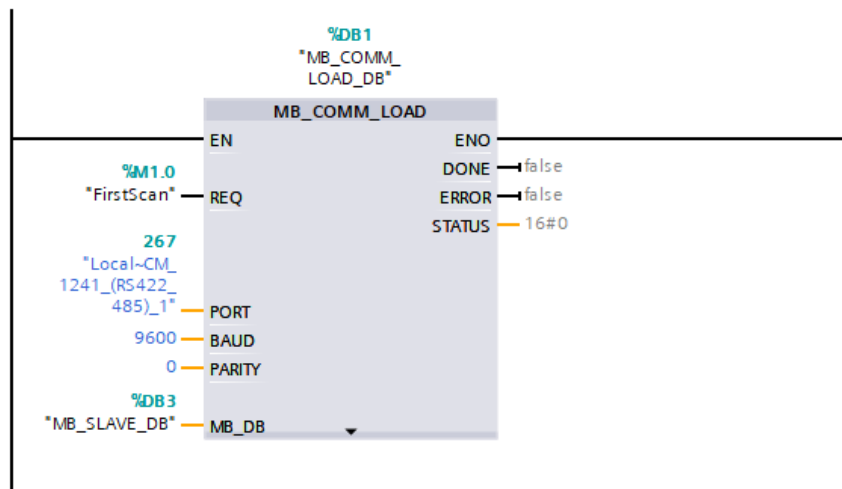




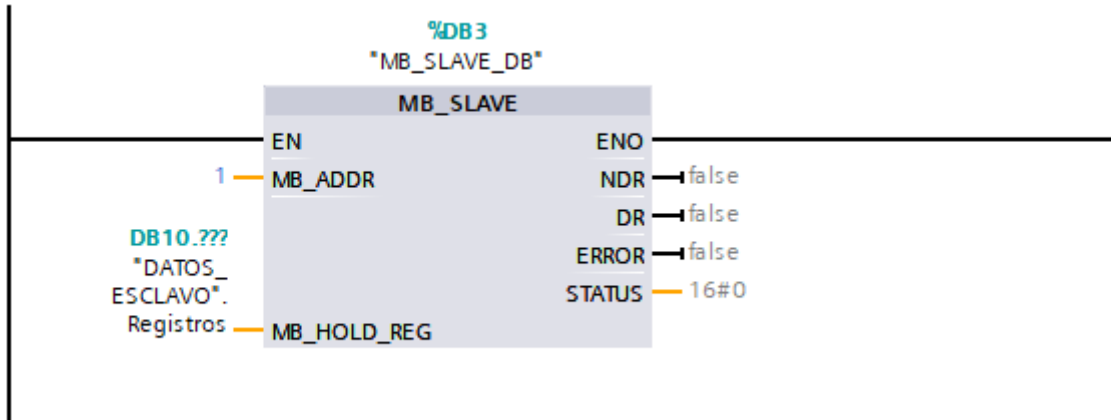
Crear el siguiente arreglo de Registros de tipo INT.

DATOS_ESCLAVO					
	Name	Data type	Offset	Start value	Retain
1	Static				<input type="checkbox"/>
2	Registros	Array[0..5] of Int	...		<input type="checkbox"/>
3	Registros[0]	Int	...	0	<input type="checkbox"/>
4	Registros[1]	Int	...	0	<input type="checkbox"/>
5	Registros[2]	Int	...	0	<input type="checkbox"/>
6	Registros[3]	Int	...	0	<input type="checkbox"/>
7	Registros[4]	Int	...	0	<input type="checkbox"/>
8	Registros[5]	Int	...	0	<input type="checkbox"/>
9	<Add new>				<input type="checkbox"/>

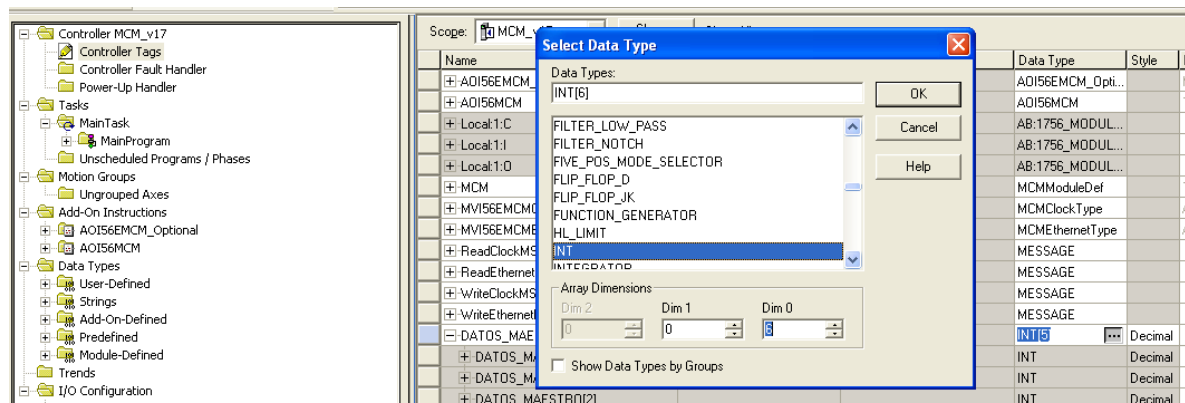
Configuración plc siemens como esclavo.



Configuración del espacio de memoria donde se recibirán los datos



Configuración del espacio de memoria del controllogix donde se almacenarán los datos.



- DATOS_MAE		{...}	{.	Decimal	INT[6]
+ DATOS_MAE[0]		0		Decimal	INT
+ DATOS_MAE[1]		0		Decimal	INT
+ DATOS_MAE[2]		0		Decimal	INT
+ DATOS_MAE[3]		0		Decimal	INT
+ DATOS_MAE[4]		0		Decimal	INT
+ DATOS_MAE[5]		0		Decimal	INT

Configuración de la base de datos del MVI56-MCM donde se almacenarán los datos obtenidos de la red modbus en este caso del plc s7-1200

WriteStartReg = 0, Dirección de inicio de los datos que van desde el controllogix al plc s7-1200.

WriteRegCnt = 6, Los espacios de memoria que se usaran para guardar los holding register.

ReadStartReg = 1000, Dirección de inicio de los datos que van desde el plc s7-1200 al controllogix.

ReadRegCnt = 6, Los espacios de memoria que se usaran para leer los holding register.

MCM	{...}	{...}		MCMModuleDef
MCM.CONFIG	{...}	{...}		MCMCONFIG
MCM.CONFIG.ModDef	{...}	{...}		MCMModule
+ MCM.CONFIG.ModDef.WriteStartReg		0	Decimal	INT
+ MCM.CONFIG.ModDef.WriteRegCnt		6	Decimal	INT
+ MCM.CONFIG.ModDef.ReadStartReg		1000	Decimal	INT
+ MCM.CONFIG.ModDef.ReadRegCnt		6	Decimal	INT
+ MCM.CONFIG.ModDef.BPFail		0	Decimal	INT
+ MCM.CONFIG.ModDef.ErrStatPtr		-1	Decimal	INT

Configuración de los parámetros del puerto 1 los cuales deben coincidir con los parámetros configurados en tía portal.

La variable Type = 0 configura el puerto 1 como maestro.

MCM	{...}		
MCM.CONFIG	{...}		
+ MCM.CONFIG.ModDef	{...}		
MCM.CONFIG.Port1	{...}		
+ MCM.CONFIG.Port1.Enabled		1	
+ MCM.CONFIG.Port1.Type		0	
+ MCM.CONFIG.Port1.FloatFlag		0	
+ MCM.CONFIG.Port1.FloatStart		0	
+ MCM.CONFIG.Port1.FloatOffset		0	
+ MCM.CONFIG.Port1.Protocol		0	
+ MCM.CONFIG.Port1.Baudrate		9600	
+ MCM.CONFIG.Port1.Parity		0	
+ MCM.CONFIG.Port1.DataBits		8	
+ MCM.CONFIG.Port1.StopBits		1	

Operating mode

Full duplex (RS422) 4-wire operation point to point
 Full duplex (RS422) 4-wire operation multipoint master
 Full duplex (RS422) 4-wire operation multipoint slave
 Half duplex (RS485) 2-wire operation

Receive line initial state

None
 Bias with R(B):-R(A)=-0V

Baud rate: 9.6 kbps

Parity: No parity

Data bits: 8 bits per character

Stop bits: 1

Flow control: None

Escritura en los holding register:

Configuran los comandos master del puerto. El Port1MasterCmd[0] se configurara para escribir en los holding register.

Enable = 1, Puerto activado

IntAddr = 0, Dirección de inicio donde se escribirán los datos de la red modbus

Count = 6, La cantidad de datos que se enviarán por la red modbus en este caso de tipo INT

Node = 1, El nodo donde está el esclavo en la red modbus.

Func = 16, Función 16 corresponde a la escritura de los holding register



- MCM	{...}
- MCM.CONFIG	{...}
+ MCM.CONFIG.ModDef	{...}
+ MCM.CONFIG.Port1	{...}
+ MCM.CONFIG.Port2	{...}
- MCM.CONFIG.Port1MasterCmd	{...}
- MCM.CONFIG.Port1MasterCmd[0]	{...}
+ MCM.CONFIG.Port1MasterCmd[0].Enable	1
+ MCM.CONFIG.Port1MasterCmd[0].IntAddr...	0
+ MCM.CONFIG.Port1MasterCmd[0].PollInt	0
+ MCM.CONFIG.Port1MasterCmd[0].Count	6
+ MCM.CONFIG.Port1MasterCmd[0].Swap	0
+ MCM.CONFIG.Port1MasterCmd[0].Node	1
+ MCM.CONFIG.Port1MasterCmd[0].Func	16
+ MCM.CONFIG.Port1MasterCmd[0].DevAd...	0

Leer registros esclavo:

Enable =2, Con la funcion 2 el puerto solo se activa cuando se detectan cambios en los holding register
IntAddr = 1000, Direccion de inicio donde se leeran los datos de la red modbus.

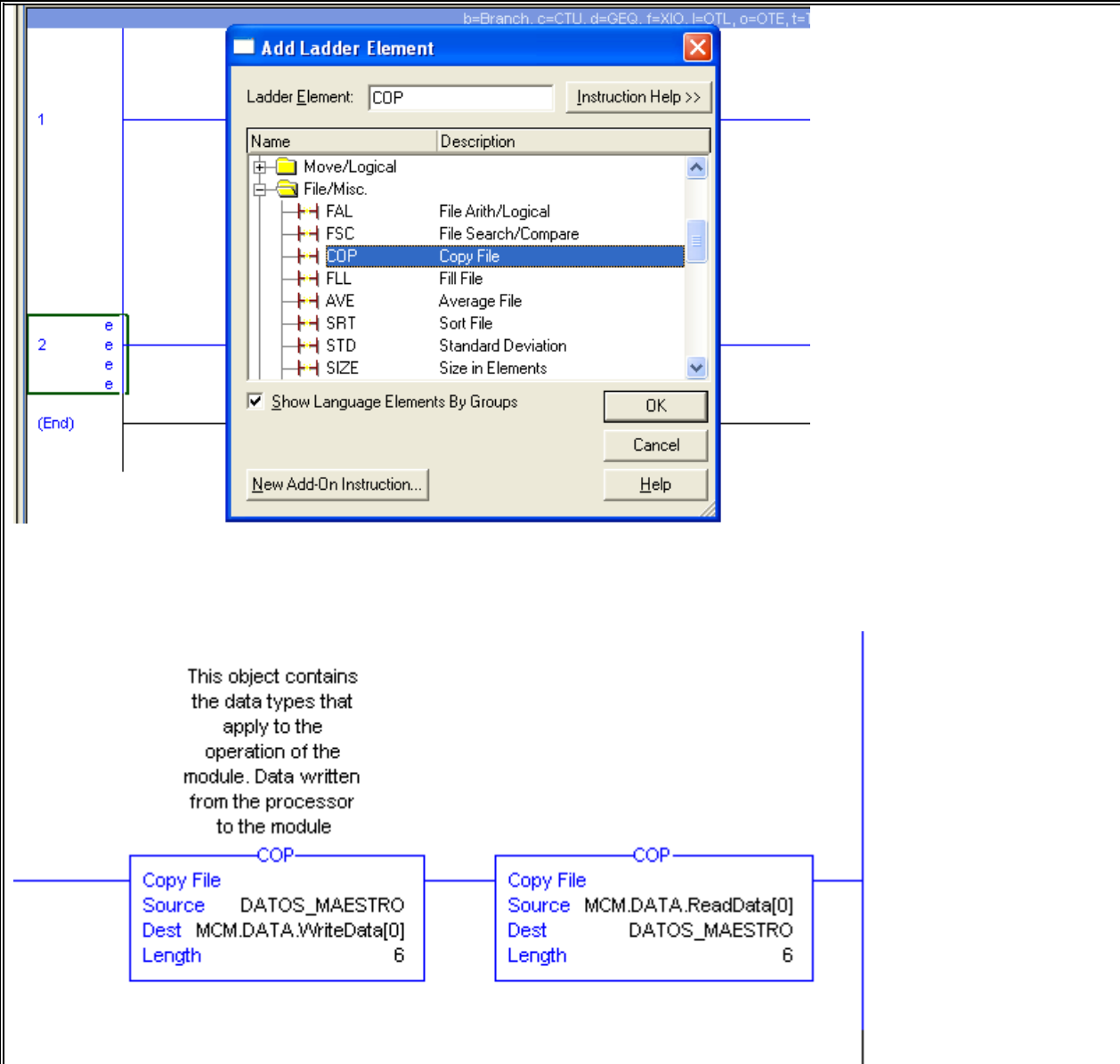
Count = 6, La cantidad de datos que leeran en la red modbus en este caso de tipo INT

Node = 1, El nodo donde esta el esclavo en la red modbus.

Func = 3, Funcion 3 corresponde a la lectura de holding register

- MCM	{...}
- MCM.CONFIG	{...}
+ MCM.CONFIG.ModDef	{...}
+ MCM.CONFIG.Port1	{...}
+ MCM.CONFIG.Port2	{...}
- MCM.CONFIG.Port1MasterCmd	{...}
+ MCM.CONFIG.Port1MasterCmd[0]	{...}
- MCM.CONFIG.Port1MasterCmd[1]	{...}
+ MCM.CONFIG.Port1MasterCmd[1].Enable	2
+ MCM.CONFIG.Port1MasterCmd[1].IntAddr...	1000
+ MCM.CONFIG.Port1MasterCmd[1].PollInt	0
+ MCM.CONFIG.Port1MasterCmd[1].Count	6
+ MCM.CONFIG.Port1MasterCmd[1].Swap	0
+ MCM.CONFIG.Port1MasterCmd[1].Node	1
+ MCM.CONFIG.Port1MasterCmd[1].Func	3
+ MCM.CONFIG.Port1MasterCmd[1].DevAd...	<input type="text" value="0"/>

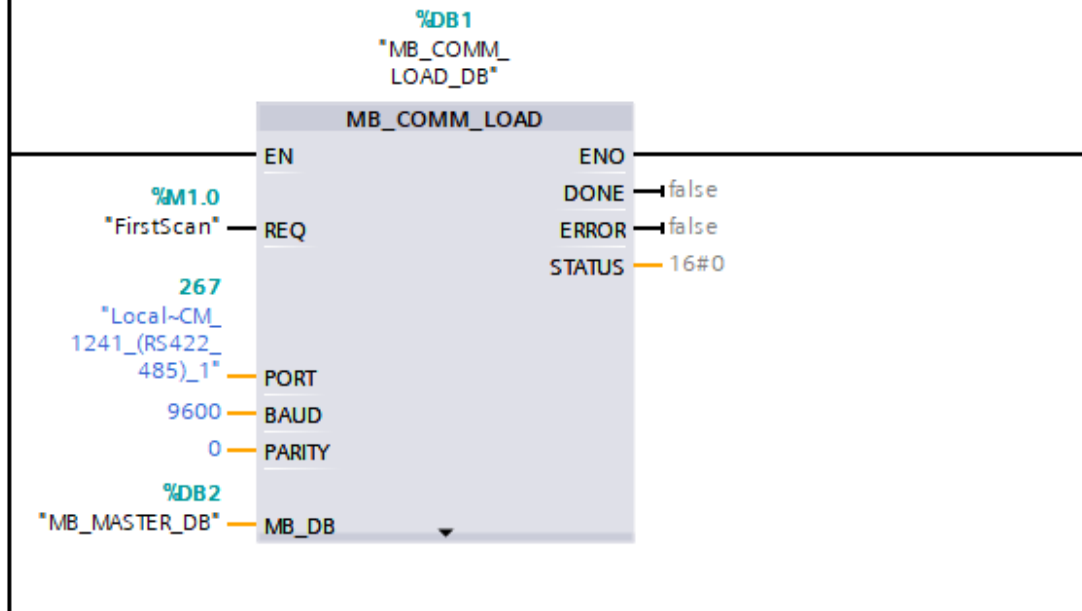
Con la funcion Copy se intercambian los datos tomados de la red modbus y guardados en la base de datos del MV156-MCM y se envian a la memoria del controllogix



This object contains the data types that apply to the operation of the module. Data written from the processor to the module

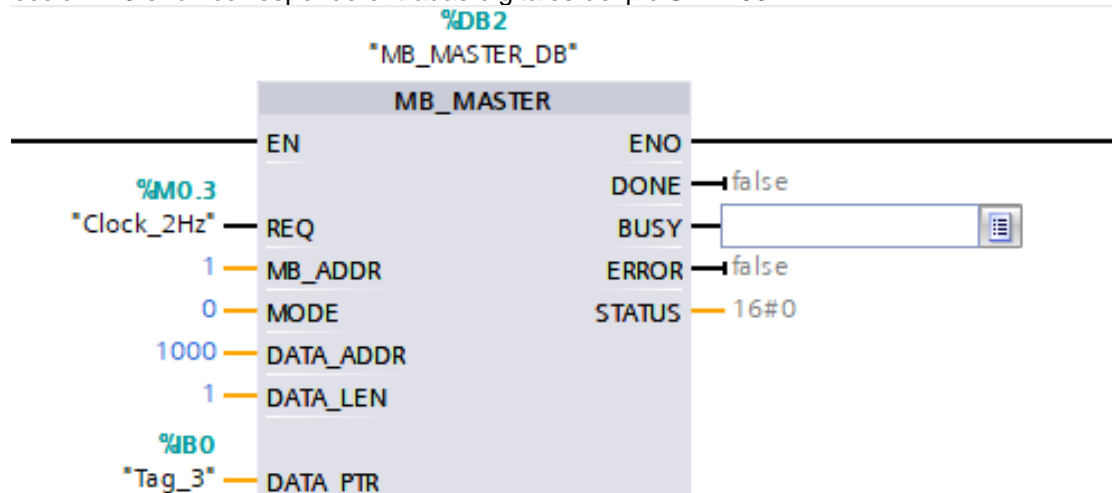
Source	Dest	Length
DATOS_MAESTRO	MCM.DATA.WriteData[0]	6
MCM.DATA.ReadData[0]	DATOS_MAESTRO	6

PLC SIEMENS S7-1200 MASTER Y RS LOGIX SLAVE
Lectura y escritura de BITS
Configuracion plc siemens como maestro.



Configuracion del maestro.

La direccion IB0 e Ib1 corresponde entradas digitales del plc S7-1200



Configuracion de la base de datos del MVI56-MCM donde se almacenaran los datos enviados desde el maestro modbus en este caso del plc s7-1200

WriteStartReg = 0, Direccion de inicio de los datos que van desde el controllogix al plc s7-1200.

WriteRegCnt = 1, Los espacios de memoria que se usaran para guardar los holding register.

ReadStartReg = 1000, Direccion de inicio de los datos que van desde el plc s7-1200 al controllogix.

ReadRegCnt = 1, Los espacios de memoria que se usaran para leer los holding register.

- MCM	{...}
- MCM.CONFIG	{...}
- MCM.CONFIG.ModDef	{...}
+ MCM.CONFIG.ModDef.WriteStartReg	0
+ MCM.CONFIG.ModDef.WriteRegCnt	1
+ MCM.CONFIG.ModDef.ReadStartReg	1000
+ MCM.CONFIG.ModDef.ReadRegCnt	1
+ MCM.CONFIG.ModDef.BPFail	0
+ MCM.CONFIG.ModDef.ErrStatPtr	-1

Configuracion de los parametros del puerto 1 los cuales deben coincidir con los parametros configurados en tia portal.

Type = 1 configura el puerto 1 como esclavo.

- MCM	{...}
- MCM.CONFIG	{...}
+ MCM.CONFIG.ModDef	{...}
- MCM.CONFIG.Port1	{...}
+ MCM.CONFIG.Port1.Enabled	1
+ MCM.CONFIG.Port1.Type	1
+ MCM.CONFIG.Port1.FloatFlag	0
+ MCM.CONFIG.Port1.FloatStart	0
+ MCM.CONFIG.Port1.FloatOffset	0
+ MCM.CONFIG.Port1.Protocol	0
+ MCM.CONFIG.Port1.Baudrate	9600
+ MCM.CONFIG.Port1.Parity	0
+ MCM.CONFIG.Port1.DataBits	8
+ MCM.CONFIG.Port1.StopBits	1

Operating mode

Full duplex (RS422) 4-wire operation point to point
 Full duplex (RS422) 4-wire operation multipoint master
 Full duplex (RS422) 4-wire operation multipoint slave
 Half duplex (RS485) 2-wire operation

Receive line initial state

None
 Bias with R(B)>R(A)>0V

Baud rate: 9.6 kbps

Parity: No parity

Data bits: 8 bits per character

Stop bits: 1

Flow control: None

En este ejemplo solo se enviaron las primeras 8 bobinas del plc s7-1200 y se recibieron en el plc de la familia controllogix.

Escritura de datos maestro esclavo (Masestro S7-1200 al controllogix processor)

Configuran los comandos master del puerto. El Port1MasterCmd[0] se configurara para escribir en los holding register.

Enable = 1, Puerto activado

IntAddr = 0, Direccion de inicio donde se escribiran los datos de la red modbus

Count = 1, La cantidad de datos que se enviaron por la red modbus en este caso de tipo INT

Node = 1, El nodo donde esta el esclavo en la red modbus.

Func = 15, Funcion 15 corresponde a la escritura de multiples bobinas



- MCM.CONFIG.Port1MasterCmd	{...}	{...}
- MCM.CONFIG.Port1MasterCmd[0]	{...}	{...}
+ MCM.CONFIG.Port1MasterCmd[0].Enable	1	
+ MCM.CONFIG.Port1MasterCmd[0].IntAddr...	0	
+ MCM.CONFIG.Port1MasterCmd[0].PollInt	0	
+ MCM.CONFIG.Port1MasterCmd[0].Count	1	
+ MCM.CONFIG.Port1MasterCmd[0].Swap	0	
+ MCM.CONFIG.Port1MasterCmd[0].Node	1	
+ MCM.CONFIG.Port1MasterCmd[0].Func	15	
+ MCM.CONFIG.Port1MasterCmd[0].DevAd...	0	
+ MCM.CONFIG.Port1MasterCmd[1]	{...}	{...}

lectura de datos maestro esclavo (Masestro S7-1200 al controllogix processor)

Configuran los comandos master del puerto. El Port1MasterCmd[0] se configurara para escribir en los holding register.

Enable = 2, La funcion 2 activa el puerto siempre que haya un cambios en las bobinas

IntAddr = 0, Direccion de inicio donde se escribiran los datos de la red modbus

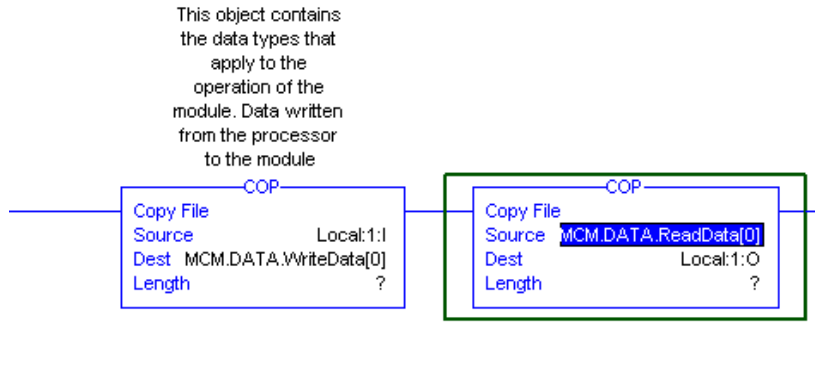
Count = 1, La cantidad de datos que se enviaron por la red modbus en este caso de tipo INT

Node = 1, El nodo donde esta el esclavo en la red modbus.

Func = 1, Funcion 1 corresponde a la lectrua del estado de las bobinas

- MCM.CONFIG.Port1MasterCmd	{...}	{...}
+ MCM.CONFIG.Port1MasterCmd[0]	{...}	{...}
- MCM.CONFIG.Port1MasterCmd[1]	{...}	{...}
+ MCM.CONFIG.Port1MasterCmd[1].Enable	2	
+ MCM.CONFIG.Port1MasterCmd[1].IntAddr...	1000	
+ MCM.CONFIG.Port1MasterCmd[1].PollInt	0	
+ MCM.CONFIG.Port1MasterCmd[1].Count	1	
+ MCM.CONFIG.Port1MasterCmd[1].Swap	0	
+ MCM.CONFIG.Port1MasterCmd[1].Node	1	
+ MCM.CONFIG.Port1MasterCmd[1].Func	1	
+ MCM.CONFIG.Port1MasterCmd[1].DevAd...	0	
+ MCM.CONFIG.Port1MasterCmd[2]	{...}	{...}

Con la funcion "Copy" pasamos la informacion de la base de datos del modulo MVI56E-MCM a la memoria del procesador controllogix, en este caso a las entradas y salidas.



Configuración de comunicación profibus entre S7-1200 y el módulo ILX56-PBM


Ir al sitio web de soporte del módulo ILX56-PBM y descargar el software de prosoft para la configuración de estos dispositivos. (<https://www.prosoft-technology.com/Products/Rockwell-Automation-In-chassis/Platform/ControlLogix/PROFIBUS-DPV1-Master-Slave-for-ControlLogix>)

Arquitecturas	Características y beneficios	Presupuesto	Hardware	Descargas	Videos	Certificaciones
---------------	------------------------------	-------------	----------	-----------	--------	-----------------

Descargas

Ficha técnica del ILX56-PBM	414,38 kB	Última modificación: 09/12/2022
Manual del usuario de ILX56-PBM	4,80 MB	Última modificación 14/06/2024
Archivo GSD ILX56-PBM	3,60 kB	Última modificación 18/05/2020
Archivos EDS del ILX56-PBM	2,54 kB	Última modificación 04/11/2024
Firmware ILX56-PBM v1.001.014	189,99 kB	Última modificación: 29/11/2022
Notas de la versión de ILX56-PBM	578,71 kB	Última modificación: 02/02/2023
Instrucciones complementarias para ILX56-PBM	10,22 kB	Última modificación: 03/04/2020
Archivos AOP ILX56-PBM / ILX56-PBS	103,26 kB	Última modificación: 07/07/2023
Dibujos CAD del modelo ILX56-PBx (STEP)	7,45 MB	Última modificación 15/09/2023
Software de configuración del paquete DTM PLX51 ILX56 HART y PROFIBUS v1.008	4,51 MB	Última modificación: 03/04/2020
Utilidad de configuración ProSoft PLX50		

En la ventana emergente ir a descargas en instalar el archivo que apareciera en la pagina.



Utilidad de configuración PLX50

Trabajador social de la unidad de cuidados intensivos (PCU)

La utilidad de configuración gratuita PLX50 proporciona toda la funcionalidad necesaria para configurar, monitorear y diagnosticar la línea de puertas de enlace PLX50.

Encuentre un distribuidor

Solicitar precios

Características y beneficios	Presupuesto	Descargas
------------------------------	-------------	-----------

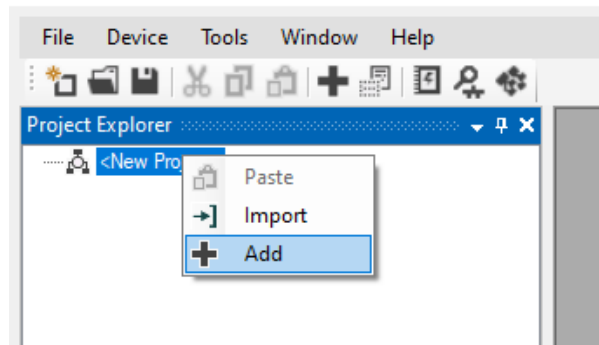
Descargas

Utilidad de configuración PLX50 1.034	72.69 MB	Última modificación 04/11/2024
---------------------------------------	----------	--------------------------------

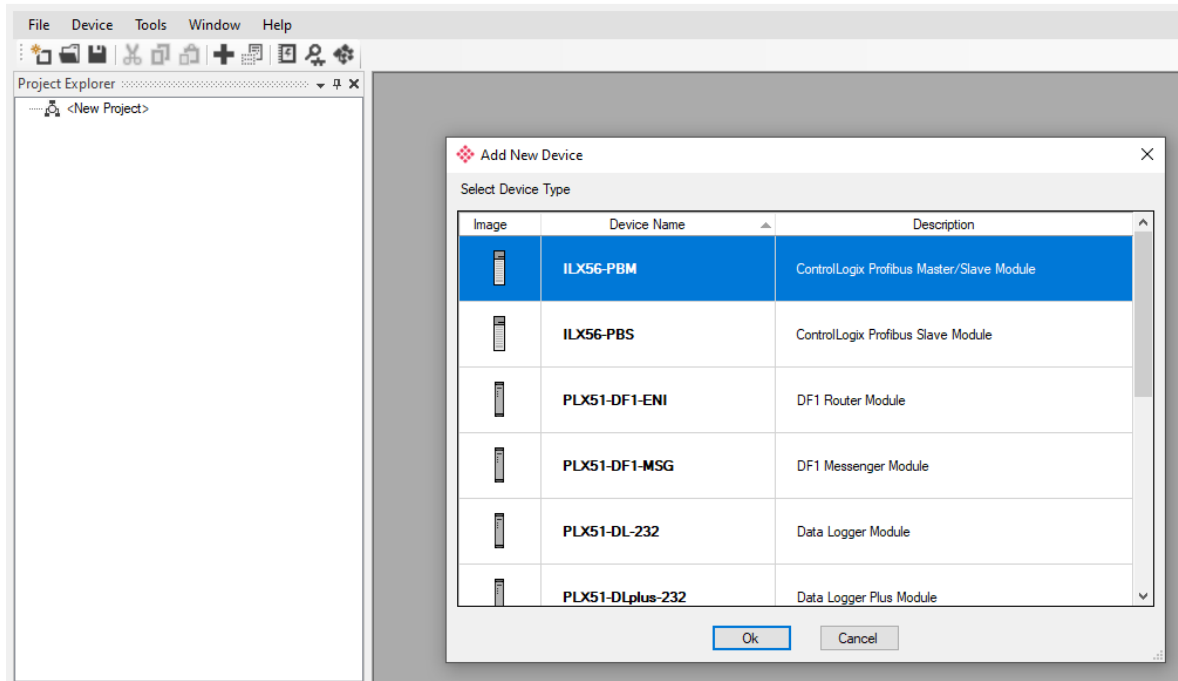
Envío de variable int desde allen brandley a siemens.
Configuración de ILX56-PBM

Añadir el modulo ILX56-PM

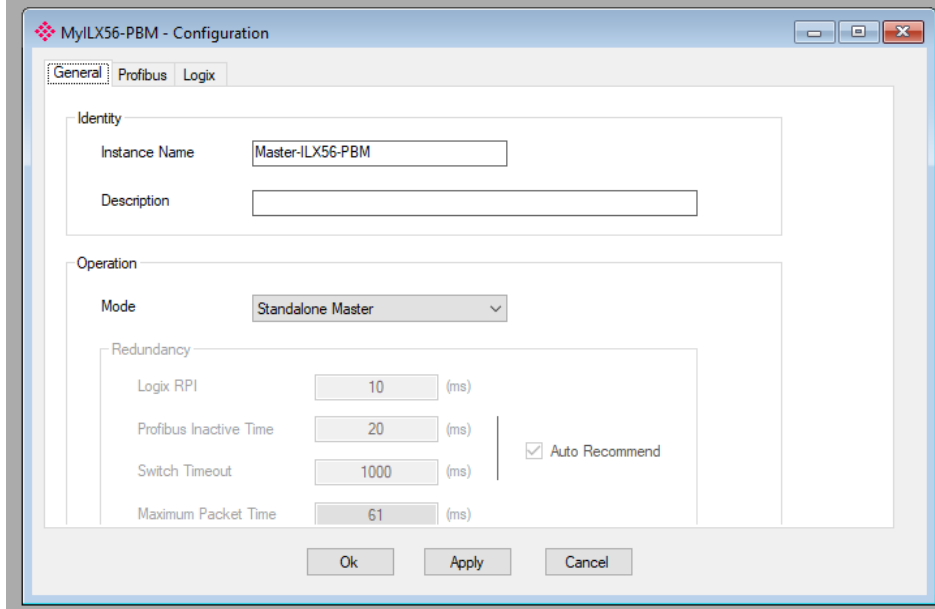
ProSoft PLX50 Configuration Utility - <New Project>*



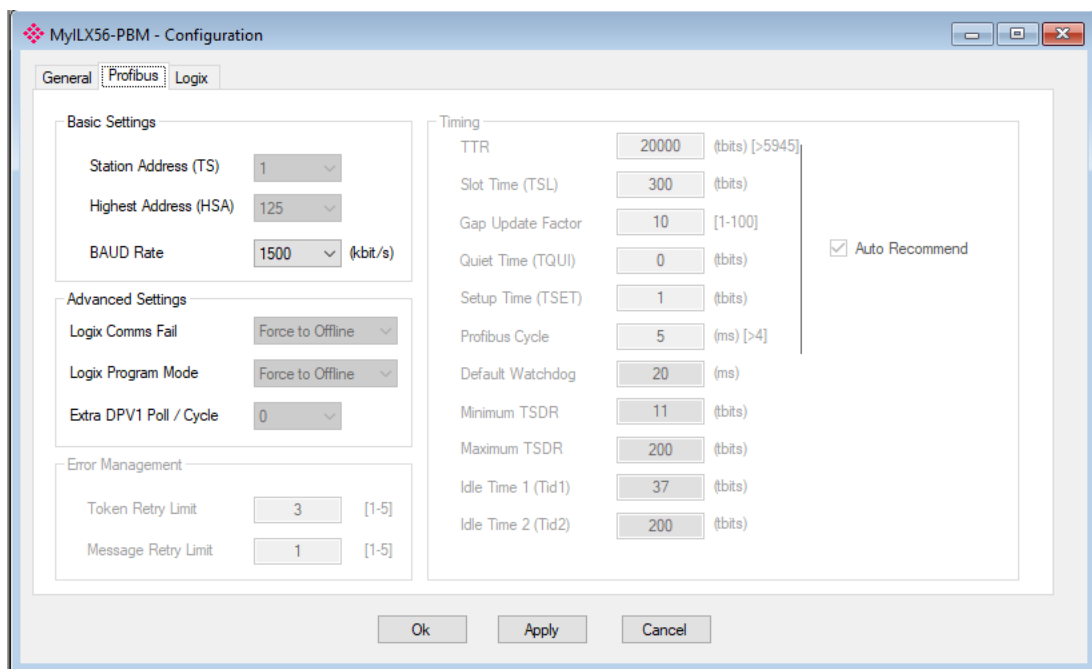
ProSoft PLX50 Configuration Utility - <New Project>*



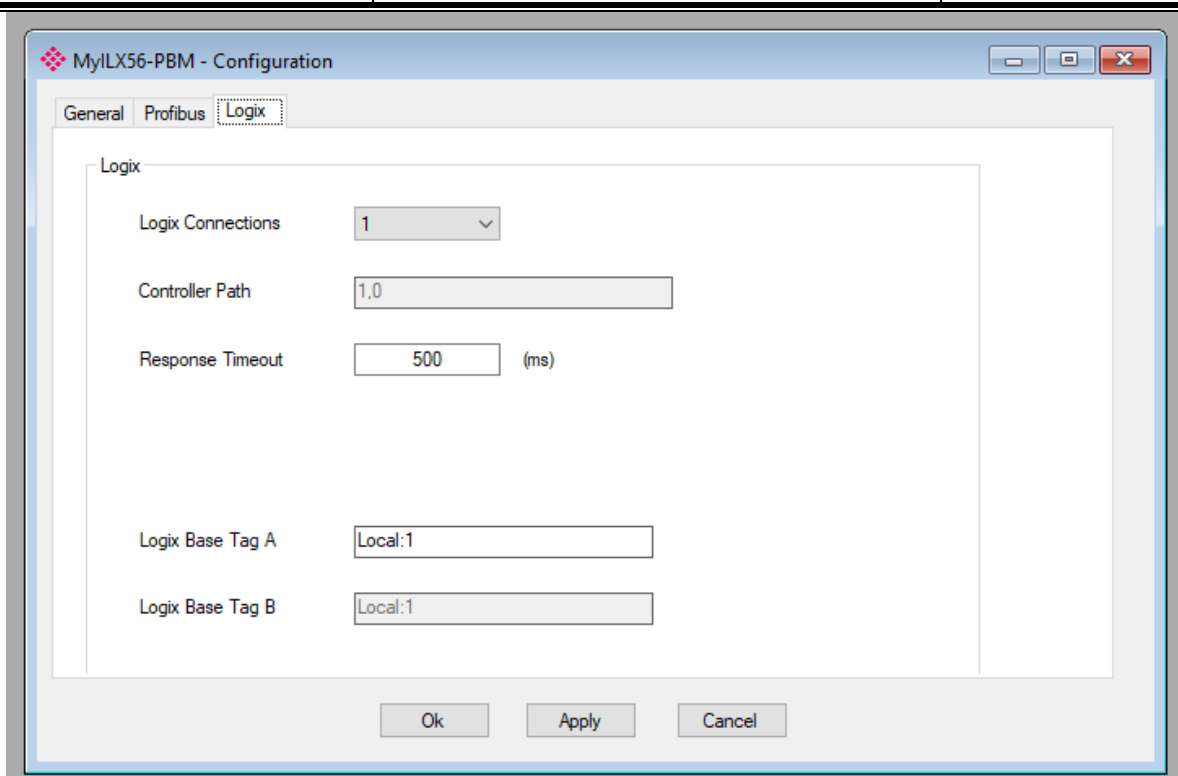
Realizar las configuraciones generales del modulo definiendolo como va a funcionar en este caso como maestro.



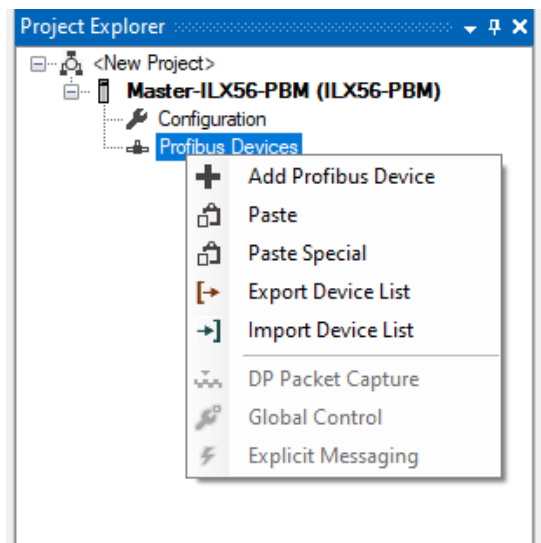
Configurar los baudios que coincidan con el modulo configurado para el plc s7-1200



En las configuración logix en "logix Base Tag A" configura el numero del local: "... " según el slot donde este instalado el mudulo.

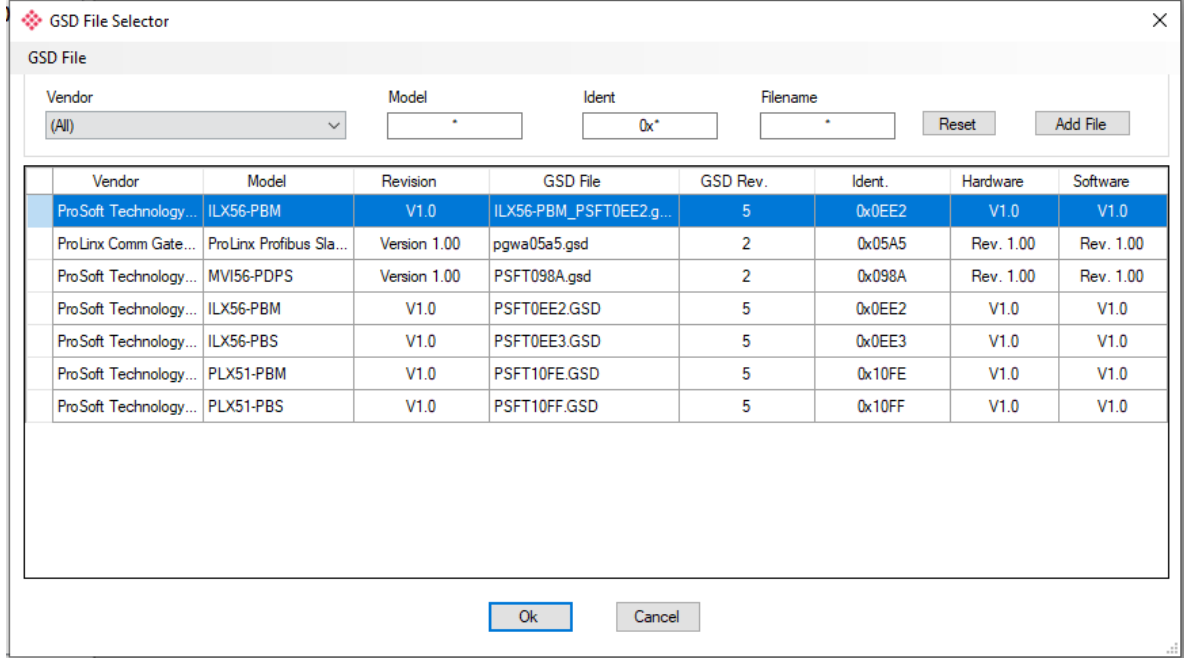


Añadir nuevo dispositivo

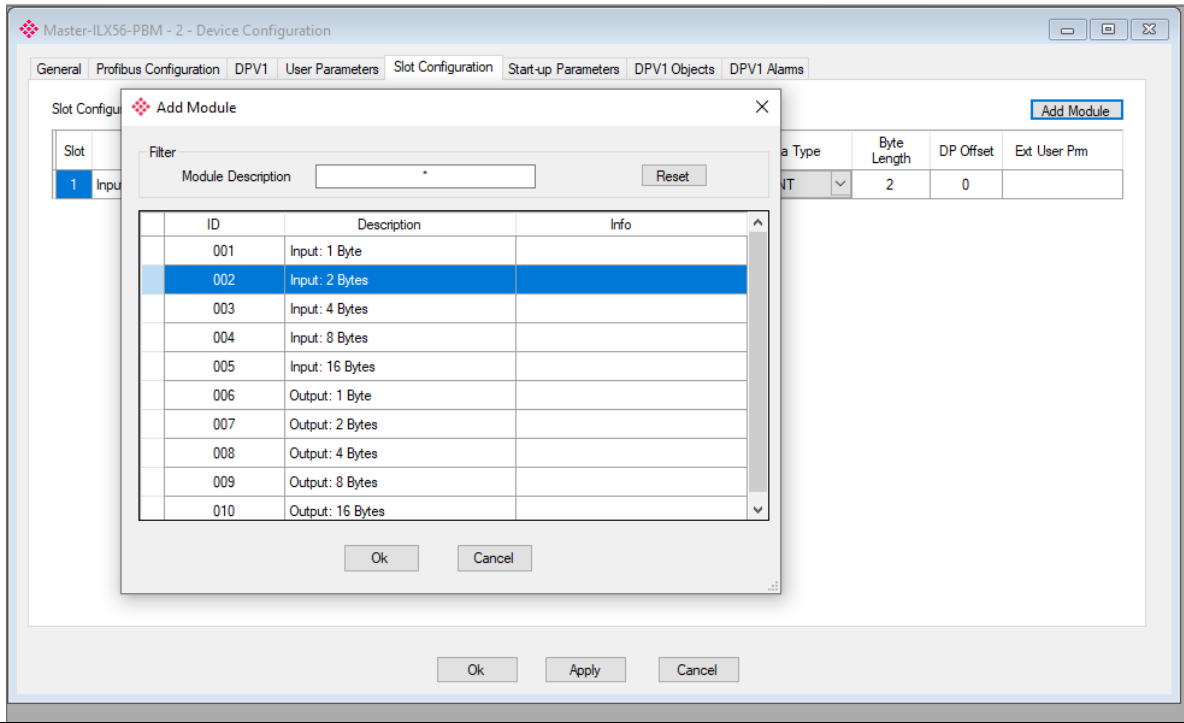


Añadir el dispositivo correspondiente al modulo ILX56-PM en este se configuraran los datos que se

desean enviar.



En la pestaña de “slot Configuration” se configura el tamaño de la entrada y la salida del módulo que leera y escribira los datos.



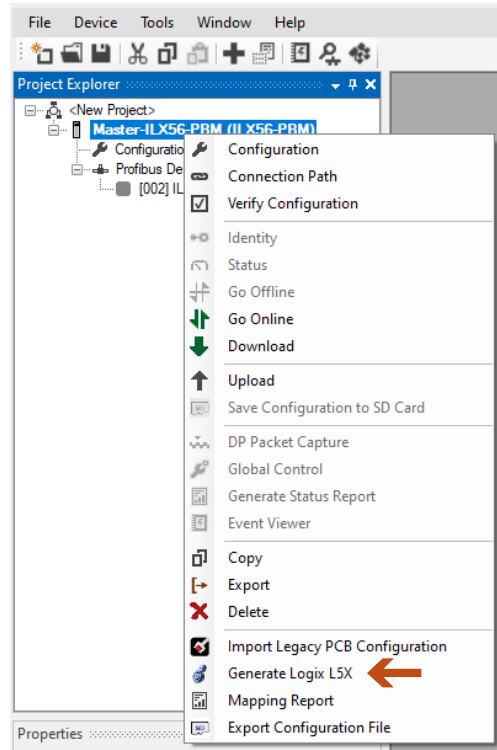
En el simbolo de “ + “ podemos añadir la cantidad de datos que se quieren enviar leer desde el mastro los cuales tendran un espacio de memoria de 2 bytes correspondientes a variables de tipo INT.

General Profibus Configuration DPV1 User Parameters Slot Configuration Start-up Parameters DPV1 Objects DPV1 Alarms


Slot Configuration Add Module

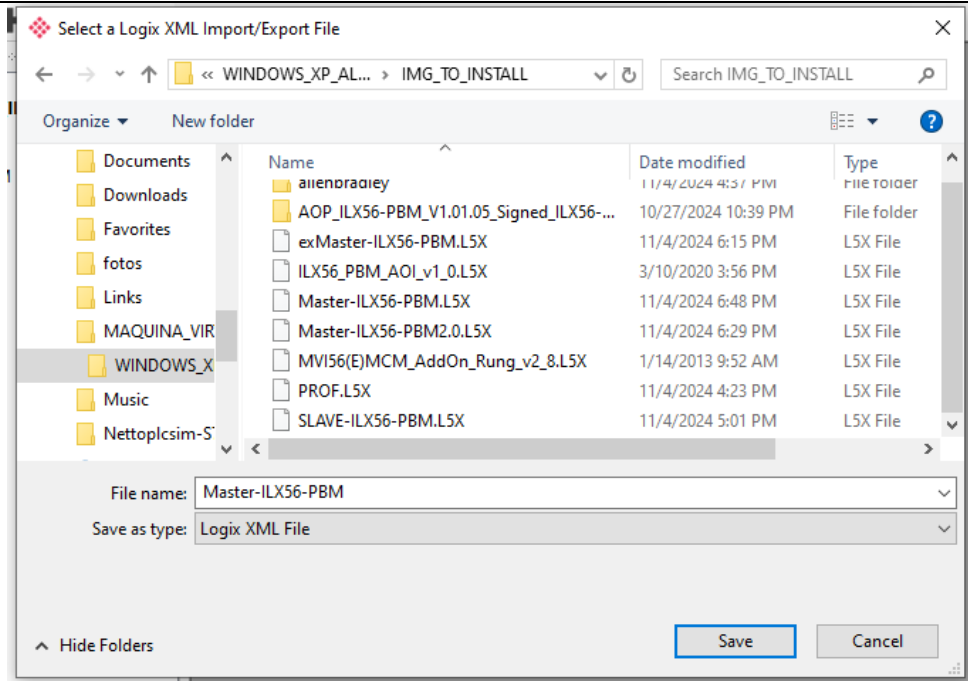
Slot	Description	Module			Data Point	Data Type	Byte Length	DP Offset	Ext User Pm
1	Input	02-Input: 2 Bytes		+	Input	SINT	2	0	(null)
	Input_1			+ X	None	None	0	0	
2	Output	07-Output: 2 Bytes		+	Output	SINT	2	0	(null)
	Output_2			+ X	None	None	0	0	

Generar un archivo .L5X el cual se exportara a la interfaz de rs logix studio 5000 con la configuraciones basicas del modulo

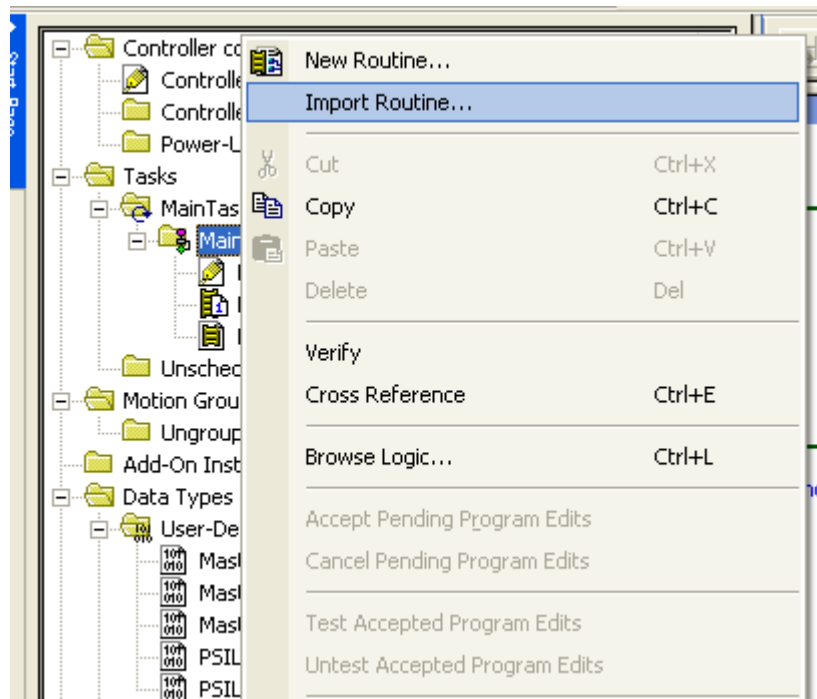


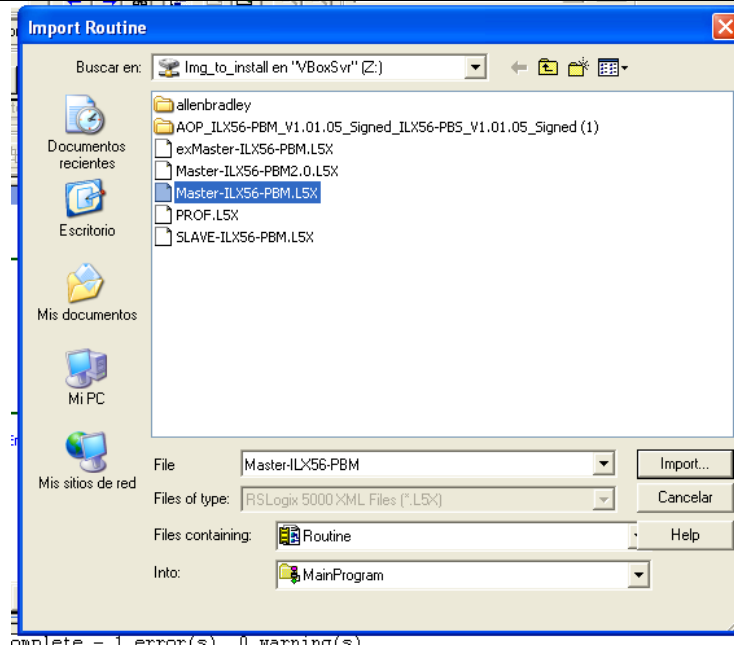
Guardar el archivo en una ubicación conocida.

 UNIVERSIDAD SANTO TOMÁS PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA BUCARAMANGA	GESTIÓN DEL CONOCIMIENTO, APRENDIZAJE Y DOCENCIA	VERSIÓN 01
	GUÍA DE LABORATORIO No. 7	CODIGO -
		Página 36 de 12



En la interfaz de Studio 5000 en un programa ya creado añadir la rutina creada en el programa prosoft technology.

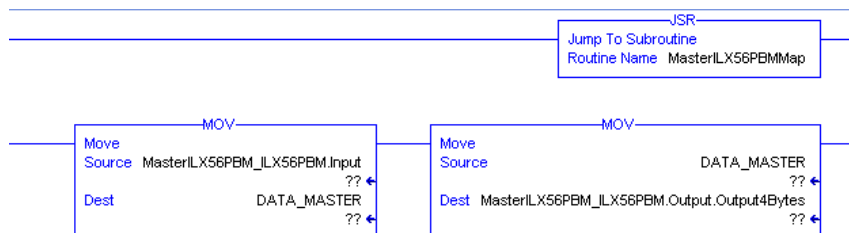




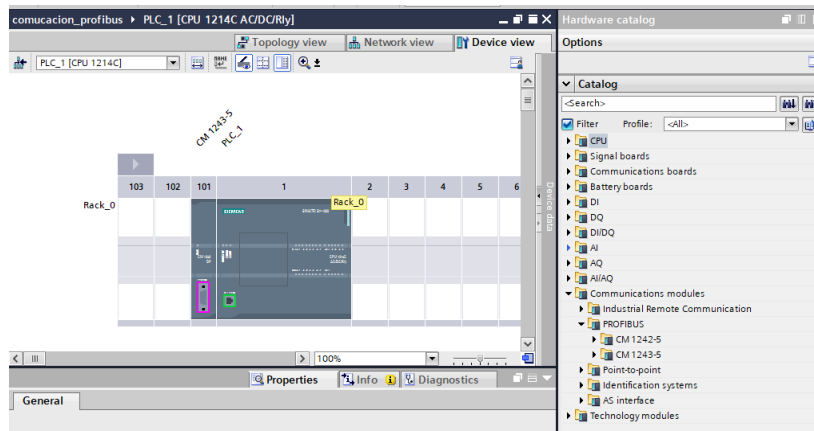
En “control Tags” se encontraran las funciones del modulo
Crear un arreglo de datos dentro de la memoria del controllogix “DATA_MASTER”

Name	Value	Force Mask	Style	Data Type
DATA_MASTER	(...)	(...)	Decimal	INT[2]
+ DATA_MASTER[0]	0		Decimal	INT
+ DATA_MASTER[1]	0		Decimal	INT
- MasterILX56PBM_ILX56PBM	(...)	(...)		MasterILX56PBM...
- MasterILX56PBM_ILX56PBM.Input	(...)	(...)		MasterILX56PBM...
+ MasterILX56PBM_ILX56PBM.Input.Status	(...)	(...)		PSILX56DPSlave...
- MasterILX56PBM_ILX56PBM.Input.Input4Bytes	(...)	(...)	Decimal	INT[2]
+ MasterILX56PBM_ILX56PBM.Input.Input4Bytes[0]	0		Decimal	INT
+ MasterILX56PBM_ILX56PBM.Input.Input4Bytes[1]	0		Decimal	INT
- MasterILX56PBM_ILX56PBM.Output	(...)	(...)		MasterILX56PBM...
+ MasterILX56PBM_ILX56PBM.Output.Control	(...)	(...)		PSILX56DPSlave...
- MasterILX56PBM_ILX56PBM.Output.Output4Bytes	(...)	(...)	Decimal	INT[2]
+ MasterILX56PBM_ILX56PBM.Output.Output4Bytes[0]	0		Decimal	INT
+ MasterILX56PBM_ILX56PBM.Output.Output4Bytes[1]	0		Decimal	INT

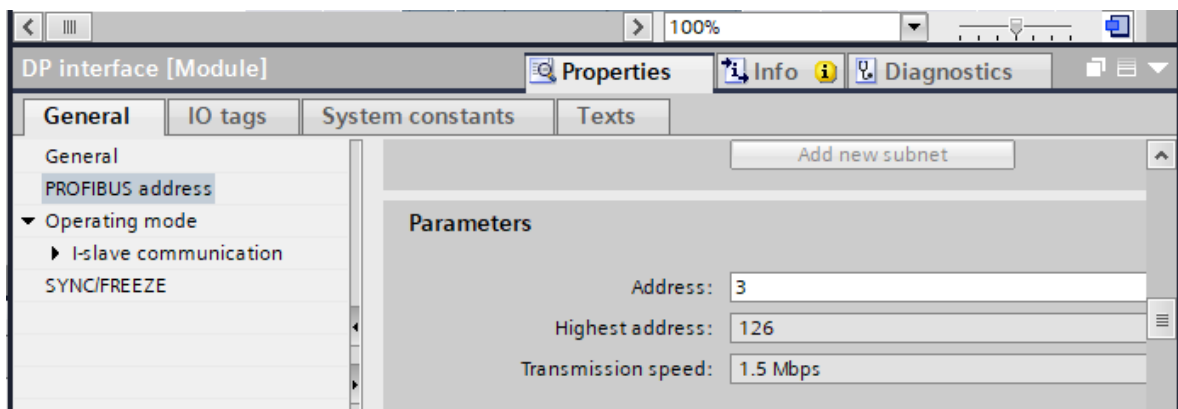
Con la función “MOVE” en el MainProgram se mueven los datos desde la base interna del modulo hasta la memoria del controllogix.



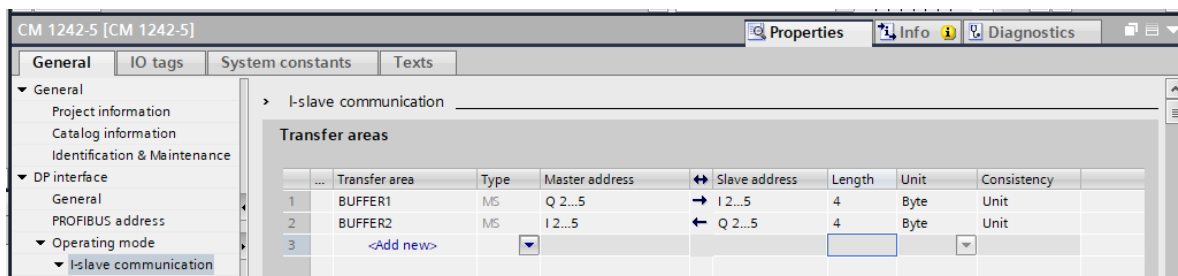
Configuración Tia portal.



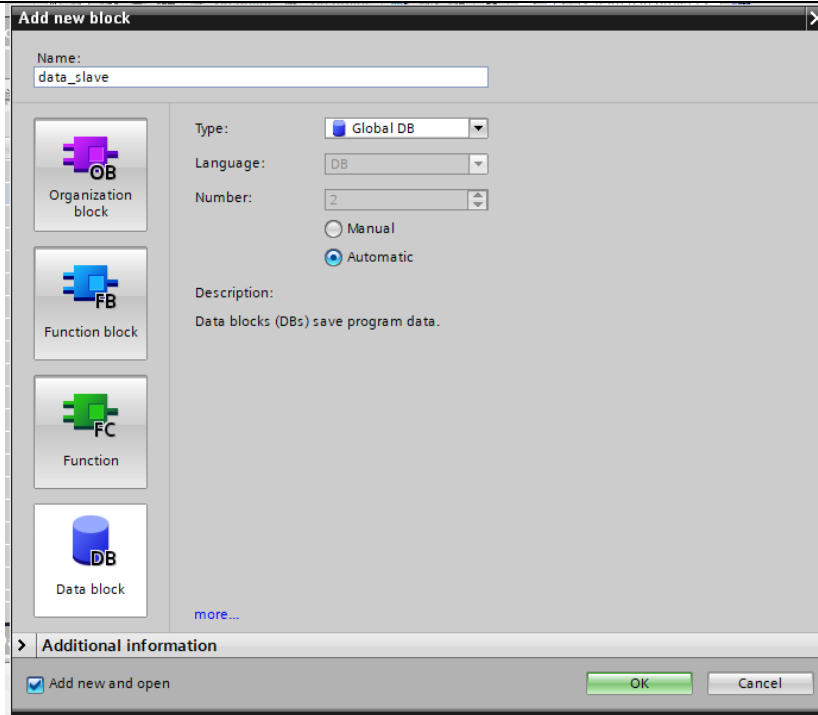
Configurar la velocidad en baudios igual del modulo ILX56-PBM



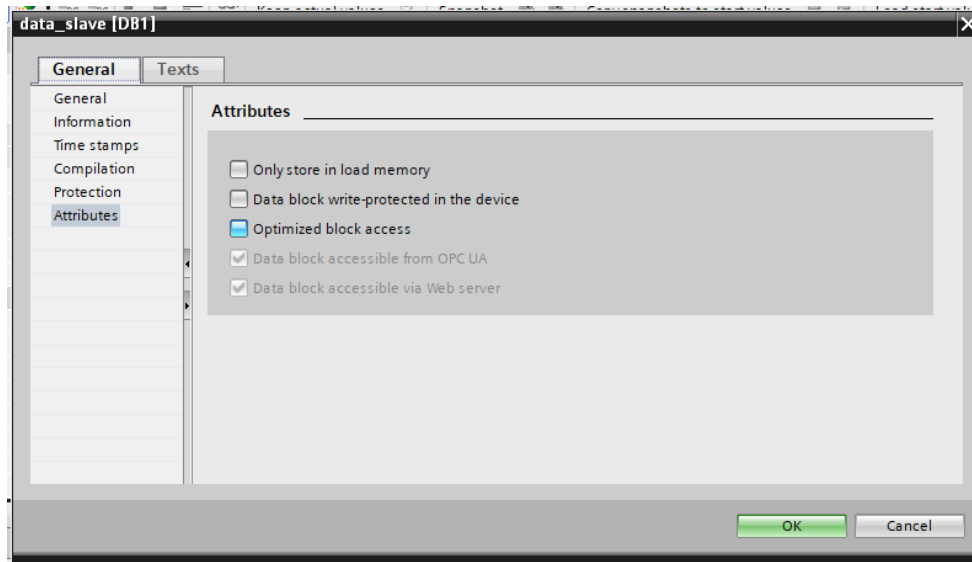
Configuración del tamaño y el flujo de datos de las entradas y salidas que se requieren en la comunicación



Crear un DB de datos



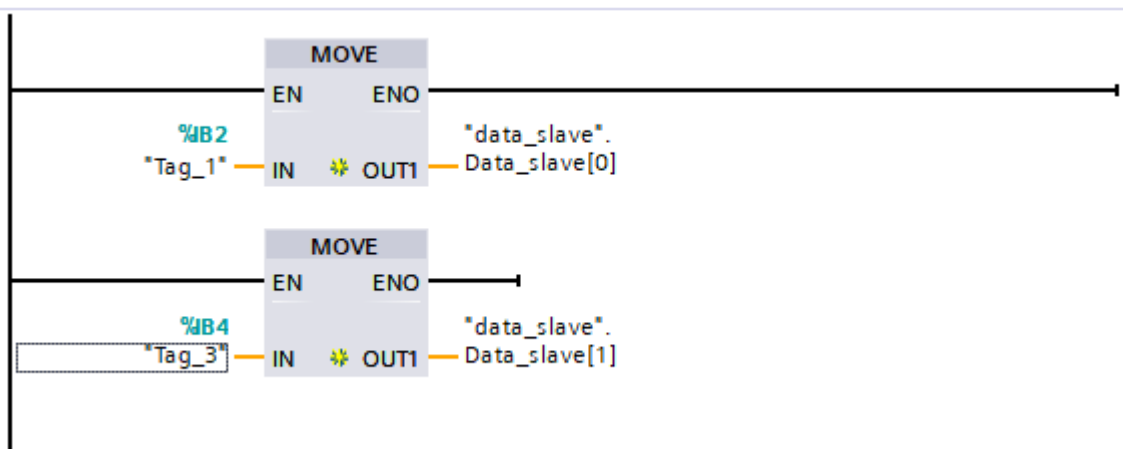
Desactivar el acceso optimizado al bloque.



Crear un arreglo de datos con variables del tipo INT

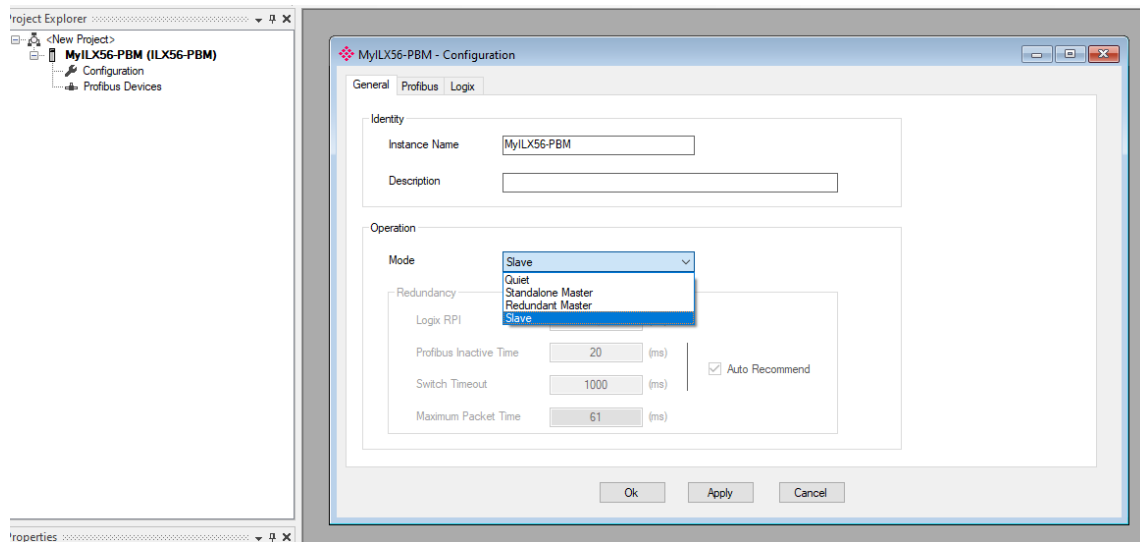
data_slave				
	Name	Data type	Offset	Start value
1	Static			
2	Data_slave	Array[0..1] o...	...	
3	Data_slave[0]	Int	...	0
4	Data_slave[1]	Int	...	0

Con la función move mover los datos que vienen desde el maestro profibus a la memoria interna del plc s7-1200

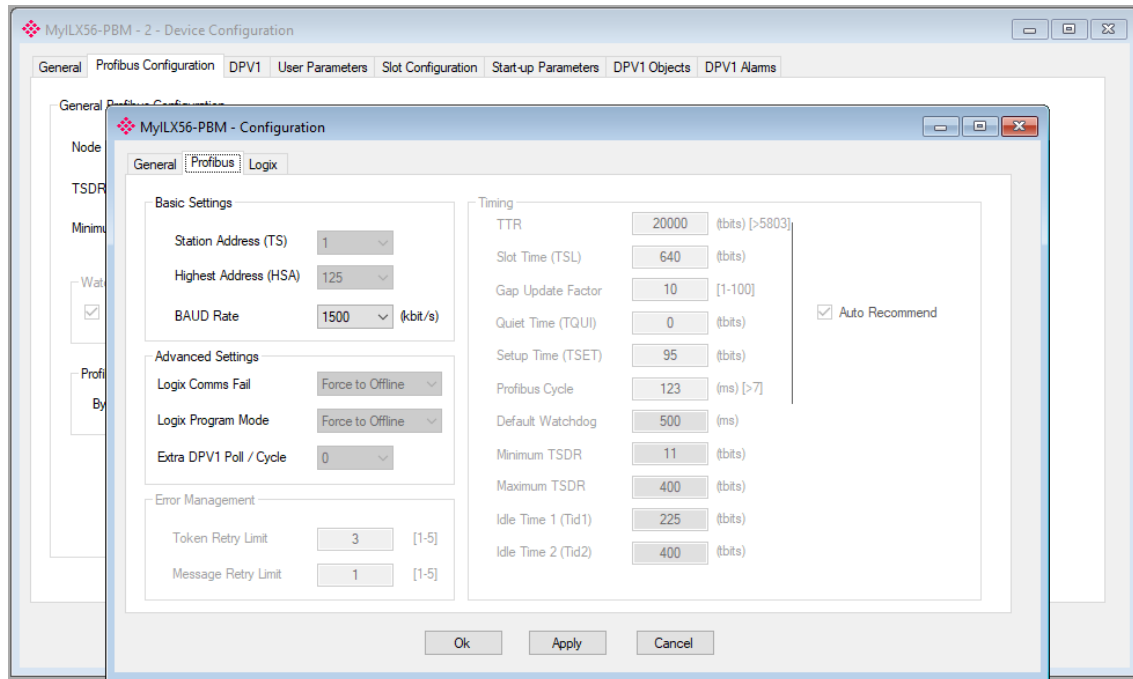


Configuración de envío de bits de desde S7-1200 al Módulo ILX56-PBM

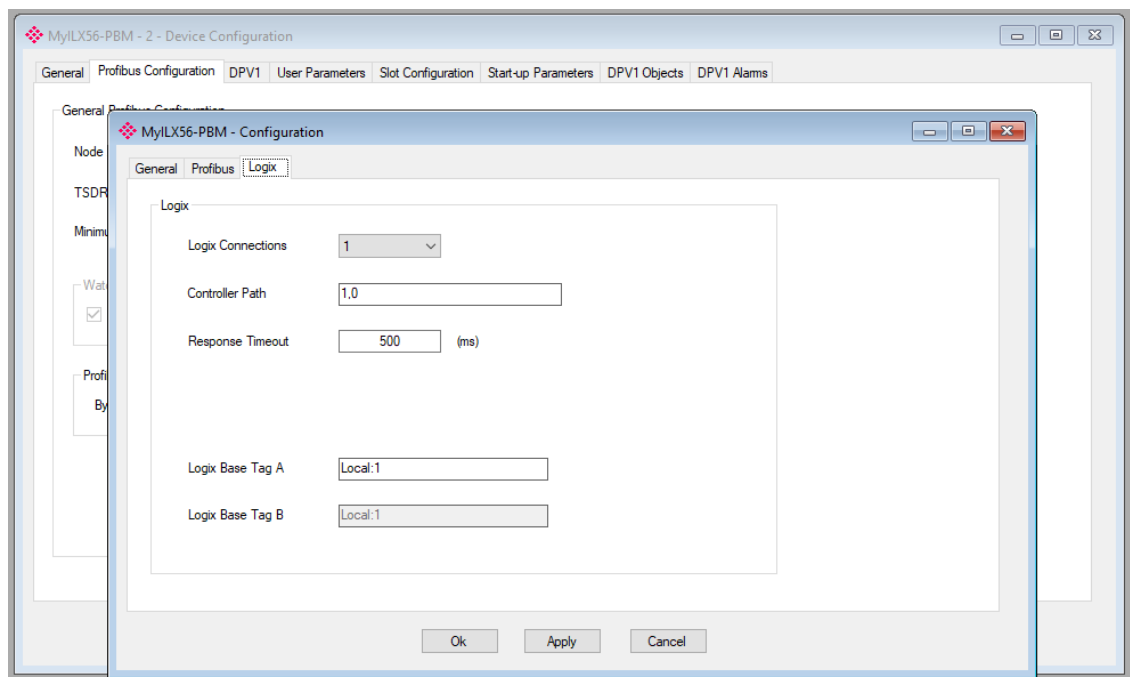
Añadir un nuevo dispositivo en la interfaz de prosoft technology y definirlo como esclavo.



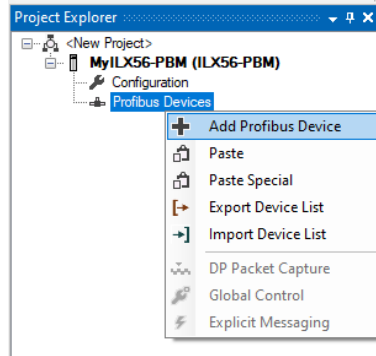
Configuración de los baudios igual que los del modulo profibus de siemens.



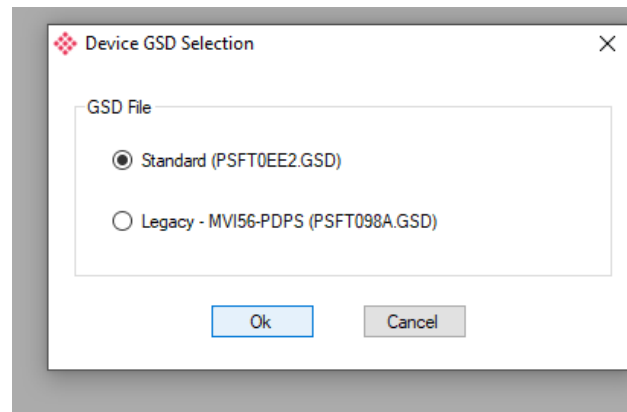
En las configuración logix en "logix Base Tag A" configura el numero del local: "... " según el slot donde este instalado el mudulo.



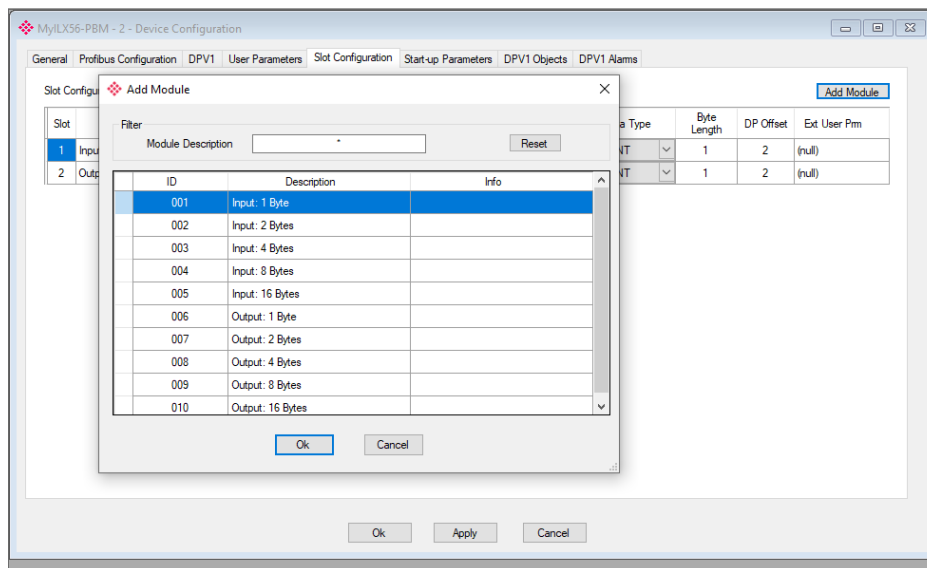
Añadir nuevo dispositivo



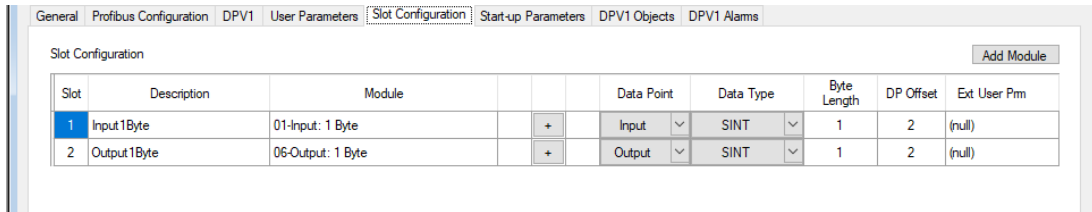
En esta venta emergente seleccionaremos el estandar el cual corresponde al ILX56-PM como esclavo.



Configuracion de las entradas y salidas que utilizara el modulo para comunicarse



Como quedan las configuradas las entradas y salidas del modulo dar click en apply y ok

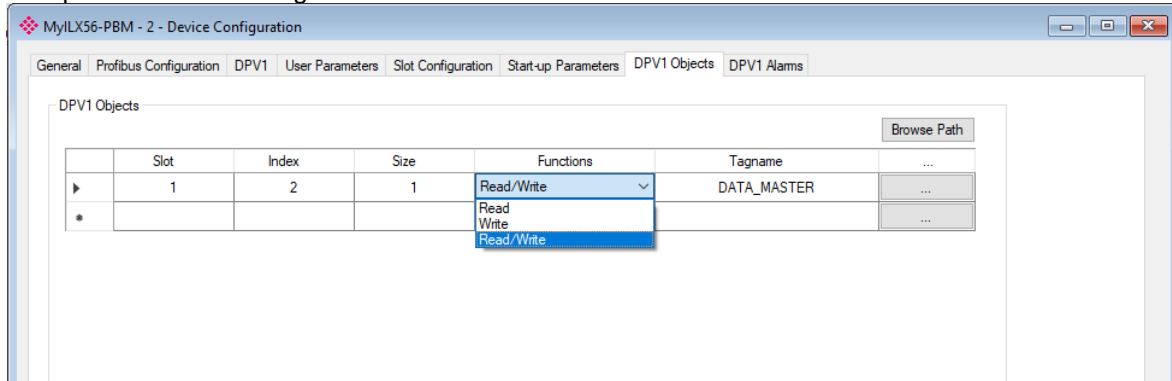


Configuracion del area de transferencia la cual se configura desde el modulo esclavo, en las siguientes imaganes se muestra la comparacion del area de transeferencia de una comunicacion profibus configurada en siemens y una para el modulo ILX56-PBM

- El numero del slot corresponde a la ubicación física del modulo.
- Index corresponde a la direccion donde iniciara la memoria para el envio de datos
- Size corresponde al tamaño que tendran los datos que se enviaran para este caso 1 byte
- Functions en como se manejara la comunicacion maestro-esclavo (write), esclavo maestro(Read) o Bidireccional.

Configuracion del mismo area de transferencia en los modulos para siemens y allen bradley.


Comparacion de la configuracion en el modulo ILX56-PBM

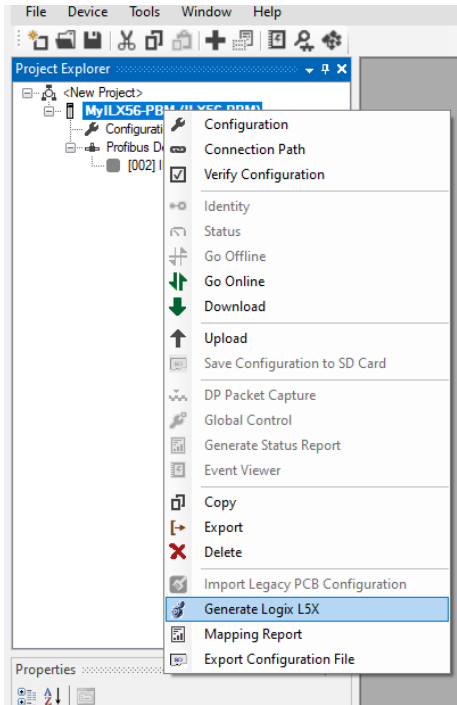


Comparacion de la configuracion en el modulo CM 1242-5

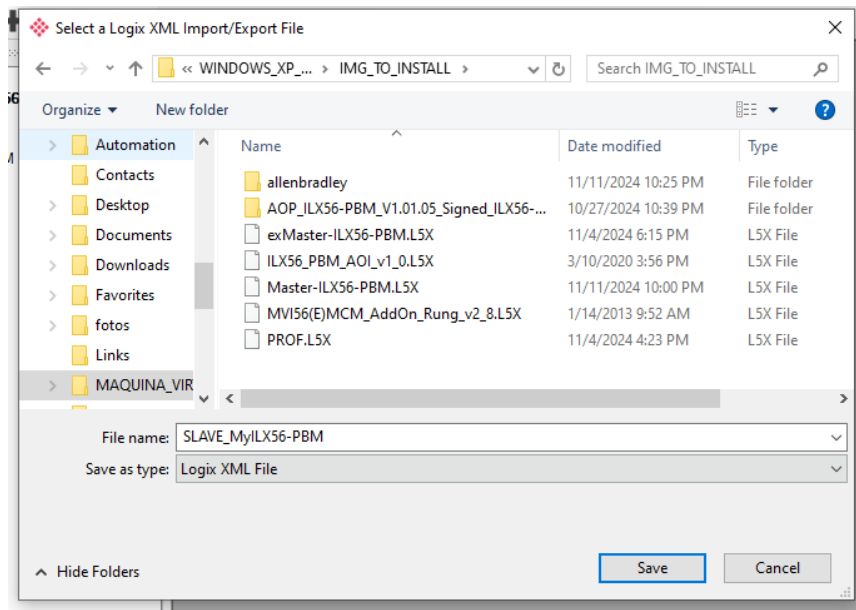
Transfer areas								
...	Transfer area	Type	Master address	↔ Slave address	Length	Unit	Consistency	
1	BUFFER1	MS	Q 2	→ I 2	1	Byte	Unit	
2	BUFFER2	MS	I 2	← Q 2	1	Byte	Unit	
3	<Add new>							

Generar un archivo .L5X el cual se exportara a la interfaz de rs logix studio 5000 con la configuraciones basicas del modulo

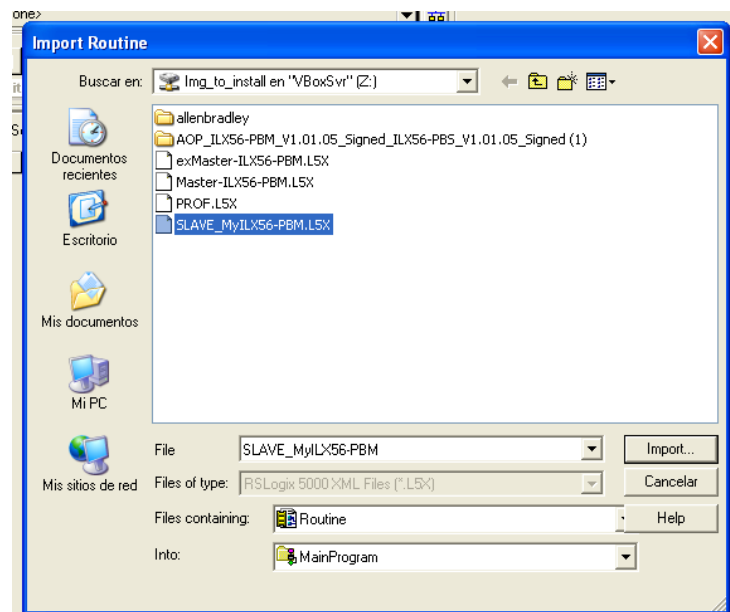
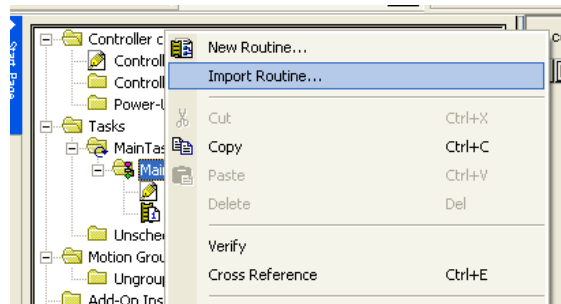
 UNIVERSIDAD SANTO TOMÁS PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA BUCARAMANGA	GESTIÓN DEL CONOCIMIENTO, APRENDIZAJE Y DOCENCIA	VERSIÓN 01
	GUÍA DE LABORATORIO No. 7	CODIGO -
		Página 44 de 12



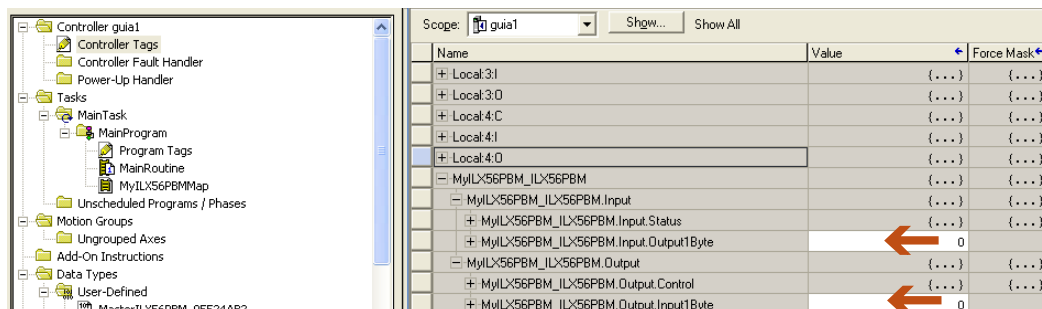
Guardar el archivo en una ubicación conocida.



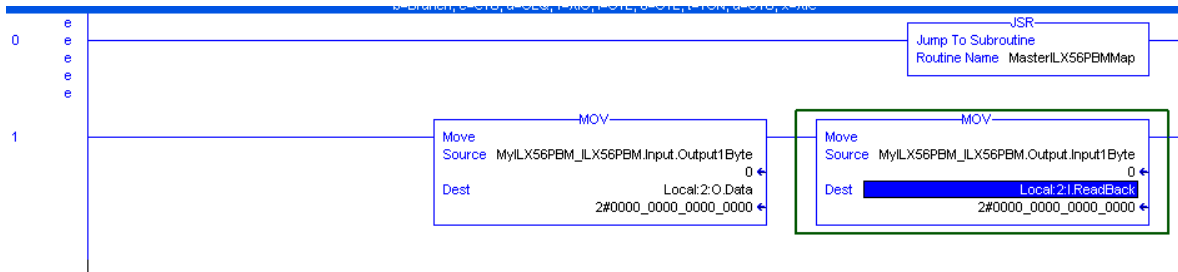
En un programa ya creado en la interfaz de estudio 5000 Designer importar la rutina creada con las configuraciones pre establecidas del modulo.



En controller tags encontraremos las variables del modulo ya creadas y pre definidas en el software de profot technology. Tambien las variable de entrada y salida donde se reciben los datos enviados por el mestros profibus (S7-1200)

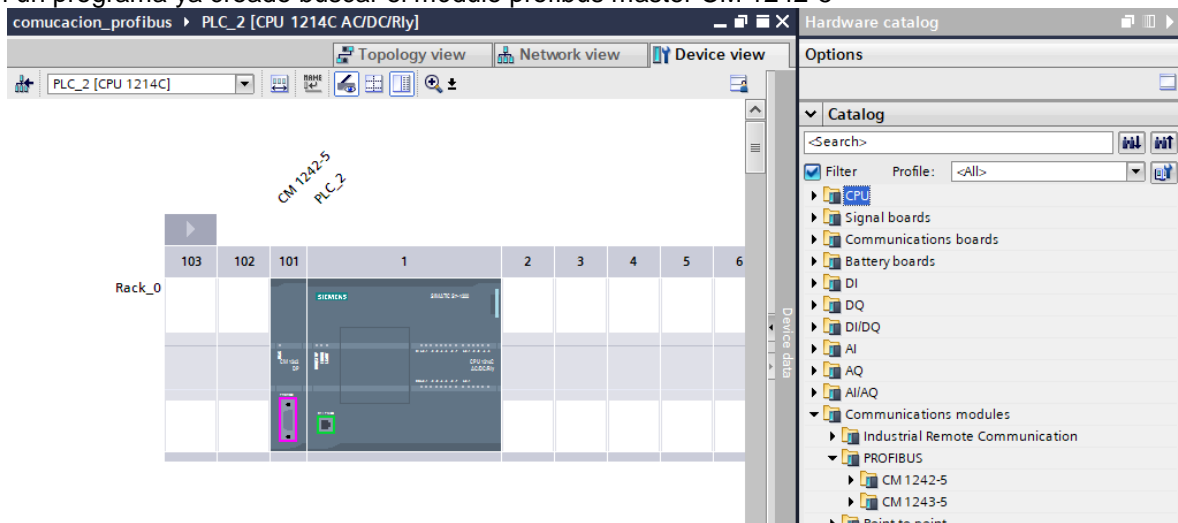


Con la función “MOVE” en el MainProgram se mueven los datos desde la base interna del modulo hasta la memoria del controllogix.

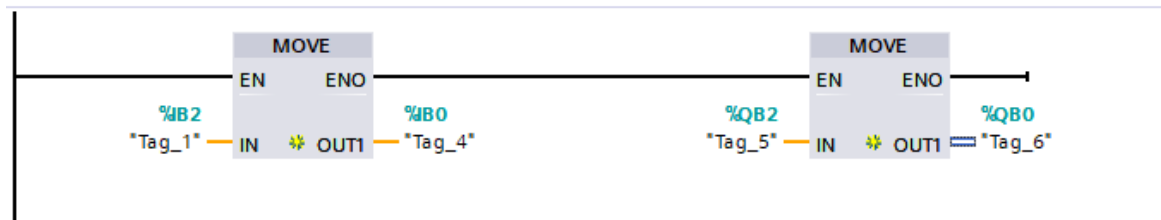



Configuración en la interfaz de Tia portal

En un programa ya creado buscar el módulo profibus master CM 1242-5



Y en el Main se realizara el envio de bits por medio de las primera 8 salidas digitales del plc y se leeran los datos del esclavo con las primeras 8 entradas digitales. Estas direccion de memoria se establecen en el area de transferencia configurado anteriormente.



 UNIVERSIDAD SANTO TOMÁS PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA BUCARAMANGA	GESTIÓN DEL CONOCIMIENTO, APRENDIZAJE Y DOCENCIA	VERSIÓN 01
	GUÍA DE LABORATORIO No. 7	CODIGO -
		Página 47 de 12

--

8. INFORME QUE DEBE PRESENTAR EL ESTUDIANTE

9. BIBLIOGRAFÍA Y OTROS RECURSOS
<p>[1] Contemporary Controls. [En línea]. Disponible: https://www.ccontrols.com/pdf/Extv9n4.pdf</p> <p>[2] “Modbus: Qué es y cómo funciona Comunicaciones Industriales”. aula21 Formación para la Industria. [En línea]. Disponible: https://www.cursosaula21.com/modbus-que-es-y-como-funciona/#:~:text=Modbus%20es%20el%20protocolo%20de,para%20conectar%20dispositivos%20electrónicos%20automatizados.</p> <p>[3] “PROFIBUS: Qué es y cómo funciona - Cursos Centro de Entrenamiento Internacional de PROFIBUS & PROFINET”. Cursos Centro de Entrenamiento Internacional de PROFIBUS & PROFINET. [En línea]. Disponible: https://profibus.com.ar/profibus_que_es_y_como_funciona/#:~:text=PROFIBUS%20es%20un%20estándar%20industrial,la%20seguridad%20de%20los%20datos.</p> <p>[4] “LA EMPRESA / Inicio - ProSoft Technology, Inc.” ProSoft Technology Inc — IIOT connectivity starts with ProSoft. Accedido el 20 de octubre de 2024. [En línea]. Disponible: https://mx.prosoft-technology.com/LA-EMPRESA</p>