Manual técnico

ELION, S.A. Farell, 5 - 08014 Barcelona Tel. 932 982 000 elion@elion.es - www.elion.es



Modbus: RTU

Características Principales

Modbus RTU es un protocolo abierto, en serie (RS-232 o RS-485) derivado de la arquitectura Master / Slave. Es un protocolo ampliamente aceptado debido a su facilidad de uso y fiabilidad. Modbus RTU se usa ampliamente en Building Management Systems (BMS) y Industrial Automation Systems (IAS). Esta amplia aceptación se debe en gran parte a la facilidad de uso de MODBUS RTU.

Los mensajes MODBUS RTU son un CRC simple de 16 bits (suma de comprobación redundante cíclica). La simplicidad de estos mensajes es garantizar la fiabilidad. Debido a esta simplicidad, la estructura de registro MODBUS RTU de 16 bits se puede usar para empaquetar puntos flotantes, tablas, texto ASCII, colas y otros datos no relacionados.

Modbus: TCP

Características Principales

MODBUS TCP / IP es una variante de la familia MODBUS de protocolos de comunicación sencillos, de proveedor neutral, destinados a la supervisión y control de equipos de automatización. Específicamente, cubre el uso de la mensajería MODBUS en un entorno 'Intranet' o 'Internet' utilizando los protocolos TCP / IP. El uso más común de los protocolos en este momento es para la conexión Ethernet de PLC, módulos de E / S y 'puertas de enlace' a otros buses de campo simples o redes de E / S.

El protocolo MODBUS TCP / IP se está publicando como un estándar de automatización ('de facto'). Como MODBUS ya es ampliamente conocido, debería haber poca información en este documento que no pudiera obtenerse en otra parte. Sin embargo, se ha intentado aclarar qué funciones dentro de MODBUS tienen valor para la interoperabilidad de los equipos de automatización en general, y qué partes son "equipaje" del uso alternativo de MODBUS como protocolo de programación para PLC.

TBEN-PLC Package

Para poder utilizar el TBEN-PLC de Turck en Codesys como dispositivo programable, deberemos instalar el Package de este el cual podemos descargar desde la web de Turck.

Enlace de descarga

Dependiendo de la versión de Codesys que utilicemos, deberemos descargar uno u otro.

| Configuration File | ^ |
|---|----------|
| CODESYS Package TBEN-Lx-PLC Version v1.0.3.0 (for CODESYS v3.5.8.10 or higher) | 15010 KB |
| EDS files | 60 KB |
| SSDML files | 11 KB |
| CODESYS Package TBEN-Lx-PLC Version v1.1.4.0 (for CODESYS v3.5.12.10 or higher) | 35840 KB |

Una vez descargado el archivo, haremos doble clic sobre él y empezará la instalación.



Seleccionaremos la opción de instalación completa y esperaremos a que finalice, cuando haya terminado, deberá verse reflejado de la siguiente forma:

| | Ordenar por: Nombre | ~ | Instalando |] |
|-------------------------|---|--|--|---|
| ersión Fecha de instala | ación Información de actualización | Informació | Desinstalando | |
| 3.1.0 21/12/2017 | | No se requie | Detalles | |
| 0.1.0 21/12/2017 | | No se requie | Detailearri | |
| 0.3.0 05/04/2018 | | No se requie | | |
| | rsión Fecha de instal 3.1.0 21/12/2017 0.1.0 21/12/2017 0.3.0 05/04/2018 | Ordenar por: Nombre rsión Fecha de instalación Información de actualización 3.1.0 21/12/2017 0.1.0 21/12/2017 0.3.0 05/04/2018 | Ordenar por: Nombre rsión Fecha de instalación Información de actualización Informació 3.1.0 21/12/2017 No se requie 0.1.0 21/12/2017 No se requie 0.3.0 05/04/2018 No se requie | Ordenar por: Nombre Instalando rsión Fecha de instalación Información de actualización Informació 3.1.0 21/12/2017 No se requie Detalles 0.1.0 21/12/2017 No se requie Detalles 0.3.0 05/04/2018 No se requie |

TBEN-PLC: Maestro Modbus RTU

En este primer ejemplo, comunicaremos una <u>BL20-E-GW-RS-MB/ET</u> con el TBEN-L5-PLC-10, para que este sea el maestro Modbus RTU de la comunicación.

Recibiremos la información de la tarjeta de 16 entradas (<u>BL20-E-16DI-24VDC-P</u>) que conectaremos a la BL20 para procesarlas y verlas en Codesys.

Una vez instalado este package, podremos abrir Codesys para crear el proyecto con el TBEN-PLC como dispositivo programable.

| Proyecto p | redeterminado | | х |
|------------|---|---|--------|
| | Está usted pro guiará para ino - Un dispositio - Un módulo o adelante - Una tarea cío - Una referenco | eparado para crear un nuevo proyecto predeterminado. Este asistente le corporar al proyecto los siguientes objetos: /o programable como se describe posteriormente de programa PLC_PRG en el lenguaje de programación que se indica má: clica que llama al PLC_PRG ia a la versión más reciente de biblioteca estándar | 5 |
| | Dispositivo: | TBEN-Lx-PLC-1x (Turck) | ~ |
| | PLC_PRG en: | Diagrama de contactos (LD) | \sim |
| | | Aceptar Cancelar | · |

Y en la vista de Dispositivos, veremos como en el PCL tenemos los dos puertos serie (COMO y COM1)



Será dentro de estos COM donde deberemos añadir los dispositivos que conectemos por Modbus RTU (RS232 o RS485) para configurar sus E/S.

En nuestro ejemplo, dentro del COM1, agregaremos la estructura de Maestro/Esclavo de Modbus RTU.



Ahora, con la BL20-RTU configurada por <u>Pactware</u> (Software gratuito de Turck) o dip-switch, haremos doble clic sobre COM1 en Codesys para configurar los parámetros de este puerto del TBEN.

Es de vital importancia que los parámetros de la BL20 coincidan con los del TBEN-PLC.

| <u>ኔ</u> ርዕM1 🗙 | | | | | | |
|---------------------|--------------------|----------------------|------------|----------------------|--------|--|
| COM1 Asignación E/S | Parámetro | Tipo | Valor | Valor predeterminado | Unidad | Descripción |
| | 🖉 🖗 Baudrate | Enumeration of UDINT | 9600 Bit/s | 9600 Bit/s | | Baudrate of the serial port. |
| COM1 Configuración | 🖤 🖗 DataBits | Enumeration of USINT | 8 bit | 8 bit | | Number of data bits |
| Estado | StopBits | Enumeration of USINT | 1 bit | 1 bit | | Number of stop bits |
| Estado | Termination active | Enumeration of USINT | no | no | | Termination of the serial port (only in RS485 mode) |
| | Biasing active | Enumeration of USINT | no | no | | Biasing of the serial port (only in RS485 mode) |
| | Power supply VAUX1 | Enumeration of USINT | off | off | | Power supply VAUX1 |
| | 🔷 🖗 Swap A/B Line | Enumeration of USINT | no | no | | Swap A and B lines of the serial port (only in RS485 mode) |
| | Operation mode | Enumeration of USINT | RS232 | RS232 | | Medium of the serial port. |
| | 🖳 🖗 Parity | Enumeration of USINT | none | none | | Parity for messages on the serial port. |

Para estar seguros de la configuración de la BL20, podemos utilizar el <u>manual de esta</u>, o revisar la configuración con Pactware mediante el USB de la BL20.

| Name | Value | |
|---|------------------------|--|
| 🛱 DTM (offline) | | |
| 🖃 Device identification | | |
| Device short name | | |
| Comment | | |
| 🖃 Settings | | |
| Force Mode activation | with confirmation | |
| Write reference list | yes | |
| 🖳 🗐 Notation | | |
| Channel name | | |
| 📮 Device | | |
| 😑 special device properties | | |
| Bus address | | |
| Software revision | | |
| Parameters | | |
| Address (DIP switch) | 0 (I/O-ASSISTANT mode) | |
| Bit rate (DIP switch) | 9.6 kbit/s | |
| Interface (DIP switch) | RS485 | |
| Terminating resistor (DIP switch) | off | |
| Address (I/O-ASSISTANT) | 1 | |
| Bit rate (I/O-ASSISTANT) | 9.6 kbit/s | |
| Electrical interface (I/O-ASSISTANT) | RS485 | |
| Terminating resistor (I/O-ASSISTANT) | off | |
| Parity | even | |
| Mode | RTU | |
| Modbus watchdog time | 0 | |

Para continuar con la configuración en Codesys, deberemos hacer doble clic sobre el dispositivo Modbus Esclavo que tenemos al final de los dispositivos de COM1, que es donde conectaremos la BL20-E-GW-RS-MB/ET.

| COM1 Modbus_Esclav | vo_puerto_COM 🗙 | | |
|--|---------------------------------------|------|---------|
| Configuración de esclavo de Modbus | Modbus-RTU/ASCII | | MODBIIS |
| Canal esclavo Modbus | Dirección de esclavo [1247] | 1 | MUDDUJ |
| Init esclavo Modbus | Tiempo de espera en la respuesta [ms] | 1000 | |
| ModbusGenericSerialSlave Asignación E/S | | | |
| Estado | | | |
| Información | | | |
| | | | |

Una vez con la dirección asignada, falta declarar las variables E/S que comunicaremos con la BL20, esta información, deberemos extraerla desde el <u>Pactware</u>.

Configurando la BL20 con todas las tarjetas que acoplemos a la cabecera, deberemos extraer el "station report" para tener conocimiento de la cantidad de información que asociaremos en Codesys.



Para realizar dicha configuración, lo podemos hacer manualmente, añadiendo los módulos que acoplemos



O mediante la función "Topology Scan", la cual nos exigirá tener conectada la BL20 mediante el USB que podemos ver fácilmente en la cabecera.



Para extraer el "station report", el cual nos dará información de los Bits, Bytes y Words que tendrá nuestra cabecera, junto con las tarjetas E/S que añadamos, haremos clic derecho sobre la cabecera RTU.



En la ventana que se abra, podremos seleccionar las opciones que queramos, aunque deberemos fijarnos en "Fieldbus report"

| Report Labels Ex-/Import | | |
|---|-------------------------------|---------------------------------|
| All chapters | | |
| Station overview | Station image | 🗹 Fieldbus report, if available |
| Station parameters | Station dimension | Device documentation (only PDF) |
| including all parameters of | f BL remote (Devicenet) nodes | |
| Station article list with prices pricelist not imported - | All gateways from bus | |
| | | |

Este Fieldbus Report, nos dirá qué cantidad de Entradas y Salidas (físicas, diagnóstico, etc.) tendrá cada componente del Slot de la BL20, tanto cabecera como tarjetas E/S.

Al tener conectada únicamente una tarjeta <u>BL20-E-16DI-24VDC-P</u>, obtenemos la siguiente información:

Station address: 1

| Adr./Slot | Name | TAG | Data Size In | Data Size Out |
|------------------|-------------------------------------|----------------------------|--------------|---------------|
| 0* | BL20-E-GW-RS-MB/ET | 1/BL20-E-GW-RS- MB/ET | 16 bit | 0 bit |
| 1 | BL20-E-16DI-24VDC-P | 01/BL20-E-16DI- 24VDC-P | 16 bit | 0 bit |
| | Local I/O data incl. status/control | | 2 Words | 0 Words |
| | Summarized diagnostics | | 1 Word | 0 Words |
| Total size for i | n/out data rounded on full words | | 3 Words | 0 Words |

*For detailed information about status/control word see online help

1.2. I/O map for input data

| Regis | ter | | Bit position | | | | | | | | | | | | | | |
|----------|------|-------|--------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| Hex | Dec | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0x0000 | 0000 | 01.15 | 01.14 | 01.13 | 01.12 | 01.11 | 01.10 | 01.09 | 01.08 | 01.07 | 01.06 | 01.05 | 01.04 | 01.03 | 01.02 | 01.01 | 01.00 |
| *0x0001 | 0001 | GW.15 | GW.14 | GW.13 | GW.12 | GW.11 | GW.10 | GW.09 | GW.08 | GW.07 | GW.06 | GW.05 | GW.04 | GW.03 | GW.02 | GW.01 | GW.00 |
| **0x0002 | 0002 | - | - | - | - | - | - | - | - | - | - | - | - | - | - | - | M00 |

Description: 1.Column=Register address, n. Column=Modul number.bitposition *) GW: gateway status-/diagnostics bits **) M: module diagnostics (1 bit for each module)

Process input data: 3 Words

- Slot 0: Cabecera BL20: Bits reservados (puede ampliar información en el manual del dispositivo)
- Slot 1: Tarjeta 16 Entradas Digitales de 24 Volteos DC. _

Estos dos Slots (0 y 1) suman un total de 2Words, una para cada Slot, se añade, además una Word extra de diagnósticos por cada tarjeta de E/S que agreguemos a la cabecera.

Con estas longitudes de variables, crearemos los canales de comunicación. Hay que tener en cuenta, que los diagnósticos del módulo no se añaden a la configuración en Codesys, para ello, abriremos el esclavo modbus RTU de Codesys y el menú de Canales de esclavos.



En la ventana que se abrirá, deberemos crear los canales de comunicación necesarios que, según el "Station Report" que hemos visto anteriormente, son dos (uno por cada Word) si no tenemos en cuenta la Word destinada a diagnósticos.

Deberemos empezar agregando un canal con los botones que veremos debajo a la derecha.

| Agregar canal | Borrar | Modificar |
|---------------|--------|-----------|
|---------------|--------|-----------|

En la ventana que se abra, será donde configuraremos los parámetros del canal de comunicación, al ser el primero, no tendrá desplazamiento inicial (offset) y la longitud, al tratarse de una Word será de 16 bits.

El FC que soporta este tipo de comunicación será el tipo 2.

| ModbusChannel | | \times |
|-------------------|--|----------|
| Canal | | |
| Nombre | Channel 2 | |
| Tipo de acceso | Read Discrete Inputs (Código funcional 2) 🛛 🗸 | |
| Desencadenador | Read Coils (Código funcional 1) Read Discrete Inputs (Código funcional 2) | |
| Comentario | Read Holding Registers (Codigo Funcional S) Read Input Registers (Código funcional 4) Write Single Coli (Código funcional 5) | |
| Registro READ | Write Single Register (Código funcional 6) Write Multiple Coils (Código funcional 15) | |
| Desplazamiento | Write Multiple Registers (Código funcional 16) Read/Write Multiple Registers (Código funcional 23) | |
| Longitud | 16 | |
| Tratamiento de er | rores Conservar el último valor 🗸 | |
| | | |
| Desplazamiento | ~ | |
| Longitud | 1 | |
| | Aceptar Cancelar | |

Añadiremos tantos canales como variables tengamos, añadiendo +1 el desplazamiento (0x0001)

| Nombre | Channel 2 | |
|-------------------|---|--|
| Tipo de acceso | Read Discrete Inputs (Código funcional 2) $\qquad \checkmark$ | |
| Desencadenador | Cyclic \checkmark Tiempo de ciclo [ms] 100 | |
| Comentario | | |
| Registro READ | | |
| Desplazamiento | 0x0001 ~ | |
| Longitud | 16 | |
| Tratamiento de er | rores Conservar el último valor 🗸 | |
| | | |
| Desplazamiento | | |
| | | |

| Modbus_Esclavo_puerto_CC | мх | | | | | | | | | |
|---|----|-----------|--|-----------------|---------------------|----------|---------------------------|----------------------|----------|------------|
| Configuración de esclavo de Modbus | | Nombre | Tipo de acceso | Desencadenador | Desplazamiento READ | Longitud | Tratamiento de errores | Desplazamiento WRITE | Longitud | Comentario |
| | 0 | Channel 0 | Read Discrete Inputs (Código funcional 02) | Cyclic, t#100ms | 16#0000 | 16 | Conservar el último valor | | | |
| Canal esclavo Modbus | 1 | Channel 1 | Read Discrete Inputs (Código funcional 02) | Cyclic, t#100ms | 16#0001 | 16 | Conservar el último valor | | | |
| Init esclavo Modbus ModbusGenericSerialSlave Asignación E/S | | | | | | | | | | |
| Estado | | | | | | | | | | |
| Información | | | | | | | | | | |
| | | | | | | | | | | |

Ahora sí que podremos descargar la configuración del proyecto y, si hemos seguido y configurado debidamente la guía, tendremos la comunicación funcionando.



| Dispositivos 👻 🖣 🗙 | Modbus_Esclavo_puerto_ | сом х | | | | | | | | |
|---|--|------------------|------------|---------------------------|--------------------|----------------------------|---------------|-------------|--------|--|
| E2M Device [Conectado] (TBEN-Lx-PLC-1x) | Configuración de esclavo de Modbus | Find | Fi | Iter Show all | | • | | | | |
| Lógica PLC Device [Co | Canal esclavo Modbus regado] (TBEN-Lx-PLC-1x) | Variable | Asignación | Canal Channel 0 | Dirección %IB28 | Tipo ARRAY [01] OF BYTE | Valor actual | Valor nuevo | Unidad | Descripción Read Discrete Inputs |
| Administrador de bibliotecas PLC_PRG (PRG) | Init esclavo Modbus | iii-¥≱ iii-¥≱ | | Channel 1 Channel 1[0] | %IB30 %IB30 | ARRAY [01] OF BYTE BYTE | 2 | | | Read Discrete Inputs Read Discrete Inputs |
| 🖻 - 🎆 Configuración de tareas 🖻 - 🕸 MainTask | ModbusGenericSerialSlave Asignación E/S | **> in2 | ** ** | Bit0 Bit1 | %IX30.0 %IX30.1 | BOOL | FALSE TRUE | | | 0x0001 0x0002 |
| PLC_PRG | Estado | ** in4 | - ×. | Bit2 Bit3 | %IX30.2 %IX30.3 | BOOL | FALSE | | | 0x0003 0x0004 |
| | Información | **) in6 | | Bit4 Bit5 | %IX30.4 %IX30.5 | BOOL | FALSE | | | 0x0005 0x0006 |
| Diagnostics (Diagnostics) OUD VAUX_control (VAUX control) | | | | Bit6 Bit7 | %IX30.6 %IX30.7 | BOOL | FALSE | | | 0x0007 0x0008 |
| | | B- 19 | · | Channel 1[1] | %IB31 | BYTE | 0 | | | Read Discrete Inputs |
| S ⊂ M1 CMI Modbus_Master_COM_Port (Modbus Master, COM P G M Modbus_Esclavo_puerto_COM (Modbus Esclavo - | 54 1 | | | | | | | | | |

TBEN-PLC: Esclavo Modbus TCP

En este segundo ejemplo, comunicaremos mediante Modbus TCP el dispositivo TBEN-PLC y la pantalla IX T7A de Beijer, donde el PLC será el esclavo de la comunicación y el panel HMI será el maestro.

En primer lugar, desde Codesys deberemos generar la estructura de **esclavo Modbus TCP** de la comunicación.



Haciendo doble clic sobre el dispositivo "Ethernet (Ethernet)", pasaremos a configurar la interfaz que conecte el TBEN-PLC con la pantalla, para ello únicamente deberemos seleccionar la interfaz con la IP que hayamos asignado al configurar el módulo.

| Canacal | | | | |
|------------|-----------------|--------------|--------------|-----------------|
| Seneral | Interface: eth0 | | | |
| Estado | Use Operating 5 | Svstem Setti | nas | |
| nformación | Change Operati | ing System S | Settings | |
| | | | _ | 7 |
| | IP address | 193 . 1 | 68.0.46 | |
| | Subnet mask | 255 . 2 | 55 . 255 . 0 | |
| | Default Gateway | 193 . 10 | 68 . 0 . 1 | |
| | | | | |
| | | Network Ac | lapters | |
| | | Interfaces: | | |
| | | Name | Description | IP Address |
| | | lo | | 127.0.0.1 |
| | | eth0 | | 193.168.0.46 |
| | | eth1 | | 0.0.0 |
| | | eth2 | | 0.0.0 |
| | | usb0 | | 0.0.0.1 |
| | | IP address | s 193 | . 168 . 0 . 46 |
| | | Subnet ma | isk 255 | . 255 . 255 . 0 |
| | | Default Ga | ateway 193 | 168 0 1 |
| | | MAC Add | 00.07 | 40-17-00-00 |
| | | MAC-Addi | 00.07 | 40.17.00.00 |
| | | | | Ok Cancel |
| | | | | |

Seguidamente, haremos doble clic sobre "Modbus_TCP_Slave_Device" y pasaremos a configurar el ID de unidad y la longitud/cantidad de E/S que comunicaremos entre PLC y HMI.

El direccionamiento de E/S que genere y podamos Modificar en Codesys, no influirá en los valores de comunicación sobre IX Developer que veremos a continuación.

| Configuración Modbus ICP | Parámetros configurados | |
|--------------------------|---|-------------|
| Modbus TCP Slave Device | Tiempo de espera: | 2000 🚖 (ms) |
| Asignación E/S | Puerto esclavo: | 502 🖨 |
| Información | ID unidad: | 2 |
| | Tamaño ensamblado de sallida (%IW): | 10 |
| | Tamaño ensamblado de entrada (%OW): | 5 |
| | Data Model Start Addresses: | |
| | Coiles | 0 |
| | Colls: | |
| | Discrete Inputs: | 0 |
| | Discrete Inputs: Holding Register: | |
| | Discrete Inputs: HoldingRegister: Input Register: | |

En el menú izquierdo, accederemos a "Modbus TCP Slave Device Asignación E/S" para modificar el direccionamiento y el "tag" asignado a estas E/S que hemos definido anteriormente.

| Configuración Modbus TCP | Find | Find Filter Show all | | | | | | | | |
|--------------------------|-----------------|----------------------|------------|-----------|--------------------|--------|--------------------------|--|--|--|
| Modbus TCP Slave Device | Variable | Asignación Canal | | Dirección | Тіро | Unidad | Descripción | | | |
| Asignación E/S | | | Inputs | 🚺 %IW1 | ARRAY [09] OF WORD | | Registros Holding Modbus | | | |
| información | 😐 🁋 IW1 | ** | Inputs[0] | %IW1 | WORD | | | | | |
| | 😐 🁋 IW2 | ** | Inputs[1] | %IW2 | WORD | | | | | |
| | 🗉 🦘 IW3 | ** | Inputs[2] | %IW3 | WORD | | | | | |
| | 💷 🍽 IW4 | ** | Inputs[3] | %IW4 | WORD | | | | | |
| | 😐 🦄 IW5 | ** | Inputs[4] | %IW5 | WORD | | | | | |
| | 😐 🦘 IW6 | ** | Inputs[5] | %IW6 | WORD | | | | | |
| | 😐 🦄 IW7 | ** | Inputs[6] | %IW7 | WORD | | | | | |
| | 😐 🁋 IW8 | ** | Inputs[7] | %IW8 | WORD | | | | | |
| | 😟 🦄 IW9 | *** | Inputs[8] | %IW9 | WORD | | | | | |
| | 😐 🦄 IW 10 | *** | Inputs[9] | %IW10 | WORD | | | | | |
| | ii - * ø | | Outputs | 🔞 %QW1 | ARRAY [04] OF WORD | | Registros Input Modbus | | | |
| | 🗎 🧖 QW1 | ** | Outputs[0] | %QW1 | WORD | | | | | |
| | 🗎 🧖 QW2 | ** | Outputs[1] | %QW2 | WORD | | | | | |
| | 🗎 🧖 QW3 | ** | Outputs[2] | %QW3 | WORD | | | | | |
| | 🗄 🧖 QW4 | ** | Outputs[3] | %QW4 | WORD | | | | | |
| | | *** | Outputs[4] | %QW5 | WORD | | | | | |

• M: Modificadas.

HMI Beijer: Maestro Modbus TCP

Al crear el proyecto en IX Developer, seleccionaremos el Driver de MODICON, Modbus Master.

| | Choose Controller | |
|------|---|---------------------------------------|
| | Choose your preferred controller or OPC server in the menu below | |
| • Co | ontrollers | |
| | Select brand | Select protocol |
| | MATSUSHITA | Modbus Master Modbus Slave RTU/TCP |
| | MELSEC | |
| | MELSERVO | |
| | MODICON | |
| 0 OF | PC UA Server | |
| | URL: | |
| 0 01 | PC Classic Server | |
| • | Localhost | |
| C | Remote Server | Browse |
| | OPC Server: | ~ Refresh |
| | | |
| | | |
| | | |
| | | |

Para configurarlo, entraremos en Tags, dentro del menú de Functions.

| ▼ Functions (4) | | | | | | |
|-------------------|--------------------|--|--|--|--|--|
| 4 | Alarm Server | | | | | |
| 9 | Multiple Languages | | | | | |
| 9 | Security | | | | | |
| 0 | Tags | | | | | |

Dentro de Tags, iremos a la pestaña de Controladores y sobre el controlador1, que será el Modbus Master, haremos clic sobre el botón de Settings, para editar los parámetros de comunicación.

| 📮 Tags | | |
|---|---|--------------------------------------|
| Tags Controllers Triggers Poll Groups Index Registers | | |
| Home Add Delete | | Controller Settings Show Selection • |
| Name | D | Active |
| > Controller1 | | V |
| | | |

Deberemos configurar el método de comunicación que utilizará el Driver, puede ser Serie o EtherNet/IP.

| - | Settings | | | | | |
|--|--|-----------------|---|--|--|--|
| | Communication mode | Ethernet TCP/IP | | | | |
| | Default station | 0 | | | | |
| | Modbus protocol | RTU | | | | |
| | 32-bit word mapping | Big-endian | | | | |
| | Addressing | Decimal | | | | |
| | Start address | 0-based | | | | |
| | Silent time (ms) | 0 | | | | |
| | Coils/input status bits per message (read) | 128 | | | | |
| | Coils/input status bits per message (wri | 1 | | | | |
| | Holding/input registers per message (re | 16 | | | | |
| | Holding/input registers per message (w | 8 | | | | |
| | Force function code 0x10 | Disable | | | | |
| | String swap | Enable | • | | | |
| String swap Enable Communication mode Serial or Ethernet TCP/IP | | | | | | |

En el menú Stations, asignaremos la dirección IP del TBEN-PLC que será con la cual comunique el HMI.

| Station | IP Address | Port | Node | | |
|---------|--------------|------|------|-----|--------|
| 0 | 193.168.0.46 | 502 | 2 | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | Add | Remove |

Direccionamiento de las variables en IX Developer.

Volviendo al direccionamiento de variables que enviará y recibirá el TBEN-PLC, las cuales hemos configurado desde Codesys, veremos qué tipo de variables son y en función de esto, asignaremos las comunicaciones en IX Developer.

| Variable | Asignación | Canal | Dirección | Тіро | Unidad | Descripción |
|----------|------------|---------|-----------|--------------------|--------|--------------------------|
| | | Inputs | 🔞 %IW1 | ARRAY [09] OF WORD | | Registros Holding Modbus |
| 😟 🍢 | | Outputs | 🔞 %QW1 | ARRAY [04] OF WORD | | Registros Input Modbus |

- 1. Registros Holding Modbus (IW en Codesys)
- 2. Registros Input Modbus (QW en Codesys)

Registros Holding Modbus en IX Developer se inician en las direcciones: 40000

Registros Input Modbus en IX Developer se inician en las direcciones: 30000

Teniendo nociones de este direccionamiento, deberemos asignar de forma incremental el valor a los Tags que vayamos a utilizar, respetando el número total de IW y QW que hayamos definido en Codesys.

| Controllers | |
|-------------|--------------|
| Data Type | Controller 1 |
| INT 16 | 40006 |
| INT16 | 40007 |
| INT 16 | 40008 |
| INT16 | 40009 |
| INT16 | 40005 |
| INT 16 | 40001 |
| INT 16 | 40002 |
| INT 16 | 40003 |
| INT16 | 40004 |
| INT16 | |
| INT16 | 30000 |
| INT16 | 30001 |
| INT16 | 30002 |

