

Guía de Implementación

IO Scanning Modbus TCP M580-M221

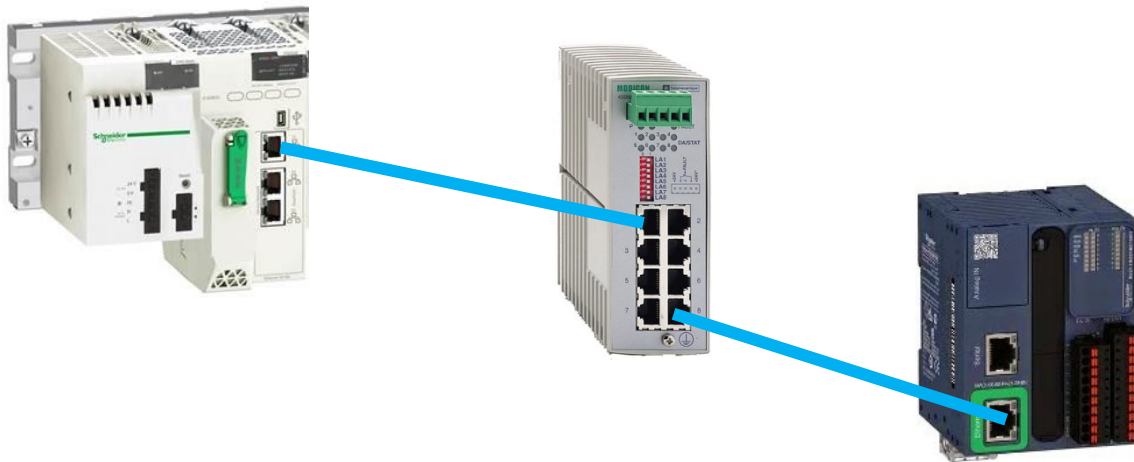
Producto y Versión:
M221 v1.3.1.0
M580 v1.13
SoMachine Basic v1.3
Unity v8.1



Revisión	Fecha	Autor	Modificaciones
1.0	02/2015	Marc Casanova	Primera versión

1. Objetivo

- Realizar un ejemplo sencillo de comunicación Modbus TCP/IP entre un M580 y un M221 haciendo uso del servicio I/O Scanning.
- La comunicación se basa en la escritura y lectura de una %MW.
- La arquitectura usada es:



2. Configuración SoMachine Basic

- Daremos una dirección IP al puerto Ethernet:

The screenshot displays the configuration interface for a Schneider Electric device. The top navigation bar includes tabs for Properties, Configuration, Programming, and Display. The left sidebar lists various components, with 'ETH1' highlighted. The main area shows the 'Ethernet' configuration for device 'M221'. The 'Fixed IP address' option is selected, and the IP address is set to 100.100.100.100, with a subnet mask of 255.255.255.0 and a gateway address of 0.0.0.0. Security parameters are also visible, with 'Programming protocol enabled', 'Modbus server enabled', and 'Auto discovery protocol enabled' all checked.

Properties Configuration Programming Display

Messages
MyController (M221CE16T)
Digital inputs
Digital outputs
Analog inputs
High Speed Counters
Pulse Generators
IO devices
ETH1
Modbus TCP
SL1 (Serial line)

Ethernet

Device name: M221

IP address by DHCP
 IP address by BOOTP
 Fixed IP address

IP address: 100 . 100 . 100 . 100
Subnet mask: 255 . 255 . 255 . 0
Gateway address: 0 . 0 . 0 . 0

Transfer Rate: AUTO

Security Parameters

Programming protocol enabled
 Modbus server enabled
 Auto discovery protocol enabled

2. Configuración SoMachine Basic

- Creamos un programa sencillo para comprobar luego que la lectura / escritura se realiza correctamente.
- En este caso se leerá la %MW10 del M221 y se escribirá sobre la %MW0:

The screenshot displays the Schneider Electric SoMachine Basic programming environment. The interface is divided into four tabs: Configuration, Programming, Display, and Commissioning. The Programming tab is active, showing a ladder logic program with three rungs.

The rungs are:

- Rung0:** Ladder logic with a normally open contact labeled `%MW0 = 1` and a coil labeled `%MW0 = 1`. The comment is `%Q0.0`.
- Rung1:** Ladder logic with a normally open contact labeled `%MW0 = 2` and a coil labeled `%MW0 = 2`. The comment is `%Q0.1`.
- Rung2:** Ladder logic with a normally open contact labeled `SHORT` and a coil labeled `%MW10 := 111`. The comment is `%MW10 := 111`.

The interface includes a toolbar with various editing tools and a left sidebar with a tree view showing the project structure.

2. Configuración SoMachine Basic

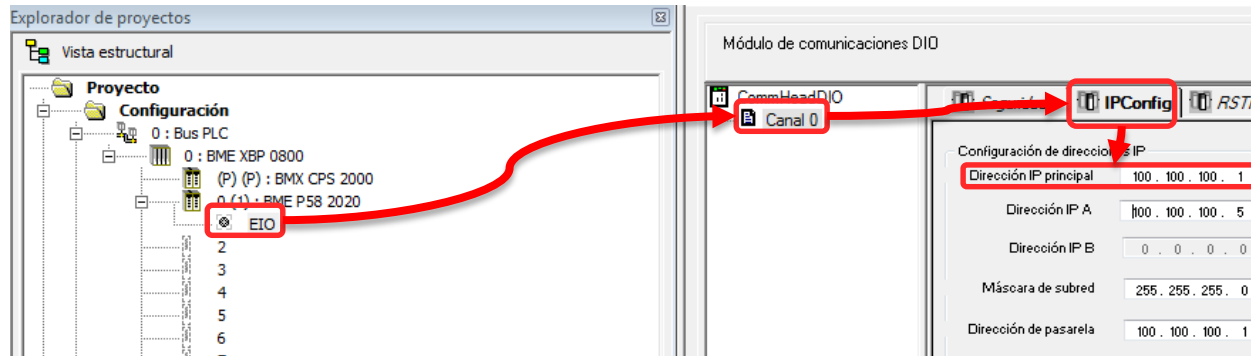
- Descargamos el programa al PLC y lo ponemos en RUN:

The screenshot displays the SoMachine Basic software interface with the following components:

- Navigation Bar:** Includes tabs for 'guration', 'Programming', 'Display', and 'Commissioning', along with a warning icon.
- Local Devices:** Lists COM11@BT, COM9@BT, COM1, and M221 Controller (USB). A checkbox for 'Keep Modbus driver parameters' is present.
- Ethernet Devices:** Shows the IP address 100.100.100.100.
- Remote Lookup:** Features input fields for IP address (0 . 0 . 0 . 0) and an 'Add' button.
- Selected Controller:** Displays 'Found: Firmware: 1.3.1.0' and 'Controller: TM221CE16T'. A yellow warning box states 'PC and controller applications are different'.
- Control Panel:** Contains buttons for 'Login', 'Logout', 'PC to Controller (download)', 'Controller to PC (upload)', and 'Stop controller'. A red arrow points from the 'PC to Controller (download)' button to a second set of buttons.
- Second Control Panel:** Shows a green success message: 'PC and Controller applications are identical Connection is established'. Below it are buttons for 'PC to Controller (download)', 'Controller to PC (upload)', 'Stop controller', and 'Start Controller'. A red arrow points from the 'Start Controller' button to the 'Start Controller' button in this panel.

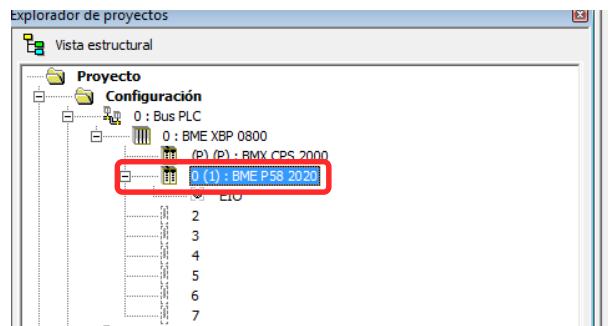
3. Configuración Unity Pro

- Configuramos la dirección IP del puerto de servicio (puerto usado para esta prueba). Debe estar en el mismo rango que la dirección IP del M221:

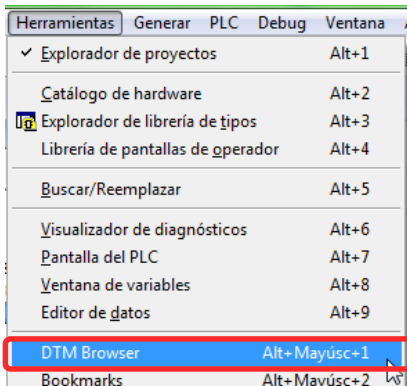


3. Configuración Unity Pro

- Crearemos un proyecto nuevo con la CPU de M580 que tengamos. En este caso una P582020:

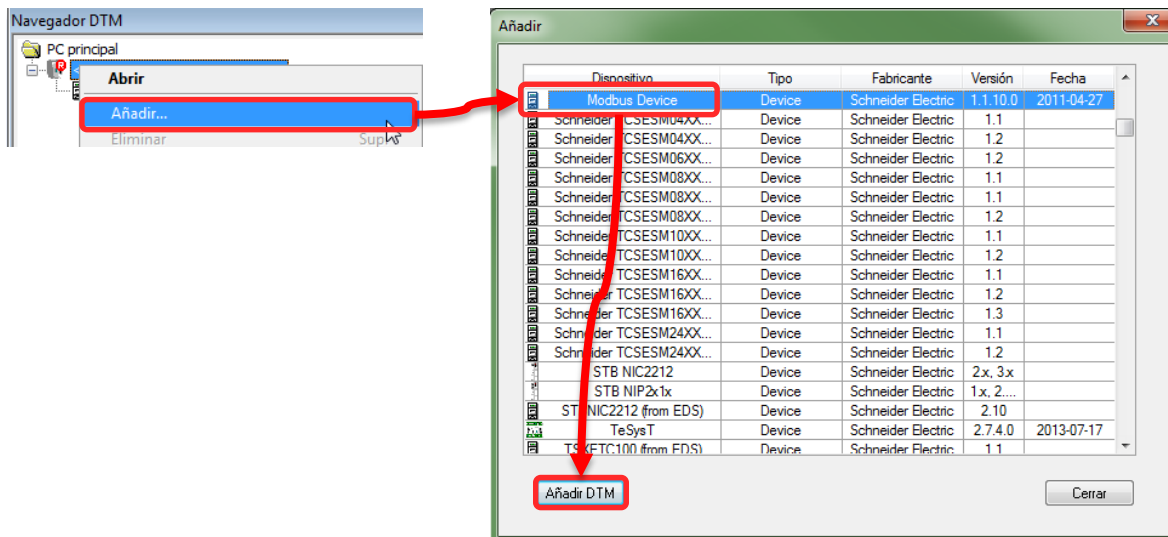


- Abriremos el DTM Browser para configurar la comunicación:



3. Configuración Unity Pro

- No hay disponible un DTM específico para M221, por lo que añadiremos un dispositivo Modbus genérico:



3. Configuración Unity Pro

- Configuraremos la comunicación. Dentro del DTM agregamos una línea de IO Scanning para escribir sobre la %MW0 y leer la %MW10 del M221:

The screenshot displays the Schneider Electric Unity Pro software interface. The left pane shows the project tree for 'BMEP58_ECPU'. A red box highlights the address '< 100.100.100.1 > BMEP58_ECPU' in the address bar. Another red box highlights the 'Lista de dispositivos' section, specifically the entry '[037] Modbus_Device <MDB: 100.100.100.100>'. The right pane shows the 'Ajuste de petición' (Request Adjustment) tab for the selected device. A table is visible with the following data:

Velocidad de repetición (ms)	Dirección LE	Longitud de lectura	Último valor	Dirección ES	Longitud de escritura
60	10	1	Conservar valor	0	1

Red arrows indicate the flow of configuration: from the address bar to the device list, and from the 'Ajuste de petición' tab to the table settings.

3. Configuración Unity Pro

- A continuación definiremos que las variables de memoria, generadas en la IODDT del dispositivo DTM añadido, sean de tipo WORD para facilitar la gestión de las comunicaciones:

Propiedades del canal
TCP/IP
Servicios
Servidor de direcciones
Esclavos EtherNet/IP locales
Esclavo local 1
Elementos
Esclavo local 2
Elementos
Esclavo local 3
Elementos
Lista de dispositivos
[197] Modbus_Device <Modbus 190.100.100.10>
Petición 001: Elementos
Registro

Entrada	(Bit de entrada)	Salida	(Bits de salida)
	0	0	
	1	1	

Definición de nombre de elemento

Nuevo tipo de datos de elementos:
WORD

Nombre de elemento (máximo de 32 caracteres):
BLOCKA_IW0

Aceptar Cancelar Ayuda

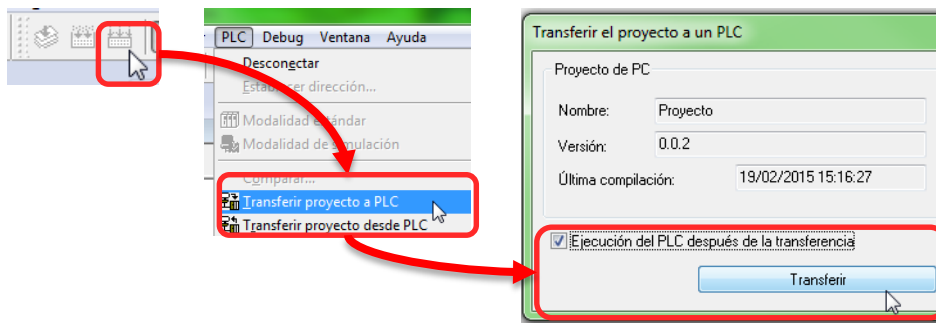
Offset/Dispositivo	Offset/Conexión	Nombre de elemento
	0	0
	1	1

- Tras hacerlo sobre las variables de entrada (lectura) haremos lo mismo con las de 'Salida'.

Modbus_Device	T_Modbus_Device
Freshness	BOOL
Freshness_1	BOOL
Inputs	T_Modbus_Device_IN
BLOCKA_IW0	WORD
Free0	ARRAY[0..1] OF BYTE
Outputs	T_Modbus_Device_O...
BLOCKA_QW0	WORD
Free1	ARRAY[0..1] OF BYTE

3. Configuración Unity Pro

- Tras compilar todo, volcaremos el programa al PLC y usaremos las variables de la IODDT para realizar las escrituras y lecturas (en una tabla de animación):



Nombre	Valor	Tipo
Modbus_Device.Inputs.BLOCKA_IW0	111	WORD
Modbus_Device.Outputs.BLOCKA_...	1	WORD

Lectura %MW10 del M221

Escritura %MW0 del M221

4. Programas ejemplo

- Se adjuntan programas ejemplo de Unity Pro y SoMachine Basic usados para realizar estas pruebas:



ioscanning_m580_m221.zef



IOscanning_M580_M221.smbp

Make the most of your energy

www.schneiderelectric.es

