

IO Scanner SoMachine Basic

- La función IO Scanner encontrarse disponible únicamente para las comunicaciones series (Modbus RTU o ASCII).
- Se pueden utilizar esa herramienta para comunicar con hasta 16 dispositivos distintos.
- Se pueden definir hasta 10 canales para cada dispositivo esclavo.
- Cada canal representa una sola petición Modbus.

SoMachine Basic suministra una serie de tipos de dispositivos predefinidos con funciones ya pre configuradas en el IO Scanner (para drivers propios de Schneider). Los equipos de este listado tienen solicitudes de inicialización predefinidas y canales pre configurados para facilitar la integración de los dispositivos en la red a ser construida.

Otra opción es utilizar el dispositivo esclavo genérico (opción “Others” - “Generic Device”), los cuales deben configurarse peticiones de inicialización y canales de forma manual. Esa opción debe ser utilizada para todos los equipos que no figuran en ese listado.

Nota: Los Drivers que no se muestran en ese listado no pueden ser reemplazados en el programa por otros drivers de funciones similares dado que presentan incompatibilidad de alguna de las funciones utilizadas por el template lo que implica que la comunicación no va a realizarse de forma correspondiente.

Funciones soportadas:

- **Funciones Modbus Serie utilizadas:**

Supported Modbus Function Code Dec (Hex)	Supported Sub-Function Code	Description
1 (1 hex) or 2 (2 hex)	-	Read multiple internal bits %M
3 (3 hex) or 4 (4 hex)	-	Read multiple internal registers %MW
5 (5 hex)	-	Write single internal bit %M
6 (6 hex)	-	Write single internal register %MW
8 (8 hex)	0 (0 hex), 10 (0A hex)...18 (12 hex)	Diagnostics
15 (0F hex)	-	Write multiple internal bits %M
16 (10 hex)	-	Write multiple internal registers %MW
23 (17 hex)	-	Read/write multiple internal registers %MW
43 (2B hex)	14 (0E hex)	Read device identification (regular service)

- **Funciones IO Scanner Modbus Serie**

Function Code Dec (Hex)	Description	Available For Configuration	Maximum Length (Bits)
1 (1 hex)	Read multiple bits (coils)	Channel	128
2 (2 hex)	Read multiple bits (discrete inputs)	Channel	128
3 (3 hex)	Read multiple words (holding registers)	Channel (default message type for channel configuration)	125
4 (4 hex)	Read multiple words (input registers)	Channel	125
5 (5 hex)	Write single bit (coil)	Channel Initialization Value	1
6 (6 hex)	Write single word (register)	Channel Initialization Value	1
15 (0F hex)	Write multiple bits (coils)	Channel Initialization Value	128
16 (10 hex)	Write multiple words (registers)	Channel Initialization Value (default message type for initialization values)	123
23 (17 hex)	Read/write multiple words (registers)	Channel	125 (read) 121 (write)

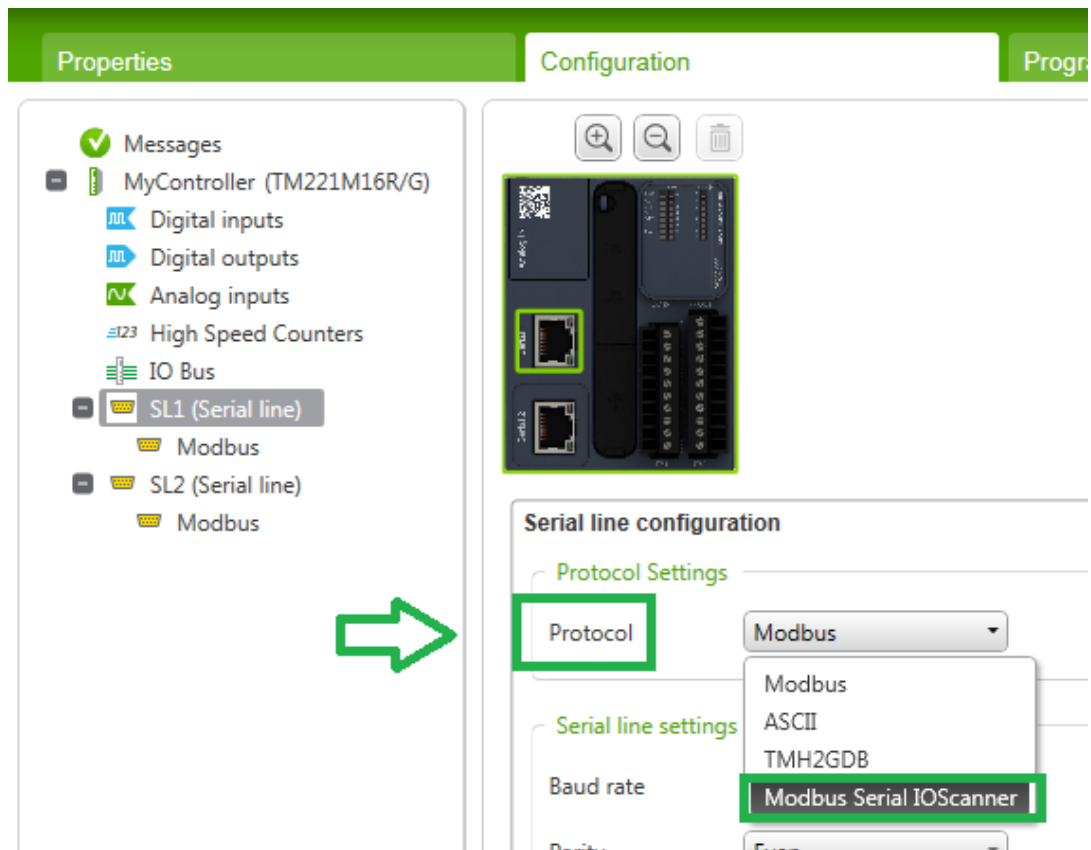
Nota: Es de fundamental importancia verificar el manual del equipo con el cual se va a realizar la comunicación a fin de asegurar que el equipo soporta las funciones utilizadas en esa herramienta.

Añadiendo un dispositivo Modbus Serie en el IOScanner:

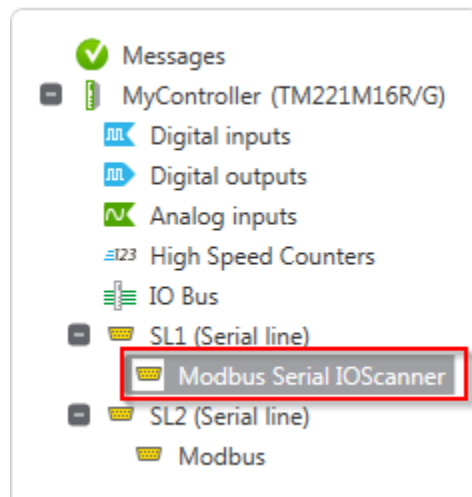
Para poder utilizar esa función primero pasamos a su declaración en el software. Para eso vaya a: “Configuration” y ahí elijas el puerto al que deseas asociar al IO scanner.



Al hacer click al puerto, en la ventana de la derecha, se van a mostrar sus propiedades (“Serial Line Configuration”). Vaya a “Protocol” y ahí despliegue las opciones → Elija “Modbus Serial IOScanner”) → Haga click en “Apply” para que los mismos sean asociados.



Una vez que el puerto tome las configuraciones el mismo quedará tal como se muestra en la imagen que sigue:

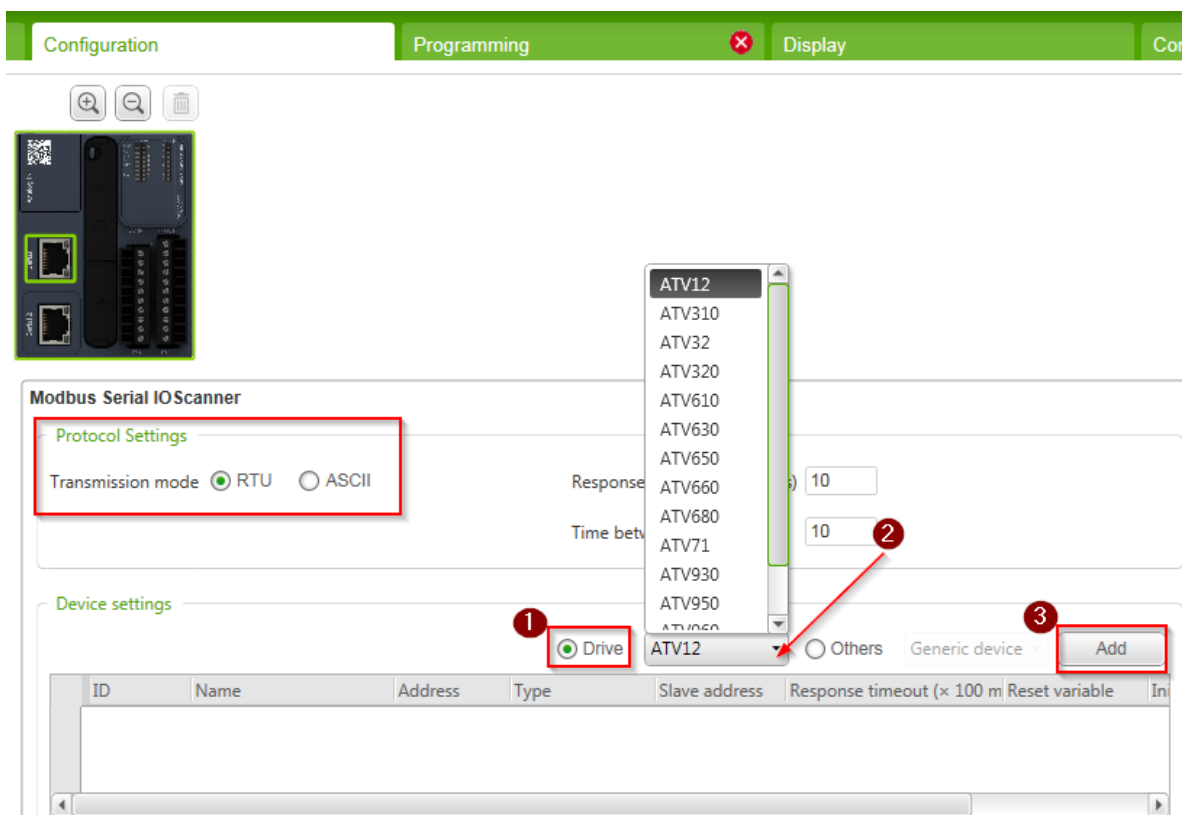


Para seguir con la configuración haga doble click en “Modbus Serial IOScanner”. Se va a mostrar la ventana que sigue. Ahí elija la forma de transmisión (en ese ejemplo vamos a trabajar con “RTU”).

A partir de ese punto pasamos a elegir la forma como vamos a utilizar esa herramienta si bien para los equipos pre configurados o si vamos a optar a utilizar una configuración personalizada.

En el caso de utilizar el **IO Scanner para manejar uno de los drivers de Schneider** tilde la opción “Drive” e indíquelo en el listado desplegable. **De no utilizarlo o de no encontrarlo habrá que tildar la opción “Others”** una vez hecho eso haga click en “ADD” para que se tomen las configuraciones.

Pasos para la configuración de un Driver del template:



Al haber elegido el dispositivo del listado → haga click en “Add” → el dispositivo va a ser mostrado tal como se muestra en la imagen que sigue. Pasamos a la configuración del Driver.

Donde se tiene los siguientes parámetros:

- **“Name”**: Es el nombre por lo cual podés identificar el dispositivo.
- El **“ADRV”** es la dirección asociada al dispositivo la cual se va a utilizar en el programa (se utilizan en los “Drivers Functions Blocks” – Bloques asociados a los drivers – validos únicamente para los drivers de Schneider).
- **“Slave Address”**: Ahí indicas la dirección Modbus del equipo asociado.
- **“Response Time”**: Se indica el tiempo de respuesta antes de que se deje de encuestar el equipo asociado sin que el responda (timeout).
- **“Reset variable”**: Especifica la dirección del bit de memoria que se utilizará para restablecer el dispositivo (reenviar las peticiones de inicialización). Cuando el bit de memoria especificado se establece en 1 por la aplicación, el dispositivo se restablece (no es un dato obligatorio para el uso de la función).
- **“Init. Requests”**: Las peticiones de inicialización son comandos específicos del dispositivo enviados por el Modbus Serial IOScanner para inicializar un dispositivo esclavo.
- **“Chanel”**: Cada canal representa una sola petición Modbus.

Para acceder a esos dos últimos parámetros hay que hacer click sobre los puntos suspensivos “...”

Device settings

Drive ATV32 Others Generic device

ID	Name	Address	Type	Slave address	Response timeout (× 100 m)	Reset variable	Init. requests	Channels
0	Device 0	%DRV0	ATV32	1	10	

La tabla de inicialización – “Initialization request Assistant”:

Name: Device 0 Address: %DRV0 Type: ATV12 Slave address: 1

Init. requests

ID	Message type	Offset	Length	Initialization value	Comment
0	Mbs 0x06 - Write single word (reg.)	8501	1	0	Switch ATV in NST State
1	Mbs 0x06 - Write single word (reg.)	12701	1	3201	Configuration of ETA register
2	Mbs 0x06 - Write single word (reg.)	12702	1	8604	Configuration of RFRD register (RPM)
3	Mbs 0x06 - Write single word (reg.)	12703	1	3206	Configuration of ETI register
4	Mbs 0x06 - Write single word (reg.)	12704	1	7200	Configuration of DP0 register
5	Mbs 0x06 - Write single word (reg.)	12721	1	8501	Configuration of CMD register
6	Mbs 0x06 - Write single word (reg.)	12722	1	8602	Configuration of LFRD register (RPM)

Buttons: Add, Ok, Cancel

- Esa asistente presenta las solicitudes de inicialización definidas por defecto. A demás de esas también es posible incluir otros parámetros a esa tabla de ser necesario pudiendo llegar hasta 20 solicitudes de inicialización para cada dispositivo esclavo.

Nota:

El Modbus Serial IO Scanner no inicia el intercambio cíclico de datos hasta que todas las solicitudes de inicialización hayan sido confirmadas por el dispositivo.

Durante la fase de inicialización, los objetos de red no se actualizan.

La tabla de inicialización también puede ser utilizada para levantar los datos pre configurados en el IO Scanner del propio variador.

En el caso de que necesites añadir parámetros a esa tabla:

- Haga clic en **“Add”** para crear nuevas solicitudes de inicialización.
- Seleccione una entrada y utilice los botones de flecha hacia arriba y hacia abajo para cambiar el orden en que se envían las solicitudes de inicialización al dispositivo.
- Cuando se hayan definido las solicitudes de inicialización, haga clic en **“OK”** para volver a la ventana Configuración del dispositivo.

La configuración de los canales:

Así como la tabla de inicialización, los equipos del “template” también ya tienen peticiones de lectura/escritura ya pre definidos. También está la posibilidad de añadir canales con distintos parámetros a esa configuración.

Utilice la ventana "**Chanel Assitant**" para configurar los canales.

Por ejemplo, el ATV12 viene con la siguiente configuración ya definida:

The screenshot displays the 'Channel assistant' software interface. At the top, it shows device information: Name: Device 0, Address: %DRV0, Type: ATV12, Slave address: 1. Below this is a table of channels. The table has columns for ID, Name, Configura, Message type, Trigger, R Offset, R Length, Error management, W Offset, W Lengtl, and Comment. One channel is listed with ID 0, Name ATV_IoScanner, Message type Mbs 0x17 - Re, Trigger Cyclic 200 ms, R Offset 12741, R Length 4, Error management Set to zero, W Offset 12761, W Lengtl 2, and Comment Main IoScanner ATV channel. An 'Add' button is located to the right of the table. Below the table, a detailed configuration window for the selected channel is shown. It includes fields for Name (ATV_IoScanner), Message type (Mbs 0x17 - Read/Write mult. words (reg.)), Trigger (Cyclic) with a Cycle time of 20 ms, and Comment (Main IoScanner ATV channel). There are 'Ok' and 'Cancel' buttons. Below this window, there are sections for 'READ objects' and 'WRITE objects'. The 'READ objects' section has fields for Offset (12741), Length (4), and Error management (Set to zero). The 'WRITE objects' section has fields for Offset (12761) and Length (2). There are 'Ok' and 'Cancel' buttons at the bottom of this section.

ID	Name	Configura	Message type	Trigger	R Offset	R Length	Error management	W Offset	W Lengtl	Comment
0	ATV_IoScanner	...	Mbs 0x17 - Re	Cyclic 200 ms	12741	4	Set to zero	12761	2	Main IoScanner ATV channel

También se puede apreciar ese otro ejemplo en donde se muestra un canal configurado para una petición de Lectura / Escritura de Varias Palabras (código de función Modbus 23). Lee una palabra del registro con el offset

16 # 0C21 y escribe dos palabras en el registro con el offset 16 # 0C20. Esta petición se ejecuta cuando hay un flanco ascendente del gatillo definido (ver tabla a continuación):

The screenshot shows the 'Channel assistant' configuration window. It features a title bar with the text 'Channel assistant' and a close button. Below the title bar is a green header. The main area is divided into sections. The 'Channel' section contains: 'Name' (Device_Channel0), 'Message type' (Mbs 0x17 - Read/Write mult. words (reg.)), 'Trigger' (Cyclic) with a 'Cycle time (x 10 ms)' field set to 20, and 'Comment' (Default channel 0). Below this are two sections: 'READ objects' and 'WRITE objects'. Each has 'Offset' (0) and 'Length' (1) fields. At the bottom right are 'OK' and 'Cancel' buttons.

Pasos para la configuración de un “Generic Device”:

Tomando el ATV312 como ejemplo, vamos a explicar el uso del IO Scanner para un dispositivo genérico. Dado que este no soporta la función Modbus Read_Write (en forma simultánea – función 23 de Modbus) que utiliza la tabla de inicialización de la plantilla y así como tampoco el ATV312 tiene la tabla de IOScanning en su memoria (Motivo por lo cual el mismo no se encuentra en el listado de equipos que viene por defecto en el software como opción de Driver . Por eso para esa familia de ATV las peticiones de lectura y/o escritura las tenés que realizar por separado. En ese caso también se podrá utilizar la función “IO Scanner” pero las construcciones de las peticiones tendrán que ser realizadas por separado y de forma manual. Añadiendo como equipo un elemento genérico (tal como lo mencionado en páginas anteriores).

Habrá que crearse un canal dedicado a la lectura y otro para la escritura. Como ese variador no tiene la tabla IO scanner incorporada en él, los registros no van a poder ser definido en un área común para las peticiones por lo que habrá que agrupar los registros a ser leído y/o escritos en cada petición de IO Scanner usando un canal para cada una de ellas (ver imágenes con ejemplo a continuación).

Una vez Elegido la opción **“Others”** y añadido el **“Generic Device”** Se hace click en los pasos suspensivos“...” para configurar los canales (Las peticiones de lectura/escritura por grupo de registros):

Modbus Serial IOScanner

Ajustes de protocolo

Modo de transmisión RTU ASCII

Timeout de respuesta (× 100 ms)

Tiempo entre tramas (ms)

Configuración del dispositivo

Unidad Otros

ATV12 Generic device **Añadir**

ID	Nombre	Dirección	Tipo	Dirección de esclavo	Timeout de respuesta (× 100 ms)	Restablecer variable	Peticiones iniciales	Canales
0	ATV312_Write		Generic device	2	10	%M0

Los canales:

Se utilizan las funciones Modbus Leer Múltiples Registros y Escribir Múltiples Registros (una canal para cada petición).

En Offset se indican las direcciones de inicio de la petición y en Longitud se indican las cantidades de registros a partir de la dirección apuntada a los cuales necesitas leer o escribir.

Nombre: ATV312_Write **Dirección:** **Tipo:** Generic device **Dirección de esclavo:** 2

Canales

ID	Nombre	Configura	Tipo de mensaje	Desencadenador	R Offset	R Longitud
0	Canal_Lectura_1	...	Mbs 0x03 - Read mult. words (holding reg.)	Cíclico 200 ms	3201	5
1	Canal_Escritura_1	...	Mbs 0x10 - Write mult. words (reg.)	Cíclico 200 ms		

Asistente de canales



Nombre: ATV312_Write Dirección: Tipo: Generic device Dirección de esclavo: 2

Canales

Añadir

ID	Nombre	Configura	Tipo de mensj	Desencadenador	R Offset	R Longiti	Gestión de errores	W Offset	W Longit	Comentario
0	Canal_Lectura_1	...	Mbs 0x03 - Re	Cíclico 200 ms	3201	5	Poner a CERO			Default channel 0
1	Canal_Escritura_1	...	Mbs 0x10 - Wi	Cíclico 200 ms				8501	2	

Aceptar

Cancelar