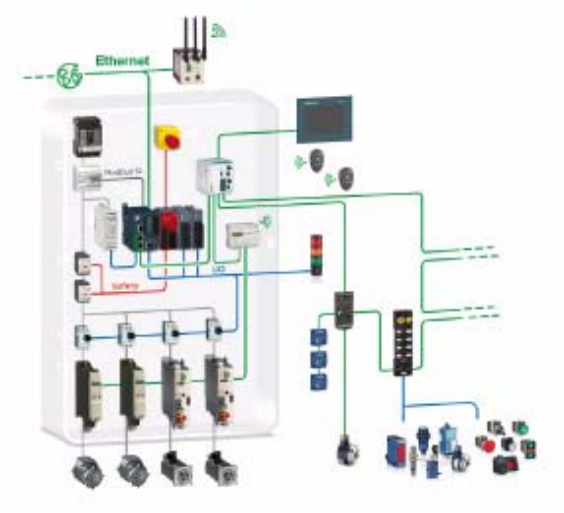
SoMachine
Modbus TCP IOScanner
Guía del usuario

09/2014



EIO0000001952.00

www.schneider-electric.com

Schneider Electric

La información que se ofrece en esta documentación contiene descripciones de carácter general y/o características técnicas sobre el rendimiento de los productos incluidos en ella. La presente documentación no tiene como objeto sustituir dichos productos para aplicaciones de usuario específicas, ni debe emplearse para determinar su idoneidad o fiabilidad. Los usuarios o integradores tienen la responsabilidad de llevar a cabo un análisis de riesgos adecuado y completo, así como la evaluación y las pruebas de los productos en relación con la aplicación o el uso de dichos productos en cuestión. Ni Schneider Electric ni ninguna de sus filiales o asociados asumirán responsabilidad alguna por el uso inapropiado de la información contenida en este documento. Si tiene sugerencias de mejoras o modificaciones o ha hallado errores en esta publicación, le rogamos que nos lo notifique.

No se podrá reproducir este documento de ninguna forma, ni en su totalidad ni en parte, ya sea por medios electrónicos o mecánicos, incluida la fotocopia, sin el permiso expreso y por escrito de Schneider Electric.

Al instalar y utilizar este producto es necesario tener en cuenta todas las regulaciones sobre seguridad correspondientes, ya sean regionales, locales o estatales. Por razones de seguridad y para garantizar que se siguen los consejos de la documentación del sistema, las reparaciones solo podrá realizarlas el fabricante.

Cuando se utilicen dispositivos para aplicaciones con requisitos técnicos de seguridad, siga las instrucciones pertinentes.

Si con nuestros productos de hardware no se utiliza el software de Schneider Electric u otro software aprobado, pueden producirse lesiones, daños o un funcionamiento incorrecto del equipo.

Si no se tiene en cuenta esta información, se pueden causar daños personales o en el equipo.

© 2014 Schneider Electric. Reservados todos los derechos.

Tabla de materias

	Información de seguridad	**5**
	Acerca de este libro	**7**
Capítulo 1	**Presentación de Modbus TCP IOScanner**	**9**
	Modbus TCP IOScanner Características generales	**10**
	Arquitectura	**12**
	Principios	**14**
Capítulo 2	**Instalación de red**	**17**
	Planificación de red	**18**
	Configuración de dirección IP	**20**
	Pruebas de red	**23**
Capítulo 3	**Configuración de Modbus TCP IOScanner**	**25**
	Adición de un esclavo al Modbus TCP IOScanner	**26**
	Configuración de un Modbus TCP IOScanner	**27**
	Configuración de un módulo de E/S distribuidas Advantys OTB en el Modbus TCP IOScanner	**29**
	Configurar un esclavo predefinido en el Modbus TCP IOScanner	**32**
	Configuración de un dispositivo genérico en el Modbus TCP IOScanner	**34**
Capítulo 4	**Operación Modbus TCP IOScanner**	**37**
	Modbus TCP IOScannerVerificación de recurso	**38**
	Modbus TCP IOScanner Modalidades de funcionamiento	**40**
	Interfaz de aplicación	**44**
Capítulo 5	**Mantenimiento de Modbus TCP IOScanner**	**47**
	Diagnósticos: SoMachine modalidad online	**48**
	Diagnósticos: servidor web	**52**
	Resolución de problemas	**54**
Apéndices		**55**
Apéndice A	**Funciones de Modbus TCP IOScanner**	**57**
	IOS_GETSTATE: leer el estado de Modbus TCP IOScanner	**58**
	IOS_START: ejecutar Modbus TCP IOScanner	**59**
	IOS_GETHEALTH: leer el valor de bit de estado	**60**
	IOS_STOP: detener el Modbus TCP IOScanner	**61**
	CONFIGURE_OTB: enviar la configuración del software del Advantys OTB	**62**

Apéndice B Tipos de datos de Modbus TCP IOScanner. 65
IosStateCodes: valores de estado de Modbus TCP IOScanner 66
CommunicationErrorCodes: códigos de error detectados 67
configurationOTBErrorCodes: se han detectado códigos de error en la configuración OTB 68
Apéndice C Representación de funciones y de bloques de funciones 69
Diferencias entre una función y un bloque de funciones 70
Cómo utilizar una función o un bloque de funciones en lenguaje IL . . 71
Cómo utilizar una función o un bloque de funciones en lenguaje ST . 75
Glosario 79
Índice 83

Información de seguridad

Información importante

AVISO

Lea atentamente estas instrucciones y observe el equipo para familiarizarse con el dispositivo antes de instalarlo, utilizarlo o realizar su mantenimiento. Los mensajes especiales que se ofrecen a continuación pueden aparecer a lo largo de la documentación o en el equipo para advertir de peligros potenciales o para ofrecer información que aclara o simplifica los distintos procedimientos.

La inclusión de este icono en una etiqueta “Peligro” o “Advertencia” indica que existe un riesgo de descarga eléctrica, que puede provocar lesiones si no se siguen las instrucciones.

Éste es el icono de alerta de seguridad. Se utiliza para advertir de posibles riesgos de lesiones. Observe todos los mensajes que siguen a este icono para evitar posibles lesiones o incluso la muerte.

⚠ PELIGRO

PELIGRO indica una situación de peligro que, si no se evita, **provocará** lesiones graves o incluso la muerte.

⚠ ADVERTENCIA

ADVERTENCIA indica una situación de peligro que, si no se evita, **podría provocar** lesiones graves o incluso la muerte.

⚠ ATENCIÓN

ATENCIÓN indica una situación peligrosa que, si no se evita, **podría provocar** lesiones leves o moderadas.

AVISO

AVISO indica una situación potencialmente peligrosa que, si no se evita, **puede provocar** daños en el equipo.

TENGA EN CUENTA

La instalación, manejo, puesta en servicio y mantenimiento de equipos eléctricos deberán ser realizados sólo por personal cualificado. Schneider Electric no se hace responsable de ninguna de las consecuencias del uso de este material.

Una persona cualificada es aquella que cuenta con capacidad y conocimientos relativos a la construcción, el funcionamiento y la instalación de equipos eléctricos y que ha sido formada en materia de seguridad para reconocer y evitar los riesgos que conllevan tales equipos.

Acerca de este libro

Presentación

Objeto

Utilice este documento para lo siguiente:
- Planifique su red de Modbus TCP IOScanner.
- Instale y configure su red de Modbus TCP IOScanner.
- Utilice y mantenga su red de Modbus TCP IOScanner.

NOTA: Lea y comprenda este documento y todos los documentos relacionados antes de llevar a cabo la instalación, utilización o realización de tareas de mantenimiento del controlador.

Campo de aplicación

Este documento ha sido actualizado con la publicación del complemento SoMachine V4.1 Modbus TCP IOScanner.

Las características técnicas de los dispositivos que se describen en este manual también se encuentran online.

Las características que se indican en este manual deben coincidir con las que figuran online. De acuerdo con nuestra política de mejoras continuas, es posible que a lo largo del tiempo revisemos el contenido con el fin de elaborar documentos más claros y precisos. En caso de que detecte alguna diferencia entre el manual y la información online, utilice esta última para su referencia.

Documentos relacionados

Título de la documentación	Número de referencia
Modicon M251 Logic Controller - Guía de programación	EIO0000001462 (ING), EIO0000001463 (FRA), EIO0000001464 (ALE), EIO0000001465 (ESP), EIO0000001466 (ITA), EIO0000001467 (CHI)
SoMachine Guía de programación	EIO0000000067 (ING); EIO0000000069 (FRA); EIO0000000068 (ALE); EIO0000000071 (ESP); EIO0000000070 (ITA); EIO0000000072 (CHI)
Guía esencial: redes, conectividad y servidores Web	DIA6ED2130205EN (ENG)

Puede descargar estas publicaciones técnicas y otra información técnica de nuestro sitio web www.schneider-electric.com.

Información relativa al producto

⚠ ADVERTENCIA

PÉRDIDA DE CONTROL

- El diseñador del esquema de control debe tener en cuenta las posibles modalidades de fallo de rutas de control y, para ciertas funciones de control críticas, proporcionar los medios para lograr un estado seguro durante y después de un fallo de ruta. Funciones de control críticas son, por ejemplo, una parada de emergencia y una parada de sobrerrecorrido, un corte de alimentación y un reinicio.
- Para las funciones de control críticas deben proporcionarse rutas de control separadas o redundantes.
- Las rutas de control del sistema pueden incluir enlaces de comunicación. Deben tenerse en cuenta las implicaciones de los retrasos de transmisión no esperados o los fallos en el enlace.
- Tenga en cuenta todas las reglamentaciones para la prevención de accidentes y las directrices de seguridad locales.[1]
- Cada implementación de este equipo debe probarse de forma individual y exhaustiva antes de entrar en servicio.

El incumplimiento de estas instrucciones puede causar la muerte, lesiones serias o daño al equipo.

[1] Para obtener información adicional, consulte NEMA ICS 1.1 (última edición), "Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control" (Directrices de seguridad para la aplicación, la instalación y el mantenimiento del control de estado estático) y NEMA ICS 7.1 (última edición), "Safety Standards for Construction and Guide for Selection, Installation and Operation of Adjustable-Speed Drive Systems" (Estándares de seguridad para la construcción y guía para la selección, instalación y utilización de sistemas de unidades de velocidad ajustable) o su equivalente aplicable a la ubicación específica.

⚠ ADVERTENCIA

FUNCIONAMIENTO IMPREVISTO DEL EQUIPO

- Utilice solo software aprobado por Schneider Electric para este equipo.
- Actualice el programa de aplicación siempre que cambie la configuración de hardware física.

El incumplimiento de estas instrucciones puede causar la muerte, lesiones serias o daño al equipo.

Capítulo 1
Presentación de Modbus TCP IOScanner

Contenido de este capítulo

Este capítulo contiene los siguiente apartados:

Apartado	Página
Modbus TCP IOScanner Características generales	10
Arquitectura	12
Principios	14

Modbus TCP IOScanner Características generales

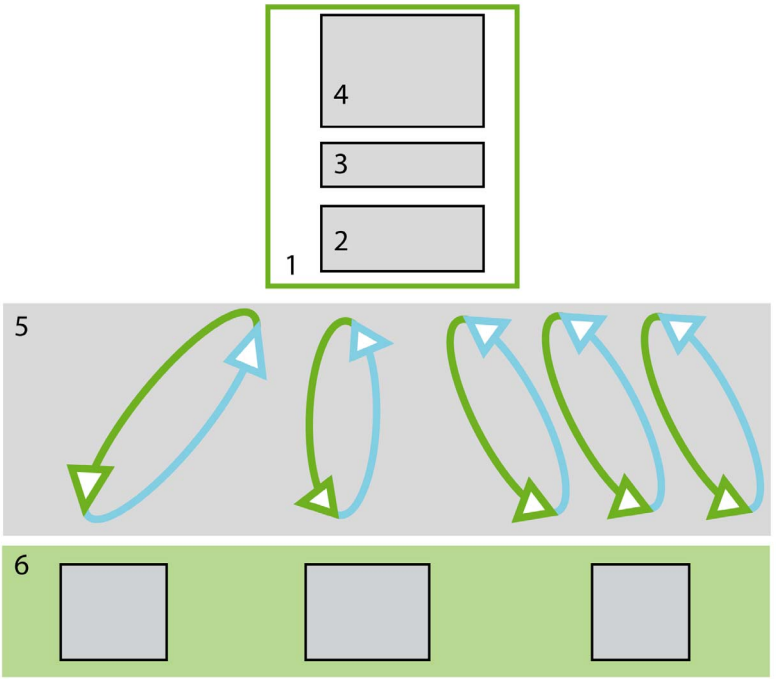
Presentación

El Modbus TCP IOScanner es un servicio basado en Ethernet que consulta continuamente a los dispositivos esclavos para intercambiar datos, estados e información de diagnóstico. En este proceso se supervisan las entradas y salidas de control de los dispositivos esclavos.

El Modbus TCP IOScanner se basa en el Modbus TCP estándar. La característica principal de este estándar es el uso de un modelo de red maestro/esclavo. El maestro único es el controlador .

La comunicación con el esclavo se consigue mediante el uso de los canales Modbus TCP *(véase página 14)*.

Principio



1 Controlador
2 Imágenes de E/S
3 Interfaz de la aplicación *(véase página 44)*
4 Aplicación
5 Canales Modbus *(véase página 14)*
6 Dispositivos esclavos *(véase página 14)*

Arquitectura del sistema

El Modbus TCP IOScanner se basa en:

- Red Ethernet incluyendo el controlador, los esclavos y el equipo de infraestructura *(véase página 12)*,
- Configuración de software *(véase página 13)*.

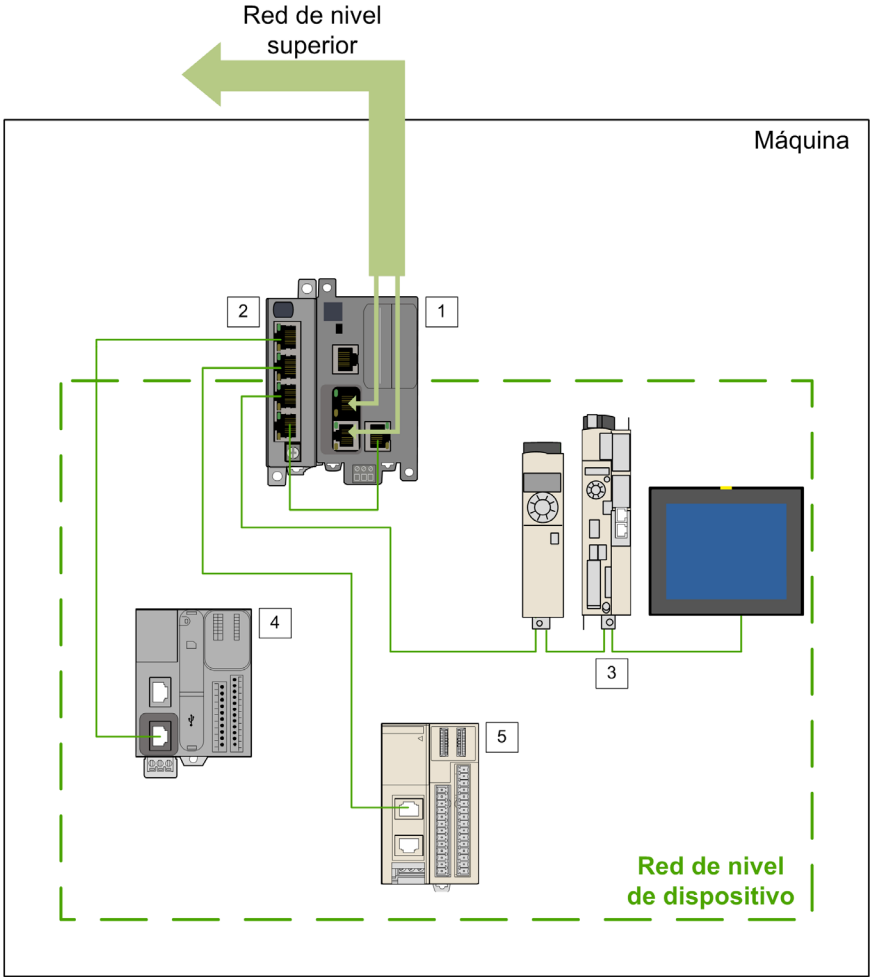
Compatibilidad del controlador

El servicio de Modbus TCP IOScanner está disponible en el controlador de TM251MESE.

Arquitectura

Red Ethernet

En la siguiente figura se presenta la arquitectura típica de un Modbus TCP IOScanner.



1 Controlador/maestro Modbus
2 TM4ES4 utilizado como un conmutador Ethernet autónomo.
3 Esclavos con encadenamiento tipo margarita
4 Esclavo Modbus
5 Isla de E/S

El controlador está conectado a la red de nivel superior, así como a la red de nivel del dispositivo.

La red de nivel del dispositivo está controlada por el Modbus TCP IOScanner.

El controlador se puede usar como una pasarela *(véase página 20)* entre las dos redes.

Configuración del software

El área de red de los esclavos y cada E/S se configuran mediante el software:

- La configuración de comunicación define los períodos de direccionamiento y comunicación.
- La configuración del dispositivo define el comportamiento del mismo.

La configuración del esclavo activa la atribución variable para optimizar la supervisión.

La configuración de E/S ajusta la calidad de la supervisión de acuerdo con el ancho de banda de la red.

Puede establecer distintos parámetros *(véase página 15)* para optimizar el rendimiento.

Servicios opcionales

El Modbus TCP IOScanner se puede asociar a varios servicios opcionales:

- DHCP: el servidor de protocolo de configuración dinámica de host asigna una dirección IP a un dispositivo esclavo cuando este lo requiere.
- FDR: el servidor de sustitución rápida de dispositivo configura un dispositivo remoto sustituido controlado por el Modbus TCP IOScanner sin detener la aplicación.
- Servidor Web *(véase página 52)*.

Principios

Descripción general

El Modbus TCP IOScanner lee las entradas y escribe las salidas de los dispositivos esclavos.

La comunicación entre el Modbus TCP IOScanner y los dispositivos esclavos se lleva a cabo mediante el uso de los canales Modbus.

La comunicación en el Modbus TCP IOScanner se configura con el software de SoMachine.

Tipos de esclavos

Hay tres tipos diferentes de esclavos Modbus TCP IOScanner en el software de SoMachine:

- Los dispositivos **esclavos Advantys OTB** se usan para las E/S digitales y analógicas remotas. Utiliza el software de SoMachine para la configuración específica del dispositivo y de los módulos de E/S asociados.
- Los dispositivos **esclavos predefinidos** son dispositivos Modbus comunes que se presentan con un conjunto predefinido de parámetros de comunicación.
  Utiliza un software dedicado y/o un HMI local para configurar los dispositivos. Con la tecnología FDT/DTM, los dispositivos esclavos predefinidos con ajustes avanzados se pueden configurar en SoMachine; consulte la Device Type Manager Guía del usuario.
- Los dispositivos **esclavos genéricos** se utilizan para todos los demás dispositivos esclavos Modbus.
  Toda la configuración del dispositivo se realiza con un software de terceros y/o con un HMI local. Con la tecnología FDT/DTM, algunos dispositivos se pueden configurar en SoMachine; consulte la Device Type Manager Guía del usuario.

Canal Modbus

Un canal Modbus transmite una petición Modbus entre el maestro y el esclavo.

El Advantys OTB y los dispositivos esclavos predefinidos utilizan un canal por dispositivo. Este canal se configura mediante el uso del software de SoMachine.

Para un dispositivo esclavo genérico, puede utilizar distintos canales. Para enviar varias peticiones distintas a un dispositivo, crea diversos canales.

Parámetros de configuración de comunicación

Configura cada esclavo en la red del Modbus TCP IOScanner mediante el uso del software de SoMachine.

En la siguiente tabla se muestran los parámetros de configuración de comunicación:

Parámetro	Descripción
Dirección IP	La dirección IP del esclavo en la red Ethernet.
Timeout de estado	El valor de tiempo se expresa en ms. Si el Modbus TCP IOScanner no detecta una respuesta por parte del esclavo pasado el retardo, se produce un error de estado.
Frecuencia de repetición	El valor de tiempo se expresa en ms. Representa el retardo entre dos transmisiones de una petición. El valor debe ser inferior al del timeout de estado.
ID de canal	Identificador único para un canal Este valor se crea automáticamente mediante el software de SoMachine cuando se añade un nuevo canal. Puede leer este valor en la ficha **Modbus TCP Channel Configuration** o, para el esclavo Advantys OTB, en la ficha **Modbus TCP Slave Configuration**.

Capítulo 2
Instalación de red

Contenido de este capítulo

Este capítulo contiene los siguiente apartados:

Apartado	Página
Planificación de red	18
Configuración de dirección IP	20
Pruebas de red	23

Planificación de red

Objetivo

Una red planificada aumenta la eficiencia de instalación a la vez que reduce el tiempo de instalación y los costes. La utilización de los materiales (interruptores, cables, puertos) se debe designar previamente para planificar la red.

Diseño de red

Para diseñar y planificar la red de Modbus TCP IOScanner, consulte la documentación correspondiente, como el Media Planning and Installation Manual, de ODVA. Puede descargar el manual desde el *sitio web de ODVA*.

Tipos de conmutadores

Dependiendo de las necesidades específicas de su red, utilice el tipo de conmutador adecuado:

Si necesita	Entonces considere utilizar
Diagnósticos de red o información del funcionamiento.	Conmutadores adecuados
Disponibilidad de las comunicaciones en caso de pérdida de conexión física	Conmutadores redundantes
Red de largo alcance (fibra óptica)	Conectar con un conector SC dúplex

NOTA: No use un concentrador para establecer una red de Modbus TCP IOScanner.

Para obtener más información acerca de los conmutadores, consulte la **Guía esencial: redes, conectividad y servidores web.**

Tipos de cable

En estas tablas se muestran las referencias de cable que se pueden usar en la red.

Para obtener más información acerca de los conmutadores, consulte la **Guía esencial: redes, conectividad y servidores web.**

En la instalación estándar, puede usar los siguientes cables:

Referencia	Descripción	Detalles	Longitud
490NTW000••	Cable blindado Ethernet para conexiones DTE	Cable estándar, equipado con conectores RJ45 en cada extremo del DTE. De conformidad con la CE	2, 5, 12, 40 o 80 m (6,56, 16,4, 39,37, 131,23 o 262,47 pies)
490NTW000••U		Cable estándar, equipado con conectores RJ45 en cada extremo del DTE. De conformidad con UL	2, 5, 12, 40 o 80 m (6,56, 16,4, 39,37, 131,23 o 262,47 pies)
TCSECE3M3M••S4		Cable para entorno severo, equipado con conectores RJ45 en cada extremo. De conformidad con la CE	1, 2, 3, 5 o 10 m (3,28, 6,56, 9,84, 16,4, 32,81 pies)
TCSECU3M3M••S4		Cable para entorno severo, equipado con conectores RJ45 en cada extremo. De conformidad con UL	1, 2, 3, 5 o 10 m (3,28, 6,56, 9,84, 16,4, 32,81 pies)

En caso de que se trate de una red de fibra óptica, puede usar los siguientes cables:

Referencia	Descripción	Detalles	Longitud
490NOC00005	Cable de fibra óptica de vidrio para conexiones DTE	1 conector SC 1 conector MT-RJ	5 m (16,4 pies)
490NOT00005		1 conector ST (BFOC) 1 conector MT-RJ	5 m (16,4 pies)
490NOR00003		2 conectores MT-RJ	3 m (9,8 pies)
490NOR00005		2 conectores MT-RJ	5 m (16,4 pies)

Configuración de dirección IP

Requisitos previos

Cada dispositivo en la red del Modbus TCP IOScanner obtiene su propia dirección IP. Todas las direcciones IP deben ser únicas.

NOTA: Asigne direcciones IP de clase C para la red del Modbus TCP IOScanner.

Configure las direcciones IP en dos etapas:

Etapa	Descripción
1	Utilizando el software de SoMachine, configure el puerto del controlador que admite el Modbus TCP IOScanner: ● Dirección IP ● Máscara de subred ● Pasarela predeterminada **NOTA:** Utilice solo direcciones fijadas. **NOTA:** No utilice el archivo de configuración de Post para direccionar los puertos del controlador.
2	Configura cada dispositivo de esclavo de la red del Modbus TCP IOScanner: ● Dirección IP ● Máscara de subred ● Pasarela predeterminada

Asignación de la dirección de esclavo

En la red del Modbus TCP IOScanner, asigne la dirección IP de los esclavos empleando el siguiente método (dependiendo del tipo de dispositivo):

- A través del servidor DHCP para gestionar todas las direcciones IP de la red del Modbus TCP IOScanner desde el software de SoMachine, o si necesita el servicio FDR.
- A través de un software de terceros o HMI local.
- A través de una configuración de ajustes avanzados en la línea de serie Modbus, mediante la tecnología FDT/DTM; consulte Device Type Manager Guía del usuario.

Servicio FDR

Algunos esclavos admiten el servicio FDR (Sustitución rápida de dispositivo).

El servicio FDR almacena parámetros de red y de funcionamiento de los dispositivos en la red. Si se sustituye un dispositivo, el servicio configura automáticamente el dispositivo de sustitución con parámetros idénticos a los del dispositivo eliminado.

Para configurar el servicio en el esclavo, consulte la documentación relativa al dispositivo esclavo.

Parámetro de dirección IP del maestro esclavo

Algunos esclavos presentan unos parámetros de **direcciones IP maestras**, de manera que solo un controlador maestro declarado tiene acceso al dispositivo esclavo.

Si el dispositivo...	Entonces...
Está configurado para utilizar el Modbus TCP IOScanner	Configure el parámetro de la **dirección IP maestra** en el dispositivo; vea los datos que figuran a continuación.
No está configurado para utilizar el Modbus TCP IOScanner	Utilice `0.0.0.0` para el parámetro de la **dirección IP maestra** en el dispositivo.

El parámetro de la **dirección IP maestra** del esclavo debe establecerse en la dirección IP del controlador que admita el Modbus TCP IOScanner (**puerto Ethernet** 2).

Para configurar este parámetro en el esclavo, consulte la documentación relativa al esclavo.

Parámetro de pasarela de esclavo

El parámetro de pasarela de los esclavos debe establecerse en la dirección IP del puerto Ethernet del controlador que admita el Modbus TCP IOScanner (**Ethernet 2**).

Una herramienta de configuración debe alcanzar los esclavos para establecer sus parámetros.

Si la herramienta de configuración...	Entonces...
Está conectada a la red de nivel superior	Actualice el parámetro de pasarela de esclavo; vea los datos que figuran a continuación.
Está conectada a la red de nivel del dispositivo	No se utiliza el parámetro de pasarela.
Utiliza un protocolo diferente a TCP/IP	No se utiliza el parámetro de pasarela.

Para configurar este parámetro en el esclavo, consulte la documentación relativa al esclavo.

NOTA: Si el servicio DHCP se utiliza para direccionar los esclavos, el parámetro de pasarela se establece en la tabla del controlador DHCP.

Rutas de acceso del PC

El PC que admite la herramienta de configuración debe estar configurado para que pueda comunicarse con los esclavos.

Si el esclavo está configurado...	Entonces...
Como un esclavo predefinido mediante ajustes avanzados (FDT/DTM)	No se necesita una parametrización del equipo específica. **NOTA:** La configuración del equipo no se altera.
Mediante otra herramienta	Actualice la tabla de rutas de acceso del PC; vea los datos que figuran a continuación.

Para actualizar la tabla de rutas de acceso del PC, detenga cualquier conexión desde el PC hacia el controlador y/o hacia otro dispositivo. A continuación, en una línea de comandos de Windows, ejecute el comando `route ADD Destination MASK Subnet_Mask Gateway`.

En la siguiente tabla se muestran los parámetros para actualizar en esta sintaxis de comando:

Parámetro	Valor
Destino	Dirección IP de la red del Modbus TCP IOScanner.
Subnet_Mask	La máscara de subred de la red del Modbus TCP IOScanner.
Pasarela	Dirección del puerto del controlador conectada a la red de nivel superior.

Para verificar estos parámetros en una línea de comando de Windows, ejecute el comando `route PRINT`.

Para eliminar esta ruta del PC, en una línea de comando de Windows, ejecute el comando `route DELETE Destination`, donde `Destination`es la dirección IP de la red del Modbus TCP IOScanner introducida anteriormente.

Pruebas de red

Objetivo

Antes de utilizar el Modbus TCP IOScanner, compruebe la red.

Verifique que:
- La configuración de la dirección de cada dispositivo está conforme a la planificación.
- Cada dispositivo está conectado correctamente.

A continuación, se presentan algunos de los métodos de prueba más habituales.

Indicador LED de estado

Dependiendo de sus dispositivos, verifique que el indicador LED de estado muestra un cableado correcto.

Verificación mediante un ordenador

Con un ordenador, verifique que cada dispositivo esclavo está conectado y direccionado:

Paso	Acción
1	Conecte el ordenador a la red de Modbus TCP IOScanner.
2	Acceda a la línea de comandos.
3	Utilice un comando `ping xxx.xxx.xxx.xxx` para alcanzar cada esclavo. `xxx.xxx.xxx.xxx` = dirección IP del esclavo que se desea probar. **NOTA:** El comando `ping -h` muestra la ayuda para el comando `ping`.

Verificación mediante un servidor web

Con el servidor web del controlador, verifique que el controlador puede comunicarse con cada dispositivo esclavo:

Paso	Acción
1	Acceda al Servidor web del controlador.
2	Abra la página de **Diagnóstico de Ethernet**.
3	Utilice la función *(véase página 52)* **ping remoto** en cada dispositivo esclavo.

Capítulo 3
Configuración de Modbus TCP IOScanner

Contenido de este capítulo

Este capítulo contiene los siguiente apartados:

Apartado	Página
Adición de un esclavo al Modbus TCP IOScanner	26
Configuración de un Modbus TCP IOScanner	27
Configuración de un módulo de E/S distribuidas Advantys OTB en el Modbus TCP IOScanner	29
Configurar un esclavo predefinido en el Modbus TCP IOScanner	32
Configuración de un dispositivo genérico en el Modbus TCP IOScanner	34

Adición de un esclavo al Modbus TCP IOScanner

Descripción general

En esta sección se describe cómo añadir un esclavo al **Modbus TCP IOScanner**.

Estos dispositivos se dividen en tres categorías:
- Dispositivos predefinidos para los dispositivos Modbus de Schneider Electric (ATV, LXM y ZBRN),
- Advantys OTB para el módulo Modbus TCP con E/S configurables,
- Dispositivos genéricos para todos los demás esclavos Modbus TCP.

Para cada dispositivo genérico, debe definir las peticiones Modbus para enviar a este dispositivo añadiendo canales. Un canal se corresponde con una petición Modbus y puede tener su propia velocidad de repetición.

Adición de un esclavo al Modbus TCP IOScanner

Para añadir un esclavo al **Modbus TCP IOScanner**, seleccione el dispositivo elegido en el **Catálogo de hardware**, arrástrelo a **Dispositivos** y suéltelo en el nodo **Ethernet_2** de **Dispositivos**.

Para obtener más información sobre cómo añadir un esclavo al proyecto, consulte:
- Utilización del método de arrastrar y colocar *(véase SoMachine, - Guía de programación)*
- Utilización del menú contextual o botón más *(véase SoMachine, - Guía de programación)*

Configuración de un Modbus TCP IOScanner

Requisitos previos

Antes de configurar el Modbus TCP IOScanner:

- Establece la dirección IP del Ethernet 2 en la **modalidad fija**. Debe ser diferente de `0.0.0.0`.
- Los dispositivos conectados deben estar en la misma subred que el puerto Ethernet 2

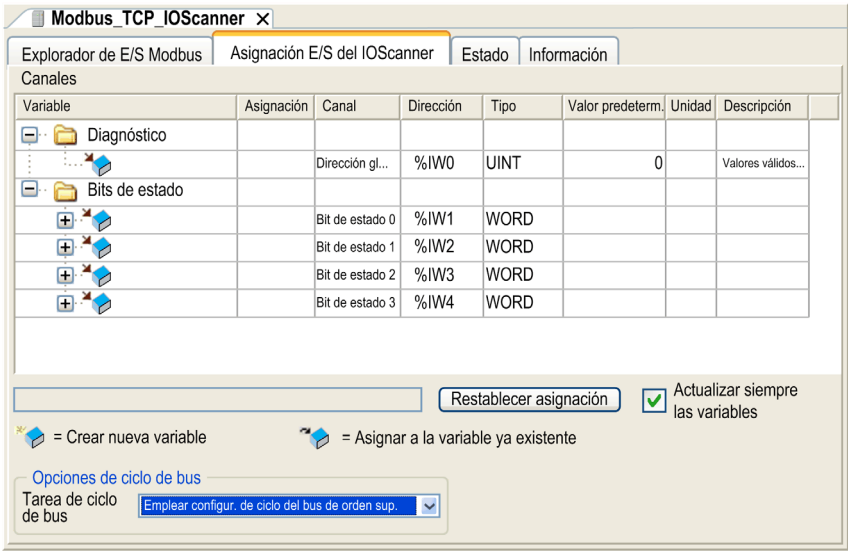
Para obtener más información sobre la dirección IP, consulte Configuración Ethernet *(véase Modicon M251 Logic Controller, Guía de programación)*.

Añadir un Modbus TCP IOScanner

El nodo Modbus TCP IOScanner se agrega automáticamente cuando se añade un esclavo en el nodo **Ethernet 2** *(véase página 26)*.

Configurar un Modbus TCP IOScanner

Para configurar un Modbus TCP IOScanner proceda de la siguiente manera:

Paso	Acción
1	En **Dispositivos**, haga doble clic en **Modbus_TCP_IOScanner**. **Resultado**: Aparece la ventana de configuración.
2	Seleccione la ficha **IOScanner I/O Mapping**. 

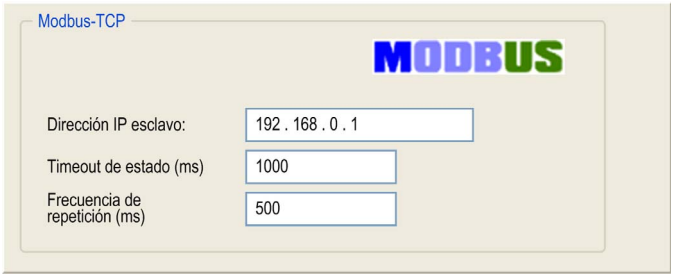
Paso	Acción
3	Seleccione la **Tarea de ciclo de bus** en la lista desplegable: ● **Emplear configuración de ciclo del bus de orden superior**(predeterminado), ● **MAST**, o ● Tarea existente del proyecto. **NOTA:** El parámetro de la **tarea de ciclo de bus** en el editor de asignación E/S del dispositivo que contiene el Modbus TCP IOScanner define la tarea como responsable de actualizar las imágenes E/S ((%QW, %IW). Estas imágenes se corresponden con la solicitud Modbus enviada a los esclavos Modbus y a los bits de estado.

NOTA: Cuando se configura elModbus TCP IOScanner, se ignora el archivo de configuración de Post para la red Ethernet 2

Configuración de un módulo de E/S distribuidas Advantys OTB en el Modbus TCP IOScanner

Configuración de un esclavo OTB

Para configurar el esclavo OTB añadido al **Modbus TCP IOScanner**, siga este procedimiento:

Paso	Acción
1	En **Dispositivos**, haga doble clic en el nodo de dispositivo Advantys OTB. Resultado: aparecerá la ventana de configuración. 
2	En el campo **Dirección IP esclavo:** introduzca la dirección IP asignada al Advantys OTB.
3	Introduzca un valor de **Timeout de estado (ms)** (predeterminado 1000). Este representa el plazo (en ms) entre una solicitud del Modbus TCP IOScanner y una respuesta del esclavo. Cuando el timeout de estado caduca, los valores de bit de estado cambian a 0. Los valores de bit de estado se pueden visualizar en la ficha IOScanner I/O Mapping *(véase página 27)* o a través del servidor Web. El timeout de estado se aplica a todos los canales del esclavo.
4	Introduzca un valor de **Frecuencia de repetición (ms)** (predeterminado 20). El valor de **Timeout de estado (ms)** debe ser mayor que el de la **Frecuencia de repetición (ms)**.
5	Configure las E/S del dispositivo Advantys OTB en la ficha **OTB I/O Configuration**.
6	Añada y configure los módulos de ampliación de TM2 adjuntos al OTB.
7	Llame al `CONFIGURE_OTB` (bloque de funciones) *(véase página 62)* para actualizar la configuración con los datos creados en los pasos anteriores.

NOTA: Las funciones expertas de Advantys OTB, como contadores, contadores rápidos y generadores de pulsos, no pueden utilizarse directamente en el Modbus TCP IOScanner.

Compatibilidad con los módulos TM3

Los módulos TM3·no son compatibles con el Advantys OTB.

Compatibilidad con los módulos TM2

En esta tabla se muestran los módulos TM2 compatibles con el Advantys OTB:

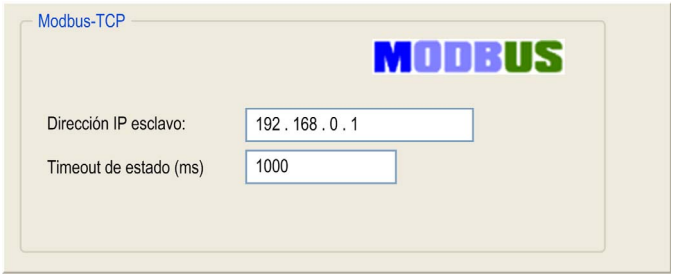
Referencia	Tipo
TM2AMI2HT	2 entradas analógicas
TM2AMI2LT	2 entradas analógicas
TM2AMI4LT	4 entradas analógicas
TM2AMI8HT	8 entradas analógicas
TM2ARI8HT	8 entradas analógicas
TM2ARI8LRJ	8 entradas analógicas
TM2ARI8LT	8 entradas analógicas
TM2AMO1HT	1 salida analógica
TM2AVO2HT	2 salidas analógicas
TM2AMM3HT	2 entradas analógicas 1 salida analógica
TM2AMM6HT	4 entradas analógicas 2 salidas analógicas
TM2ALM3LT	2 entradas analógicas 1 salida analógica
TM2DAI8DT	8 entradas digitales Tipo de señal: tipo CA
TM2DDI8DT	8 entradas digitales Tipo de señal: común positivo/negativo
TM2DDI16DT	16 entradas digitales Tipo de señal: común positivo/negativo
TM2DDI16DK	16 entradas digitales Tipo de señal: común positivo/negativo
TM2DDI32DK	32 entradas digitales Tipo de señal: común positivo/negativo
TM2DRA8RT	8 contactos en 1 línea común Tipo de salida: relé (contactos NO)
TM2DRA16RT	16 contactos en dos líneas comunes Tipo de salida: relé (contactos NO)
TM2DDO8UT	8 salidas de transistor en 1 línea común Tipo de señal: común negativo
TM2DDO8TT	8 salidas de transistor en 1 línea común Tipo de señal: común positivo
TM2DDO16UK	16 salidas de transistor en 1 línea común Tipo de señal: común negativo

Referencia	Tipo
TM2DDO16TK	16 salidas de transistor en 1 línea común Tipo de señal: común positivo
TM2DDO32UK	32 salidas de transistor en 2 líneas comunes Tipo de señal: común negativo
TM2DDO32TK	32 salidas de transistor en 2 líneas comunes Tipo de señal: común positivo
TM2DMM8DRT	4 entradas digitales Tipo de señal: común positivo/negativo 1 línea común con 4 contactos Tipo de salida: relé (contactos NO)
TM2DMM24DRF	16 entradas digitales Tipo de señal: común positivo/negativo 2 líneas comunes con 8 contactos cada una Tipo de salida: relé (contactos NO)

Configurar un esclavo predefinido en el Modbus TCP IOScanner

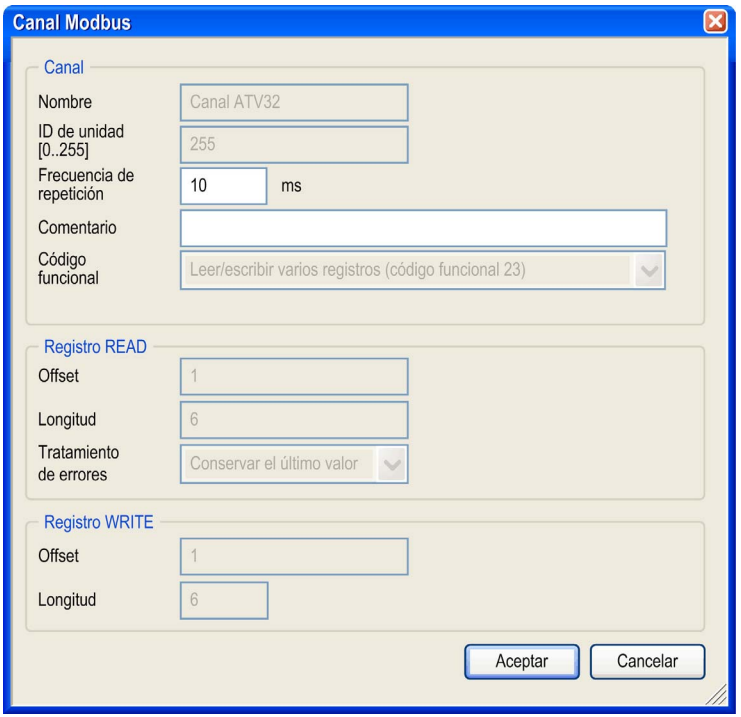
Configuración de un esclavo predefinido añadido al Modbus TCP IOScanner

Para configurar el esclavo predefinido añadido al Modbus TCP IOScanner, siga este procedimiento:

Paso	Acción
1	En **Dispositivos**, haga doble clic sobre el nodo esclavo añadido. Resultado: aparecerá la ventana de configuración. 
2	En **Dirección IP esclavo**, introduzca la dirección IP del esclavo Modbus.
3	Introduzca un valor de **Timeout de estado (ms)** (predeterminado 1000). Este representa el plazo (en ms) entre una solicitud del Modbus TCP IOScanner y una respuesta del esclavo. Cuando el timeout de estado caduca, los valores de bit de estado cambian a 0. Los valores de bit de estado se pueden visualizar en la ficha IOScanner I/O Mapping *(véase página 27)* o a través del servidor Web. El timeout de estado se aplica a todos los canales del esclavo.
4	Para los dispositivos con configuración avanzada, es posible que se soliciten algunos ajustes adicionales. Consulte la Device Type Manager Guía del usuario

Editar canales Modbus TCP

Para editar los parámetros del Canal Modbus para un esclavo predefinido, siga este procedimiento:

Paso	Acción
1	En **Dispositivos**, haga doble clic sobre el nodo esclavo añadido.
2	Seleccione la ficha **Modbus TCP Channel Configuration** y haga clic en el botón **Editar...**. Resultado: se muestra la ventana del **Canal Modbus**. 
3	Introduzca la **Frecuencia de repetición** para el canal. La frecuencia de repetición es el intervalo de consulta de las solicitudes Modbus.
4	Puede introducir un **Comentario** para el canal.
5	Haga clic en **Aceptar**.

Configuración de un dispositivo genérico en el Modbus TCP IOScanner

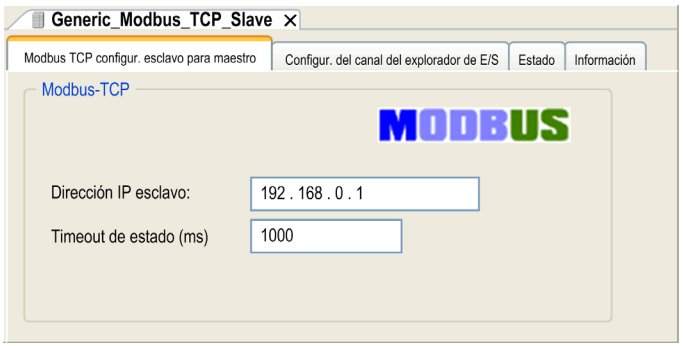
Descripción general

Para configurar un dispositivo genérico añadido al Modbus TCP IOScanner, complete los parámetros de las dos fichas siguientes:

- **Modbus TCP Slave Configuration**
- **IO Scanner Channel Configuration**

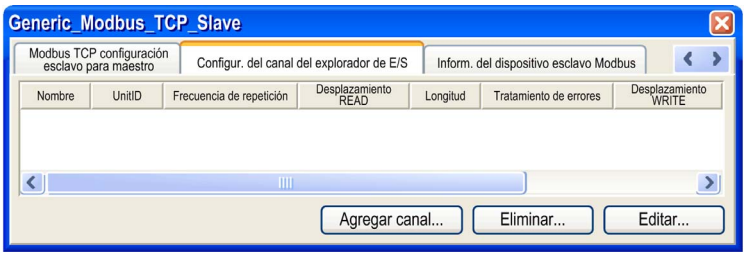
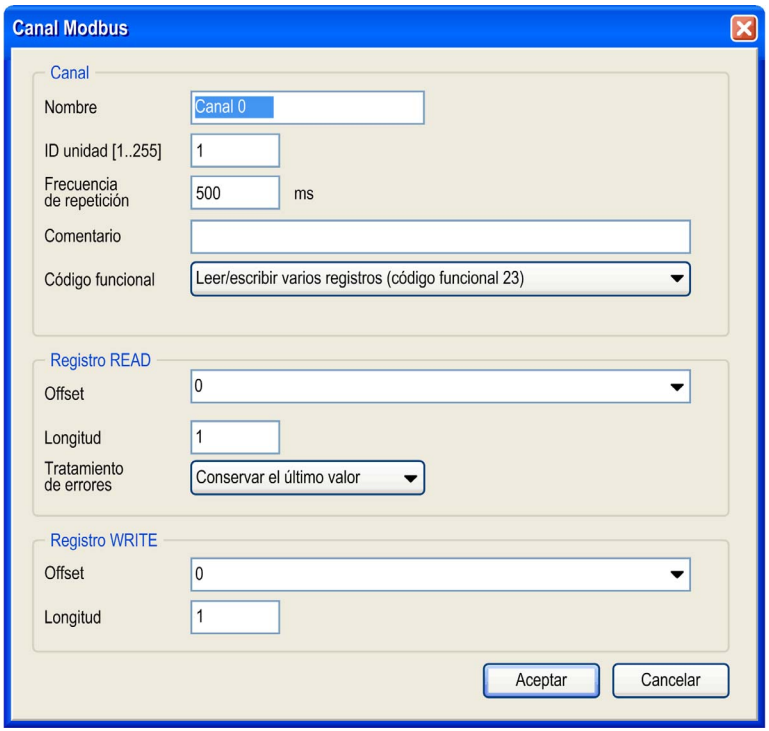
Ficha Modbus TCP Slave Configuration

Para configurar los parámetros de la ficha **Modbus TCP Slave Configuration**, proceda de la siguiente manera:

Paso	Acción
1	En **Dispositivos**, haga doble clic en **Generic_Modbus_TCP_Slave**. Resultado: Aparecerá la ventana de configuración. 
2	Introduzca un valor de **Dirección IP esclavo** (predeterminado 192.168.0.1).
3	Introduzca un valor de **Timeout de estado (ms)** (predeterminado 1000). Este representa el plazo (en ms) entre una solicitud del Modbus TCP IOScanner y una respuesta del esclavo. Cuando el timeout de estado caduca, los valores de bit de estado cambian a 0. Los valores de bit de estado pueden visualizarse en la ficha Asignación E/S de IOScanner *(véase página 27)*.

Ficha IO Scanner Channel Configuration

Para configurar los parámetros en la ficha **IO Scanner Channel Configuration**, proceda de la siguiente manera:

Paso	Acción
1	Haga clic en la ficha **IO Scanner Channel Configuration**: 
2	Para eliminar un canal, selecciónelo y haga clic en **Eliminar**.
3	Para modificar los parámetros de un canal, selecciónelo y haga clic en **Editar**.
4	Para agregar un canal, haga clic en **Agregar canal**. Se muestra este cuadro de diálogo: 

Paso	Acción

5	En el área del **Canal**, puede definir:
	● **Nombre**: cadena opcional para dar nombre al canal
	● **ID de unidad [1..255]**: ID de unidad del dispositivo esclavo Modbus TCP (predeterminado 255). Ver nota.
	● **Frecuencia de repetición**: intervalo de consulta de la petición Modbus (predeterminado 20 ms)
	● **Comentario**: campo opcional para describir el canal
	● **Código funcional**: tipo de petición Modbus:
	  ● **Leer/escribir varios registros (código funcional 23)**(predeterminada)
	  ● **Lectura de registros de mantenimiento (código funcional 03)**
	  ● **Escritura de registros múltiples (código funcional 16)**
	
	En el área del **registro READ**, puede definir:
	● **Offset**: número de registro de inicio para leer de 0 a 65.535
	● **Longitud**: número de los registros que se van a leer (según el código funcional).
	● **Tratamiento de errores**: define el valor de retorno en el caso de una interrupción de la comunicación:
	  ● **Conservar el último valor** (predeterminado) mantiene el último valor válido
	  ● **Poner a cero** restablece todos los valores a 0
	
	En el área del **registro WRITE**, puede definir:
	● **Offset**: número de registro de inicio para escribir de 0 a 65.535
	● **Longitud**: número de los registros que se van a escribir (según el código funcional).

6	Haga clic en **Aceptar** para validar la configuración del canal.
7	Repita los pasos del 4 al 6 para crear otros canales que definan la comunicación Modbus con el dispositivo. Para cada petición Modbus, debe crear un canal.

NOTA: El identificador de unidad se emplea con los dispositivos Modbus TCP compuestos de varios dispositivos Modbus, por ejemplo, desde Modbus TCP hasta Modbuspasarelas RTU. En tal caso, el identificador de unidad permite alcanzar la dirección de esclavo del dispositivo detrás de la pasarela. De forma predeterminada, los dispositivos Modbus/compatibles con TCP ignoran el parámetro del identificador de la unidad.

Capítulo 4
Operación Modbus TCP IOScanner

Contenido de este capítulo

Este capítulo contiene los siguiente apartados:

Apartado	Página
Modbus TCP IOScannerVerificación de recurso	38
Modbus TCP IOScanner Modalidades de funcionamiento	40
Interfaz de aplicación	44

Modbus TCP IOScannerVerificación de recurso

Objetivo

Los recursos de la ficha **Recursos del IOScanner TCP Modbus** permiten estimar la carga en la funcionalidad del Modbus TCP IOScanner. Verifique la carga antes de utilizar la máquina.

Para gestionar la carga, puede manipular uno o varios de los siguientes factores de carga:
- número de esclavos
- número de canales
- frecuencia de repetición

Estimación de la carga

Esta ecuación permite estimar la carga en el componente del Modbus TCP IOScanner.

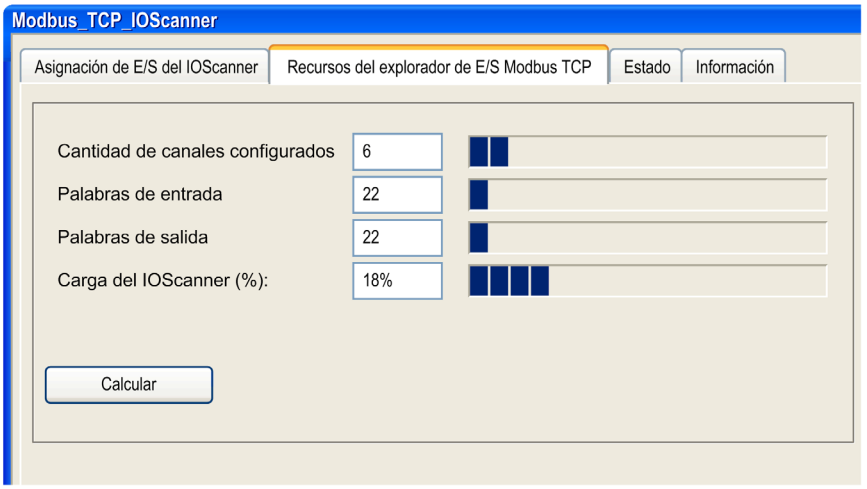
$$\text{Carga del IOScanner (\%)} = \sum_{\text{canal} = 1}^{\text{N.º de canales}} \frac{50}{\text{Frecuencia de repetición}_{\text{canal}}}$$

En el software de SoMachine, un cálculo automático de la carga está disponible:

Paso	Acción
1	En **Dispositivos**, haga doble clic en el nodo **Modbus_:TCP_IOScanner**.
2	Seleccione la ficha **Recursos del IOScanner TCP Modbus**.
3	Hacer clic en **Calcular**.

Descripción

La siguiente imagen muestra la ficha **Recursos del IOScanner TCP Modbus**:



NOTA: La carga debe ser inferior a 100 %

Modbus TCP IOScanner Modalidades de funcionamiento

Estados de Modbus TCP IOScanner

El estado del Modbus TCP IOScanner define el comportamiento de los distintos dispositivos en la red del Modbus TCP IOScanner. Para cada estado, la supervisión de la información (bit de estado, estados de comunicación, etc.) es específica.

El estado del Modbus TCP IOScanner depende del estado del controlador:

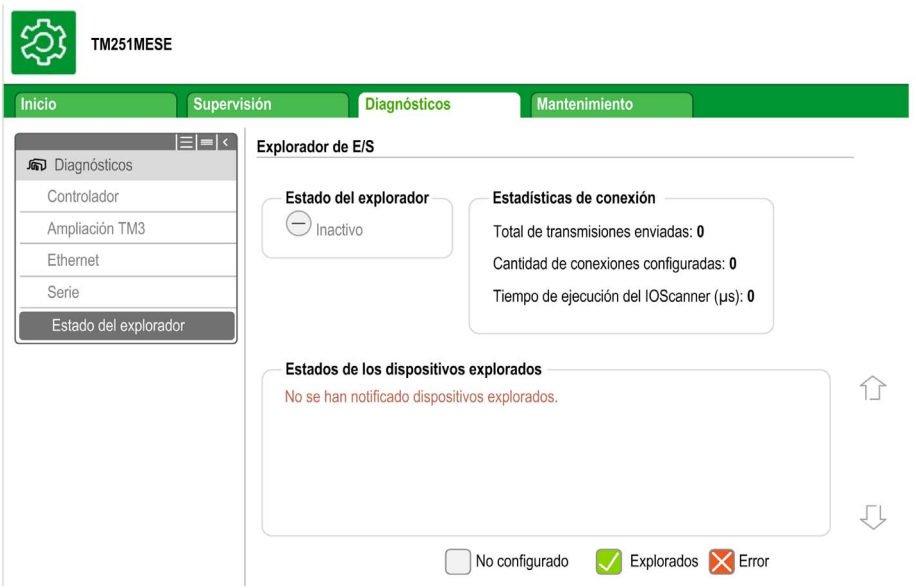
Estado del controlador	Estado del Modbus TCP IOScanner
EMPTY	IDLE
STOPPED	STOPPED
HALT	STOPPED
RUNNING	OPERATIONAL
RUNNING with breakpoint	OPERATIONAL con un comportamiento específico

Estado EMPTY del controlador

Las conexiones TCP/IP están cerradas.

Los estados del dispositivo esclavo se gestionan de acuerdo con su modalidad de funcionamiento individual. En el estado EMPTY, el Modbus TCP IOScanner no se ha creado. Por tanto, ni los bits de estado ni las imágenes de E/S están disponibles.

En esta imagen se muestra la página del servidor de web en ese estado:



Estado STOPPED del controlador

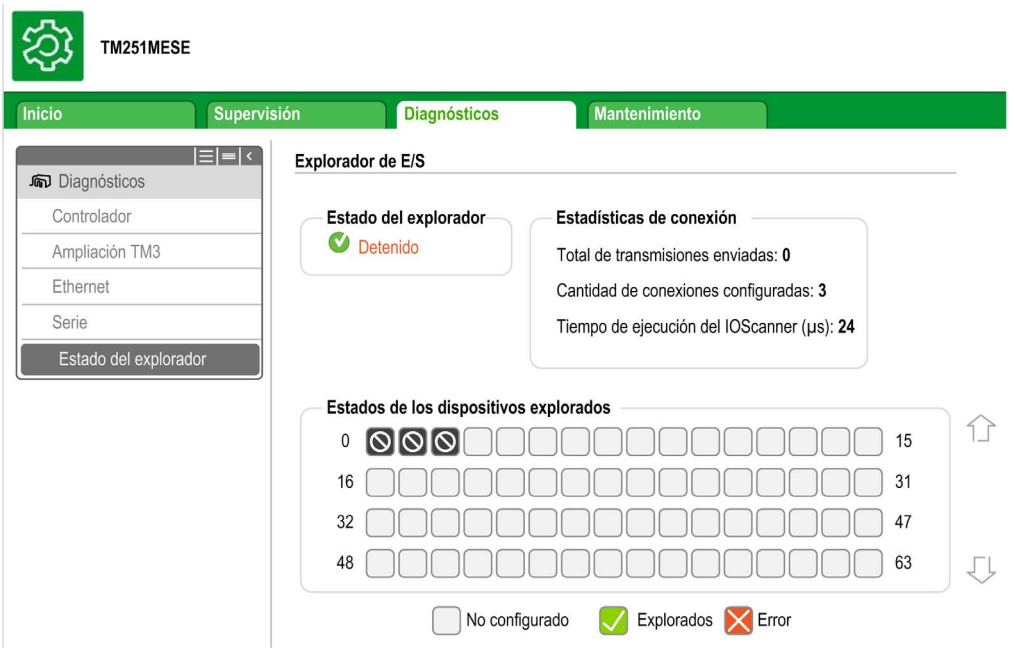
Las conexiones TCP/IP están cerradas. Cuando el Modbus TCP IOScanner cambia del estado OPERATIONAL al estado STOPPED, todas las conexiones con los esclavos se cierran en la modalidad unilateral.

Los dispositivos esclavos se gestionan de acuerdo con su modalidad de funcionamiento individual.

En la siguiente tabla se muestran las variables de SoMachine:

Variable	Valor	Comentarios
Valor de bit de estado	0	-
Imagen de entrada	0 o el último valor leído	Los valores de entrada dependen del parámetro **Tratamiento de errores**. Los valores de entrada son aquellos que se producen cuando el controlador cambia al estado STOPPED (DETENIDO), por lo que puede no reflejar el estado actual de la entrada a partir de ese momento.

En esta imagen se muestra la página del servidor de web en ese estado:



Estado HALT del controlador

Presenta el mismo comportamiento que el del estado del controlador STOPPED.

Estado RUNNING del controlador

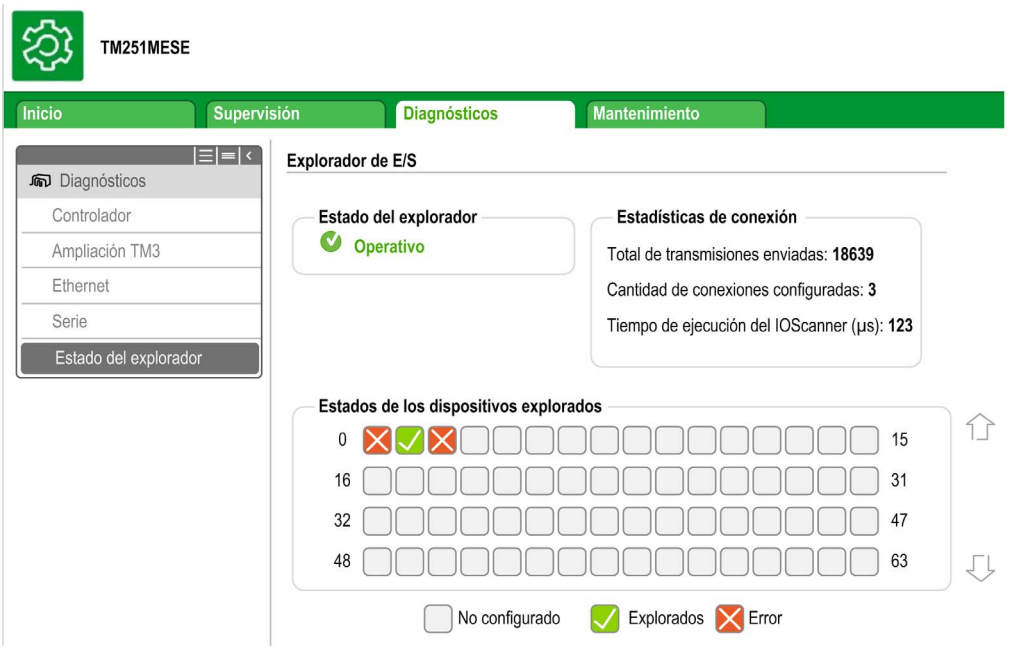
Las conexiones TCP/IP están abiertas.

Los dispositivos esclavos se gestionan mediante el controlador.

En la siguiente tabla se muestran las variables de SoMachine:

Variable	Valor	Comentarios
Valor de bit de estado	De 0 a 1	0: No hay respuesta por parte del esclavo antes de que el timeout se agote. 1: Las peticiones se envían y se responden antes de que el timeout se agote.
Imagen de entrada	Último valor leído	Los valores se actualizan de manera sincronizada en tarea *(véase página 44)* que ejecuta el Modbus TCP IOScanner.

En esta imagen se muestra la página del servidor de web en ese estado:



Controlador RUNNING con estado de punto de interrupción

Las conexiones TCP/IP están abiertas.

Los dispositivos esclavos se gestionan mediante el controlador.

En la siguiente tabla se muestran las variables de SoMachine:

Variable	Valor	Comentarios
Valor de bit de estado	1	-
Imagen de entrada	Último valor leído	Los valores de entrada son aquellos que se producen cuando el controlador introduce el estado RUNNING con punto de interrupción, por lo que puede no reflejar el estado actual de la entrada a partir de ese momento

Interfaz de aplicación

Descripción general

La interfaz de aplicación es un conjunto de funciones y variables que habilitan la comunicación entre la aplicación y el Modbus TCP IOScanner:

- Tarea de ciclo de bus
- Variables de estado
- Variables de imagen de E/S
- Bloques de funciones

Tarea de ciclo de bus

El Modbus TCP IOScanner y la aplicación intercambian datos en cada ciclo de una tarea de aplicación.

El parámetro **Tarea de ciclo de bus** le permite seleccionar la tarea de aplicación que gestiona el Modbus TCP IOScanner:

- **Emplear configuración de ciclo del bus de orden superior**: asocia el Modbus TCP IOScanner con la tarea de aplicación que gestiona al controlador.
- **MAST**: asocia el Modbus TCP IOScanner con la tarea MAST.
- Otra tarea existente: puede crear una tarea y asociarla con el Modbus TCP IOScanner.

Para obtener mayor información acerca de las tareas de aplicación, consulte SoMachine Guía de programación.

Variables de estado

Hay dos tipos de variables de estado:

- **Bits de estado**: son variables que indican el estado de comunicación de los canales. Hay un bit de estado por canal.
- **Estado del explorador global**: es una variable que indica el estado del Modbus TCP IOScanner.

En esta tabla se muestran los valores de bit de estado:

Valor de bit de estado	Estado de comunicación del canal
0	El timeout de estado expira sin recibir una respuesta.
1	No se han detectado errores. Se ha recibido la petición y la respuesta.

Variables de imagen de E/S

El Modbus TCP IOScanner recopila y escribe datos desde/hacia los dispositivos esclavos. A estas variables se les llama imágenes de E/S.

Direcciones de variables

Cada variable posee su propia dirección:

Variable	Tipo	Cantidad
Imagen de E/S	**%IW** para las entradas **%QW** para las salidas	Se crea una matriz de palabras para cada canal.
Bit de estado	**%IW**	Cuatro palabras consecutivas
Estado del explorador global	**%IW**	Una palabra

La asignación de E/S *(véase página 49)* organiza las variables:

- Verificando las direcciones
- Asignando nombres a las variables (asignación)

Bloques de función para controlar el Modbus TCP IOScanner

La aplicación usa varios bloques de funciones *(véase página 57)* para comunicarse con el controlador y los esclavos:

- `CONFIGURE_OTB`
- `IOS_GETSTATE`
- `IOS_START`
- `IOS_GETHEALTH`
- `IOS_STOP`

Bloques de función para controlar el ATV y los dispositivos Lexium

Utilice los bloques de función PLC Open para controlar el ATV y los dispositivos Lexium. Se puede acceder a estos bloques de funciones desde la biblioteca Altivar del Modbus TCP. Para obtener más información, consulte ATV Modbus TCP Function blocks Library Guide y LXM Modbus TCP Function blocks Library Guide.

Capítulo 5
Mantenimiento de Modbus TCP IOScanner

Contenido de este capítulo

Este capítulo contiene los siguiente apartados:

Apartado	Página
Diagnósticos: SoMachine modalidad online	48
Diagnósticos: servidor web	52
Resolución de problemas	54

Diagnósticos: SoMachine modalidad online

Descripción general

En la modalidad online, puede supervisar el Modbus TCP IOScanner en SoMachine usando los siguientes métodos:

- Los iconos en **Dispositivos**
- La ficha de estado del Modbus TCP IOScanner y los diferentes dispositivos esclavos
- La ficha de asignación de E/S del Modbus TCP IOScanner
- Las fichas de asignación E/S de los dispositivos esclavos
- Ficha de recursos del Modbus TCP IOScanner

Dispositivos

El estado de comunicación de Modbus TCP IOScanner y los esclavos se presentan mediante iconos en **Dispositivos**:

Icono	Significado
	La comunicación con el dispositivo es correcta. **NOTA:** Modbus TCP IOScanner siempre se presenta con este icono.
	El controlador no puede comunicarse con el dispositivo. **NOTA:** Cuando el Modbus TCP IOScanner está en STOPPED, todos los dispositivos se presentan con este icono.

Asignación de Modbus TCP IOScanner

Asignación de E/S del IOScanner | Recursos del explorador de E/S Modbus TCP | Información

Canales

Variable	Asignación	Canal	Dirección	Tipo	Predet...	Actua...	Prepar...
Diagnóstico							
		Direcc. gl...	%I...	UINT	0	2	
Bits de estado							
		Bit de est...	%I...	WORD		63	
Healthbits_OTB1EODM9LP		Bit 0	%IX...	BOOL	FALSE	TRUE	
Healthbits_Altivar32		Bit 1	%IX...	BOOL	FALSE	TRUE	
Healthbits_Lexium32M		Bit 2	%IX...	BOOL	FALSE	TRUE	
Healthbits_Generic_Slave_channel3		Bit 3	%IX...	BOOL	FALSE	TRUE	
Healthbits_Generic_Slave_channel4		Bit 4	%IX...	BOOL	FALSE	TRUE	
Healthbits_Generic_Slave_channel5		Bit 5	%IX...	BOOL	FALSE	TRUE	
		Bit 6	%IX...	BOOL	FALSE	FALSE	
		Bit 7	%IX...	BOOL	FALSE	FALSE	
		Bit 8	%IX...	BOOL	FALSE	FALSE	
		Bit 9	%IX...	BOOL	FALSE	FALSE	
		Bit 10	%IX...	BOOL	FALSE	FALSE	
		Bit 11	%IX...	BOOL	FALSE	FALSE	
		Bit 12	%IX...	BOOL	FALSE	FALSE	
		Bit 13	%IX...	BOOL	FALSE	FALSE	
		Bit 14	%IX...	BOOL	FALSE	FALSE	
		Bit 15	%IX...	BOOL	FALSE	FALSE	
		Bit de est...	%I...	WORD		0	
		Bit de est...	%I...	WORD		0	
		Bit de est...	%I...	WORD		0	

Columna		Uso	Comentario
Variable	**Diagnóstico**	Asigna un nombre a la variable de estado del explorador global.	–
	Bits de estado	Asigna un nombre a cada uno de los bits de estado. Por ejemplo, nombra a un bit de estado con el nombre del dispositivo asociado.	Los bits de estado se agrupan en 4 subcarpetas de 16 bits.

Columna	Uso	Comentario
Dirección	Recupera la dirección de cada variable.	Las direcciones se deben modificar cuando se cambie la configuración.
Valor actual	Supervisa la red de Modbus TCP IOScanner	Para valores booleanos (bit de estado): ● **TRUE** = 1 ● **FALSE** = 0

Asignación de esclavo

En esta figura se muestra el ejemplo de una ficha de asignación de E/S para un dispositivo esclavo Advantys OTB:

Modbus TCP configuración esclavo | Configuración de E/S de OTB | Asignación de E/S del ModbusIOScanner | Información

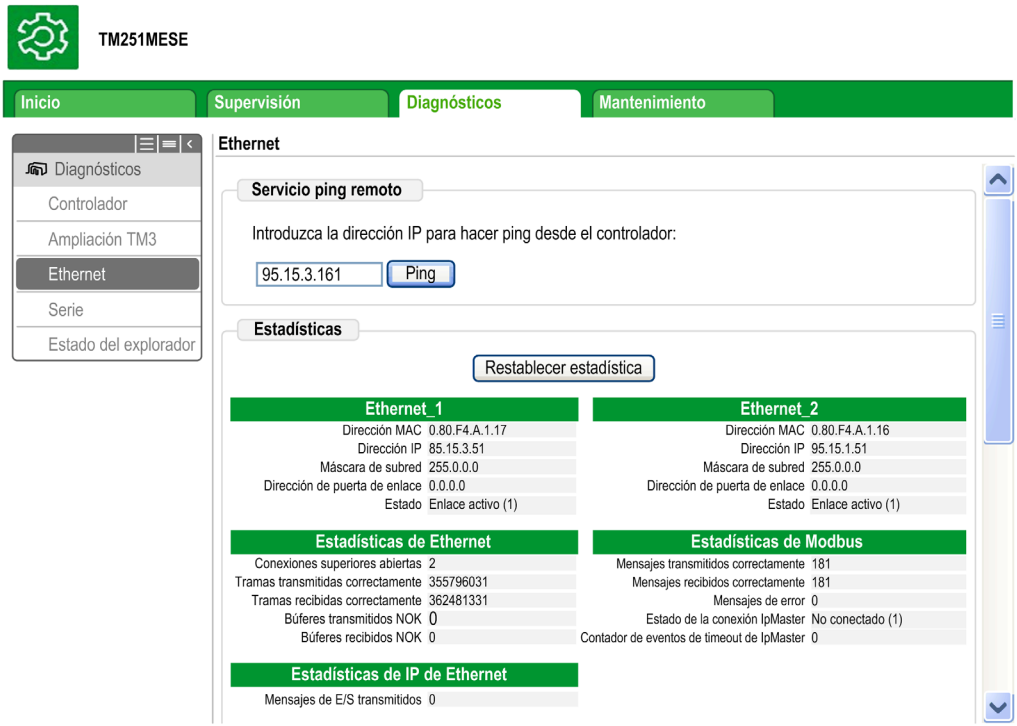
Canales

Variable	Asignac.	Canal	Dirección	Tipo	Valor predeterm.	Valor actual
Entradas						
iwOTB1EODM9LP_Read_Inputs		Leer entradas	%IW18	WORD		2049
		Bit 0	%IX3...	BOOL	FALSE	TRUE
		Bit 1	%IX3...	BOOL	FALSE	FALSE
		Bit 2	%IX3...	BOOL	FALSE	FALSE
		Bit 3	%IX3...	BOOL	FALSE	FALSE
		Bit 4	%IX3...	BOOL	FALSE	FALSE
		Bit 5	%IX3...	BOOL	FALSE	FALSE
		Bit 6	%IX3...	BOOL	FALSE	FALSE
		Bit 7	%IX3...	BOOL	FALSE	FALSE
		Bit 8	%IX3...	BOOL	FALSE	FALSE
		Bit 9	%IX3...	BOOL	FALSE	FALSE
		Bit 10	%IX3...	BOOL	FALSE	FALSE
		Bit 11	%IX3...	BOOL	FALSE	TRUE
Salidas						
qwOTB1EODM9LP_Output_commands		Config. de...	%QW1...	WORD		255
		Bit 0	%QX2.0	BOOL	FALSE	TRUE
		Bit 1	%QX2.1	BOOL	FALSE	TRUE
		Bit 2	%QX2.2	BOOL	FALSE	TRUE
		Bit 3	%QX2.3	BOOL	FALSE	TRUE
		Bit 4	%QX2.4	BOOL	FALSE	TRUE
		Bit 5	%QX2.5	BOOL	FALSE	TRUE
		Bit 6	%QX2.6	BOOL	FALSE	TRUE
		Bit 7	%QX2.7	BOOL	FALSE	TRUE

Columna		Uso	Comentario
Variable	**Entradas**	Asigna un nombre para cada entrada del dispositivo.	Asimismo, todos los bits se pueden asignar.
	Salidas	Asigna un nombre para cada salida del dispositivo.	
Dirección		Recupera la dirección de cada variable.	Las direcciones se deben modificar cuando se cambie la configuración.
Valor actual		Sigue el valor de las entradas del dispositivo en tiempo real.	Para valores booleanos (bit de estado): ● **TRUE** = 1 ● **FALSE** = 0

Diagnósticos: servidor web

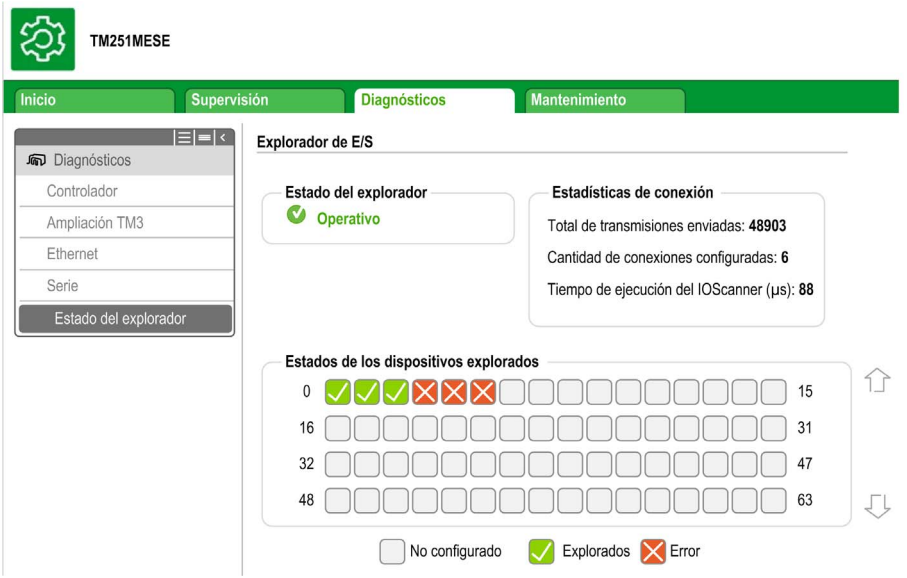
Página Ethernet



En la siguiente tabla se muestra el resultado de la prueba de ping en la página **Ethernet**.

Icono	Significado
	La prueba de comunicación es correcta.
	El controlador no puede comunicarse con la dirección IP definida.

Página de estado del explorador



En la siguiente tabla se muestran los diferentes estados de los canales presentados en la página de **estado del explorador**:

Icono	Valor de bit de estado	Significado
	1	La petición y la respuesta se llevan a cabo a la vez.
	0	Se ha detectado un error, las comunicaciones se han cerrado.
	–	El ID no corresponde a un canal configurado.
	0	El Modbus TCP IOScanner se ha detenido, las comunicaciones se han cerrado.

NOTA: Haciendo clic en un icono, se abre el sitio web del esclavo (si lo hubiera). Para acceder a este sitio web, el equipo debe alcanzar el esclavo. Para obtener más información, consulte las rutas de acceso del PC *(véase página 21)*.

Resolución de problemas

Problemas principales

Síntomas	Causas posibles	Resolución
El Modbus TCP IOScannerse presenta con un triángulo rojo en **Dispositivos**.	La configuración no es compatible con la versión del controlador.	● Limpiar ● Reconstruir todo ● Asegúrese de que el controlador cuenta con la última versión del firmware.
Un esclavo se presenta con un triángulo rojo en **Dispositivos**.	El controlador no puede comunicarse con el esclavo.	● Verifique el cableado del esclavo y el arranque. ● Verifique la dirección IP (utilizando el Servicio ping remoto *(véase página 52)* en la dirección IP del dispositivo esclavo). ● Verifique si el esclavo es compatible con la petición de lectura/escritura. ● Verifique si los registros a los que se ha accedido son relevantes para ese esclavo en cuestión. ● Verifique si los registros a los que se ha accedido están protegidos contra la escritura. ● Verifique que el servicio FDR (sustitución rápida de dispositivo) está configurado correctamente en el esclavo. ● Verifique que el parámetro de la **dirección IP maestra** está configurado correctamente en el esclavo.
Un esclavo/canal se muestra en rojo **temporalmente**.	El cableado es inestable.	Verifique el cableado.
	La configuración requiere algún ajuste.	● Aumente el valor de timeout de estado. ● Aumente el valor de frecuencia de repetición.
	La carga es demasiado importante para el Modbus TCP IOScanner.	Verifique la ficha **Recursos del IOScanner TCP Modbus**.
Algunos de los estados del esclavo no están presentes en la aplicación.	La frecuencia de repetición es demasiado lenta (el valor es demasiado alto).	Disminuya el valor de la frecuencia de repetición para los canales asociados a ese esclavo.
	La tarea de ciclo de bus no es lo suficientemente rápida.	● Asocie otra tarea al Modbus TCP IOScanner. ● Disminuya el valor de ciclo de la tarea asociada.

Apéndices

Contenido de este anexo

Este anexo contiene los siguientes capítulos:

Capítulo	Nombre del capítulo	Página
A	Funciones de Modbus TCP IOScanner	57
B	Tipos de datos de Modbus TCP IOScanner	65
C	Representación de funciones y de bloques de funciones	69

Apéndice A
Funciones de Modbus TCP IOScanner

Descripción general

En este capítulo se explican las funciones incluidas en la biblioteca de `ModbusTCPIOScanner`

Contenido de este capítulo

Este capítulo contiene los siguiente apartados:

Apartado	Página
IOS_GETSTATE: leer el estado de Modbus TCP IOScanner	58
IOS_START: ejecutar Modbus TCP IOScanner	59
IOS_GETHEALTH: leer el valor de bit de estado	60
IOS_STOP: detener el Modbus TCP IOScanner	61
CONFIGURE_OTB: enviar la configuración del software del Advantys OTB	62

IOS_GETSTATE: leer el estado de Modbus TCP IOScanner

Descripción de la función

Esta función devuelve el valor correspondiente al estado del Modbus TCP IOScanner.

Representación gráfica

Representación IL y ST

Para ver la representación general en lenguaje IL o ST, consulte el capítulo Representación de funciones y de bloques de funciones *(véase página 69)*

Descripción de variables de E/S

En la tabla se describe la variable de salida:

Salida	Tipo	Comentario
`IOS_GETSTATE`	IosStateCodes *(véase página 66)*	Valores de retorno: `IosStateCodes enum`

Ejemplo

Este es un ejemplo de llamada de esta función:

```
mystate := IOS_GETSTATE() ; (* 0=NOT CONFIGURED 2=OPERATIONAL or
3=STOPPED. *)
```

IOS_START: ejecutar Modbus TCP IOScanner

Descripción de la función

Esta función inicia el Modbus TCP IOScanner.

Permite el control del tiempo de ejecución de la ejecución del Modbus TCP IOScanner. De forma predeterminada, el Modbus TCP IOScanner se inicia como la aplicación.

Esta llamada a función espera a que el Modbus TCP IOScanner se inicie físicamente, así que puede durar hasta 5 ms.

Iniciar un Modbus TCP IOScanner que ya se ha iniciado no tiene ningún efecto.

Representación gráfica

Representación IL y ST

Para ver la representación general en lenguaje IL o ST, consulte el capítulo Representación de funciones y de bloques de funciones *(véase página 69)*.

Descripción de variables de E/S

En la tabla se describe la variable de salida:

Salida	Tipo	Comentario
`IOS_START`	UDINT	0 = inicio correcto Otro valor = inicio incorrecto

Ejemplo

Este es un ejemplo de llamada de esta función:

```
rc := IOS_START() ;
IF rc <> 0 THEN (* Abnormal situation to be processed at application level
*)
```

IOS_GETHEALTH: leer el valor de bit de estado

Descripción de la función

Esta función devuelve el valor de bit de estado de un canal específico.

Representación gráfica

Representación IL y ST

Para ver la representación general en lenguaje IL o ST, consulte el capítulo Representación de funciones y de bloques de funciones *(véase página 69)*.

Descripción de variables de E/S

En esta tabla se describe la variable de entrada:

Entrada	Tipo	Comentario
`channelID`	UINT	ID de canal *(véase página 15)* del canal que se supervisará.

En esta tabla se describe la variable de salida:

Salida	Tipo	Comentario
`IOS_GETHEALTH`	UINT	0: los valores de E/S del canal no están actualizados 1: los valores de E/S del canal están actualizados

Ejemplo

Este es un ejemplo de llamada de esta función:

```
chID:=1 ;
channelHealth := IOS_GETHEALTH(chID)(* Get the health value (1=OK, 0=Not
OK) of the channel number chID. The channel ID is displayed in the
configuration editor of the device *)
```

IOS_STOP: detener el Modbus TCP IOScanner

Descripción de la función

Esta función inicia el Modbus TCP IOScanner.

Permite el control del tiempo de ejecución de la ejecución del Modbus TCP IOScanner. De forma predeterminada, el Modbus TCP IOScanner se detiene cuando el controlador está en `STOPPED`.

El Modbus TCP IOScanner debe detenerse en el primer ciclo de aplicación, ya que esta función es una llamada sincrónica y, por lo tanto, puede durar algún tiempo.

Esta llamada de función puede durar unos 5 ms mientras espera que el Modbus TCP IOScanner se detenga físicamente.

Detener un Modbus TCP IOScanner que ya se ha detenido no tiene ningún efecto.

Representación gráfica

Representación IL y ST

Para ver la representación general en lenguaje IL o ST, consulte el capítulo Representación de funciones y de bloques de funciones *(véase página 69)*.

Descripción de variables de E/S

En la tabla se describe la variable de salida:

Salida	Tipo	Comentario
`IOS_STOP`	UDINT	0 = detención correcta Otro valor = detención incorrecta

Ejemplo

Este es un ejemplo de llamada de esta función:

```
rc := IOS_STOP() ;
IF rc <> 0 THEN (* Abnormal situation to be processed at application level
*)
```

CONFIGURE_OTB: enviar la configuración del software del Advantys OTB

Descripción de la función

Esta función envía los datos de configuración del SoMachine de un Advantys OTB al dispositivo físico a través del Modbus TCP.

Permite la actualización de los parámetros de configuración de una isla de E/S sin un software de terceros.

El Modbus TCP IOScanner debe detenerse antes de llamar a esta función.

La ejecución de este bloque es asíncrona. Para verificar que se complete la configuración, los indicadores de salida `Done`, `Busy` y `Error` se deben comprobar en cada ciclo de aplicación.

Representación gráfica

Representación IL y ST

Para ver la representación general en lenguaje IL o ST, consulte el capítulo Representación de funciones y de bloques de funciones *(véase página 69)*.

Descripción de variables de E/S

En esta tabla se describen las variables de entrada:

Entrada	Tipo	Comentario
`Execute`	BOOL	Entrada de activación. Inicia la configuración en el flanco ascendente.
`sAddr`	STRING	OTB Dirección IP. El formato de la cadena debe ser `3{xx.xx.xx.xx}`

En esta tabla se describen las variables de salida:

Salida	Tipo	Comentario
`Done`	BOOL	Se establece en TRUE cuando se completa correctamente la configuración.
`Busy`	BOOL	Se establece en TRUE cuando la configuración está en curso.
`Error`	BOOL	Se establece en TRUE cuando la configuración finaliza con la detección de un error.
`ConfError`	configurationOTBError-Codes *(véase página 68)*	Valores de retorno: `configurationOTBErrorCodes`
`CommError`	CommunicationError-Codes *(véase página 67)*	Valores de retorno: `CommunicationErrorCodes`

Ejemplo

Este es un ejemplo de llamada de esta función:

```
VAR
```

(*Bloque de funciones para configurar OTB , es necesario detener el IOscanner antes de la ejecución del FB*)

```
configure_OTB1: CONFIGURE_OTB;
```

(*Valor inicial diferente de 16#00000000 , IO_start_done=0 cuando hay un inicio correcto*)

```
IO_start_done: UDINT := 1000;
```

(*Valor inicial diferente de 16#FFFFFFFF , IO_start_done=16#FFFFFFFF cuando hay una detención correcta*)

```
IO_stop_done: UDINT := 1000;
```

(*Configure_OTB_done= true cuando se configura correctamente el OTB, entonces se puede iniciar el IOScanner*)

```
Configure_OTB_done: BOOL;
myBusy: BOOL;
myError: BOOL;
myConfError: configurationOTBErrorCodes;
myCommError: UINT;
myExecute: BOOL;
END_VAR
```

(*Primero, detenga el IOScanner antes de configurar el OTB*)

```
IF NOT myExecute THEN
IO_stop_done:=IOS_STOP();
END_IF
```

(*Envíe los datos de configuración al OTB, en la dirección 95.15.3.1, cuando myExecute se encuentre en TRUE*)

```
configure_OTB1(
Execute:= myExecute,
sAddr:='3{95.15.3.1}' ,
Done=> Configure_OTB_done,
Busy=> myBusy,
Error=&gt; myError,
ConfError=&gt; myConfError,
CommError=&gt; myCommError);
```

(*Una vez que el OTB esté configurado correctamente, inicie el IOScanner)

```
IF Configure_OTB_done THEN
IO_start_done:=IOS_START();
END_IF
```

Apéndice B
Tipos de datos de Modbus TCP IOScanner

Descripción general

En este capítulo se explican los tipos de datos de la biblioteca del `ModbusTCPIOScanner`.

Contenido de este capítulo

Este capítulo contiene los siguiente apartados:

Apartado	Página
IosStateCodes: valores de estado de Modbus TCP IOScanner	66
CommunicationErrorCodes: códigos de error detectados	67
configurationOTBErrorCodes: se han detectado códigos de error en la configuración OTB	68

IosStateCodes: valores de estado de Modbus TCP IOScanner

Descripción del tipo de enumeración

El tipo de datos de enumeración de los `IosStateCodes` contiene los siguientes valores:

Enumerador	Valor	Comentario
`IosErr`	0	El Modbus TCP IOScanner está en estado de error.
`IosIdle`	1	El Modbus TCP IOScanner está en estado IDLE (inactivo). La configuración está vacía o no es compatible.
`IosOperationnal`	2	El Modbus TCP IOScanner está en estado OPERATIONAL (operativo).
`IosStopped`	3	El Modbus TCP IOScanner está en estado STOPPED (detenido).

CommunicationErrorCodes: códigos de error detectados

Descripción del tipo de enumeración

El tipo de datos de enumeración de los `CommunicationErrorCodes` contiene los siguientes valores:

Enumerador	Valor	comentario
`CommunicationOK`	hex 00	Intercambio correcto.
`TimedOut`	hex 01	Se ha detenido el intercambio debido al timeout.
`Canceled`	hex 02	Se ha detenido el intercambio a petición del usuario.
`BadAddress`	hex 03	El formato de dirección es incorrecto.
`BadRemoteAddr`	hex 04	La dirección remota es incorrecta.
`BadMgtTable`	hex 05	El formato de la tabla de gestión es incorrecto.
`BadParameters`	hex 06	Los parámetros específicos son incorrectos.
`ProblemSendingRq`	hex 07	Se ha detectado un error al enviar la petición al destinatario.
`RecvBufferTooSmall`	hex 09	El tamaño del búfer de recepción es demasiado pequeño.
`SendBufferTooSmall`	hex 0A	El tamaño del búfer de transmisión es demasiado pequeño.
`SystemResourceMissing`	hex 0B	Falta el recurso del sistema.
`BadTransactionNb`	hex 0C	El número de transacción es incorrecto.
`BadLength`	hex 0E	La longitud es incorrecta.
`ProtocolSpecificError`	hex FE	El error de funcionamiento detectado contiene un código específico del protocolo.
`Refused`	hex FF	Se ha rechazado la transacción.

configurationOTBErrorCodes: se han detectado códigos de error en la configuración OTB

Descripción del tipo de enumeración

El tipo de datos de enumeración de los `configurationOTBErrorCodes` contiene los siguientes valores:

Enumerador	Valor	comentario
`ConfigurationOK`	hex 00	La configuración OTB se ha realizado correctamente.
`IPAddrErr`	hex 01	El parámetro de entrada `sAddr` no es correcto.
`ChannelNbErr`	hex 02	No hay valor de inicialización del canal OTB para la dirección IP.
`ChannelInitValueErr`	hex 03	No ha podido determinarse el valor de inicialización del canal OTB.
`CommunicationErr`	hex 04	La configuración OTB se ha detenido debido a un error detectado.
`IosStateErr`	hex 05	El Modbus TCP IOScanner está en ejecución. El Modbus TCP IOScanner debe detenerse antes de ejecutar el bloque de funciones `CONFIGURE_OTB`.

Apéndice C
Representación de funciones y de bloques de funciones

Descripción general

Cada función se puede representar en los lenguajes siguientes:
- IL: Lista de instrucciones
- ST: Texto estructurado
- LD: Diagrama de contactos
- FBD: Diagrama de bloques de funciones
- CFC: Diagrama de función continua

En este capítulo se proporcionan funciones y ejemplos de representación de bloques de funciones y se describe cómo utilizarlas en lenguajes IL y ST.

Contenido de este capítulo

Este capítulo contiene los siguiente apartados:

Apartado	Página
Diferencias entre una función y un bloque de funciones	70
Cómo utilizar una función o un bloque de funciones en lenguaje IL	71
Cómo utilizar una función o un bloque de funciones en lenguaje ST	75

Diferencias entre una función y un bloque de funciones

Función

Una función:

- Es una POU (Unidad de organización de programa) que devuelve un resultado inmediato.
- Se le llama directamente por su nombre (y no a través de una instancia).
- No tiene un estado persistente desde una llamada hasta la otra.
- Se puede utilizar como un operando en otras expresiones.

Ejemplos: operadores booleanos (`AND`), cálculos, conversión (`BYTE_TO_INT`)

Bloque de funciones

Bloque de funciones

- Es una POU (Unidad de organización de programa) que devuelve una o más salidas.
- Debe llamarse a través de una instancia (copia del bloque de funciones con nombre y variables dedicados).
- Todas las instancias tienen un estado persistente (salidas y variables internas) de una llamada a otra desde un bloque de funciones o programa.

Ejemplos: temporizadores, contadores

En el ejemplo, `Timer_ON` es una instancia del bloque de funciones `TON`:

```
1  PROGRAM MyProgram_ST
2  VAR
3      Timer_ON: TON;  // Function Block Instance
4      Timer_RunCd: BOOL;
5      Timer_PresetValue: TIME := T#5S;
6      Timer_Output: BOOL;
7      Timer_ElapsedTime: TIME;
8  END_VAR
```

```
1  Timer_ON(
2      IN:=Timer_RunCd,
3      PT:=Timer_PresetValue,
4      Q=>Timer_Output,
5      ET=>Timer_ElapsedTime);
```

Cómo utilizar una función o un bloque de funciones en lenguaje IL

Información general

En esta sección se describe cómo implementar una función y un bloque de funciones en lenguaje IL.

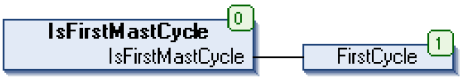
Las funciones `IsFirstMastCycle` y `SetRTCDrift` y el bloque de funciones `TON` se utilizan como ejemplos para mostrar implementaciones.

Uso de una función en lenguaje IL

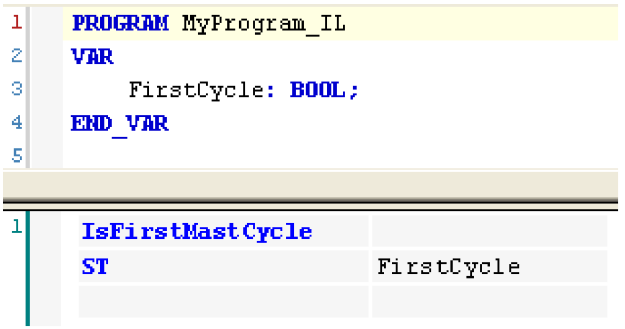
En este procedimiento se describe cómo insertar una función en lenguaje IL:

Paso	Acción
1	Abra o cree una nueva POU en el lenguaje de Lista de instrucciones (IL). **NOTA:** Aquí no se detalla el procedimiento para crear una POU. Para obtener más información, consulte Adding and Calling POUs *(véase SoMachine, - Guía de programación)*.
2	Las variables de entrada son los parámetros de entrada requeridos por la función.
3	Si la función tiene 1 o más entradas, empiece a cargar la primera entrada utilizando la instrucción LD.
4	Inserte una nueva línea abajo y: ● escriba el nombre de la función en la columna de operadores (campo izquierdo); o ● Utilice la opción **Accesibilidad** para seleccionar la función (seleccione **Insertar llamada de módulo** en el menú contextual).
5	Si la función tiene más de una entrada y se utiliza Accesibilidad, se crea automáticamente el número necesario de líneas con `???` en los campos de la derecha. Sustituya los `???` por el valor o la variable adecuada que corresponda al orden de las entradas.
6	Inserte una nueva línea para almacenar el resultado de la función en la variable adecuada: Escriba la instrucción ST en la columna de operadores (campo de la izquierda) y un nombre de variable en el campo de la derecha.

Para ilustrar el procedimiento, considere las funciones `IsFirstMastCycle` (sin parámetro de entrada) y `SetRTCDrift` (con parámetros de entrada) que se representan gráficamente a continuación:

Función	Representación gráfica
sin parámetros de entrada: `IsFirstMastCycle`	
con parámetros de entrada: `SetRTCDrift`	

En lenguaje IL, el nombre de la función se utiliza directamente en la columna de operadores:

Función	Representación en el Editor POU IL de SoMachine
Ejemplo en IL de una función sin parámetros de entrada: `IsFirstMastCycle`	

Función	Representación en el Editor POU IL de SoMachine

Ejemplo en IL de una función con parámetros de entrada: `SetRTCDrift`	`PROGRAM MyProgram_IL`
	`VAR`
	`  myDrift: SINT (-29..29) := 5;`
	`  myDay: DAY_OF_WEEK := SUNDAY;`
	`  myHour: HOUR := 12;`
	`  myMinute: MINUTE;`
	`  myDiag: RTCSETDRIFT_ERROR;`
	`END_VAR`
	
	`LD  myDrift`
	`SetRTCDrift  myDay`
	`  myHour`
	`  myMinute`
	`ST  myDiag`

Uso de un bloque de funciones en lenguaje IL

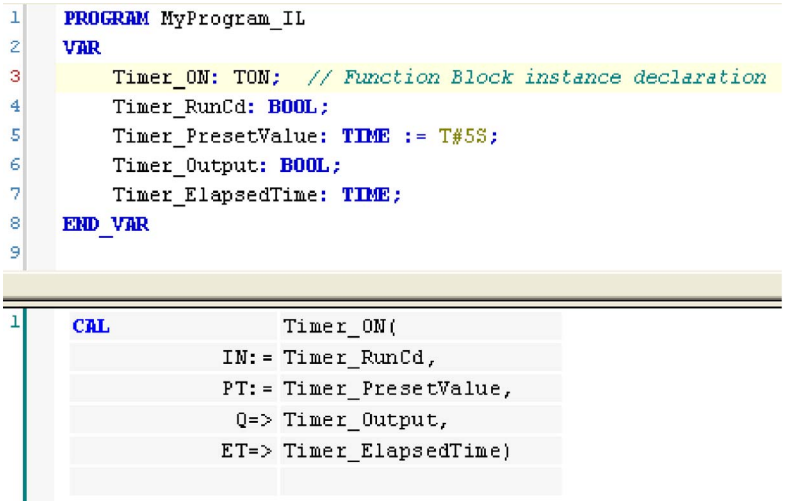
En este procedimiento se describe cómo insertar un bloque de funciones en lenguaje IL:

Paso	Acción
1	Abra o cree una POU nueva en el lenguaje de Lista de instrucciones (IL). **NOTA:** Aquí no se detalla el procedimiento para crear una POU. Para obtener más información, consulte Adding and Calling POUs *(véase SoMachine, - Guía de programación)*.
2	Cree las variables que necesita el bloque de funciones, incluido el nombre de instancia.
3	Se llama a los bloques de funciones utilizando una instrucción `CAL`: ● Utilice la opción **Accesibilidad** para seleccionar el bloque de funciones (botón derecho del ratón y seleccionar **Insertar llamada de módulo** en el menú contextual). ● La instrucción `CAL` y la E/S necesaria se crean automáticamente. Cada parámetro (E/S) es una instrucción: ● Los valores de las entradas se establecen con "`:=`". ● Los valores de las salidas se establecen con "`=>`".
4	En el campo de la derecha `CAL`, sustituya `???` por el nombre de la instancia.
5	Sustituya otros `???` por una variable apropiada o un valor inmediato.

Para ilustrar el procedimiento, considere este ejemplo con el bloque de funciones `TON` que se representa gráficamente a continuación:

3	Representación gráfica
TON	

En lenguaje IL, el nombre del bloque de funciones se utiliza directamente en la columna de operadores:

Bloque de funciones	Representación en el Editor POU IL de SoMachine
TON	

Cómo utilizar una función o un bloque de funciones en lenguaje ST

Información general

En esta sección se describe el modo de implementar una función y un bloque de funciones en lenguaje ST.

La función `SetRTCDrift` y el bloque de funciones `TON` se utilizan como ejemplos para mostrar implementaciones.

Uso de una función en lenguaje ST

En este procedimiento se describe cómo insertar una función en lenguaje ST:

Paso	Acción
1	Abra o cree una POU nueva en el lenguaje de Texto estructurado (ST). **NOTA:** Aquí no se detalla el procedimiento para crear una POU. Para obtener más información, consulte Adding and Calling POUs *(véase SoMachine, - Guía de programación)*.
2	Las variables de entrada son los parámetros de entrada requeridos por la función.
3	Utilice la sintaxis general en el **Editor POU ST** para el lenguaje ST de una función. La sintaxis general es: `FunctionResult:= FunctionName(VarInput1, VarInput2,.. VarInputx);`

Para ilustrar el procedimiento, considere la función `SetRTCDrift` que se representa gráficamente a continuación:

Función	Representación gráfica
`SetRTCDrift`	

El lenguaje ST de esta función es este:

Función	Representación en el Editor POU IL de SoMachine

`SetRTCDrift`	`PROGRAM MyProgram_ST`
	`VAR myDrift: SINT(-29..29) := 5;`
	`myDay: DAY_OF_WEEK := SUNDAY;`
	`myHour: HOUR := 12;`
	`myMinute: MINUTE;`
	`myRTCAdjust: RTCDRIFT_ERROR;`
	`END_VAR`
	`myRTCAdjust:= SetRTCDrift(myDrift, myDay, myHour, myMinute);`

Uso de un bloque de funciones en lenguaje ST

En este procedimiento se describe cómo insertar un bloque de funciones en lenguaje ST:

Paso	Acción
1	Abra o cree una POU nueva en el lenguaje de Texto estructurado (ST). **NOTA:** Aquí no se detalla el procedimiento para crear una POU. Para obtener más información sobre la adición, declaración y llamadas de POU, consulte la documentación *(véase SoMachine, - Guía de programación)* relacionada.
2	Cree las variables de entrada y salida y la instancia requeridas para el bloque de funciones: ● Las variables de entrada son los parámetros de entrada requeridos por el bloque de funciones ● Las variables de salida reciben el valor devuelto por el bloque de funciones
3	Utilice la sintaxis general en el **Editor POU ST** para el lenguaje ST de un bloque de funciones. La sintaxis general es: `FunctionBlock_InstanceName(Input1:=VarInput1, Input2:=VarInput2,... Ouput1=>VarOutput1, Ouput2=>VarOutput2,...);`

Para ilustrar el procedimiento, considere este ejemplo con el bloque de funciones `TON` que se representa gráficamente a continuación:

3	Representación gráfica
`TON`	

En esta tabla se muestran ejemplos de una llamada de bloque de funciones en lenguaje ST:

Bloque de funciones	Representación en el Editor POU IL de SoMachine

TON	1 PROGRAM MyProgram_ST
	2 VAR
	3 Timer_ON: TON; // Function Block Instance
	4 Timer_RunCd: BOOL;
	5 Timer_PresetValue: TIME := T#5S;
	6 Timer_Output: BOOL;
	7 Timer_ElapsedTime: TIME;
	8 END_VAR
	
	1 Timer_ON(
	2 IN:=Timer_RunCd,
	3 PT:=Timer_PresetValue,
	4 Q=>Timer_Output,
	5 ET=>Timer_ElapsedTime);

Glosario

B

bit de estado

Variable que indica el estado de comunicación de los canales.

byte

Un tipo que está codificado en un formato de 8 bits que, en el formato hexadecimal, va de `16#00` a `16#FF`.

C

canal Modbus

Traslado de comunicaciones que lleva una solicitud de Modbus entre el maestro y un esclavo.

CFC

(*diagrama de función continua*) Un lenguaje de programación (una ampliación del estándar IEC 61131-3) basado en el lenguaje de diagrama de bloque de funciones (FBD) y que funciona como un diagrama de flujo. Sin embargo, no se utiliza ninguna red y es posible un posicionamiento libre de elementos gráficos, lo que permite bucles de realimentación. En cada bloque, las entradas se sitúan a la izquierda y las salidas, a la derecha. Las salidas del bloque se pueden conectar a las entradas de otros bloques para formar expresiones complejas.

D

DHCP

(*protocolo de configuración dinámica del host*) Una ampliación avanzada de BOOTP. DHCP es más avanzado, pero tanto DHCP como BOOTP son habituales. (DHCP puede manejar las solicitudes de clientes BOOTP).

diagrama de bloques de funciones

Uno de los cinco lenguajes para lógica o control que cumplen con el estándar IEC 61131-3 para sistemas de control. El diagrama de bloques de funciones es un lenguaje de programación orientado gráficamente. Funciona con una lista de redes en la que cada red contiene una estructura gráfica de cuadros y líneas de conexión que representa una expresión lógica o aritmética, la llamada de un bloque de funciones, un salto o una instrucción de retorno.

F

FB

(*bloque de funciones*) Un práctico mecanismo de programación que consolida un grupo de instrucciones de programación para realizar una acción específica y normalizada, por ejemplo, el control de velocidad, el control de intervalo o el conteo. Un bloque de funciones se puede componer de datos de configuración, un conjunto de parámetros de funcionamiento internos o externos y, normalmente, una o diversas entradas y salidas de datos.

FDR

(*sustitución rápida de dispositivo*)

frecuencia de repetición

Intervalo de consulta de la solicitud Modbus enviada.

I

IL

(*lista de instrucciones*) Un programa escrito en lenguaje que se compone de una serie de instrucciones basadas en texto y ejecutadas secuencialmente por el controlador. Cada instrucción incluye un número de línea, un código de instrucción y un operando (consulte IEC 61131-3).

INT

(*entero*) Un número entero con codificación de 16 bits.

L

LD

(*diagrama de contactos*) Una representación gráfica de instrucciones de un programa de controlador con símbolos para contactos, bobinas y bloques en una serie de escalones ejecutados de forma secuencial por un controlador (consulte IEC 61131-3).

P

POU

(*unidad de organización de programas*) Una declaración variable en el código fuente y el conjunto de instrucciones correspondiente. Las POUs facilitan la reutilización modular de programas de software, funciones y bloques de funciones. Una vez declaradas, cada una de las POUs está disponible para las otras.

S

ST

(*texto estructurado*) Un lenguaje que incluye instrucciones complejas y anidadas (por ejemplo, bucles de repetición, ejecuciones condicionales o funciones). ST cumple con IEC 61131-3.

T

Timeout de estado

Representa el tiempo máximo (en ms) entre una petición del explorador de E/S Modbus y una respuesta del esclavo.

V

variable

Una unidad de memoria direccionada y modificada por un programa.

Índice

A

arquitectura
    Modbus TCP IOScanner, *12*

B

bloques de función
    Modbus TCP IOScanner, *45*

C

cálculo de la carga
    Modbus TCP IOScanner, *38*
características generales
    Modbus TCP IOScanner, *10*
CommunicationErrorCodes
    tipos de datos, *67*
ConfigurationOTBErrorCodes
    tipos de datos, *68*
CONFIGURE_OTB
    funciones, *62*

D

direcciones IP
    Modbus TCP IOScanner, *20*

E

estados
    Modbus TCP IOScanner, *40*

F

ficha Recursos del IOScanner TCP Modbus
    Modbus TCP IOScanner, *38*
funciones
    cómo utilizar una función o un bloque de funciones en lenguaje IL, *71*
    cómo utilizar una función o un bloque de funciones en lenguaje ST, *75*
    CONFIGURE_OTB, *62*
    diferencias entre una función y un bloque de funciones, *70*
    IOS_GETHEALTH, *60*
    IOS_GETSTATE, *58*
    IOS_START, *59*
    IOS_STOP, *61*

I

interfaz de aplicación
    Modbus TCP IOScanner, *44*
IOS_GETHEALTH
    funciones, *60*
IOS_GETSTATE
    funciones, *58*
IOS_START
    funciones, *59*
IOS_STOP
    funciones, *61*
IosStateCodes
    tipos de datos, *66*

M

modalidades de funcionamiento
    Modbus TCP IOScanner, *40*
Modbus TCP IOScanner
    añadir un dispositivo, *26*
    añadir y configurar, *27*
    arquitectura, *12*
    bloques de función, *45*
    cálculo de la carga, *38*
    características generales, *10*
    configurar un dispositivo genérico, *34*
    configurar un dispositivo OTB, *29*
    configurar un dispositivo predefinido, *32*
    direcciones IP, *20*
    estados, *40*
    ficha Recursos del IOScanner TCP Mod-

bus, *38*
interfaz de aplicación, *44*
modalidades de funcionamiento, *40*
planificación de red, *18*
principios, *14*
prueba de red, *23*
resolución de problemas, *54*
servidor web M251, *52*
supervisión a través de SoMachine , *48*
tarea de ciclo de bus, *44*

P

planificación de red
Modbus TCP IOScanner, *18*
principios
Modbus TCP IOScanner, *14*
prueba de red
Modbus TCP IOScanner, *23*

R

resolución de problemas
Modbus TCP IOScanner, *54*

S

servidor web M251
Modbus TCP IOScanner, *52*
supervisión a través de SoMachine
Modbus TCP IOScanner , *48*

T

tarea de ciclo de bus
Modbus TCP IOScanner, *44*
tipos de datos
CommunicationErrorCodes, *67*
ConfigurationOTBErrorCodes, *68*
IosStateCodes, *66*