

Manual Twido Suite

Manual de introducción al uso de la herramienta de configuración, programación y depuración de controladores programables de la gama Twido.



ADVERTENCIA

Los productos presentados en este manual son susceptibles de evolución en cuanto a sus características de presentación, de funcionamiento o de utilización. Su descripción en ningún momento puede revestir un aspecto contractual.

El Instituto Schneider Electric de Formación, acogerá favorablemente cualquier solicitud con fines didácticos exclusivamente, de utilización de gráficos o de aplicaciones contenidas en este manual.

Cualquier reproducción de este manual está totalmente prohibida sin la autorización expresa del Instituto Schneider Electric de Formación.

Manual de formación TwidoSuite

Creado: Instituto Schneider Electric de Formación

Bac de Roda 52, Edificio A – 1ª Planta

Fecha: 23 de Julio de 2008

Versión: 2.1

SCHNEIDER ELECTRIC ESPAÑA

Índice

| | Página |
|--|--------|
| 1. Presentación general del Hardware Twido_____ | p. 7 |
| 1.1. Controladores compactos _____ | p. 7 |
| 1.2. Controladores modulares_____ | p. 10 |
| 1.3. Módulos de ampliación_____ | p. 11 |
| 1.4. Descripción general de las comunicaciones | p. 16 |
| 2. Instalación del software Twidosuite_____ | p. 23 |
| 2.1. Introducción_____ | p. 23 |
| 2.2. Requisitos mínimos y recomendados_____ | p. 24 |
| 2.3. Instalación del software de programación Twidosuite Explorador de proyectos_____ | p. 24 |
| 3. Operaciones básicas con el PLC_____ | p. 27 |
| 3.1. Abrir por primera vez el Twidosuite_____ | p. 27 |
| 3.2. Registro del Twidosuite_____ | p. 28 |
| 3.3. Crear un proyecto nuevo_____ | p. 29 |
| 3.4. Navegación por el espacio de trabajo del Twidosuite_____ | p. 30 |
| 3.5. Configuración básica del Hardware Twido __ | p. 31 |
| 3.6. Edición de Datos_____ | p. 34 |
| 3.7. Escribir Programa_____ | p. 38 |
| 4. Operaciones avanzadas_____ | p. 67 |
| 4.1. Operaciones con señales analógicas _____ | p. 67 |
| 4.2. Bloque generador de pulsos %PLS_____ | p. 77 |
| 4.3. Instrucciones de contador rápido FC _____ | p. 79 |
| 4.4. Instrucciones de contaje muy rápido VFC__ | p. 43 |

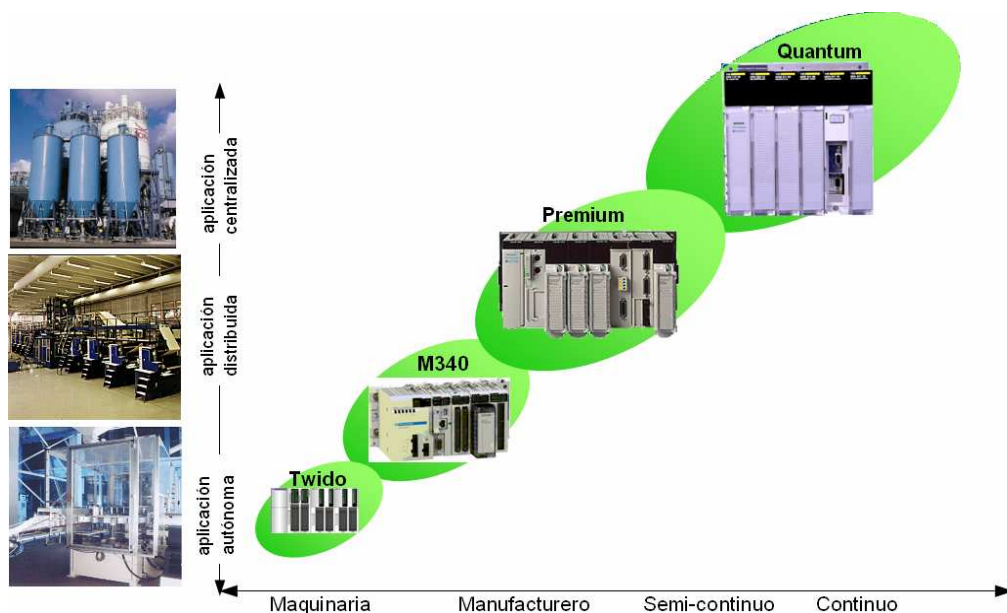
5. Comunicaciones_____ p. 83

- 5.1. Configuración de una red Modbus _____ p. 83
- 5.2. Macros de comunicaciones_____ p. 86
- 5.3. Configuración de una red ASI_____ p. 89
- 5.4. Configuración de una red CanOpe _____ p. 93
- 5.5. Macros Drive _____ p. 96
- 5.6. Implementación del Twido en una red
Ethernet _____p. 103
- 5.7. Implementación de una red RemoteLink___p. 108

6. Glosario_____ p. 113

1. Presentación general del Hardware Twido

La elección de un modelo u otro de autómatas vendrá dada por la tipología y complejidad de la aplicación que se desea automatizar.



Esquema de campo de aplicación de los diferentes autómatas de SCHNEIDER ELECTRIC

Los Controladores programables **Twido**, han sido optimizados para las instalaciones sencillas y las máquinas pequeñas: aplicaciones estándar de **10 a 100 E/S** (máx. 252 E/S). Donde el Twido ofrece una flexibilidad y sencillez a la hora de automatizar este tipo de aplicaciones.

El controlador Twido dispone de dos modelos: **Compacto o Modular**.

1.1 Controladores Compactos:

La gama de controladores programables compactos Twido ofrece una solución “**todo en uno**” con unas dimensiones reducidas, lo que permite reducir el tamaño de las consolas o de los cofres en las aplicaciones donde el espacio ocupado resulta primordial.

Los controladores de tipo compacto tiene integradas en el mismo cuerpo las entradas y salidas, este dependerá del modelo, pudiendo elegir: **10 E/S, 16 E/S, 24 E/S y 40 E/S**.

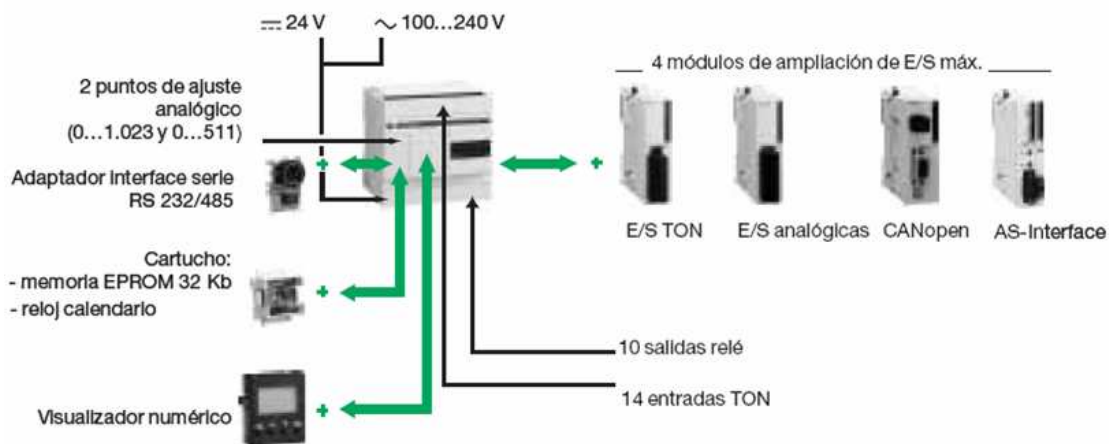
Los controladores de **24 E/S** y **40 E/S** admiten módulos de ampliación que nos confieren una mayor flexibilidad a la hora de elegir el tipo de controlador.



Gama de controladores Twido compactos

En los controladores de **24 E/S** es posible colocar hasta **4 módulos** de ampliación y en los de **40 E/S** hasta **7 módulos**, dependiendo siempre de que no se supere los límites de consumo de potencia, este se puede controlar a través del software **TwidoSuite**. Dependiendo del tipo de módulo de ampliación se puede llegar hasta **152 E/S** con el controlador de **24 E/S** y hasta **264 E/S** como máximo con el de **40 E/S**.

Los controladores Twido compactos ya tienen integrada la fuente de alimentación y utilizan: Una alimentación de corriente alterna comprendida entre **100 y 240 Vca** (que garantiza la alimentación **24 Vcc** de los captadores), o una alimentación de corriente continua comprendida entre 19,2 y 30 Vcc (prever una alimentación auxiliar externa tipo Phaseo para la alimentación de los captadores).



Módulos de ampliación y accesorios para los controladores compactos

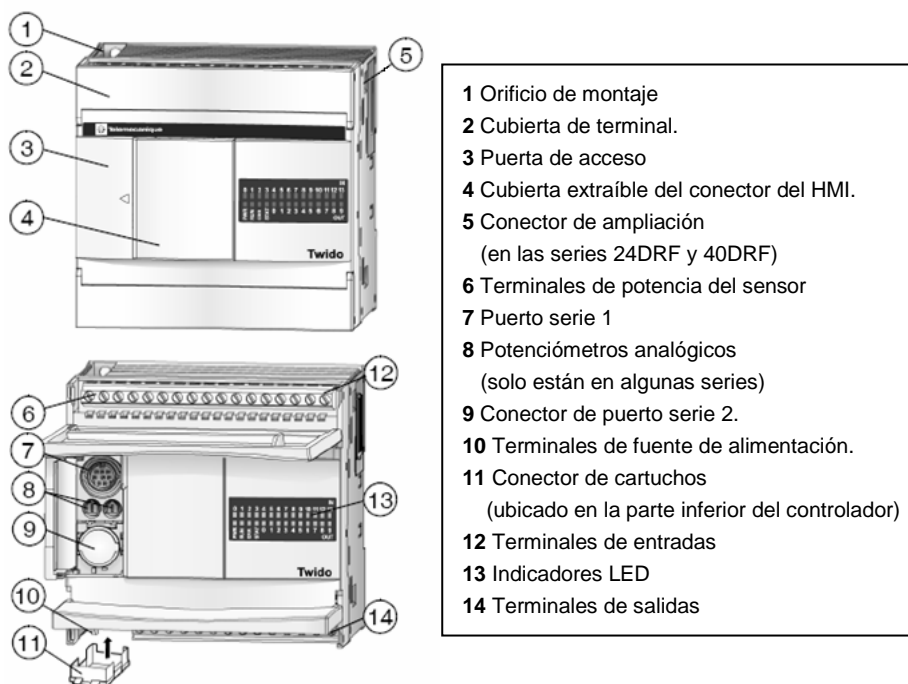
Además de los módulos de ampliación, los controladores compactos Twido, también disponen de módulos opcionales, como visualizador numérico, cartucho de ampliación de memoria, cartucho de reloj calendario y puerto de comunicación RS 485 o RS 232C suplementario, que permiten ajustarse a las necesidades de la aplicación.

Los controladores base compactos de las series **TWDLCAA40DRF** y **TWDLCAE40DRF** integran funciones avanzadas:

- Puerto de red **Ethernet 100Base-TX** integrado: sólo para TWDLCAE40DRF
- Reloj de tiempo real (RTC) integrado.
- Un cuarto contador rápido (FC).
- Soporte de batería externa.

1.1.1 Descripción de los componentes de un controlador compacto.

Los controladores Twido compactos están formados por los siguientes componentes, teniendo en cuenta que hay pequeñas diferencias entre modelos de controlador, pero que los componentes siempre serán los mismos:



Partes de un controlador compacto

1.2 Controladores Modulares:

La gama de controladores modulares ofrece **5 bases**, que se diferencian entre sí por la capacidad de tratamiento y el número y tipo de entradas/ salidas integradas (**20 o 40 E/S**). Todas ellas pueden ampliarse con cualquier módulo de entradas/salidas, hasta 7 módulos (configuración máxima). Todas las bases modulares se alimentan a **24 Vcc.**



20 E/S conexión TELEFAST



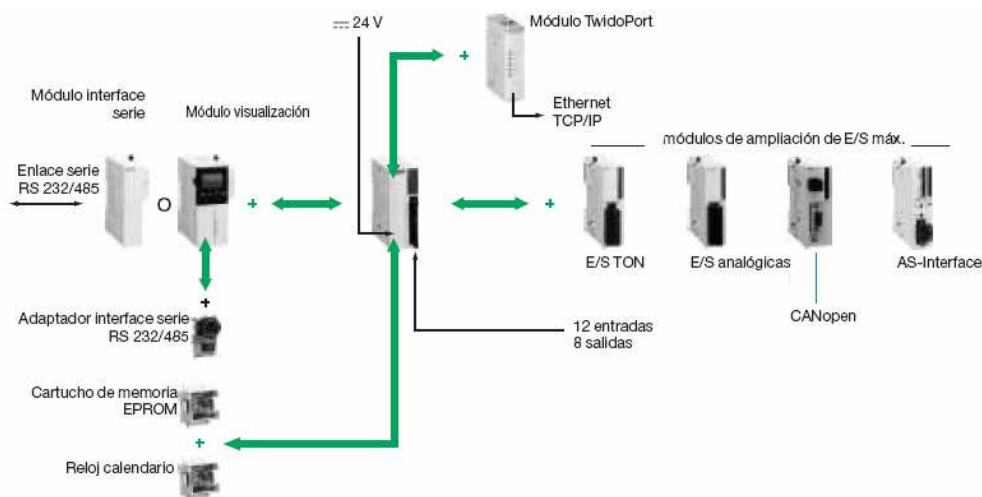
20 E/S conexión Bornero



40 E/S conexión TELEFAST

Estas bases ofrecen una **modularidad** que se adapta a las necesidades de la aplicación a partir de una base que admite hasta **4 o 7 módulos de ampliación** de entradas/salidas TON y/o analógicas (según el modelo), de esta manera se pueden configurar hasta un máximo de **244 E/S** con los módulos de **20 E/S** integradas y de **264 E/S** en los de **40 E/S** integradas.

Es posible añadir E/S adicionales al controlador mediante módulos de E/S de ampliación hay diferentes tipos: módulos de **E/S digitales** o tipo de relé y módulos de **E/S analógicas**.

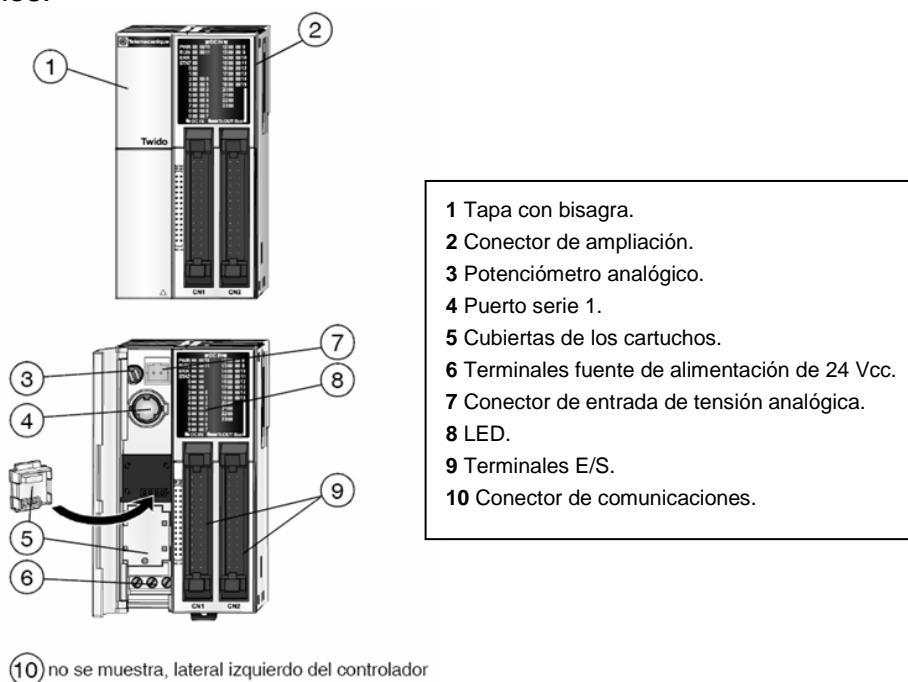


Módulos de ampliación y accesorios para los controladores modulares

Existe una oferta de **módulos opcionales** que proporciona al usuario el mismo grado de flexibilidad que las plataformas de automatismos de mayor entidad. Las bases modulares **TWD LMDA** admiten simultáneamente los módulos opcionales cartucho de memoria, reloj calendario y los módulos visualizador numérico o interface serie, cada uno de los cuales admite a su vez un segundo puerto de comunicación RS 485 o RS 232C.

1.2.1 Descripción de los componentes de un controlador modular.

Los controladores Twido modulares están formados por los siguientes componentes, teniendo en cuenta que hay pequeñas diferencias entre modelos de controlador, pero que los componentes siempre serán los mismos:



Partes de un controlador modular

1.3 Módulos de ampliación:

Los módulos de ampliación nos confieren a nuestro controlador Twido la mayor adaptabilidad posible a las diferentes aplicaciones dentro de su campo de acción, pudiéndose ajustar el dispositivo lo máximo posible a cada una de las aplicaciones concretas con el ahorro de coste que ello comporta.

1.3.1 Módulos de E/S digitales:

Existen una amplia gama de módulos distintos de entradas/salidas TON Twido que permiten completar las entradas/salidas integradas tanto en las bases compactas ampliables como en las bases modulares, pudiendo cada usuario adaptar el controlador a las necesidades de su aplicación, optimizando así los costes.



16 Entradas con
bornero extraíble



32 Entradas con
conector TELEFAST



8 Salidas relé con
bornero extraíble



16 Entradas / 8 Salidas
con conexión por resorte

La descripción de los 15 tipos de tarjetas, se divide en módulos de entradas, módulos de salida y módulos híbridos de entradas y salidas, también diferenciándose por el tipo de conexión:

- Módulos de entradas TON con alimentación a **24 Vcc**.
 - **8** Entradas de bornero extraíble.
 - **16** Entradas de bornero extraíble.
 - **8** Entradas de conector TELEFAST.
 - **16** Entradas de conector TELEFAST.

- Módulo de entradas TON con alimentación a **120 Vac**.
 - **8** Entradas de bornero extraíble.

- Módulos mixtos de entradas y salidas con alimentación a 24 Vcc.
 - **16** Entradas/**8** Salidas **Relé** 2 A, de conexión por resorte.
 - **4** Entradas/**4** Salidas **Relé** 2 A, de bornero extraíble.

- Módulos de salidas con bornero extraíble:
 - **8** Salidas **Relé** 2 A.
 - **16** Salidas **Relé** 2 A.
 - **8** Salidas Transistor 0,3 A **PNP** con alimentación a **24Vcc**.
 - **8** Salidas Transistor 0,3 A **NPN** con alimentación a **24 Vcc**.

■ Módulos de salidas con conector TELEFAST:

- 16 Salidas Transistor 0,1 A **PNP** con alimentación a **24Vcc**.
- 16 Salidas Transistor 0,1 A **NPN** con alimentación a **24 Vcc**.
- 32 Salidas Transistor 0,1 A **PNP** con alimentación a **24Vcc**.
- 32 Salidas Transistor 0,1 A **NPN** con alimentación a **24 Vcc**.

Los elementos electrónicos internos y las vías de entradas/salidas de todos los módulos de entradas/salidas TON están aislados mediante fotoacoplador.

1.3.2 Módulos de E/S analógicas:

Los módulos de ampliación de entradas analógicas Twido permiten obtener diferentes valores analógicos presentes en las aplicaciones industriales. Los de salidas analógicas se utilizan para dirigir los preaccionadores en unidades físicas, como variadores de velocidad, válvulas y aplicaciones que requieran control de procesos.



4 Entradas analógicas



2 Salidas analógicas



4 Entradas y 2 Salidas analógicas

Las entradas y salidas analógicas vendrán definidas en función del valor analógico, puede ser la corriente o la tensión, incluso valores de resistencia para los casos de temperatura.

Los módulos de entradas/salidas analógicas, tendrán la conexión del tipo bornero extraíble.

Los módulos de entradas/salidas analógicas se elegirán teniendo en cuenta los siguientes parámetros:

Por el número de entradas/salidas: 2, 4 y 8

Por el tipo de señal analógica:

- Señal de **Tensión** de **0...10 V**
- Señal de **corriente** de **0...20 mA**
- Señal de **corriente** de **4...20 mA**
- Señal de entrada **termopar** del tipo **K, J y T**.
- Señal de entrada **PTC** del tipo **Pt 100/1000, Ni100/1000** rango de temperatura entre 50...150 °C.

Existe un tipo de módulo mixto que presentan entradas y salidas analógicas en el mismo cuerpo.

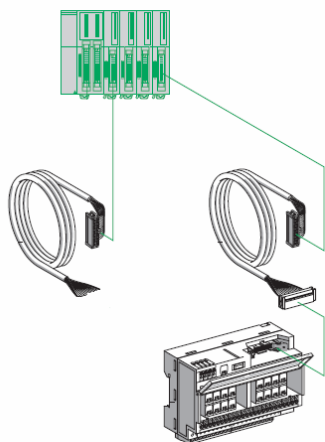
Los módulos analógicos Twido ofrecen una **resolución de 10 bits, 11 bits + signo y 12 bits**, con conexión mediante bornero con tornillo desenchufable.

Es necesario instalar una alimentación externa de **24 Vcc** para cada módulo analógico.

Los elementos electrónicos internos y las vías de entradas/salidas de todos los módulos de entradas/salidas analógicas están aislados mediante fotoacoplador.

1.3.3 Tipos de conexionado:

Se observa que hay diferentes tipos de conexionado en los diferentes módulos de ampliación para facilitar el cableado de los diferentes elementos de la aplicación a automatizar (sensores, actuadores) al controlador Twido. Estos 3 tipos de conexionado son:



Sistema de conexionado TELEFAST

- Borneros con tornillos extraíbles.
- Conectores de tipo resorte que permiten realizar un cableado sencillo, rápido y seguro.
- El sistema de precableado Advantys **Telefast ABE 7** permite conectar los módulos con **conectores de tipo HE 10**: a los cables preequipados con hilos libres en uno de los extremos o al sistema de cableado Advantys Telefast ABE 7 para controlador Twido (conjunto de cables de conexión y bases ABE 7).

1.3.4 Accesorios:

Existen varias opciones que pueden agregarse a los controladores base, para incrementar las prestaciones en su justa medida, en función de la aplicación.

- Cartuchos de ampliación de memoria de 32 Kb o 64 Kb, para backup del proyecto o almacenamientos de datos de la aplicación.
- Cartucho de reloj de tiempo real (RTC), para la utilización dentro del software de programación TwidoSuite de los 16 bloques de función horaria extendida.
- Adaptadores de comunicaciones es una opción disponible para todas las CPUs (excepto la de 10 E/S) ofrece la posibilidad de comunicación RS-232 o RS-485, varios tipos de conexionado (conector Mini DIN o bornero para RS-485), pudiéndose configurar como: Modbus Maestro/Esclavo, ASCII y Remote Link.
- Display HMI (diferente en función si el Twido es compacto o modular), permite el ajuste de la máquina sin necesidad de utilizar el software de programación y funciones de diagnóstico durante el funcionamiento tales como: acceso a variables internas (temporizadores, contadores, hora, etc), puesta en RUN, STOP, inicialización de memoria y el bloqueo por programa.



Ampliación de memoria



Cartucho Reloj de tiempo real



Adaptadores de comunicaciones



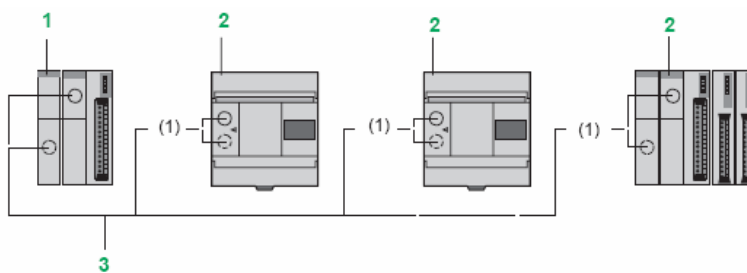
Monitor de operación

1.4 Descripción general de las comunicaciones:

Los controladores Twido disponen de un puerto serie, o de un segundo puerto opcional, que se utiliza para servicios en tiempo real o de administración de sistemas. Los servicios en tiempo real proporcionan funciones de distribución de datos para intercambiar datos con dispositivos de E/S, así como funciones de administración para comunicarse con dispositivos externos. Los servicios de administración de sistemas controlan y configuran el controlador por medio de TwidoSoft. Cada puerto serie se utiliza para cualquiera de estos servicios, pero sólo el puerto serie 1 es válido para comunicarse con TwidoSoft.

Para poder utilizar estos servicios, existen tres protocolos implícitos disponibles en cada controlador:

- **Remote Link (Conexión remota):** permite realizar una comunicación entre autómatas Twido vía RS-485, utilizado para ver **E/S a distancia** (sin programa en las CPUs deportadas) o para red de Twidos con programa, con una longitud máxima de **200 m** y hasta **8 equipos** en una red (**maestro + 7 esclavos**).



- 1 Controlador de base compacta o modular denominado "maestro".
- 2 Bases compacta o modular Twido utilizadas en ampliación de E/S o "réflex" local.
- 3 Cable RS 485, 3 hilos, a partir del puerto de enlace serie integrado o del 2.º puerto de enlace serie opcional.

Esquema ejemplo de Conexión Remota

- **ASCII:** permite comunicar el autómata, vía **RS-485 y RS-232**, con un gran número de dispositivos: **impresoras** (para la impresión periódica de raports de producción), lectores de códigos de barras y módems.

- **Modbus:** comunicación Modbus Maestro/Esclavo por ambos puertos (RS485 o 232), permite conectar Twido a un gran número de equipos industriales, basado en mensajería aperiódica entre equipos.

Nota: El controlador compacto TWDLCAE40DRF proporciona un puerto de comunicación RJ45 Ethernet integrado. Las comunicaciones **Ethernet** implementan el protocolo **TCP/IP Modbus**.

Además, se pueden implementar más comunicaciones agregando módulos de comunicación, existen varios tipos en función del protocolo que se desee:

- Modulo de comunicación Maestro **ASI**.
- Modulo de comunicación Maestro **CANOpen**.
- Modulo de comunicación Maestro/Esclavo **Modbus**.



Maestro CANopen

Maestro AS-Interface

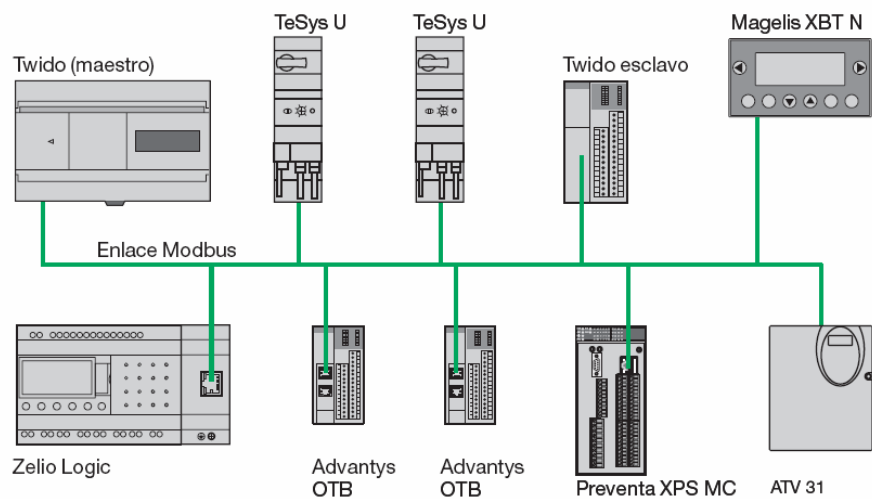
Enlace serie asíncrono
integrado o opcional

1.4.1 Bus de comunicación Modbus:

El enlace serie Modbus permite responder a las arquitecturas maestro/esclavo (no obstante, es necesario comprobar que los servicios Modbus útiles para la aplicación se implanten en los equipos implicados).

El bus está constituido por una estación maestro y por estaciones esclavo. Sólo la estación maestro puede iniciar el intercambio (la comunicación directa entre estaciones esclavo no es posible). Existen dos mecanismos de intercambio:

- **Pregunta/respuesta**, las peticiones del maestro se dirigen a un esclavo determinado. El esclavo interrogado espera de vuelta la respuesta.
- **Difusión**, el maestro difunde un mensaje a todas las estaciones esclavo del bus. Éstas últimas ejecutan la orden sin emitir respuesta.



Esquema ejemplo de comunicación MODBUS

- **Modo maestro de Modbus:** el modo maestro de Modbus permite que el controlador pueda iniciar una transmisión de solicitudes Modbus, esperando una respuesta desde un esclavo Modbus.
- **Modo esclavo Modbus:** el modo esclavo Modbus permite que el controlador pueda responder a las solicitudes de Modbus desde un maestro Modbus. Se trata del modo de comunicación predeterminado si no existe ninguna comunicación configurada.

La comunicación Modbus Maestro/Esclavo se puede realizar por ambos puertos (RS485 o 232).

Este protocolo permite conectar Twido a un gran número de equipos industriales, como variadores de velocidad, arrancadores de motor, sensores...etc.

1.4.2 Bus de comunicación ASI-interface:

El bus AS-I (la abreviatura del término inglés Actuator-Sensor-Interface) es un bus de campo (nivel 0) que se emplea para realizar interconexiones de sensores/ accionadores. Permite ejecutar información de tipo «binario» o analógico entre un equipo «maestro» de bus y equipos «esclavo». El bus AS-Interface está formado por tres elementos básicos principales:

- Una fuente de alimentación específica que suministra una tensión de 30 VDC.
- Un maestro de bus
- Uno o varios equipos esclavos (sensores, accionadores y otros).



Esquema ejemplo de comunicación ASI

El módulo maestro **TWD NOI 10M3** gestiona con el perfil AS-I M3:

- Equipos esclavo “**Todo o Nada**” (**62 equipos máx.** organizados en 2 bancos A y B de 31 direcciones cada uno).
- Equipos **analógicos** (**7 equipos máx.** en banco A). El perfil AS-Interface M3 admite el perfil analógico **7.3** (7 esclavos), pero no así el perfil analógico **S-7.4**.

El número máximo de módulos de ampliación **TWD NOI 10M3** por controlador Twido es de **2**.

La alimentación AS-Interface resulta indispensable para alimentar los distintos equipos conectados a AS-Interface, que debe situarse preferentemente cerca de las estaciones que consuman más energía.

1.4.2 Bus de comunicación CANopen:

Desarrollado originalmente para sistemas de vehículos automotores, el bus de comunicación CAN se utiliza en la actualidad en muchos campos, como por ejemplo: transporte, dispositivos de piezas móviles, dispositivos médicos, control automático industrial. La implementación bus CAN ofrece unas ventajas a la aplicación como: sistema de asignación del bus, detección de errores y fiabilidad en el intercambio de datos.

Las comunicaciones en el bus se realizan **peer to peer**; en cualquier momento, cada dispositivo puede enviar una solicitud en el bus y la respuesta de los dispositivos pertinentes. La prioridad de las solicitudes que circulan por el bus la determina un identificador en el nivel de mensaje individual.

El módulo maestro **TWD NCO1M** para bus CANopen permite al controlador programable Twido, asegurar la función maestro CANopen.



Ejemplo de arquitectura CANopen

El bus consta de una estación maestro, el controlador Twido y varias estaciones esclavo. El maestro se encarga de la configuración, los intercambios y el diagnóstico hacia los esclavos. El bus CANopen es un bus de tipo comunicación y permite gestionar diversos esclavos como:

- Esclavos “Todo o Nada”,
- Esclavos analógicos,
- Variadores de velocidad,
- Arranques motor,
- Servomotores.

El maestro CANopen Twido controla hasta 16 esclavos, cada uno con un objeto de datos de proceso PDO (Process Data Object) en la entrada y un objeto de datos de proceso PDO en la salida. Si un esclavo dispone de más de un objeto de datos de proceso PDO, se reduce entonces el máximo de esclavos.

1.4.3 Bus de comunicación Ethernet:

TwidoPort ConneXium aporta conectividad Ethernet a la línea de productos Twido. Es la pasarela entre un único dispositivo Twido Modbus/RTU (RS-485) y la capa física de las redes Modbus/TCP en el modo esclavo. Este módulo de pasarela sólo admite el modo esclavo.



Ejemplo de arquitectura Ethernet

Las principales características del módulo TwidoPort son las siguientes:

- Clase **A10** (Sin servidor web, ni I/O scanning).
- **Mensajería Modbus** (lectura/escritura de palabras de datos).
- Función **BOOTP**.
- Admite la configuración manual utilizando Telnet.
- Interface físico 10BASE-T/100BASE-TX, con conector normalizado de tipo RJ45.
- Caudal binario **10/100 Mbits/s**, con reconocimiento automático de par trenzado

La gama de los controladores programables Twido ofrece una base compacta con puerto Ethernet integrado. **TWD LCAE 40DRF** y **TWD LCDE 40DRF**.

El módulo TwidoPort **499 TWD 01100** es un interface Ethernet fácil de utilizar y destinado al controlador programable Twido versión ≥ 3.0 , compacto o modular.

2. Instalación del software TwidoSuite

2.1 Introducción:

TwidoSuite es el primer software que está organizado según el ciclo de desarrollo del proyecto. La navegación por el software es tan sencilla que se convierte en innata.

TwidoSuite es un entorno de desarrollo gráfico, lleno de funciones para crear, configurar y mantener aplicaciones de automatización para los autómatas programables Twido de Telemecanique. TwidoSuite permite crear programas con distintos tipos de lenguaje, después de transferir la aplicación para que se ejecute en un autómata.



Plataforma TwidoSuite

TwidoSuite es un programa basado en **Windows** de 32 bits para un ordenador personal (PC) que se ejecuta en los sistemas operativos **Microsoft Windows 2000 y XP Professional**.

Las principales funciones del software TwidoSuite son:

- Interface de usuario intuitiva y orientada a proyectos.
- Diseño de software sin menús. Las tareas y funciones del paso seleccionado de un proyecto siempre se encuentran visibles.
- Soporte de programación y configuración.
- Comunicación con el autómata.
- Ayuda de primera mano acerca del nivel de tareas que ofrece enlaces relevantes a la ayuda en línea.

TwidoSuite es un software fácil de usar que necesita poco o nada de aprendizaje. Este software tiene por objeto reducir de forma

significativa el tiempo de desarrollo de los proyectos simplificando todas las intervenciones.

2.2 Requisitos mínimos y recomendados:

La configuración mínima necesaria para utilizar TwidoSuite es la siguiente:

- Se recomienda un equipo compatible con PC y procesador Pentium a **466 MHz** o superior.
- **128 MB** de RAM o más,
- **100 MB** de espacio libre en el disco duro.
- Sistema operativo: **Windows 2000 o Windows XP.**
- Evite el uso de los parches 834707-SP1 (corregido por el parche 890175) y 896358 que producen problemas de visualización en la ayuda en línea.
- Se recomienda **Service Pack 2** o superior. Disponible para su descarga en el sitio web www.microsoft.com.



La conexión Autómata-PC hace uso del protocolo TCP/IP. Este protocolo debe estar instalado en el PC.

2.3 Instalación del software de programación TwidoSuite:

El TwidoSuite es un software de programación utilizado para la configuración, programación y depuración de la gama de controladores programables Twido.

El TwidoSuite es un software gratuito que se puede descargar desde la página del ISEFONLINE, a través de la siguiente dirección.

www.isefonline.com

Para comenzar la instalación, abrir el archivo ejecutable que previamente se ha descargado.



Aparecerá una ventana flotante con la información de los derechos usuario de la licencia del software TwidoSuite, la aceptamos pulsando el botón “Accept”.

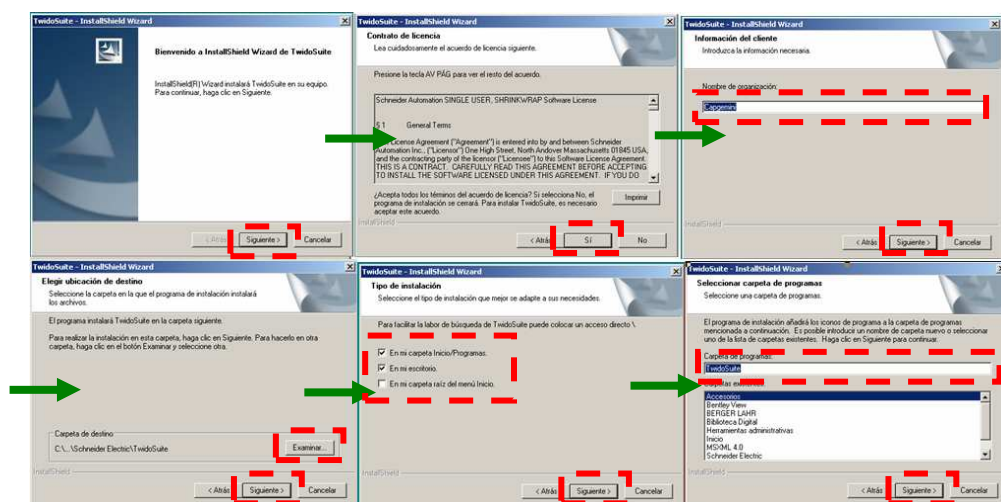
Una vez aceptada la licencia, se abre una ventana nueva donde se puede colocar la ruta donde se quiere descomprimir el instalador. (Sino cambiamos la ruta, el creara por defecto “C:\Burndisk”). Cuando se haya especificado la ruta, pulsar el botón “Install” para comenzar la descomprimir el archivo.

Abrimos la ruta especificada donde se ha creado la carpeta de instalación y pulsamos sobre el icono “Setup”, que mostrará la ventana inicial de instalación.



Si se dispone de otra versión de TwidoSuite instalada en el PC, desinstalarla previamente a la instalación de la nueva versión.

Durante el proceso de instalación, tendremos que aceptar el contrato de licencia, colocar el nombre de vuestra organización, especificar la ruta donde se desea instalar el programa, si deseamos colocar un icono en el escritorio o en la barra de inicio rápido y desde que carpeta de la barra de programas se desea colocar el software.

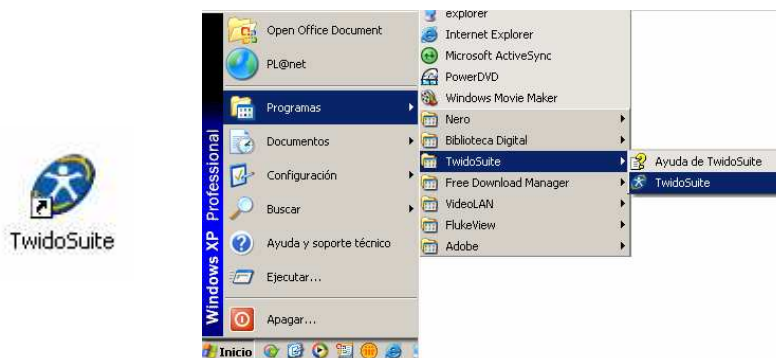


Ventanas del proceso de instalación

3. Operaciones básicas

3.1 Abrir por primera vez el Twidosuite:

Una vez finalizada la instalación del proyecto, abriremos por primera vez el Twidosuite, para ello iremos al icono que se ha generado en el escritorio (si lo hemos seleccionado en la instalación) o lo buscaremos dentro de la barra de programas de nuestro PC.



Opciones de apertura de Twidosuite

Se abrirá la pantalla inicial de Twidosuite, aparecen tres opciones principales:

- **Modo “Programación”**: Modo estándar para la creación de una aplicación.
- **Modo “Vigilancia”**: Este modo nos permite conectarse a un autómata en modo vigilancia, donde podrá comprobar su funcionamiento sin necesidad de sincronizar su aplicación con la que hay cargada en la memoria del autómata.
- **Actualización de autómatas**: Es un programa que indica todos los pasos necesarios para actualizar el **Firmware** Executive del controlador programable Twido.



Pantalla inicial de Twidosuite



En la pantalla de inicial del Twidosuite, además de los tres modos principales podemos seleccionar uno de los dos idiomas que hemos seleccionado en la instalación, apretando en los iconos que aparecen en la parte inferior izquierda de la ventana inicial.

3.2 Registro del TwidoSuite

Una vez descargado, instalado y abierto por primera vez el Twidosuite, el paso siguiente es proceder a registrar el software. El registro es gratuito y le mantendrá informado de las noticias sobre los productos más recientes, actualizaciones de software y firmware para su controlador Twido.

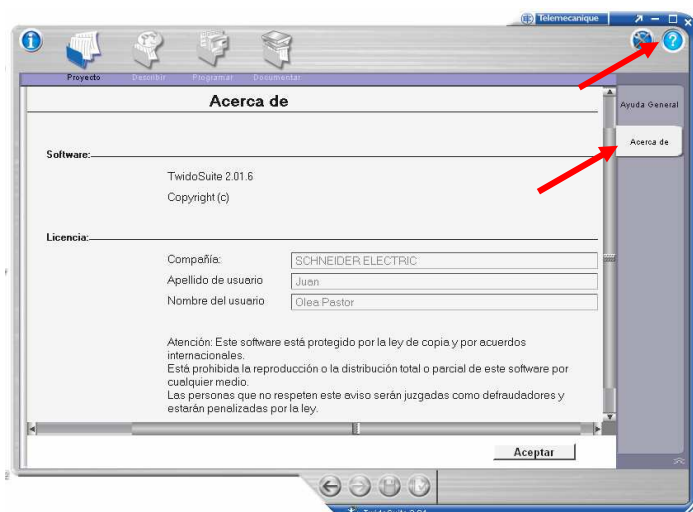


Sino se registra, solo se dispone de un periodo de prueba de 30 días para este software. Para poder seguir utilizando este software después de que caduque el periodo de prueba, deberá registrarlo.

Para registrar TwidoSuite, abrirlo en “**Modo programación**” y cuando aparezca el espacio de trabajo de la ventana principal, pulsar en el icono de “**Otros**” que se encuentra situado en la esquina superior derecha.



Cuando aparezca la ventana, hacer clic en “**Acerca de**” en la barra de tareas de la parte derecha de la pantalla. Rellene el formulario Licencia con la información «Empresa», «Usuario» y «Número de serie» y pulse “**Ahora**”.



Ventana de la opción “Otros”

Se inicia el Asistente de registro TwidoSuite. Siga las instrucciones en pantalla para registrar la copia del software TwidoSuite.



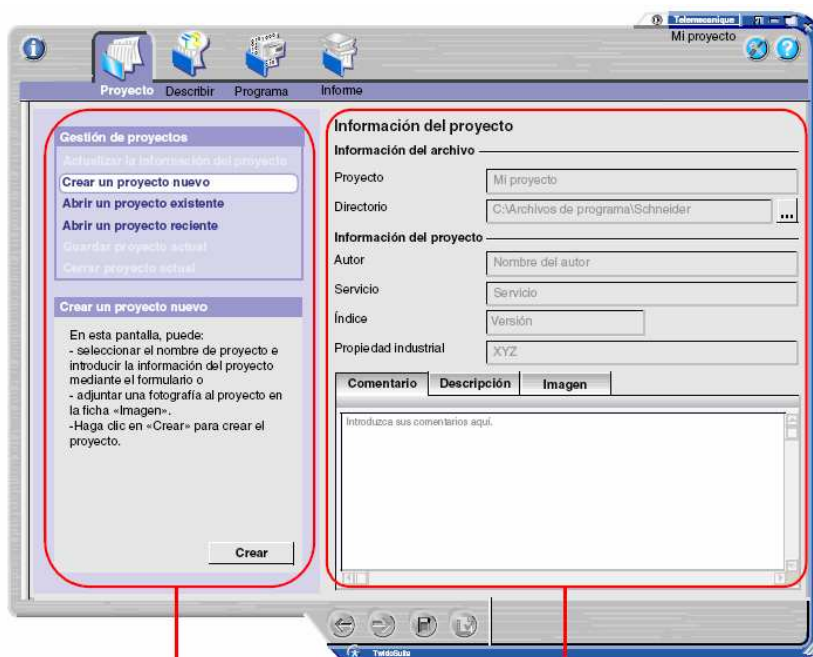
Hay cuatro formas de registrarse: por la Web, el teléfono, el fax y por correo electrónico.

Leer las “**Condiciones del servicio**”; aceptarlas para poder continuar y enviar el registro.

3.3 Crear un proyecto nuevo:

Para crear nuestro primer proyecto, seleccionar el “**Modo Programación**” y nos aparecerá el espacio de trabajo de la ventana principal de Twidosuite. Siempre que entramos aparece por defecto la ventana de proyecto, donde podremos realizar la gestión de nuestros proyectos (Crear, abrir, guardar y cerrar un proyecto).

Pulsar en “**Crear un proyecto nuevo**” dentro del marco de acciones de la ventana, acto seguido rellenar los campos de información general de nuestro proyecto, como puede ser: el nombre del proyecto, la ruta donde lo deseamos guardar, el autor, la versión, la compañía...etc.



Marco de acciones

Panel principal con la información del proyecto

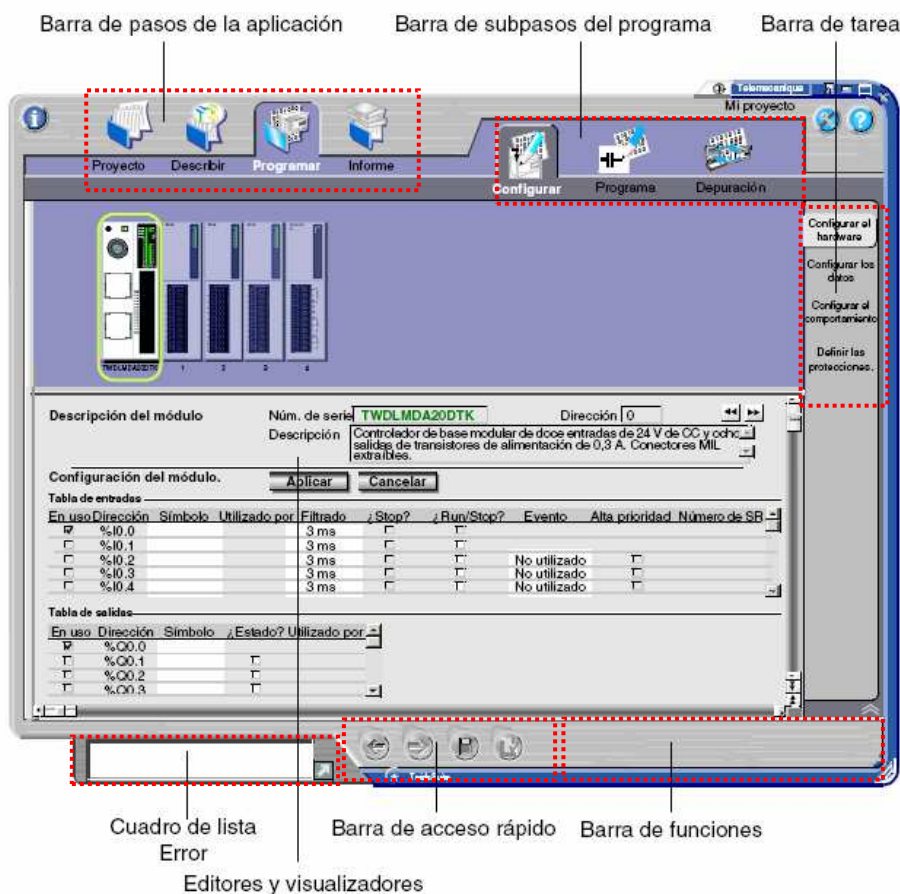
Ventana de la pestaña de “Proyecto”

Opcionalmente, también se dispone en la parte de debajo de tres pestañas que se pueden rellenar para detallar más información de la aplicación, como son: “**Comentarios**” para introducir los comentarios de la aplicación que se desean descripción de funcionamiento, notas...etc, “**Descripción**” donde aparece gráficamente la configuración de nuestra aplicación (equipos, redes) y la pestaña “**Imagen**” donde podemos cargar una imagen que se desee (logo de la compañía, esquemas).

Una vez se haya introducido la información del proyecto generaremos el archivo pulsando el botón “**Crear**” que aparece en la parte de abajo del marco de acciones.

3.4 Navegación por el espacio de trabajo de TwidoSuite

La navegación por el interface del Twidosuite es muy intuitiva y gráfica ya que sigue los pasos de ciclo de desarrollo natural de una aplicación de automatización, por eso la navegación y la comprensión de que se realiza en esa ventana es tan sencilla.



En el espacio de trabajo general siempre tendremos una serie de barras, pestañas y menús que tendrán las siguientes funciones:

- **Barra de pasos de la aplicación:** Muestra los cuatro pasos de la aplicación TwidoSuite (Proyecto, Describir, Programar, Documentar).
- **Barra de subpasos del programa:** Muestra los tres subpasos del programa (Configurar, Programa, Depuración). Aparece únicamente cuando el paso Programa está seleccionado.
- **Barra de tareas:** Proporciona acceso a todas las tareas que puede realizar en el paso o subpaso seleccionado de la aplicación.
- **Barra de funciones:** Proporciona acceso a funciones especiales asociadas a la tarea seleccionada.
- **Barra de acceso rápido:** Muestra los comandos Anterior/Siguiente y los accesos directos a Guardar y a Analizar programa en todo momento.
- **Editores y visualizadores:** Se trata de ventanas de TwidoSuite que organizan los controles de programación y configuración de manera que las aplicaciones puedan desarrollarse correctamente.
- **Barra del cuadro de lista Error:** Muestra información acerca de los posibles errores o advertencias de la aplicación.

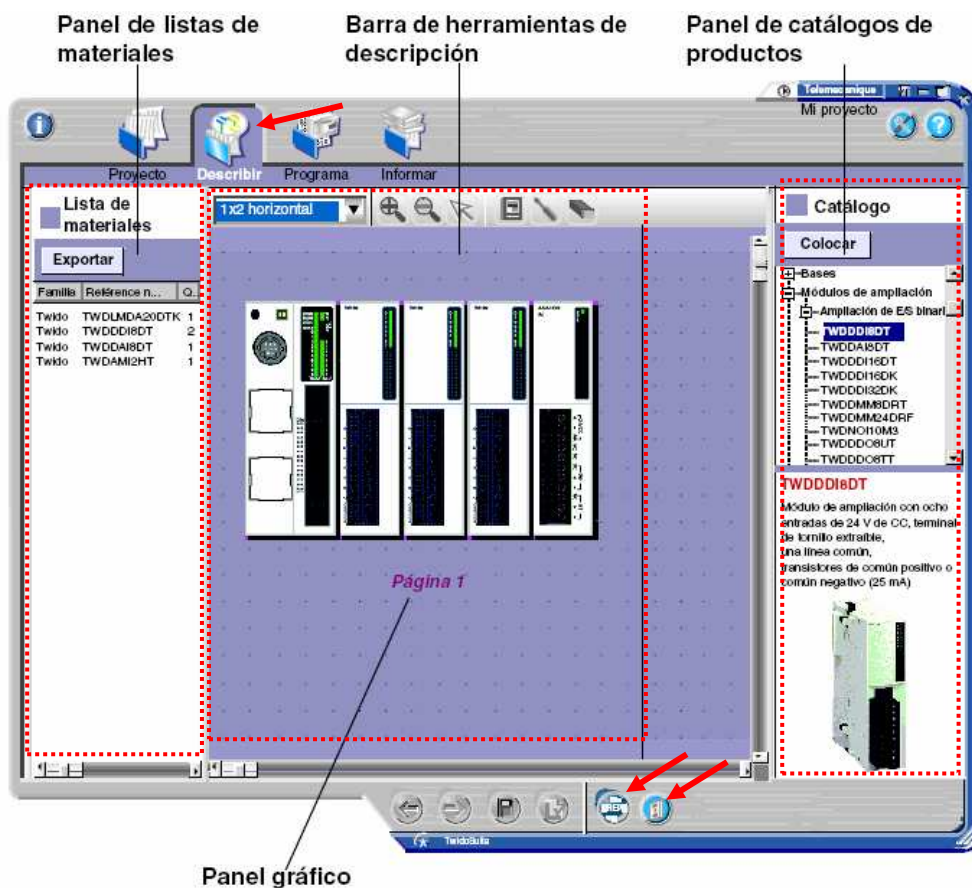
3.5 Configuración básica del Hardware Twido:

Lo primero que hay que hacer cuando se inicia la tarea de realizar un proyecto de automatización, es la configuración o descripción del hardware que se necesitará para dicho propósito, por lo tanto en función de ciertas premisas como son: El número de entradas y salidas (así como el tipo), la necesidad de memoria y velocidad en la CPU, necesidad de buses de comunicación...etc.

Todo este proceso de descripción de la aplicación desemboca en la elección de un hardware determinado que se ajuste a las necesidades de la aplicación. Siendo distinto de una aplicación a otra.

Por esta razón se tendrá que configurar el hardware en el software antes de empezar a programar.

Iniciar la configuración haciendo clic sobre el icono “**Describir**” de la barra de pasos del Twidosuite. Se abrirá la ventana de configuración, donde observaremos lo siguiente:



Ventana de configuración de Hardware “Describir”

- El **Panel gráfico** es el área donde se construye la representación gráfica de la configuración agregando elementos de hardware.
- El **Panel de catálogos de productos** muestra el catálogo de elementos de hardware Twido, incluidos la base compacta y los autómatas modulares, módulos de expansión, opciones, elementos de red en estructura con forma de árbol. La parte inferior del panel proporciona una breve descripción del elemento de hardware seleccionado.

- El **Panel de listas de materiales** es un área adicional que muestra una lista de materiales de elementos de hardware que actualmente forman el proyecto abierto. Puede hacer clic cómodamente en Exportar para guardar la lista de materiales en un archivo con formato .CSV.



La visualización del catálogo



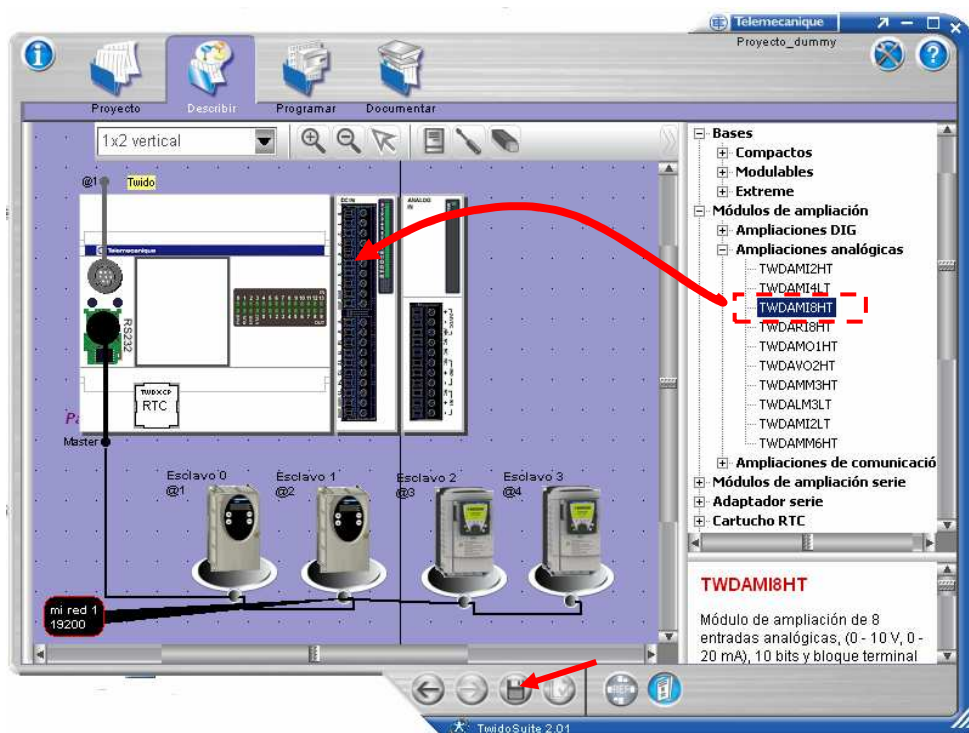
y



del panel de lista de

materiales se puede activar y desactivar haciendo clic en los iconos que se encuentran en la barra de funciones.

Para ir creando la configuración, arrastre los elementos de hardware seleccionados del panel de catálogos al panel gráfico para construir gradualmente el sistema de automatización (incluido los buses de comunicación y los equipos conectados).



Pasos de configuración Hardware



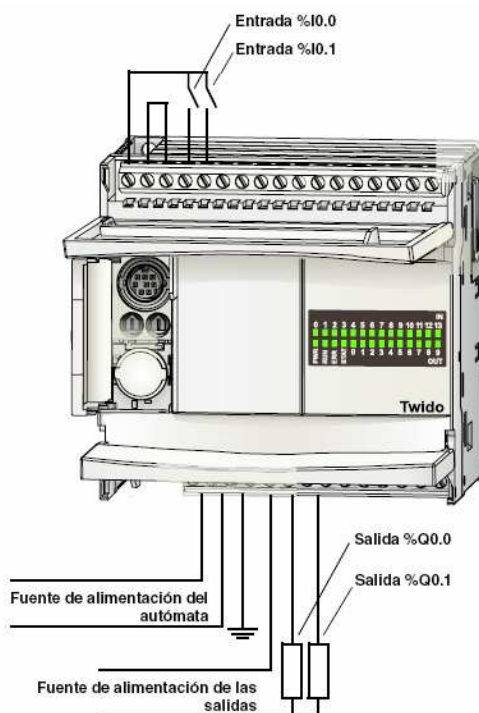
Por defecto, al abrir la ventana aparece una CPU predeterminada, se puede sustituir seleccionando y posicionando la correcta, encima de la predeterminada.

Se tendrá que ir colocando los diferentes elementos tal y como se tienen situados físicamente, cuando se haya finalizado la configuración habrá que guardarla, haciendo clic en el icono de **“Guardar”** de la barra de acceso rápido.

3.6 Edición de datos:

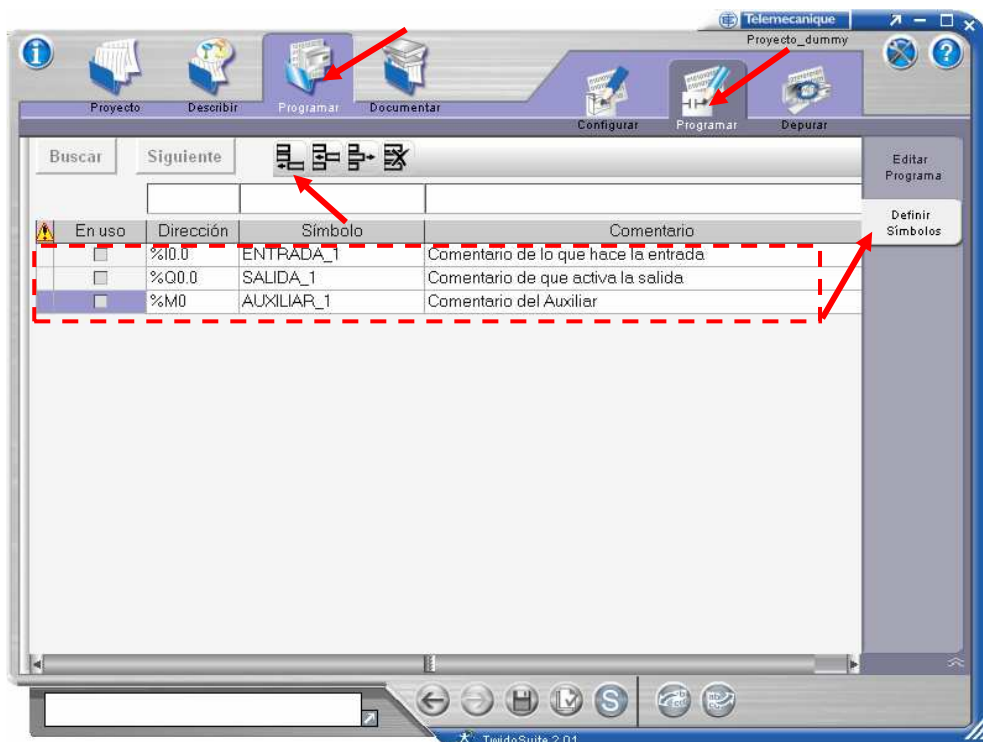
Antes de programar la aplicación, debe definir las especificaciones del cableado de las E/S de la aplicación.

Para ello se le asigna a cada una de las entradas y salidas físicas un símbolo que nos indique que realiza y nos ayude después a la mejor comprensión del programa.



Posible cableado de entradas y salidas físicas

Para definir los símbolos de las E/S de la aplicación, se irá a la pestaña **“Programar”** de la barra de pasa de la aplicación, donde aparece directamente en la pestaña **“Programa”** de la barra de subpasos del programa y tendremos que hacer clic en la opción **“Definir símbolos”** que es situada en la barra de tareas en la parte derecha de la ventana.



Ventana para la creación de símbolos

En ese momento aparecerá en el área de trabajo la ventana para definir los diferentes símbolos que utilizaremos más tarde en la programación. Para crear un nuevo símbolo, se tendrá que pulsar en el icono de creación de símbolos. En ese momento aparecerá una nueva fila en el área inferior donde rellenaremos los diferentes campos.

La ventana consta de cuatro columnas donde se pone el “**Símbolo**” (nombre descriptivo que nos aporte información de lo que realiza la señal), “**Dirección**” posición de memoria a la que está direccionada esa variable, “**Comentario**” amplía la información descriptiva de la señal y “**En uso**” que indica si ese símbolo está siendo usado en el programa.

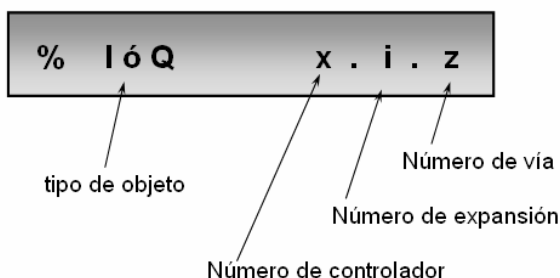
3.6.1 Tipos de variables básicas:

Una variable es una entidad de memoria de los tipos BOOL, WORD, INT, REAL...etc. Según la información que muestran se pueden definir diferentes tipos de objetos:

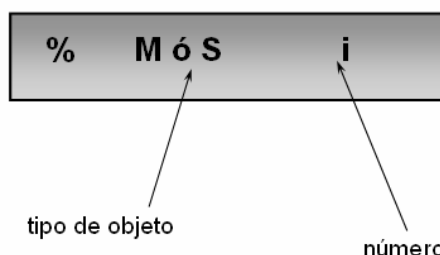
- **Los Objetos bit:** son variables binarias y como tal pueden ser consultadas por instrucciones booleanas. La información que suministran al autómata puede ser 0 ó 1 (falso ó verdadero).

Tipos de objetos de bit:

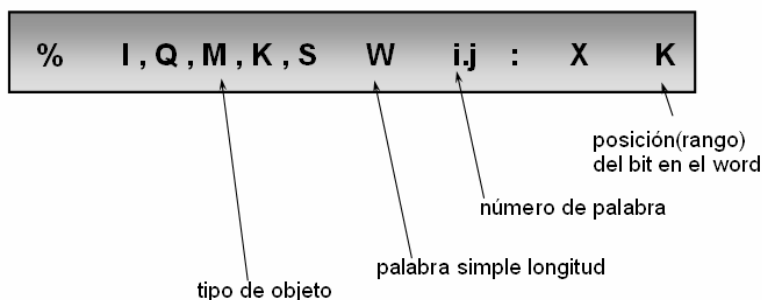
- **Bits de E/S:** Estos bits son las "imágenes lógicas" de los estados eléctricos de las E/S. Las entradas estarán direccionadas con la letra "%I" y las salidas "%Q".



- **Bits internos:** Los bits internos son áreas de memoria interna utilizadas para almacenar valores intermedios durante la ejecución de un programa. Los bits internos estarán diseccionados con la letra "%M".
- **Bits de sistema:** Los bits de sistema de %S0 a %S127 supervisan el funcionamiento correcto del autómata y la correcta ejecución del programa de aplicación. La letra para direccional los bits de sistema será la "%S".



- **Bits extraídos de palabras:** Uno de los 16 bits de algunas palabras puede extraerse como bit de operando.



- **Los Objetos de palabra:** Los objetos de palabra están direccionados en forma de palabras de **16 bits**, almacenados en la memoria de datos y que pueden contener un valor entero comprendido entre **-32.768 y 32.767**.

El contenido de las palabras o los valores se almacena en la memoria de usuario en código binario de 16 bits mediante la convención que aparece a continuación.

| | | | | | | | | | | | | | | | | | |
|---|-------|------|------|------|------|-----|-----|-----|----|----|----|---|---|---|---|---------------|------------------|
| | F | E | D | C | B | A | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Posición del bit |
| | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | Estado del bit |
| + | 16384 | 8192 | 4096 | 2048 | 1024 | 512 | 256 | 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 | Valor del bit | |

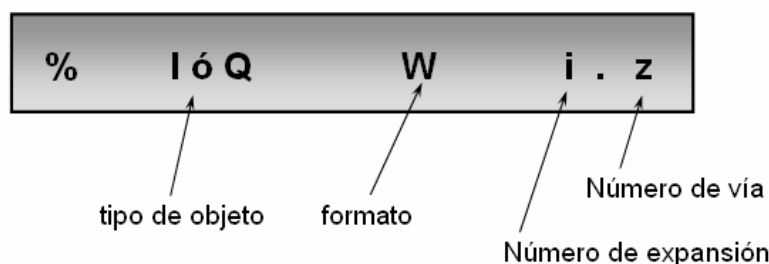
En la notación binaria con señal, el **bit 15** se asigna por convención a la señal valor codificado: Si el bit 15 se establece en **0**, el contenido de la palabra es un **valor positivo**, si en cambio es **1**, el contenido de la palabra es un **valor negativo**.

El valor de la palabra puede estar codificado en diferentes formatos: Formato **Decimal** (Mín.: -32.768, máx.: 32.767, por ejemplo, 1.579), formato **Hexadecimal** (Mín.: 16#0000, máx.: 16#FFFF, por ejemplo, 16#A536), formato de caracteres **ASCII** (por ejemplo "OK").

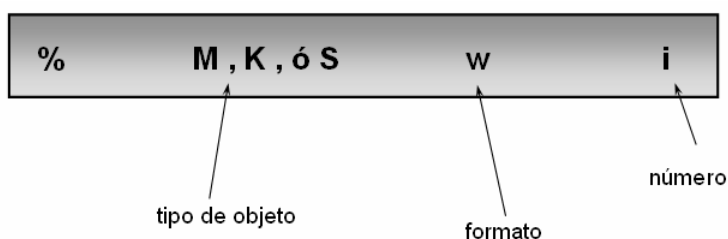
| | | | |
|------------------|----------------|--------------------|--------------|
| Binario | Decimal | Hexadecimal | ASCII |
| 0000010101001100 | = 1356 | = 16#54C | = |

Tipos de objetos de palabra:

- **E/S analógicas:** Es el valor eléctrico reflejado en las E/S analógicas. Las entradas estarán direccionadas con la letra **"%IW"** y las salidas **"%QW"**.



- **Memoria interna:** Palabras empleadas para almacenar valores durante la operación en la memoria de datos, se direcciona “%MW”.
- **Constantes:** Almacenan constantes o mensajes alfanuméricos. Su contenido solo se puede escribir o modificar mediante el Twidosuite durante la configuración. Se acceden a la zona de constantes a través de “%KW”.
- **Palabras de sistema:** Palabras del sistema que ofrecen diferentes funciones, como proporcionar acceso directo a los estados internos del autómata. La letra para direccionar los bits de sistema será la “%SW”.



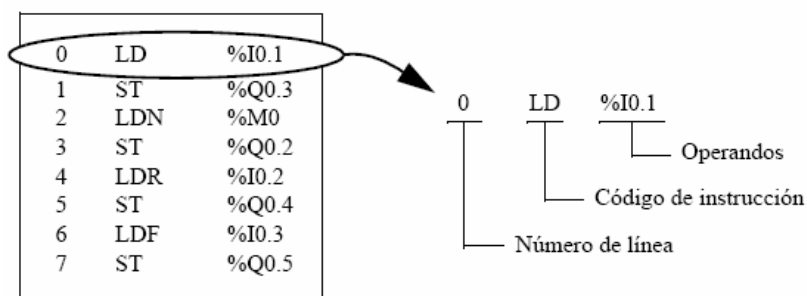
3.7 Escribir Programa:

El Twidosuite proporciona instrucciones para utilizar los lenguajes de programación **Ladder** (diagrama de contactos) y **Lista** (Instrucciones de lista) y **GRAFSET**.

- **Lenguaje Ladder:**
 Los diagramas Ladder o de contacto son similares a los diagramas lógicos de relé que representan circuitos de control de relé. Las principales diferencias entre los dos son las siguientes funciones de la programación de Ladder que no aparecen en los diagramas de lógica de relé.
 - Todas las entradas están representadas por símbolos de contactos (—|—).
 - Todas las salida están representadas por símbolos de bobinas (—(—).
 - Las operaciones numéricas están incluidas en el conjunto de instrucciones de Ladder gráficas.

- Instrucciones de Lista:**

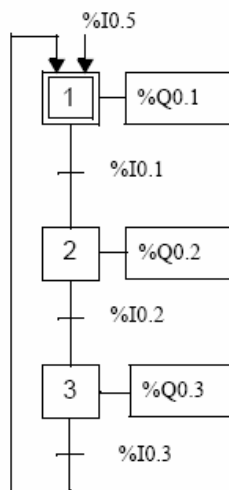
Un programa escrito en lenguaje de lista está formado por una serie de instrucciones que el autómata ejecuta de forma secuencial. Cada instrucción de lista está representada por una línea de programa y tiene tres componentes: Número de línea, Código de instrucción y Operandos.



Ejemplo de programa en Listado de instrucciones

- GRAFCET (Gráfica de control de secuencias de programación):**

Las instrucciones Grafcet de TwidoSuite ofrecen un método sencillo para traducir una secuencia de ajuste (diagrama Grafcet).



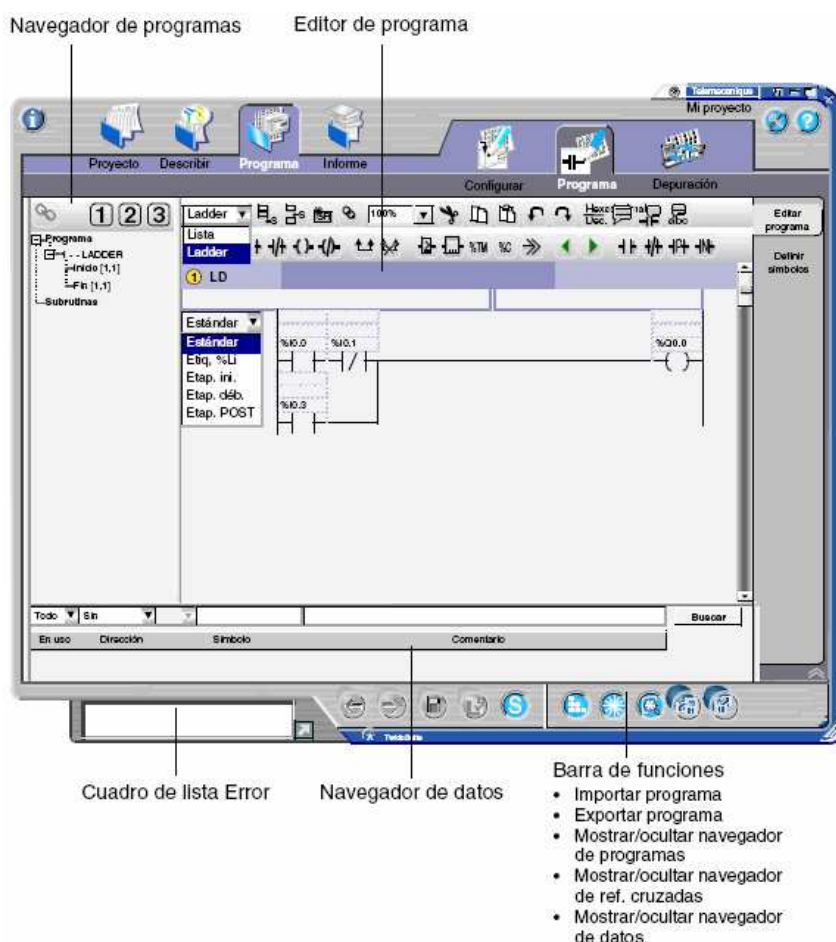
Ejemplo de programa en GRAFCET

La cantidad máxima de pasos Grafcet depende del tipo de autómata Twido. La cantidad de pasos activados en cualquier momento están limitados sólo por la cantidad total de pasos.

3.7.1 Abrir el editor de programa:

El “Editor de programa” es el área de trabajo donde crearemos la programación en el lenguaje que se haya escogido.

Para abrir el editor de programas se pulsara la pestaña de “Programar” en la barra de pasos de la aplicación y asegurarse que en la barra de subpasos del programa estamos en la pestaña de “Programa”, aparece la ventana para la edición de programa.



Ventana del editor de programa

- **Navegador de programas:** En esta área se puede observar las diferentes secciones del programa, así como las subrutinas.
- **Editor de programa:** Es el área donde se realizará la programación propiamente dicha de nuestra aplicación.
- **Barra de funciones:** Barra donde se sitúan las funciones adicionales asociadas a la programación (como exportar o importar programas).

- **Navegador de datos:** Esta ventana nos permite buscar donde están siendo usados los símbolos o direcciones, en que partes del programa.



En el editor también aparecerá una barra de instrucciones que cambiará en función del lenguaje de programación seleccionado.

3.7.1.1 Creación de un programa en lenguaje ladder:

Existen tres barras de herramientas de programación de Ladder disponibles que permiten editar programas e introducir instrucciones de Ladder gráficamente de forma sencilla:

- **barra de herramientas del programa,**
- **barra de herramientas de la paleta de Ladder,**
- **paleta de Ladder ampliada.**

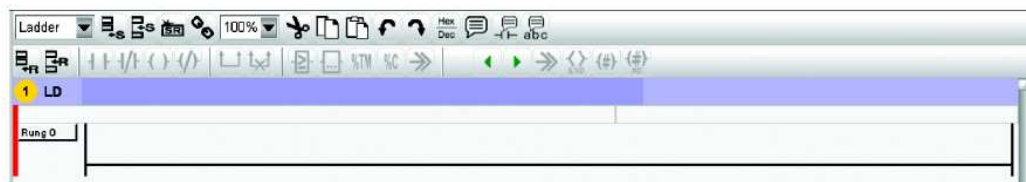
| | |
|------------------------------|--|
| 1 Alternar modo Ladder/Lista | 8 Copiar |
| 2 Agregar sección | 9 Pegar |
| 3 Insertar sección | 10 Deshacer |
| 4 Agregar subrutina | 11 Rehacer |
| 5 Cortar en dos secciones | 12 Visualizar en hexadecimal o decimal |
| 6 Porcentaje del zoom | 13 Mostrar/ocultar comentarios del escalón |
| 7 Cortar | 14 Mostrar/ocultar comentarios del Ladder |
| | 15 Mostrar/ocultar símbolos |

Barra de Herramientas Ladder

Para comenzar a programar se tiene que introducir la primera sección del programa para ello en la barra de herramientas pulsaremos el icono de “agregar una sección”.



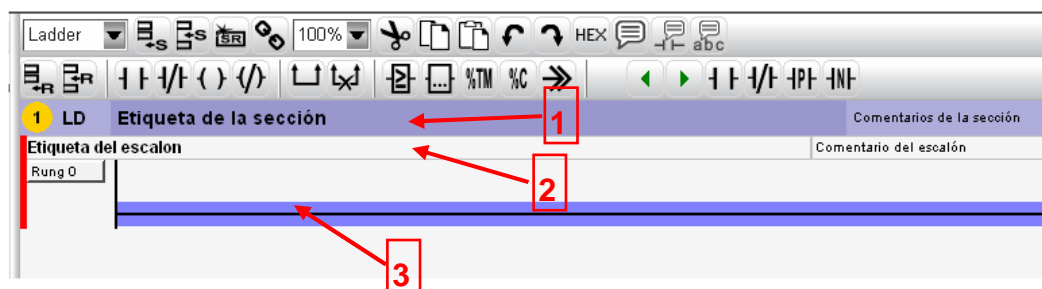
La primera sección se inserta en el Editor de Ladder Logic, de forma que muestra el primer escalón vacío, como figura a continuación. (Tenga en cuenta que un programa debe contener al menos una sección.)



Haga doble clic en el encabezado de la sección para introducir el título de la sección y los comentarios.

Haga doble clic en la cabecera de escalón para introducir título de escalón, comentarios y declaración de tipos de escalón.

Seleccione la conexión horizontal vacía que se ejecuta entre las líneas de alimentación derecha e izquierda del escalón.



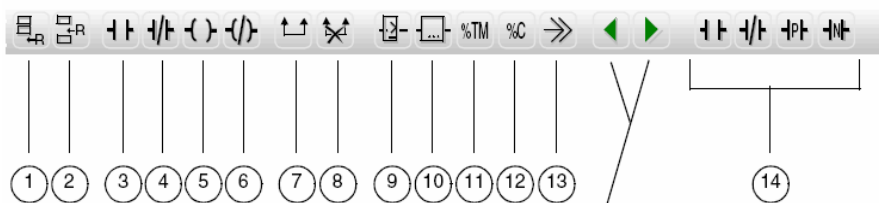
La conexión seleccionada se marca en color azul.

3.7.1.2 Inserción de instrucciones básicas:

Para la inserción de las instrucciones básicas se dispone de una paleta de Ladder para que la inserción de las instrucciones básicas sea lo más rápida posible.

Las instrucciones más utilizadas en un programa ladder son:

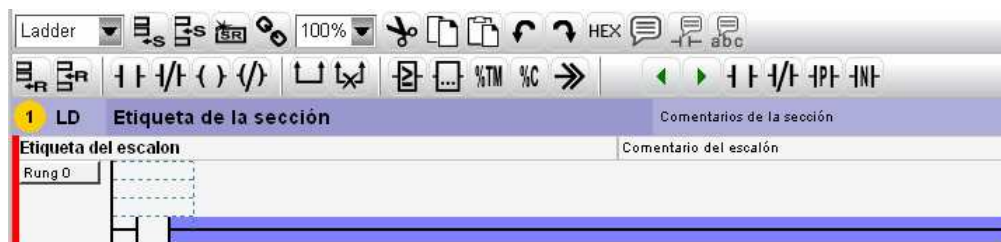
- Contacto normalmente abierto
- Contacto normalmente cerrado
- Bobina
- Bloque de comparación
- Temporizadores
- Contadores



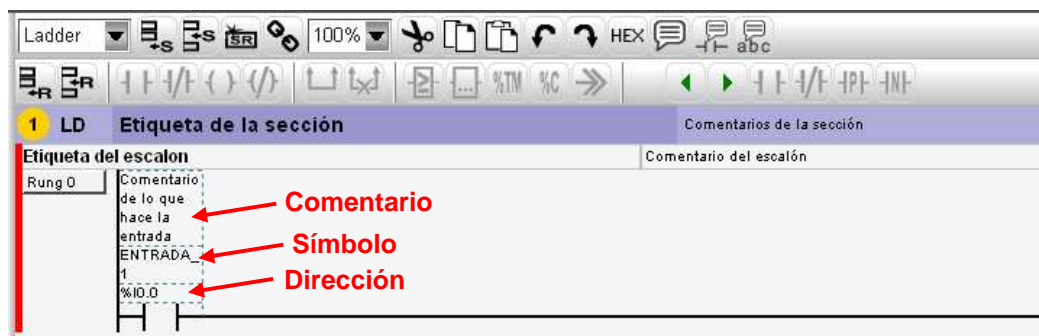
Alternar secciones de la paleta de Ladder ampliada

| | |
|---|--|
| 1 Agregar escalón | 2 Insertar escalón |
| 3 Agregar contacto normal abierto | 4 Agregar contacto normal cerrado |
| 5 Agregar bobina | 6 Agregar bobina inversa |
| 7 Agregar conexión | 8 Eliminar conexión |
| 9 Agregar bloque de comparación | 10 Agregar bloque de operación |
| 11 Agregar bloque de temporizador | 12 Agregar bloque de contador |
| 13 Agregar llamada de subrutina o salto | 14 Botones de la paleta de Ladder ampliada |

Ahora haga clic en el icono Contacto normal de la paleta de Ladder para insertar la instrucción gráfica de entrada. Y el contacto normal se inserta en la parte más hacia la izquierda del escalón.



Ahora hay que definir este contacto con que variable está relacionado, para ello nos posicionamos con el ratón en una de los dos recuadros punteados que aparecen encima del contacto y hacemos clic sobre él. Si elegimos el de justo encima tendremos que poner la dirección específica de la memoria. Si elegimos el inmediatamente superior se tendrá que poner el símbolo de la variable.



Si el símbolo de la variable está definido, aparecerá también rellenado automáticamente el otro campo y el comentario.

3.7.2 Programación ejemplo con instrucciones básicas:

Realizaremos un ejemplo de programación de una aplicación para coger la dinámica de utilización del editor de programación. Para concentrarnos únicamente en la programación ya se habrá definido una configuración y una tabla de símbolos.

La aplicación es una guillotina industrial donde tenemos que hacer trozos de plástico de una medida determinada:

- Una vez pulsemos el **botón de marcha** el motor de la cinta transportadora que suministra la materia prima se pondrá en marcha, cuando el **final de carrera** que se encuentra en la entrada de señal de que la materia prima ya ha llegado a la zona de corte, al cabo de **10 segundos** bajará la guillotina, cortando a la distancia correcta la materia prima. Este proceso se repetirá hasta que el **sensor de la cinta transportadora de salida** haya contado que en la zona de embalaje existen **100 trozos** cortados listos para embalar.

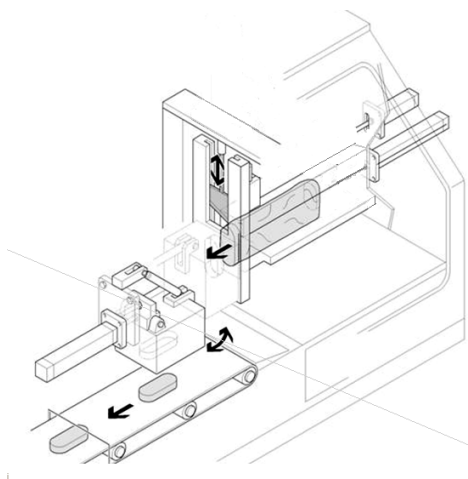


Tabla de símbolos

Entradas:

- %I0.0 – Botón de marcha
- %I0.1 – Sensor de entrada a guillotina.
- %I0.2 – Sensor de cinta de salida

Salidas:

- %Q0.0 – Cinta transportadora
- %Q0.1 – Guillotina

En este caso el programa lo podemos dividir en dos secciones:

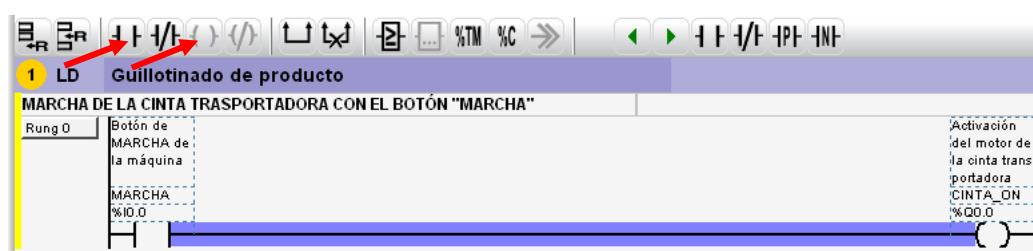
1. Sección que se encargará del proceso de corte.
2. Sección de la gestión del contador de piezas cortadas.



La descomposición de una aplicación en trozos más pequeños (estructuración) facilita enormemente la programación y la posterior depuración del mismo.

Para ello crearemos la primera sección donde programaremos el proceso de corte. Entrando en la ventana del editor de programación, insertaremos la primera sección “**Guillotinado de producto**”.

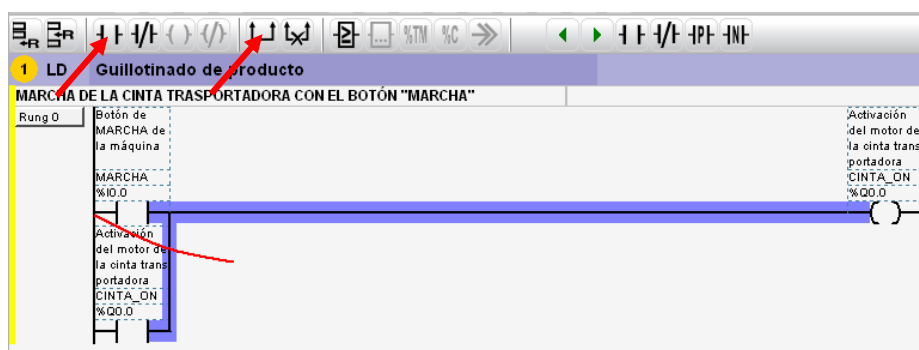
En el primer escalón programaremos la puesta en marcha de la cinta transportadora, la condición que se había definido para la puesta en marcha de la cinta era pulsar el botón de marcha. Por lo que colocaremos un contacto normalmente abierto que será la entrada del botón de marcha y una activación o bobina que será la salida de activación de la cinta.




Ahora para que la cinta permanezca en marcha sin tener que tener el botón pulsado constantemente, enclavaremos el botón de marcha con la señal de salida, creando otra rama en el escalón con la condición en paralelo.

Pulsar en el trozo de línea que queda entre el contacto de marcha y la línea vertical, y manteniéndolo pulsado, moverse hacia el otro extremo del contacto, observaremos que sale una especie de arco de color rojo que se torna verde cuando toca el otro extremo de la línea, cuando esté en verde dejar de pulsar y se creará el segmento.

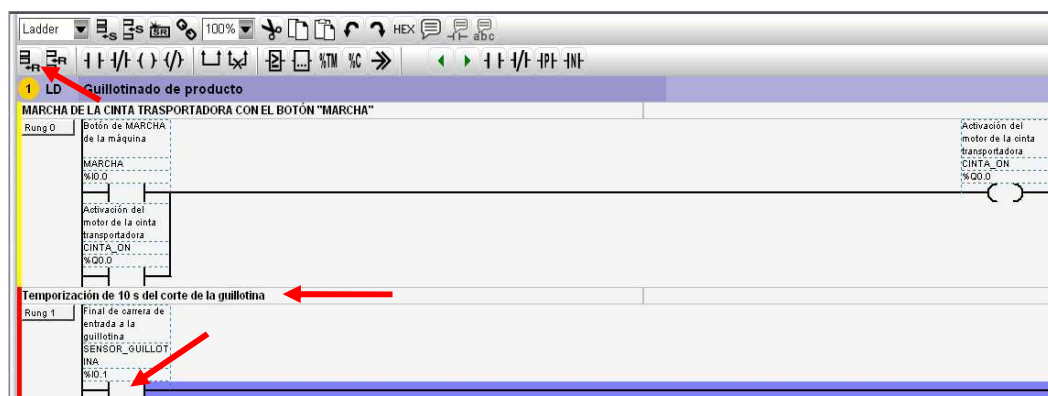
Seleccionar el segmento e introducir un contacto normalmente abierto de la señal de salida de la cinta.




Ahora una vez pulsado el botón ya se podrá soltar, ya que la bobina queda alimentada con su propia condición de activación, más adelante colocaremos una condición para desenclavar y por lo tanto parar la cinta (cuando se hayan cortado 100 piezas).

Ahora crearemos otro escalón (No otra sección) donde realizaremos la temporización de 10 s. Para insertarlo pulsaremos el icono de escalón. 

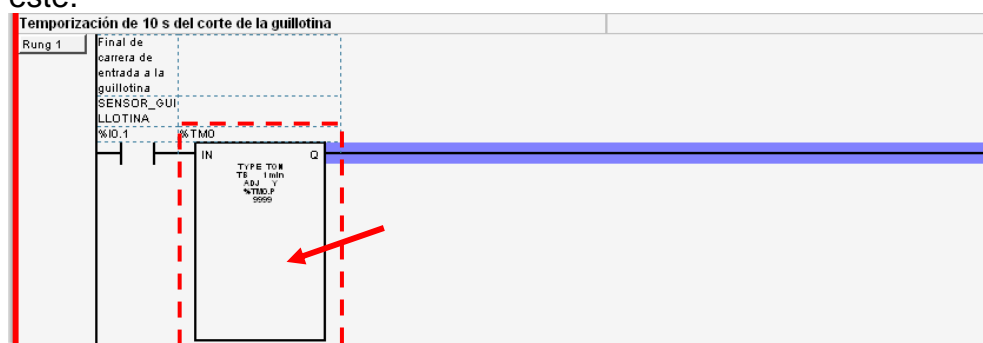
Después introduciremos un contacto normalmente abierto que será el sensor de entrada a la guillotina que será lo que activará la temporización.



3.7.2.1 Introducción de un bloque temporizador:

Los temporizadores son instrucciones que tienen su propio icono de inserción en el programa, para ello tendremos que pulsar en el icono. 

Con esta acción, lo único que hemos hecho es introducir el bloque temporizador, que acto seguido tendremos que configurar su comportamiento. Haciendo doble clic sobre el bloque de temporización nos lleva directamente a la ventana de configuración de este.



En la ventana de configuración tendremos las siguientes opciones de configuración de temporizadores:



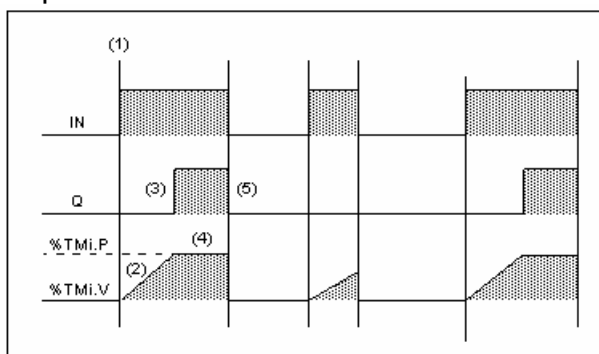
En la ventana de configuración, aparecen todos los temporizadores disponibles.

Seleccionamos la primera fila y en la columna “**Símbolo**” introducimos el nombre que deseamos que tenga nuestro configurador (Ej. T_distancia).

En la columna tipo elegiremos entre las tres funciones de temporización que se disponen. Estas son:

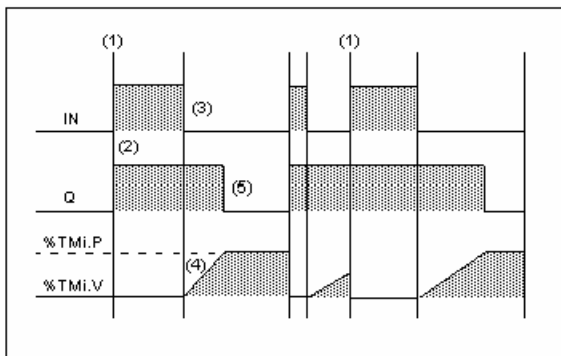
- **TON (predeterminado) –“Temporizador de retardo a la conexión”:**

(1) Cuando hay un flanco de subida en la entrada del temporizador “IN” de 0»1, (2) el temporizador comienza a contar el tiempo que le hemos definido “%TMI.P”, (3) cuando ese tiempo se ha cumplido el temporizador pone su salida “Q” de 0»1, (5) la salida queda activada mientras la entrada también siga a 1, en el momento que la entrada cae de 1»0 también lo hace el temporizador.



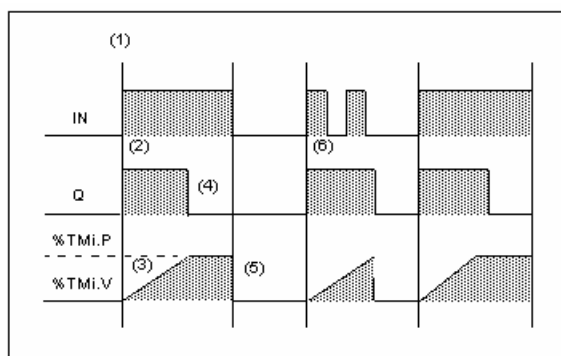
○ **TOF – Temporizador de retardo a la desconexión:**

(1) Cuando hay un flanco de subida en la entrada del temporizador “IN” de 0»1 , (2) La salida del temporizador “Q” pasa también de 0»1, (3) cuando la señal de entrada pasa de 1»0, (4) el temporizador comienza a contar el tiempo que se le ha configurado “%TMI.P” , (5) en el momento que se cumple el tiempo de preselección, la salida del temporizador pasa de 0»1.



○ **TP - Pulso a la temporización:**

(1) Cuando hay un flanco de subida en la entrada del temporizador “IN” de 0»1 , (2) La salida del temporizador “Q” pasa también de 0»1, (3) y el temporizador comienza a contar el tiempo que se le ha configurado “%TMI.P” , (4) en el momento que se cumple el tiempo de preselección, la salida del temporizador pasa de 0»1, (5) aunque el temporizador permanecerá activo hasta que la señal de entrada pase de 0»1.



De estos tres tipos de temporizador elegiremos el de retardo a la desconexión “TOF”.

Una vez seleccionado el tipo de temporizador, en la columna de “**Base**” se tiene que elegir la base de tiempos en la que se va entrar el tiempos entre: minutos, segundos, 100 milisegundos, 10 milisegundos o 1 milisegundo. En este caso elegiremos la base de tiempos de segundos (**1 s**).

Para acabar en la columna de “**Preselección**” se tiene que colocar el tiempo que deseamos, en nuestro caso (**10 segundos**).

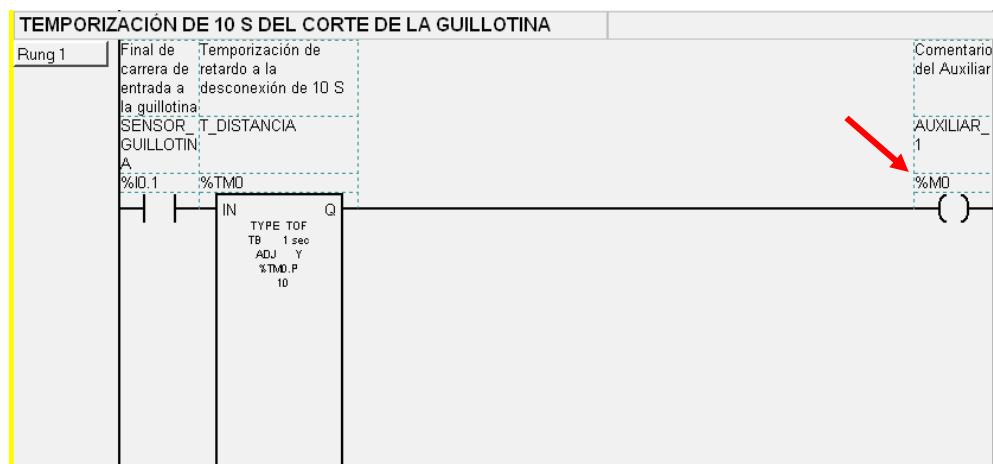
| Usos | %TM | Símbolo | Tipo | Base | Preselección | Ajustable |
|--------------------------|------|-------------|------|------|--------------|-------------------------------------|
| <input type="checkbox"/> | %TMO | T_DISTANCIA | TOF | 1 s | 10 | <input checked="" type="checkbox"/> |

La columna de “**Ajustable**” si está habilitada nos permite cambiar el tiempo de preselección desde el programa a través de valor “**%TMi.P**”.

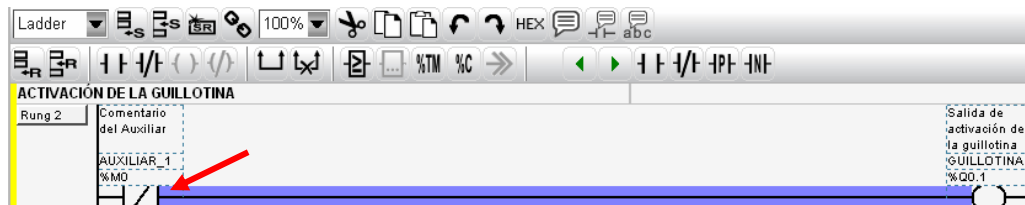
Para volver a la ventana de edición de programa, tendremos que pulsar nuevamente la pestaña de “Programa” en la barra de subpasos de programa.



Para acabar el escalón introduciremos en la bobina un bit de memoria auxiliar “**%M0**” con símbolo “**AUXILIAR_1**” para que nos ayude a la activación de la guillotina. Quedándonos el escalón de la siguiente manera.



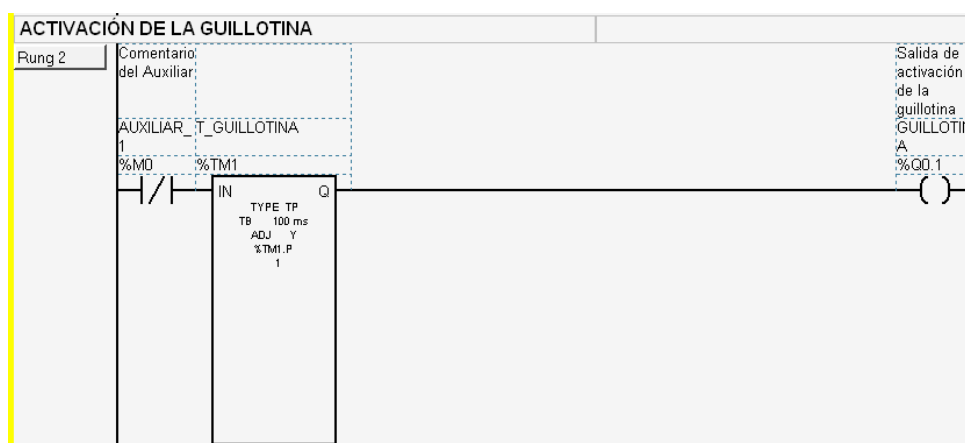
Crearemos un nuevo escalón para la activación de la guillotina. La activación de la guillotina tendrá como condición el contacto normalmente cerrado del bit de memoria auxiliar “%M0”.



Pero la guillotina tiene un tiempo determinado para que baje, corte bien y después se retraiga para volver a dejar a pasar la materia prima. Como es un funcionamiento **temporizado por pulso**, intercalaremos entre el contacto y la bobina otro temporizador “T_GUILLOTINA”, pero esta vez será del tipo “TP” con un tiempo de preselección de “100 ms”. Quedando el temporizador configurado:

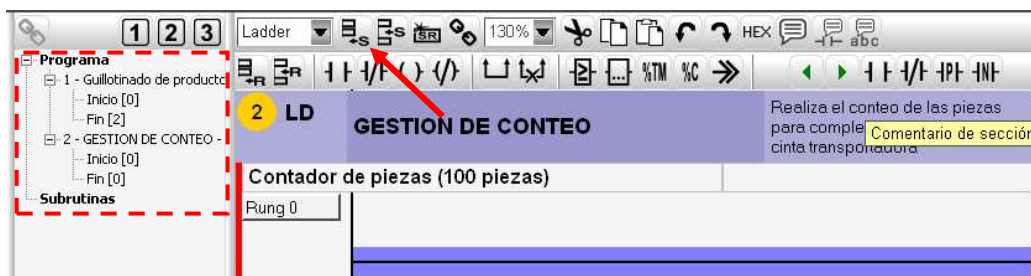
| Uso | %TM | Símbolo | Tipo | Base | Preselección | Ajustable |
|-------------------------------------|------|--------------|------|--------|--------------|-------------------------------------|
| <input checked="" type="checkbox"/> | %TM0 | T_DISTANCIA | TOF | 1 s | 10 | <input checked="" type="checkbox"/> |
| <input checked="" type="checkbox"/> | %TM1 | T_GUILLOTINA | TP | 100 ms | 1 | <input checked="" type="checkbox"/> |


Finalmente el escalón queda de la siguiente manera:



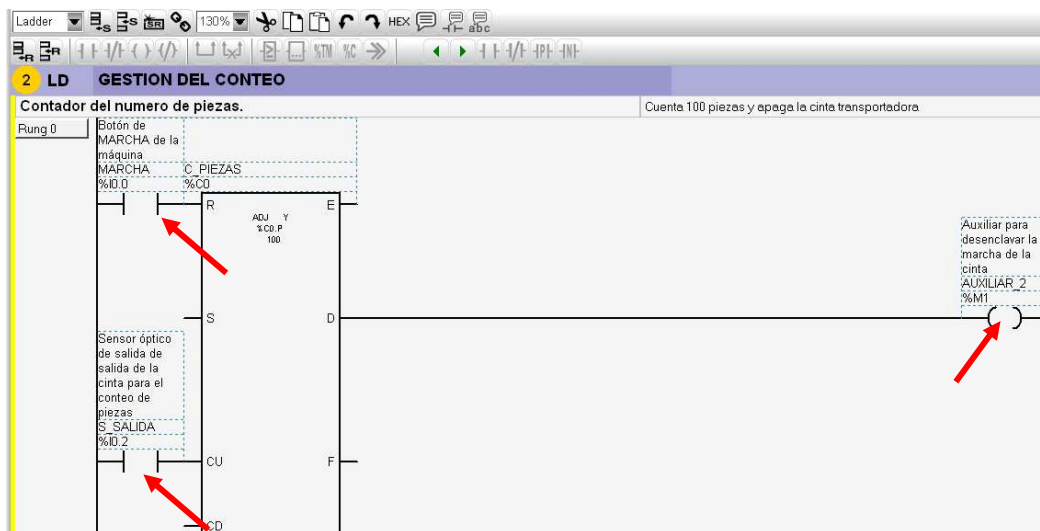
Una vez programado la sección del funcionamiento de la guillotina, iniciaremos la programación de la siguiente sección que es el conteo de piezas para la parada posterior de la cinta transportadora.

Crearemos una nueva sección a la que llamaremos “Gestión de conteo”.



Agregaremos una instrucción de contador seleccionando la línea y pulsando el icono. 

La detección del número de piezas se realiza a través de un sensor óptico de barrera que se encuentra al final de la cinta transportadora. Este será la condición que haga incrementar el contador. Por lo que, colocaremos un contacto normalmente abierto con la señal del sensor de salida en la entrada del bloque contador “CU” que cada vez que recibe una señal incrementa este. También colocaremos la señal de marcha en la entrada “R”, para que cada vez que iniciemos la guillotina ponga el contador a 0.

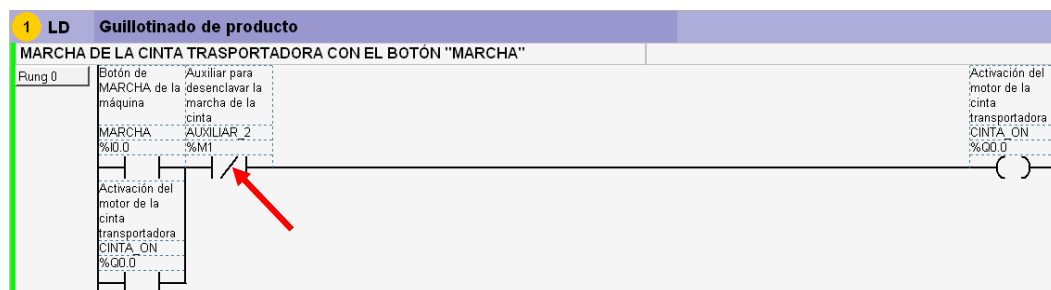


Finalmente colocaremos una bobina con un bit auxiliar “%M1” que cuando el contador llegue a las **100 piezas** y de señal, pare o “desenclave” la cinta.

Ahora tendremos que configurar el contador, para ello hacemos doble clic en el bloque y nos aparecerá la ventana de configuración, una vez en ella seleccionamos la primera fila de todos lo que están disponibles y en ella rellenamos la columna “**Símbolo**” con el nombre que deseamos que tenga (C_PIEZAS), en la columna “**Preselección**” colocaremos el número de piezas en la que queremos que el contador nos de señal (100). Y como en los temporizadores si seleccionamos la casilla de “**Ajustable**” permitiremos que dicha preselección se pueda cambiar a través de programa desde “**%C0.P**”.

| Usd | %C | Símbolo | Preselección | Ajustable |
|--------------------------|-----|----------|--------------|-------------------------------------|
| <input type="checkbox"/> | %C0 | C_PIEZAS | 100 | <input checked="" type="checkbox"/> |
| <input type="checkbox"/> | %C1 | | 9999 | <input checked="" type="checkbox"/> |
| <input type="checkbox"/> | %C2 | | 9999 | <input checked="" type="checkbox"/> |

Para finalizar la programación, solo tendremos que ir a la primera sección y en el primer escalón colocar una condición de normalmente cerrada del bit auxiliar que hemos utilizado para que de señal de que el contador ha llegado a la preselección.

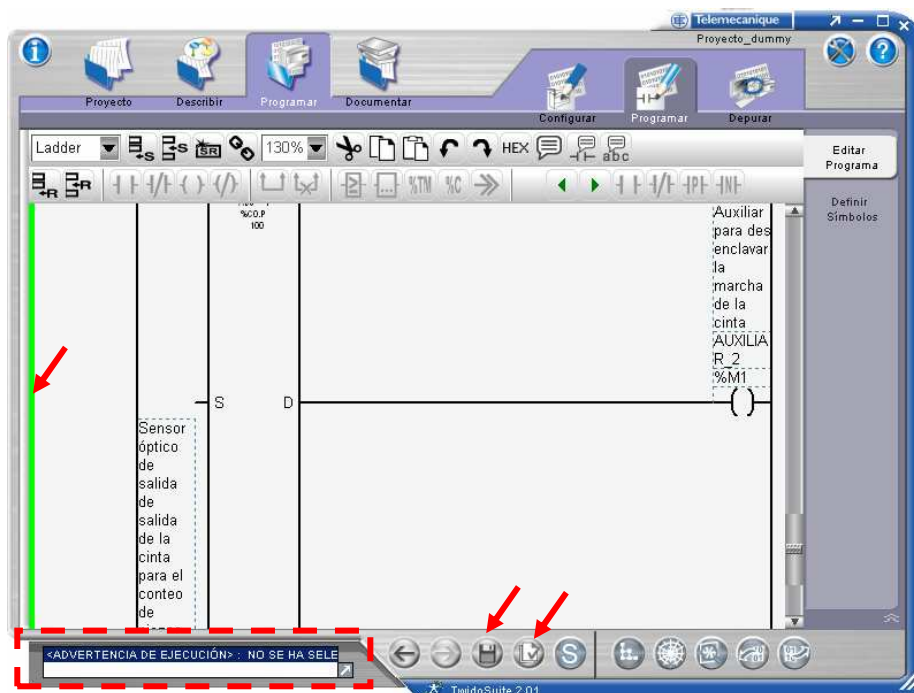


Con esta pequeña modificación en la primera sección, ya habríamos finalizado el programa. Observamos que en el borde de cada uno de los escalones hay una barra de color. En función del estado del escalón el color pueda cambiar entre rojo>amarillo>verde, el significado de los colores es el siguiente:

- **Rojo:** para los escalones con error.
- **Amarillo:** para los escalones no validados.
- **Verde:** para los escalones validados y correctos.

3.7.3 Validación y guardado del programa:

Cuando se ha realizado el programa, tendremos que asegurarnos que este no tiene errores antes de cargarlo en el autómata. Por eso una vez finalizado la programación ó durante el proceso de creación del mismo, pulsando el icono situado en la barra de acceso rápido “Analizar Programa”.



Si hay algún error en el programa, aparecerá en el la ventana del **cuadro de lista de error** y el color del lateral del escalón que contiene el error permanecerá en **rojo**. Sino hay ningún error, no aparecerá nada en el cuadro de error y el lateral de todos los escalones se pondrán en **verde**.

Sólo nos dejará guardar el programa, una vez que no contenga errores, para guardarlo pulsar sobre el icono de “**Guardar**” que se encuentra en la barra de acceso rápido.

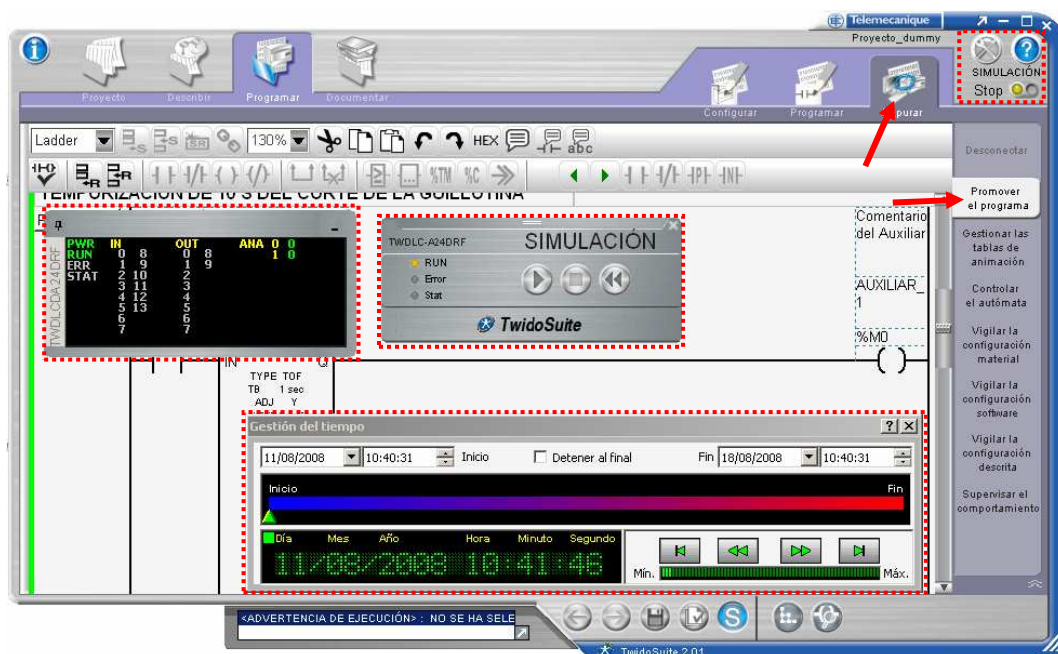


3.7.4 Simulación del programa:

TwidoSuite incorpora una función de simulador que permite probar un programa recién escrito sin tener que cargarlo en el controlador. El Simulador Twido permite ejecutar el programa y probar diferentes parámetros para ver si se dan las respuestas esperadas.

Inicio del Simulador Twido

Para iniciar el Simulador Twido, seleccione **Programa** → **Promover Programa** y haga clic en el botón del Simulador Twido en la esquina inferior derecha de la pantalla TwidoSuite.



El TwidoSuite cambia automáticamente a **Programa** → **Depuración** → **Promover el programa**. También aparecerá en la esquina superior derecha que estamos en modo simulación y en que estado se encuentra “RUN” o “STOP”.

Si estamos en modo simulación en el Twidosuite aparecerá un icono en la barra de tareas de Windows.

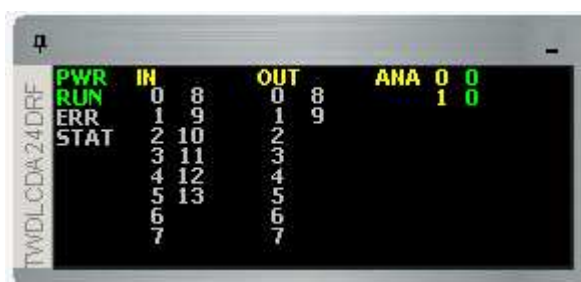


En el área de trabajo, aparece la ventana de descripción general del Simulador Twido. Emerge el escritorio digital. Si se configura o incrusta un RTC, también aparece el panel Gestión del tiempo.

El Simulador carga la configuración que ha creado en la pantalla **Describir**.

En el modo simulación se pueden realizar las mismas funciones que en el modo depuración cuando conectamos directamente al autómeta Twido. Pero con la diferencia que en este caso no necesitamos físicamente el autómeta.

La ventana de descripción general del Simulador Twido muestra el estado de las entradas/salidas de todos los módulos de ampliación y de controlador base, descritos.



En esta ventana se puede visualizar los indicadores del estado de la CPU y las activaciones de las entradas y salidas digitales configuradas (**Blanco=Desactivado; Verde=Activado**) y el valor de las entradas y salidas analógicas.

También aparece la **consola de control**, para poder controlar el “autómeta virtual”, al igual que en el modo online, nos permite poder ejecutar o parar el programa.



Los indicadores luminosos **RUN**, **ERR** y **STAT** se simulan en el escritorio digital del Simulador Twido tal y como aparecerían en un controlador base conectado (para obtener más detalles sobre estos indicadores luminosos)

La simulación del autómatas se hace de la misma manera que en el modo de depuración online, es decir, poder **forzar y modificar las variables**, creando una tabla de animación y veremos la ejecución del programa de manera gráfica con la única diferencia que no estamos conectados físicamente a un autómatas Twido real.

El **panel de control Gestión del tiempo** permite controlar la temporización de la simulación (una acción en el panel de control Gestión del tiempo modifica el tiempo RTC en consecuencia e inmediatamente).



1. Tiempo de inicio: tiempo al inicio de la simulación.
2. Período de simulación: Si esta casilla está seleccionada, cuando se llegue al tiempo final, el simulador se detiene y aparece una ventana emergente con el texto siguiente: Tiempo final alcanzado. El simulador se cambiará al modo DETENER. Si esta casilla no está marcada, la simulación permanece en modo EJECUTAR.
3. Tiempo final: tiempo al final de la simulación.
4. Un cursor indica la progresión de tiempo actual. Puede mover manualmente el cursor para fijar la hora actual.
5. Estado del simulador: Detener, EJECUTAR, avance rápido, retroceso rápido.
6. Indica la hora actual de la simulación.
7. Volver al tiempo de inicio.
8. Retroceder rápidamente en el tiempo.
9. Avanzar rápidamente en el tiempo.
10. Ir al tiempo final.
11. Fijar el nivel de avance rápido/retroceso rápido en el tiempo.



La hora inicial predeterminada de la simulación es la hora de su PC.

La duración predeterminada de la simulación es de siete días.

3.7.5 Descarga del programa en el PLC:

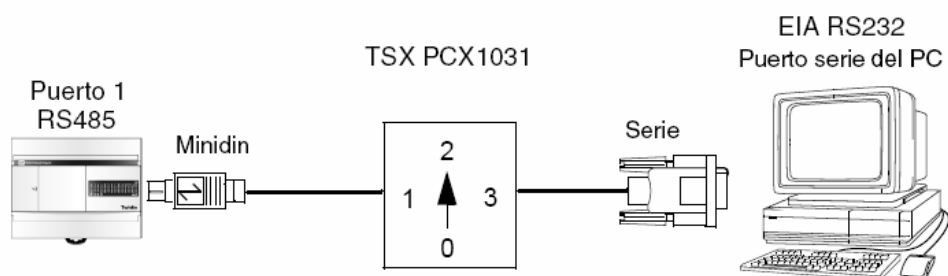
En esta sección, aprenderá a cómo transferir el programa al PLC Twido y empezar a ejecutar el programa en el controlador. Tenga en cuenta que un programa debe guardarse antes de poder transferirse al PLC.

○ **Transferencia del programa**

Para ejecutar la aplicación cuando vaya a depurarla, primero debe transferirla a la RAM del controlador.

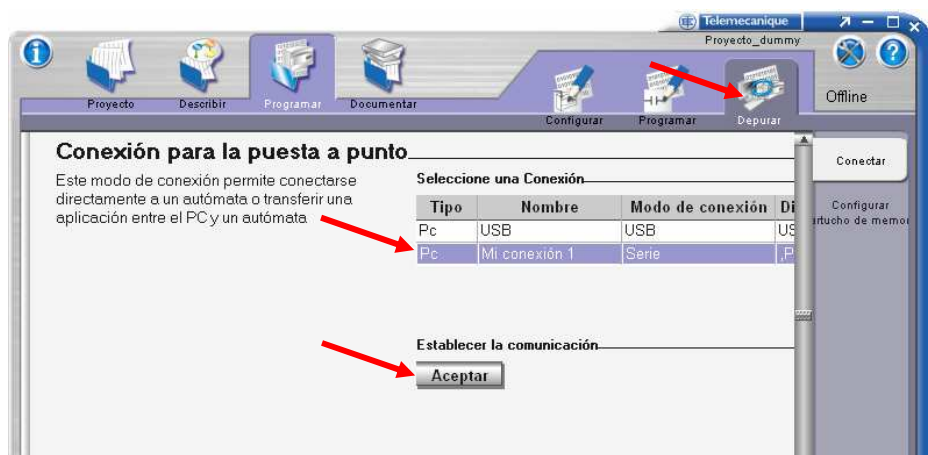
El puerto serie EIA RS232C de su PC se puede conectar al **puerto 1** del autómatas con el cable **TSX PCX1031**. Este cable convierte las señales comprendidas entre EIA RS232 y EIA RS485.

El cable TSX PCX1031 incorpora un conmutador rotativo de cuatro posiciones para seleccionar distintos modos de funcionamiento. El conmutador designa las cuatro posiciones como "0-3" y el ajuste apropiado de TwidoSuite para el autómatas Twido es la posición **2**.



Para transferir el programa de aplicación al PLC, si aún no ha conectado el PC al controlador, seleccione una conexión en la tabla de conexiones de la tarea **Programa** → **Depuración** → **Conectar** y haga clic en **Aceptar**.

El TwidoSuite intenta establecer una conexión con el controlador y realiza comprobaciones de sincronización entre el PC y las aplicaciones del PLC.



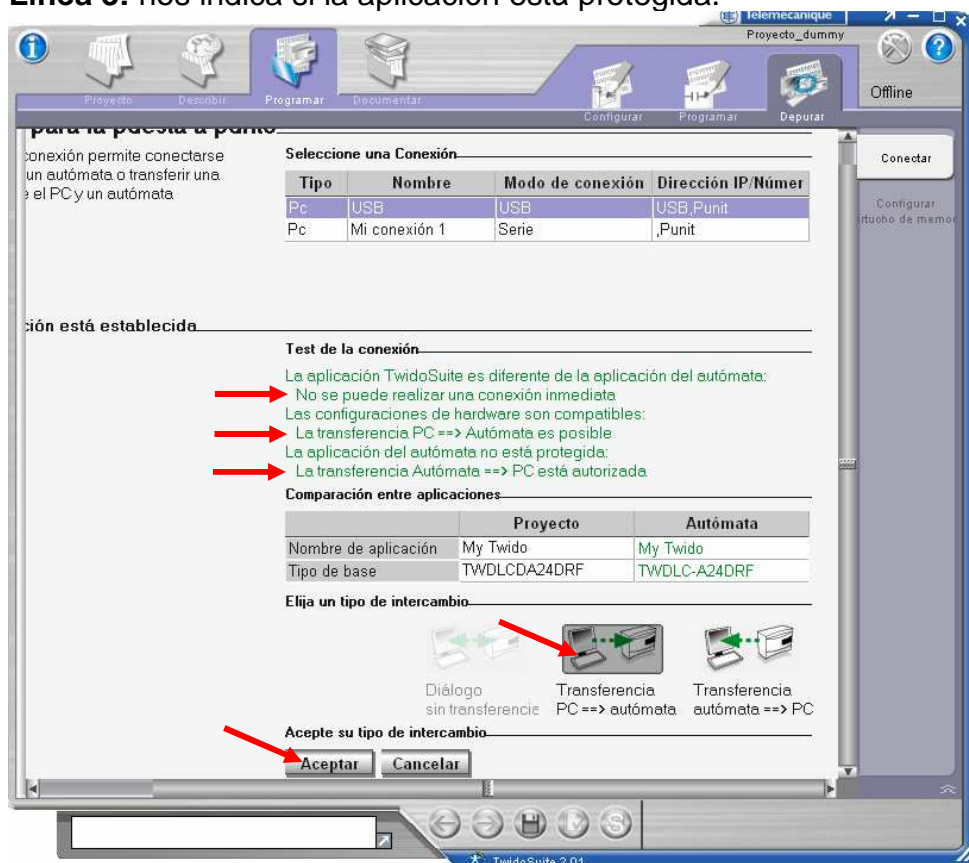
Hay diferentes tipos de conexión para conectarse al Twido, vía puerto serie COM, vía cable USB, vía Ethernet. En nuestro caso utilizaremos el puerto COM (serie) para comunicarnos.

El **Test de conexión** que aparece tras pulsar “Aceptar” muestra los resultados de tres comprobaciones de sincronización.

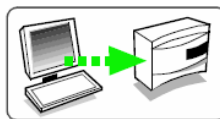
Línea 1: comparación entre el PC y las aplicaciones del PLC.

Línea 2: comprobación de compatibilidad del hardware.

Línea 3: nos indica si la aplicación está protegida.



Si el PC y las aplicaciones del PLC son diferentes, seleccione el icono del **“Transferir PC => Controlador”** que ha quedado habilitado.



*Si la aplicación del controlador está protegida con una contraseña, recibirá un aviso para introducir la contraseña. Escriba la contraseña y seleccione **“Aceptar”** para continuar con la transferencia.*

Si se está ejecutando el controlador, aparecerá un cuadro de advertencia que indica que, para continuar con la transferencia, deberá detenerse el controlador. La transferencia comienza cuando se detiene el controlador. Si el controlador está en ejecución, seleccione **“Aceptar”** para detenerlo y continuar con la transferencia. Para cerrar el cuadro y cancelar la transferencia, seleccione **“Cancelar”**.

Aparecerá un cuadro de diálogo de advertencia que indica que se sobrescribirá el contenido de la aplicación del controlador. Seleccione **“Aceptar”** para continuar con la transferencia.




Los comentarios referentes a los símbolos y al programa no se transfieren al controlador.


○ **Ejecución del programa en el Twido:**


Cuando haya acabado la transferencia del programa aparecerá una consola de control, que nos dirá el **estado del Twido Online**.



Esta estará formada con tres botones para activar los estados del PLC de modo Online:

 **Ejecutar:** Cuando se está ejecutando el autómeta, también lo estará haciendo el programa de aplicación. Las entradas del autómeta se actualizan y los valores de datos se establecen con arreglo a las instrucciones de la aplicación. Éste es el único estado en el que se actualizan las salidas reales.

 **Detener:** Cuando se detiene el autómeta, la aplicación no se estará ejecutando. Las entradas del autómeta se actualizan y los datos internos se mantienen con los últimos valores. Las salidas no se actualizan en este estado.

 **Inicializar:** Cuando se inicializa el autómeta, la aplicación no se ejecutará. Las entradas del autómeta se actualizan y los valores de datos se establecen con su estado inicial. Las salidas no se actualizan en este estado.

También consta de tres pilotos de estado RUN, ERR y STAT en el autómeta base se simulan en el Panel del autómeta:

| RUN | ERR | Descripción |
|-----------------------|-------------------|--|
| Apagado | Rojo parpadeante | No existe configuración en el autómeta |
| Luz amarilla continua | Apagado | Autómeta detenido |
| Luz verde continua | Apagado | El autómeta está en funcionamiento |
| Verde parpadeante | Rojo parpadeante | Autómeta detenido |
| Apagado | Luz roja continua | Error de hardware o de sistema. Reiniciarse en frío. |

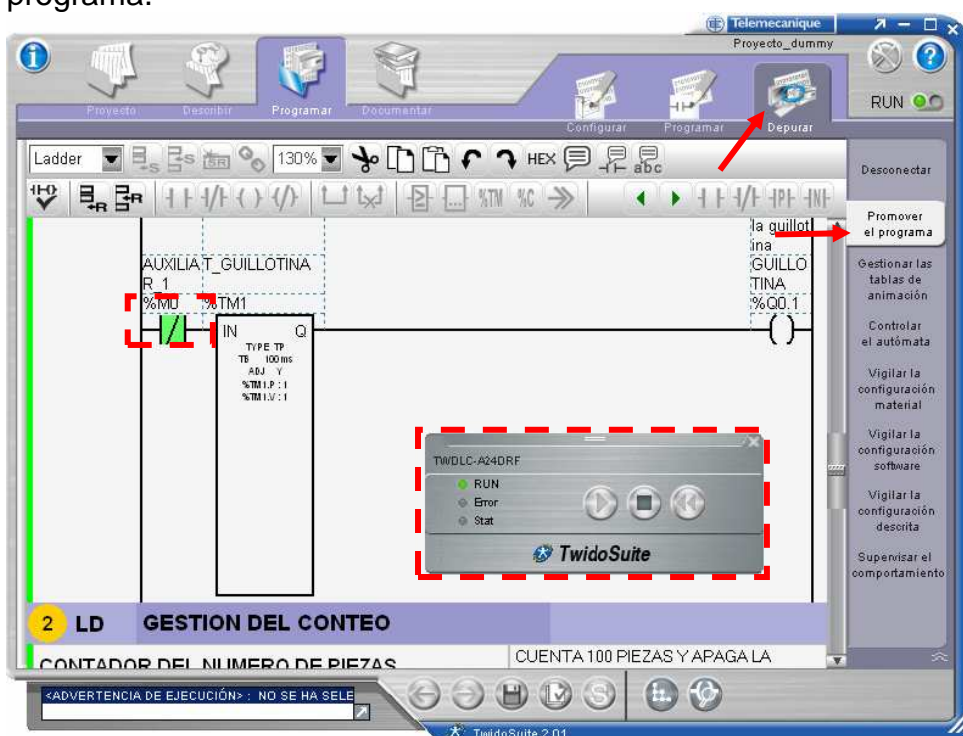
3.7.6 Depuración del programa del PLC:

Tras cargar el programa en el PLC y estar conectado en modo online con el autómeta, el Twidosuite dispone de varias herramientas para poder verificar el correcto funcionamiento del programa y realizar los ajustes necesarios para que la aplicación real vaya como se desea.

○ **Monitorización del programa del PLC:**

Puede animar los programas de Ladder y Lista cuando esté en línea, pudiendo supervisar el funcionamiento del programa de una manera gráfica. Para ir al programa animado, una vez conectado, seleccione la tarea **Programa** → **Depurar** → **Promover Programa** para abrirlo.

El editor de programa muestra elementos de Ladder animados. Observe los elementos de Ladder animados durante la ejecución del programa.



Puede observar cómo se enciende la luz de fondo de los objetos lógicos como contactos, bobinas y objetos especiales cuando su valor lógico pasa a ser 1.

Puede realizar un seguimiento de las variables de datos de los bloques de función, de comparación y de operación, incluidos los valores actuales y los preseleccionados.



Los operandos binarios se muestran como 0 ó 1, mientras que el resto de los valores se muestran en forma hexadecimal o decimal, según el formato seleccionado en Preferencias.

○ **Tablas de animación:**

Las tablas de animación se utilizan para ver y ajustar las variables del programa **en línea** cuando un PLC esté en modo Ejecutar o Detener.

Con una tabla de animación, puede:

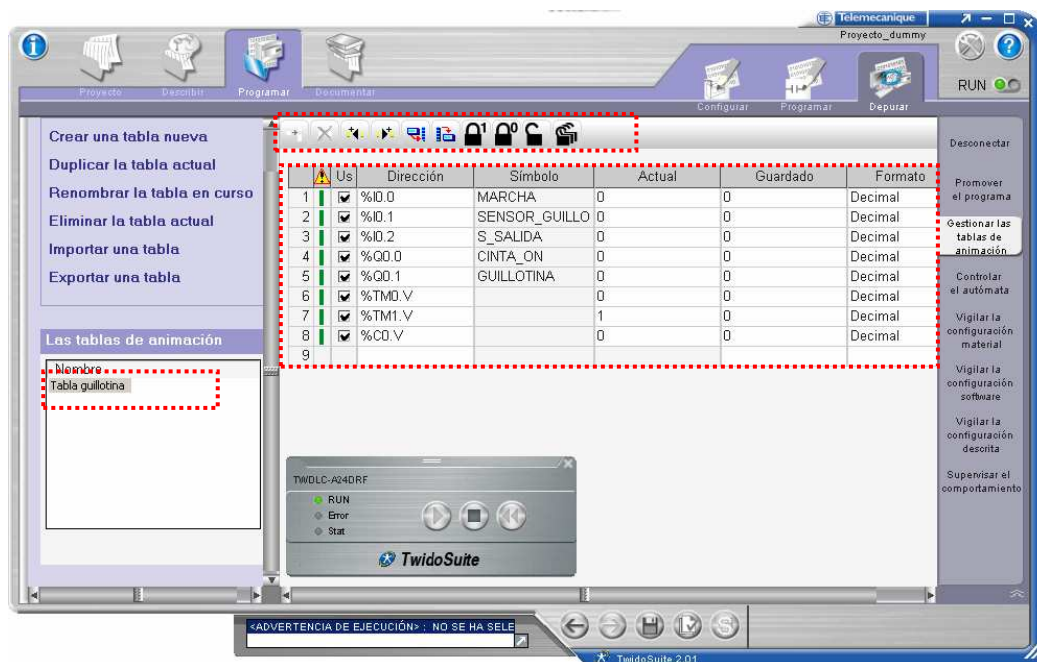
- Supervisar el valor real de cualquier variable del controlador, mientras el programa está en ejecución, y compararlo con el valor esperado;
- Cambiar el valor de las variables independientes de los valores calculados, hasta que se actualicen de nuevo;
- Realizar una copia de seguridad de los valores actuales de todas las variables de una tabla de animación;
- Restaurar los valores guardados de todas las variables de una tabla de animación en el controlador y
- Forzar entradas y salidas.

Para crear una tabla de animación, una vez conectado, seleccione la tarea **Programa** → **Depurar** → **Gestionar las tablas de animación** para abrir el editor.



Para crear una tabla de animación se tiene que pulsar la opción de **“Crear una tabla nueva”** dentro del marco de tareas de gestión de las tablas que se encuentra en la parte izquierda del editor.

Una vez se le ponga un nombre a la tabla, se colocan las diferentes señales que nos interesa poder observar y probar dentro de nuestra aplicación. Para nuestro ejemplo serán los siguientes:



En la columna de **“Dirección”** es donde se va introduciendo las direcciones de las señales que nos interesan, sino también se puede colocar el símbolo en la columna de **“Símbolo”**. Cuando se haya colocado la dirección o el símbolo se refrescarán las columnas de **“Guardado”** con el valor guardado y en la columna **“Actual”** el valor actual de la señal.

Para variables que **no sean booleanas** se puede elegir en la columna **“Formato”** la codificación en la que la se quiere visualizar en **Decimal** o **Hexadecimal**. En este caso por ejemplo se ha colocado los valores actuales de los temporizadores **“%TMO.V”**, **“%TM1.V”** y el valor actual del contador **“%CO.V”** en formato decimal.

En la tabla de animación es también posible modificar/forzar los valores de las variables desde el PC.

o *Diferencia entre Modificar y Forzar una variable:*

Si se modifica una variable, el usuario puede cambiar el valor de la variable pero el control lo tendrá el PLC, es decir, que el PLC puede volver a cambiar el valor. Si se fuerza una variable, el control del valor la tiene el usuario, es decir, el PLC no podrá cambiarla.

Para **modificar** el valor de una variable solo se tiene que seleccionar y cambiar el valor en la columna “Actual”. Por ejemplo cambiar el valor del contador de piezas del 0 actual a 23.

| Us | Dirección | Símbolo | Actual | Guardado | Formato | |
|----|-------------------------------------|---------|----------------|----------|---------|---------|
| 1 | <input checked="" type="checkbox"/> | %IO.0 | MARCHA | 0 | 0 | Decimal |
| 2 | <input checked="" type="checkbox"/> | %IO.1 | SENSOR_GUILLO | 0 | 0 | Decimal |
| 3 | <input checked="" type="checkbox"/> | %IO.2 | S_SALIDA | 0 | 0 | Decimal |
| 4 | <input checked="" type="checkbox"/> | %Q0.0 | CINTA_ON | 0 | 0 | Decimal |
| 5 | <input checked="" type="checkbox"/> | %Q0.1 | GUILLOTINA | 0 | 0 | Decimal |
| 6 | <input checked="" type="checkbox"/> | %TMD.V | | 0 | 0 | Decimal |
| 7 | <input checked="" type="checkbox"/> | %TM1.V | T_GUILLOTINA.V | 1 | 0 | Decimal |
| 8 | <input checked="" type="checkbox"/> | %CO.V | 0 | 0 | Decimal | |

| Us | Dirección | Símbolo | Actual | Guardado | Formato | |
|----|-------------------------------------|---------|----------------|----------|---------|---------|
| 1 | <input checked="" type="checkbox"/> | %IO.0 | MARCHA | 0 | 0 | Decimal |
| 2 | <input checked="" type="checkbox"/> | %IO.1 | SENSOR_GUILLO | 0 | 0 | Decimal |
| 3 | <input checked="" type="checkbox"/> | %IO.2 | S_SALIDA | 0 | 0 | Decimal |
| 4 | <input checked="" type="checkbox"/> | %Q0.0 | CINTA_ON | 0 | 0 | Decimal |
| 5 | <input checked="" type="checkbox"/> | %Q0.1 | GUILLOTINA | 0 | 0 | Decimal |
| 6 | <input checked="" type="checkbox"/> | %TMD.V | | 0 | 0 | Decimal |
| 7 | <input checked="" type="checkbox"/> | %TM1.V | T_GUILLOTINA.V | 1 | 0 | Decimal |
| 8 | <input checked="" type="checkbox"/> | %CO.V | 23 | 0 | Decimal | |





Para **forzar una variable**, se selecciona y se pulsa uno de los comandos de forzado de la barra de herramientas. Por ejemplo forzar la señal de salida “%Q0.0” de 0 a 1.

| Us | Dirección | Símbolo | Actual | Guardado | Formato | |
|----|-------------------------------------|---------|---------------|----------|---------|---------|
| 1 | <input checked="" type="checkbox"/> | %IO.0 | MARCHA | 0 | 0 | Decimal |
| 2 | <input checked="" type="checkbox"/> | %IO.1 | SENSOR_GUILLO | 0 | 0 | Decimal |
| 3 | <input checked="" type="checkbox"/> | %IO.2 | S_SALIDA | 0 | 0 | Decimal |
| 4 | <input checked="" type="checkbox"/> | %Q0.0 | CINTA_ON | 0 | 0 | Decimal |
| 5 | <input checked="" type="checkbox"/> | %Q0.1 | GUILLOTINA | 0 | 0 | Decimal |

| Us | Dirección | Símbolo | Actual | Guardado | Formato | |
|----|-------------------------------------|---------|---------------|----------|---------|---------|
| 1 | <input checked="" type="checkbox"/> | %IO.0 | MARCHA | 0 | 0 | Decimal |
| 2 | <input checked="" type="checkbox"/> | %IO.1 | SENSOR_GUILLO | 0 | 0 | Decimal |
| 3 | <input checked="" type="checkbox"/> | %IO.2 | S_SALIDA | 0 | 0 | Decimal |
| 4 | <input checked="" type="checkbox"/> | %Q0.0 | F 1 | 0 | Decimal | |
| 5 | <input checked="" type="checkbox"/> | %Q0.1 | GUILLOTINA | 0 | 0 | Decimal |


En la columna con la etiqueta **Actual** de la tabla de animación, la variable forzada aparece con un prefijo **F** junto al estado forzado 0 ó 1. F 0 o F 1.

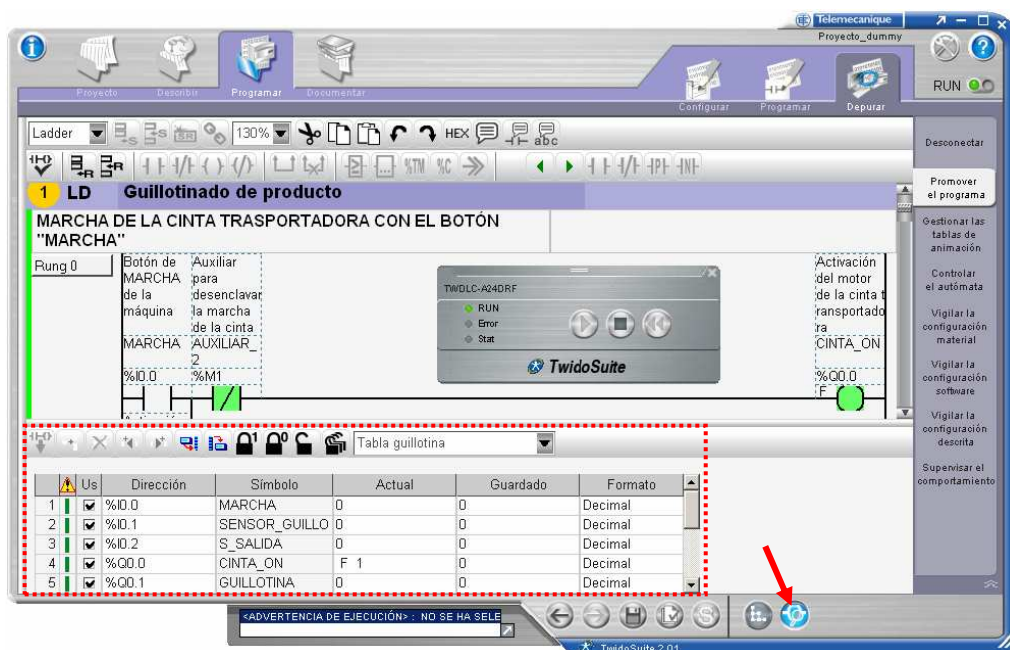
En la ventana de herramientas existen diferentes comandos de forzado:


-  Forzar a 1
-  Forzar a 0
-  Borrar forzado
-  Borrar todos los forzados



A la hora de forzar variables tener en cuenta de no dejar valores forzados una vez acabada la depuración del programa..

Se pueden **Mostrar/ocultar las tablas de animación**, dentro de la ventana del editor del programa. Haciendo clic en el icono de la barra de funciones para mostrar y ocultar las tablas de animación situadas en la parte inferior de la ventana. 



También se puede Copiar los objetos de escalón en una tabla de animación, seleccionando el escalón y pulse (botón situado más a la izquierda de la barra de herramientas de animación), como se muestra en la figura que aparece a continuación. 

1 LD Guillotinado de producto

MARCHA DE LA CINTA TRANSPORTADORA CON EL BOTÓN "MARCHA"

Rung 0

Botón de MARCHA de la máquina

Auxiliar para desenclavar la marcha de la cinta MARCHA_AUXILIAR_2

%I0.0 %M1

Activación del motor de la cinta transportadora CINTA_ON

%Q0.0 F

Copiar los objetos del rung en la tabla de animación

| | Us | Dirección | Símbolo | Actual | Guardado | Formato |
|----|-------------------------------------|-----------|----------------|--------|----------|---------|
| 7 | <input checked="" type="checkbox"/> | %TM1.V | T_GUILLOTINA.V | 1 | 0 | Decimal |
| 8 | <input checked="" type="checkbox"/> | %C0.V | C_PIEZAS.V | 23 | 0 | Decimal |
| 9 | <input checked="" type="checkbox"/> | %M1 | AUXILIAR_2 | 0 | 0 | Decimal |
| 10 | <input checked="" type="checkbox"/> | | MARCHA | | | Decimal |

<ADVERTENCIA DE EJECUCIÓN> NO SE HA SELE

TwidoSuite 2.01

4. Operaciones avanzadas

4.1 Operaciones con señales analógicas:

En este capítulo se describe el modo de gestionar los potenciómetros analógicos incorporados y los módulos de ampliación analógicos.

○ **Potenciómetros analógicos:**

Los controladores Twido cuentan con un o dos potenciómetros analógicos. Los potenciómetros aportan unos valores numéricos, de **0 a 1.023** para el potenciómetro analógico 1 y de **0 a 511** para el potenciómetro analógico 2, correspondientes a los valores analógicos que indican estos potenciómetros, forman parte de las dos palabras de entrada siguientes:

- **%IW0.0.0** para el potenciómetro analógico 1 (a la izquierda)
- **%IW0.0.1** para el potenciómetro analógico 2 (a la derecha)

Estas palabras se pueden utilizar en operaciones aritméticas. Se pueden emplear para cualquier tipo de ajuste (preselección de un retardo o de un contador, ajuste de la frecuencia del generador de pulsos o de la duración del precalentamiento de una máquina, etc.).

○ **Módulos analógicos:**

Además del potenciómetro integrado de 10 bits y el canal analógico de 9 bits, todos los autómatas Twido que admiten ampliaciones de E/S pueden configurar módulos de E/S analógicos y comunicarse con ellos.

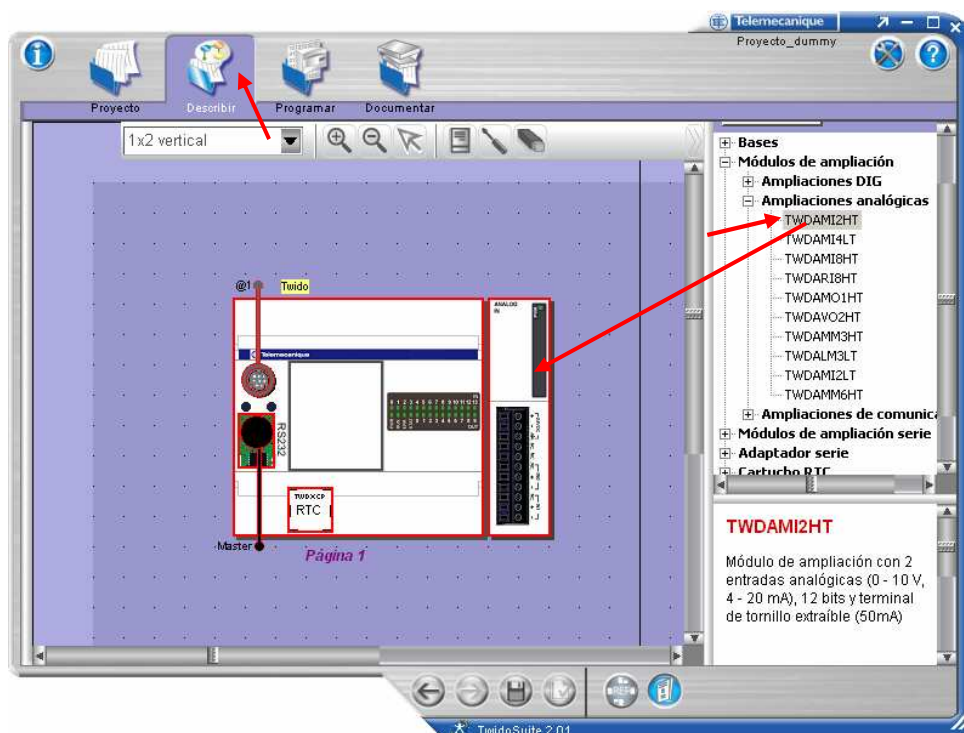
Funcionamiento de los módulos analógicos:

Las palabras de entrada y de salida (%IW y %QW) se emplean para intercambiar datos entre la aplicación de usuario y los canales analógicos. La actualización de estas palabras se realiza de manera sincronizada con la exploración del autómata en modo RUN.

Direccionamiento de entradas y salidas analógicas

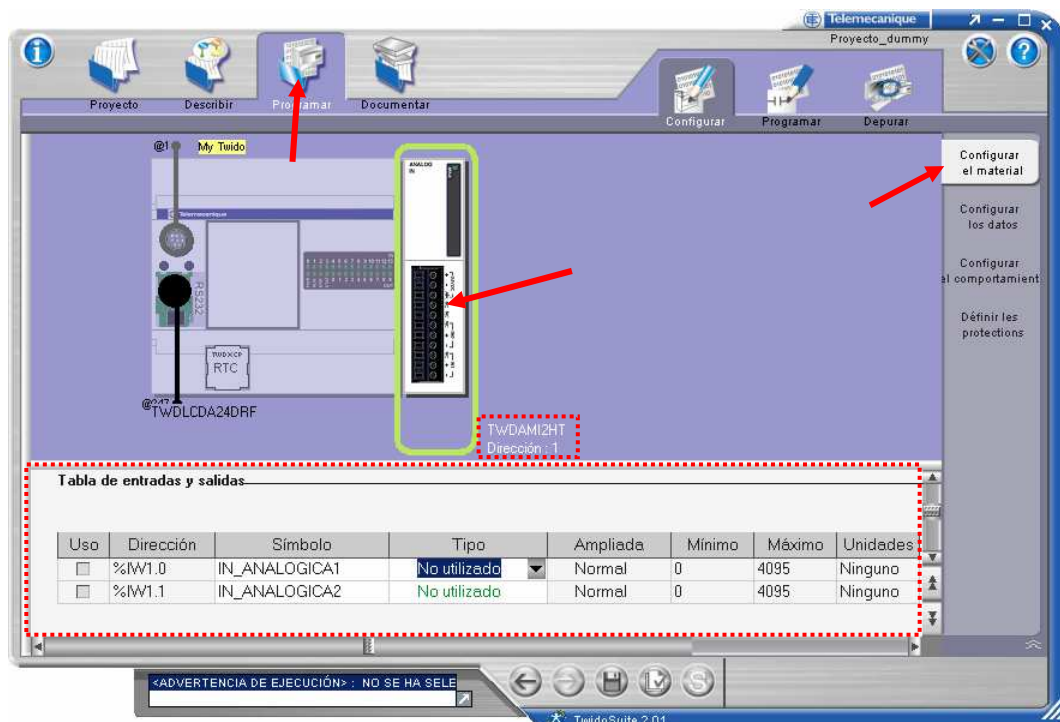
En la pestaña “**Describir**” de la barra de pasos de la aplicación se configura el hardware de la aplicación, dentro del **catalogo de hardware/Módulos de ampliación/Ampliaciones analógicas** están las diferentes referencias de todos los módulos analógicos. Para introducirlo en la configuración solo se tiene que seleccionar y arrastrar a la posición deseada en la ventana de configuración.

Cuando se tenga el módulo analógico en la posición, se tendrá que **configurar**, para ello haremos doble clic en el módulo y nos iremos a la zona de configuración hardware.



En **Programa** → **Configurar** → **Configurar el hardware**, el panel de configuración muestra una zona de **Descripción** que contiene el número de referencia y una breve descripción del módulo seguida de la tabla de configuración de dicho módulo.

Si dispone de varios módulos en el sistema y desea mostrar la tabla de configuración de uno en concreto, haga clic en el módulo deseado en el panel gráfico superior.



En la tabla de configuración de los módulos de ampliación analógicos, consta de las siguientes columnas, con más o menos filas en función del número de entradas y salidas que dispone el módulo analógico configurado.

| Uso | Dirección | Símbolo | Tipo | Ampliada | Mínimo | Máximo | Unidades |
|--------------------------|-----------|---------------|--------------|----------|--------|--------|----------|
| <input type="checkbox"/> | %IW1.0 | IN_ANALOGICA1 | No utilizado | Normal | 0 | 4095 | Ninguno |
| <input type="checkbox"/> | %IW1.1 | IN_ANALOGICA2 | No utilizado | Normal | 0 | 4095 | Ninguno |

En la columna de “Dirección” ya viene preestablecida en forma “%IW” para las entradas y “%QW” para las salidas.

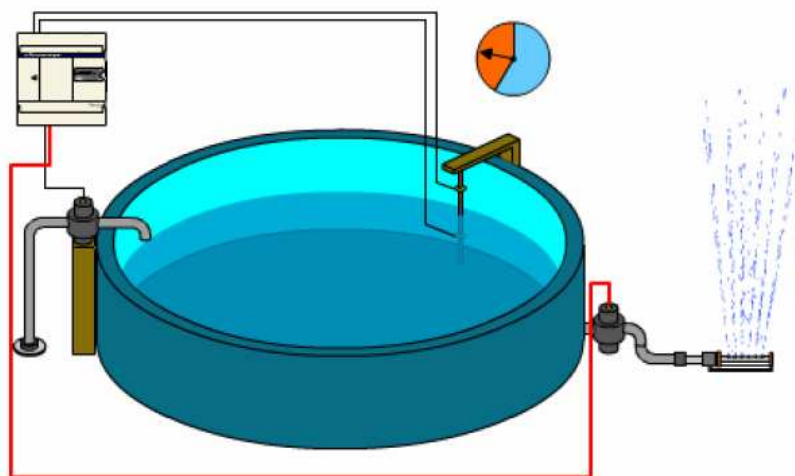
La columna “Símbolo” podemos asignar un nombre a la variable para identificarla o hacer la llamada desde el programa.

La columna “Tipo” se define el tipo de entrada y salida. Identifica el modo de un canal. Las opciones dependen del canal y del tipo de módulo. (Por ejemplo de tensión 0-10 Vdc, de corriente 4-20 mA).

La columna “**Rango**” Identifica el rango de valores de un canal. Las opciones dependen del tipo específico de canal y de módulo. Una vez se ha configurado el “**Tipo**”, también puede ajustar el **Rango** correspondiente. La tabla muestra los valores “**Mínimo**” y “**Máximo**” admitidos (fijos o definidos por el usuario, en función de la opción escogida en la columna “**Ampliada**” que se puede elegir entre **Normal** o **Personalizada**) junto con la “**Unidad**”, en caso necesario.

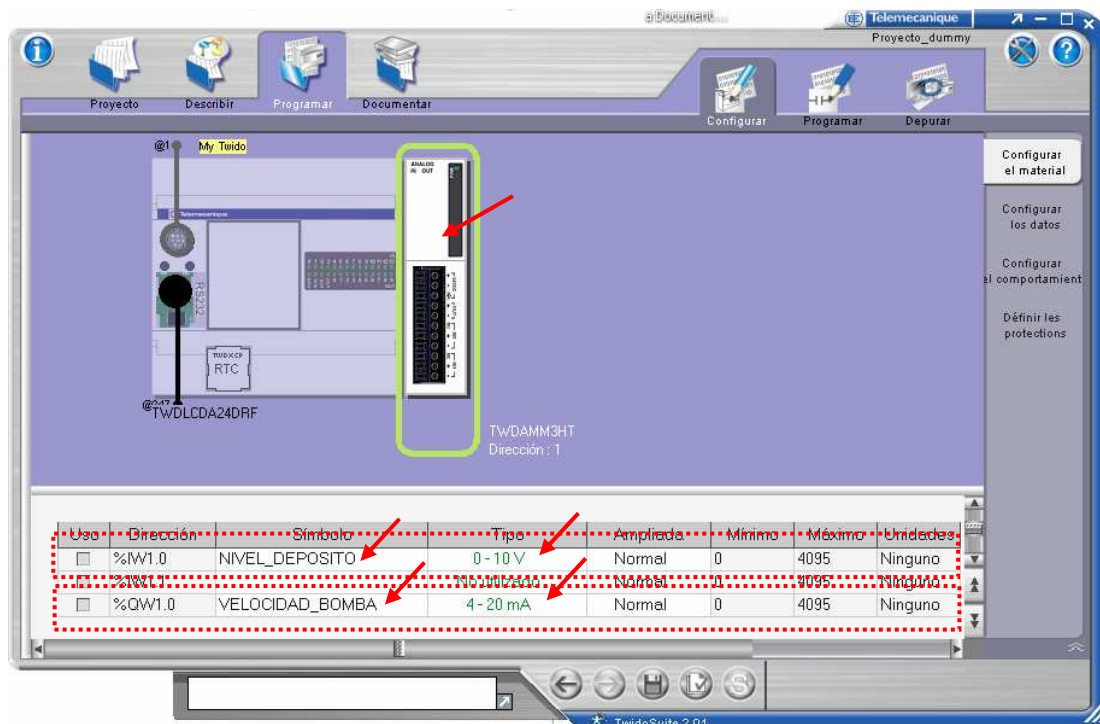
- *Ejemplo de utilización de señales analógicas:*



Gestión de la velocidad de la bomba de riego en función del nivel de agua, se dispone de una entrada analógica de corriente que nos indica el nivel de agua en el depósito. Y se controla la velocidad de la bomba a través de una salida analógica, para ello disponemos de un módulo de ampliación mixto con entradas y salidas analógicas.

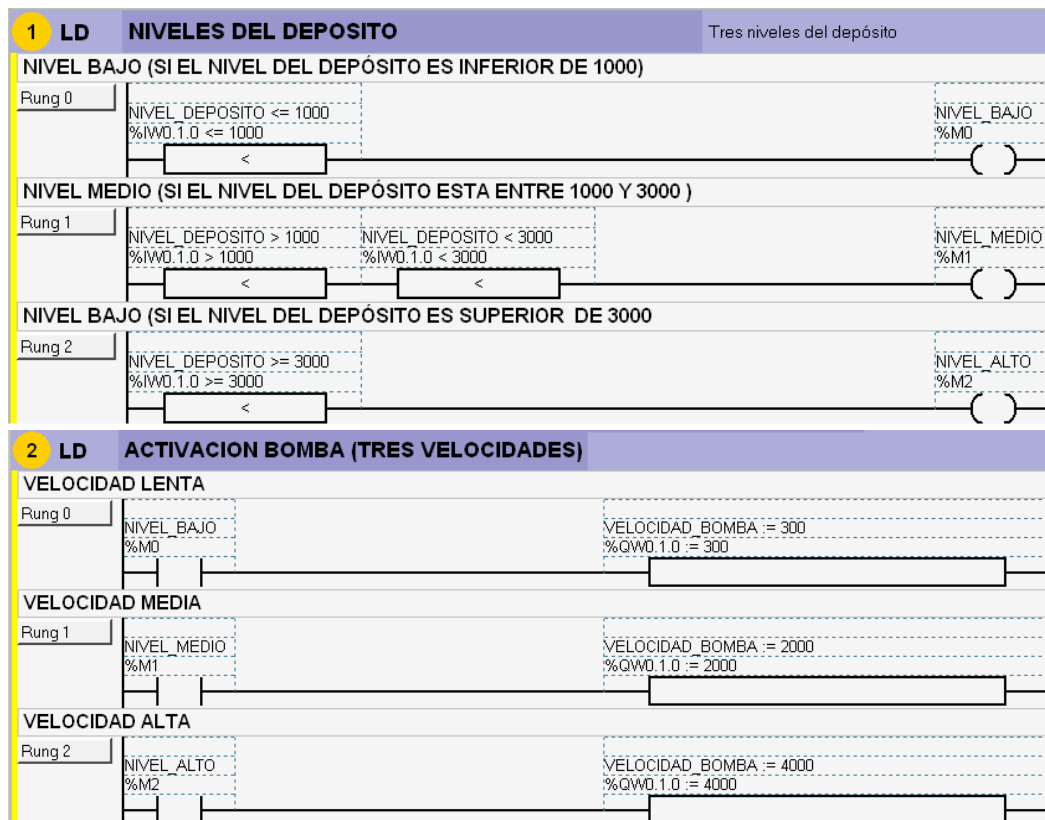


Primero tendremos que configurar el equipo, con el Twido y el módulo de ampliación analógico con referencia **TWDAMM3HT**, que dispone de **2 entradas analógicas** y **1 salida analógica**.

Tras haber realizado la configuración, hacer doble clic para realizar la “**configuración**” del modulo analógico. Configuraremos la entrada analógica con el símbolo de “**Nivel_deposito**” del tipo **0 - 10 V** y una salida analógica que la llamaremos “**Velocidad_bomba**” que será de **4 - 20 mA**.



Utilizaremos bloques de comparación  para determinar los niveles del depósito e insertaremos bloques de operación  para cargar la velocidad de la salida analógica, el programa serán de la siguiente manera:



o **Regulación continua PID:**

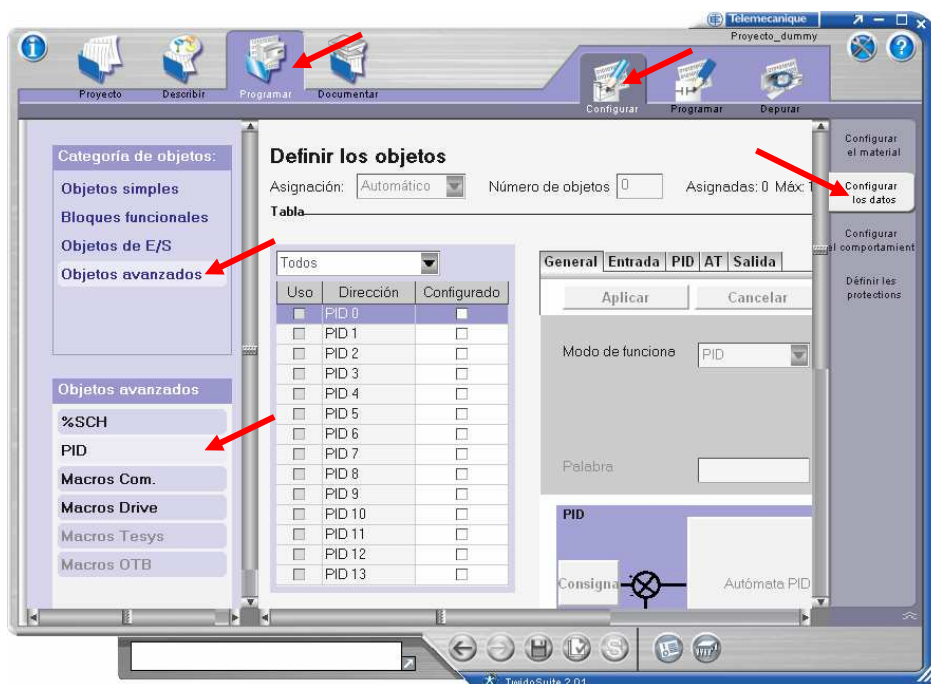
El Twidosuite dispone la posibilidad de implementar para procesos industriales algoritmos de **regulación continua PID**, utilizados para regulación de nivel, temperatura, caudal, presión...etc, donde recibiremos una **señal analógica de entrada** con la medida que queremos regular (nivel, caudal, temperatura) y según los cálculos del algoritmo PID implementado en el control “Twido” el sistema actuará sobre una **señal analógica de salida** que controlará por ejemplo la velocidad de una bomba...etc.



El sistema de regulación continua PID es un tema demasiado extenso para ser explicado en este manual, donde nos centraremos en como configurar este sistema en el software Twidosuite.

Configuración en el Twidosuite de una regulación PID:

Para la implementar un regulador PID en el Twidosuite una vez definido en el sistema la entrada y salida analógica para la regulación, se tiene que configurar el regulador PID, para ello elegir la pestaña “**Configurar**” dentro de la barra de subpasos de programa y en la barra de tareas elegir la opción “**Configurar los datos**”, aparecerá el área de trabajo, en la columna de “Categoría de objetos” elegir “**Objetos avanzados**”, aparecerá una nueva columna en la parte inferior con las diferentes funciones de “**objetos avanzados**” elegir la función de “**PID**”.



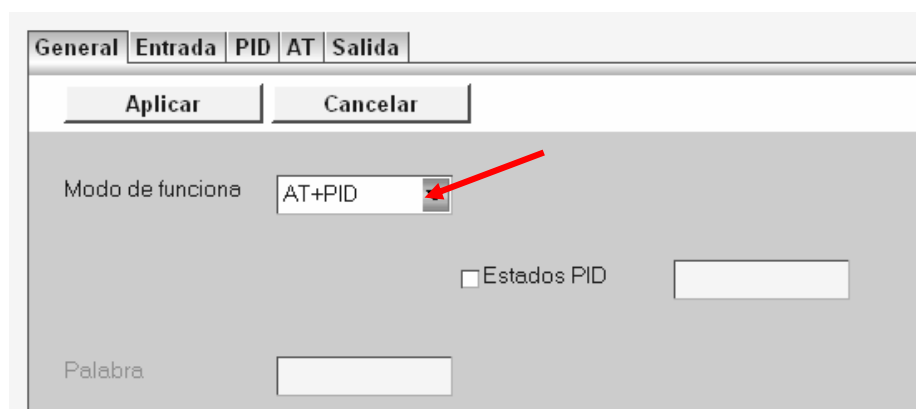
Aparecerá el área para declarar las diferentes instancias PID. Entre las **14** instancias que se pueden configurar, se elige una seleccionándola en la columna de “**Configurado**”.



Importante: Las pestañas deben completarse en el orden en el que aparecen en el cuadro de diálogo PID: primero General, Entrada, PID, AT y, finalmente, Salida.

En la pestaña general se elige entre los diferentes modos de funcionamiento.

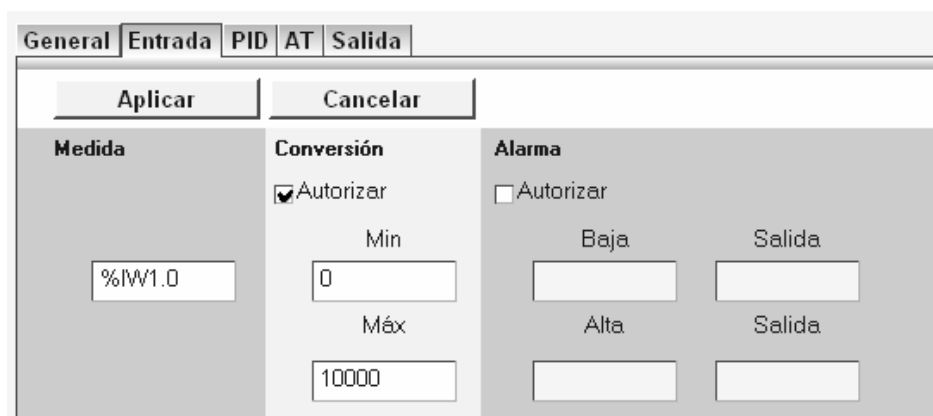
- **PID:** Autómata PID simple.
- **AT + PID:** La función de Auto-Tuning se activa al iniciar el PID e introduce automáticamente los valores de ganancia **Kp**, **Ti**, **Td** (pestaña **PID**) y el tipo de acción PID (pestaña **Salida**). Al final de la secuencia de Auto-Tuning, el autómata pasa al modo PID para la consigna ajustada y utiliza los parámetros establecidos por AT.
- **AT:** La función de Auto-Tuning se activa al iniciar el PID e introduce automáticamente los valores de ganancia **Kp**, **Ti**, **Td** (pestaña **PID**) y el tipo de acción PID (pestaña **Salida**). Al final de la secuencia, el PID se detiene y espera.
- **Dirección de palabra:** Este tipo de configuración a través de la dirección de palabra permite que el usuario pueda gestionar el modo de funcionamiento del autómata PID mediante el programa de aplicación, de manera que sea posible adaptarse a los requisitos finales.



En la pestaña de **Entrada** se tiene que introducir el canal analógico utilizado como medida en el campo asociado (En este ejemplo: se ha elegido **%IW1.0**).

Activar la casilla **Conversión** si se desea convertir la variable del proceso especificada como entrada del PID. Si se selecciona esta casilla, se podrá acceder a los dos campos **Valor mín.** y **Valor máx.** La conversión es lineal y convierte un valor comprendido entre **0** y **10.000** en un valor cuyos mínimos y máximos están comprendidos entre **-32.768** y **+32.767**.

Cuando sea necesario, establecer las alarmas en los umbrales de medida alto y bajo activando las casillas y completando los campos correspondientes.



En la pestaña **PID**, introducir el valor que debe utilizarse para definir la consigna del autómatas. En general, este valor es una dirección de memoria o una consigna de una entrada analógica.

Sólo se puede seleccionar el **tipo de corrector** si anteriormente se ha elegido el modo de funcionamiento PID en la pestaña General.

Definir los parámetros **Kp, Ti, Td**. **Importante:** En caso de seleccionar el modo **AT** o **AT+PID**, es necesario completar los campos Kp, Ti y Td con **direcciones de memoria**, para habilitar así la función de Auto-Tuning e introducir automáticamente los valores encontrados.

Introducir el **Período de muestreo** del PID. El controlador utiliza este valor para adquirir medidas y actualizar salidas.

En la pestaña **AT**, activar la casilla **Autorizar** si se desea utilizar **AT**. Introducir el valor de Límite de la medida. Se trata del valor de límite que no debe superar la medida durante la AT. Introducir el valor de **Consigna de salida**, que es el valor de salida del autómatas enviado para generar la **AT**.

En la pestaña **Salida**, introducir la selección de la lista desplegable **Acción**. Esta selección depende del sistema configurado:

- **Acción directa:** la salida del autómatas disminuye cuando el valor de variación (consigna - medida) aumenta (autómatas en frío).
- **Acción inversa:** Acción directa: la salida del autómatas disminuye cuando el valor de variación (consigna - medida) aumenta (autómatas en caliente).



Al utilizar la función AT, esta lista selecciona automáticamente Dirección de bit. La función AT determina el modo de funcionamiento, y en este caso se introduce el bit asociado con este campo.

Cuando sea necesario, introducir los valores de umbral de la salida del autómatas en el campo **Alarmas**. Esta función puede ser necesaria en determinadas aplicaciones para gestionar las alarmas de proceso en las que se han superado los umbrales.

Definir el modo de funcionamiento en **Modo manual**. La lista desplegable contiene varias opciones:

- **Inhibir** = sin modo manual.
- **Autorizar** = el autómatas funciona sólo en modo manual.
- **Dirección de bit** = el valor del bit se utiliza para modificar el funcionamiento del modo manual (si el bit se establece en 0 = modo automático, si el bit se establece en 1 = modo manual).

Usar el modo manual para realizar pruebas con el fin de determinar la **limitación de salida mín./máx.** o la **consigna de salida de AT** más precisa.

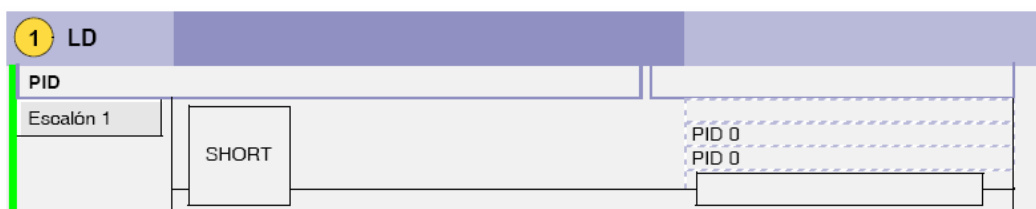
Ajustar la palabra de **Salida binaria**. El autómatas emplea esta palabra para enviar la consigna de control. Puede enviarse directamente a un canal de salida analógica (%QW..) o a una palabra de memoria (%MWxx) para un procesamiento adicional.

Definir la **Salida PWM**, si lo requiere el sistema. Activar la casilla **Autorizar** si desea controlar el sistema mediante un impulsor PWM.

Confirmar la configuración del autómata haciendo clic en **Aceptar** en la parte inferior izquierda de la pantalla.

Llamada del PID configurado en el programa.

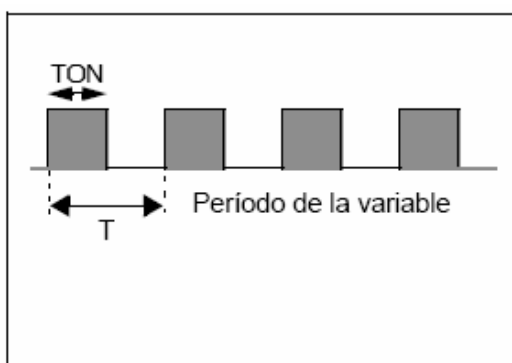
Para utilizar el objeto de regulación configurado previamente, solo se tiene que hacer la llamada al número de instancia en el programa.






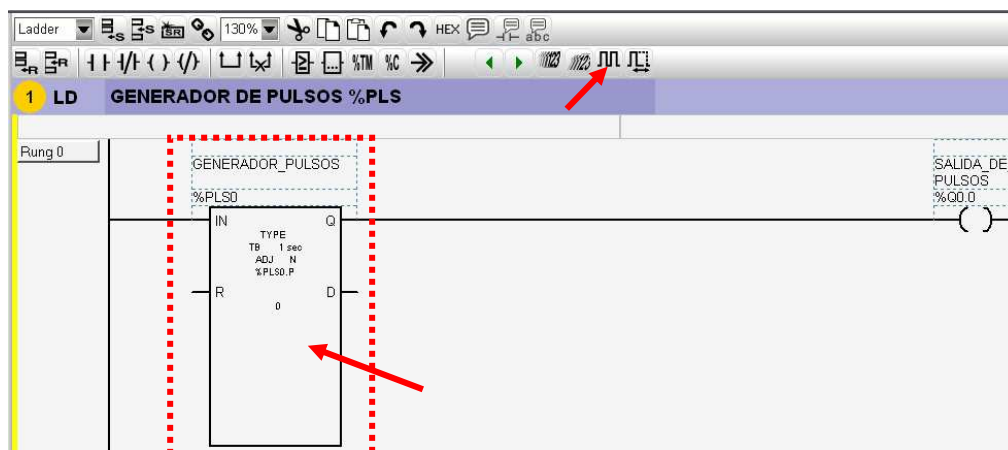
4.2 Bloque generador de pulsos %PLS:

El bloque de función **%PLS** se utiliza para generar señales de onda cuadradas.

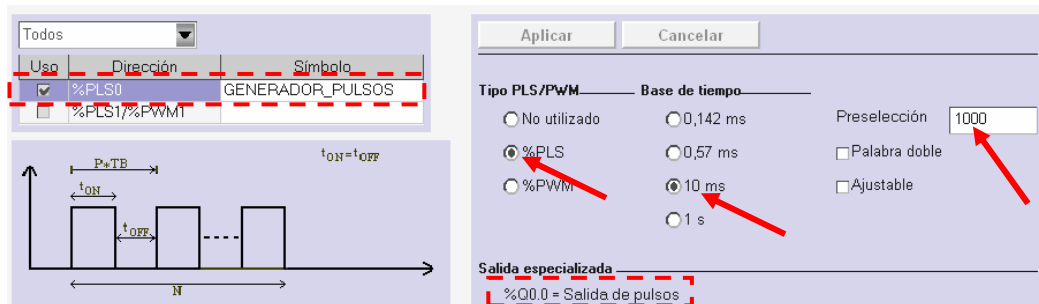
Existen dos funciones “%PLS” disponibles en los canales de salidas especializadas “%Q0.0” o “%Q0.1”. El bloque de función **%PLS** permite un ancho de señal (o ciclo de servicio) único del **50%**. Puede limitar el número de pulsos o el período en el que se ejecutará el tren de pulso. Éste se puede determinar en el momento de la configuración o de la actualización desde la aplicación de usuario.



Para introducir el generador de pulsos dentro de un programa, en el editor crearemos un escalón, seleccionaremos la rama e iremos desplazándonos con los iconos   por los diferentes bloques de función avanzados, hasta encontrar el icono del generador de pulsos , al pulsar aparecerá el bloque en el escalón seleccionado.

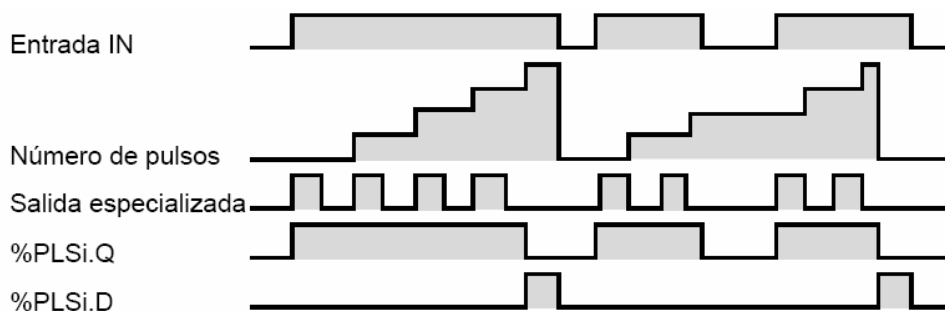


Para la configuración del bloque generador de pulsos basta con hacer doble clic en el mismo y aparece la ventana de configuración. En la columna de símbolo le pondremos un nombre, seleccionaremos el tipo de bloque de función en **%PLS** (generador de pulsos), le diremos también la **base de tiempos** (en este caso **10 ms**) y la **preselección** (**1000**) en la parte inferior encontramos que la salida especializada será la **%Q0.0** que no se podrá utilizar en el programa para otro motivo.



Número de pulsos %PLSi.N ó %PLSi.ND: El número de pulsos que se va a generar en el período T puede limitarse al rango $0 < = \%PLSi.N < = 32.767$ en el modo estándar o $0 < = \%PLSi.ND < = 4.294.967.295$ en modo de palabra doble. El **bit predeterminado** se establece en **0**. Para generar un número ilimitado de pulsos, %PLSi.N o %PLSi.ND se establece en cero. El número de pulsos siempre puede modificarse, independientemente del valor de la opción Ajustable.

El funcionamiento se explica en la ilustración siguiente:



Entrada de generación de pulsos IN: En estado 1, la generación de pulsos se realiza en el canal de salida especializada. En estado 0, el canal de salida se establece en 0.

Restablecer entrada R: En el estado 1, las salidas %PLSi.Q y %PLSi.D se establecen en 0. El número de pulsos generado en el período T se establece en 0.



Generación de salida de pulsos actual %PLSi.Q: El estado 1 indica que la señal de pulsos se genera en el canal de salida especializada configurado.

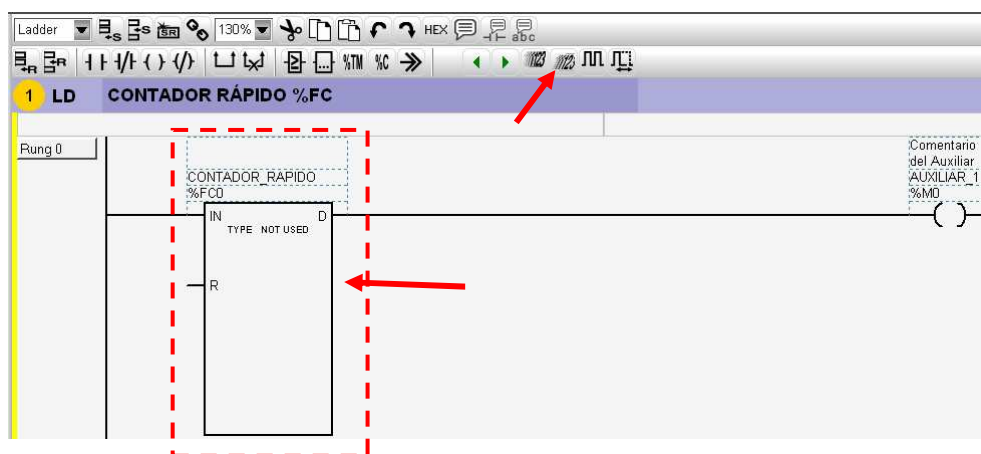
Salida de generación de pulsos concluida %PLSi.D: En estado 1, la generación de la señal ha concluido. Se ha alcanzado el número de pulsos deseados.

4.3 Instrucciones de contador rápido FC:

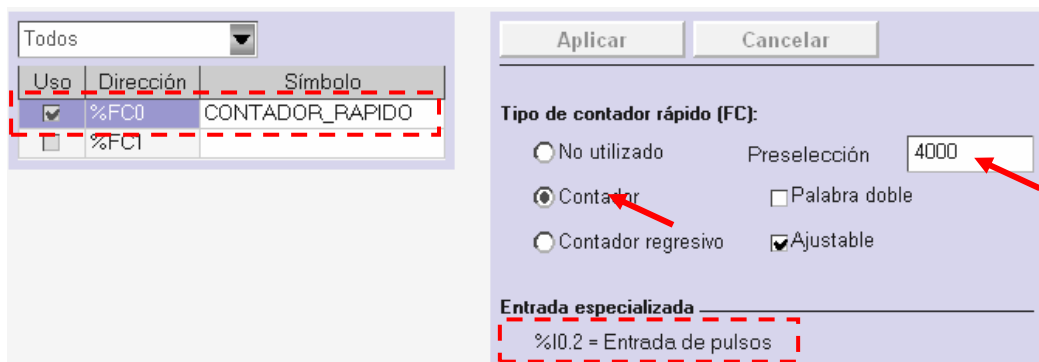
El bloque de función de contador rápido (%FC) se puede utilizar como contador progresivo o regresivo. Puede contar el flanco ascendente de las entradas binarias hasta frecuencias de 5 kHz (1) en modo computacional de palabra o de palabra doble. Dado que los contadores rápidos (FC) se gestionan mediante interrupciones de hardware específicas.

Los controladores compactos **TWDLCA••40DRF** admiten hasta cuatro contadores rápidos, mientras que las demás series de controladores compactos pueden configurarse para usar un máximo de tres contadores rápidos. Los controladores modulares sólo pueden usar un máximo de dos. Los bloques de función de contador rápido %FC0, %FC1, %FC2 y %FC3 utilizan las entradas especializadas %I0.2, %I0.3, %I0.4 y %I0.5, respectivamente. Estos bits no están reservados para el uso exclusivo de estos bloques de función. Para su asignación se debe tener en cuenta el uso de estos recursos especializados por parte de otros bloques de función.

Para introducir el contador rápido dentro del programa, en el editor crearemos un escalón, seleccionaremos la rama e iremos desplazándonos con los iconos  por los diferentes bloques de función avanzados, hasta encontrar el icono del contador rápido , al pulsar aparecerá el bloque en el escalón seleccionado.



Para la configuración del contador rápido basta con hacer doble clic en el mismo y aparece la ventana de configuración.



En la columna de símbolo le pondremos un nombre, seleccionaremos el tipo de contador rápido entre: **no utilizado, contador “progresivo” y contador regresivo**. Introduciremos la preselección **(4000)**, en la parte inferior encontramos que la entrada de contaje rápido especializada será la **%I0.2** que no se podrá utilizar en el programa para otro motivo.

El funcionamiento si está configurado para un conteo progresivo, el valor actual se incrementa en uno con cada flanco ascendente que aparezca en la entrada especializada. Cuando se alcanza el valor preestablecido **%FCi.P o %FCi.PD**, el bit de salida Hecho **%FCi.D** se establece en **1**.

Si está configurado para conteo regresivo, el valor actual disminuye en uno con cada flanco ascendente que aparezca en la entrada especializada. Si el valor es igual a **cero**, el bit de salida Hecho **%FCi.D** se pone a **1**.

4.4 Instrucciones de contaje muy rápido VFC:

El bloque de función de contador muy rápido (**%VFC**) puede configurarse mediante TwidoSuite para realizar una de las funciones siguientes:

- Contador progresivo/regresivo
- Contador progresivo/regresivo bifásico
- Contador progresivo
- Contador regresivo individual
- Frecuencímetro

El **%VFC** admite el conteo de la entrada binaria hasta frecuencias de **20 kHz** en modo computacional de palabra o de palabra doble. Los autómatas compactos TWDLC••40DRF admiten hasta dos contadores muy rápidos, mientras que las demás series de autómatas compactos admiten un contador muy rápido (**%VFC**).

Los autómatas modulares admiten hasta dos contadores muy rápidos (**%VFC**).

Asignaciones de E/S especializadas.

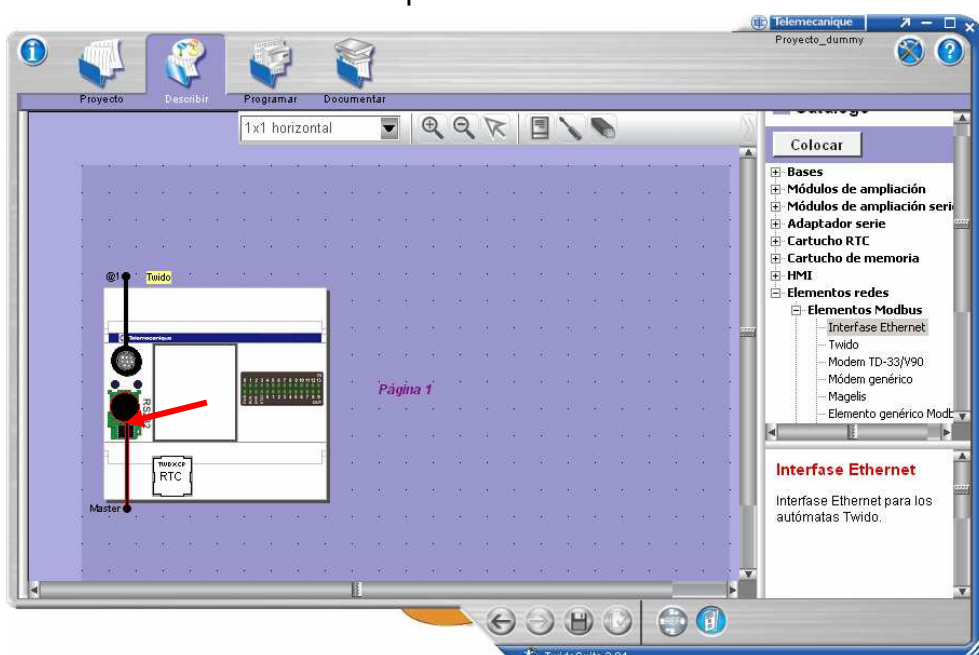
Los bloques de función de contadores muy rápidos (%VFC) utilizan entradas especializadas y entradas y salidas auxiliares. Estas entradas y salidas no están reservadas para el uso exclusivo de estos bloques de función. Para su asignación se debe tener en cuenta el uso de estos recursos especializados por parte de otros bloques de función.

5. Comunicaciones

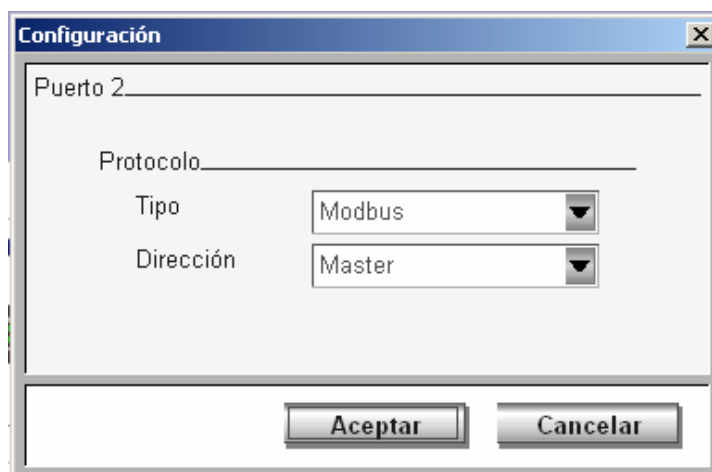
5.1 Configuración de una red Modbus:

Para la configuración de una red Modbus, con el software TwidoSuite, el usuario tendrá que seguir los siguientes pasos:

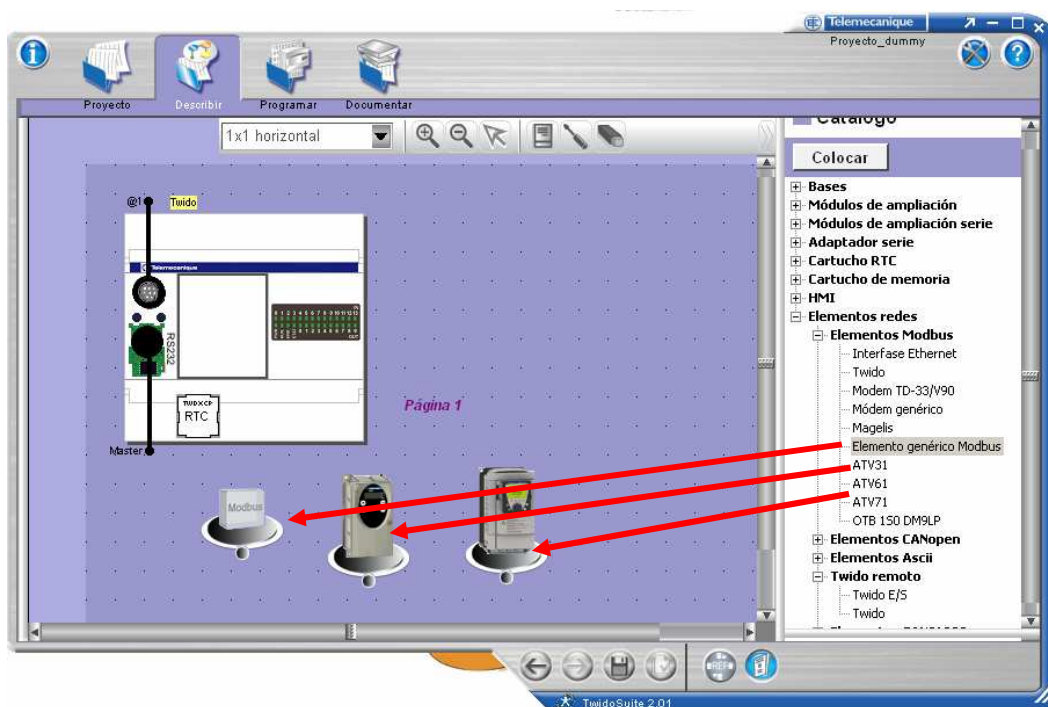
Configurar uno de los puertos del Twido (puerto 1 o puerto 2 si está instalado) como maestro Modbus en la pestaña de “**Describir**”, haciendo doble clic sobre el puerto deseado.



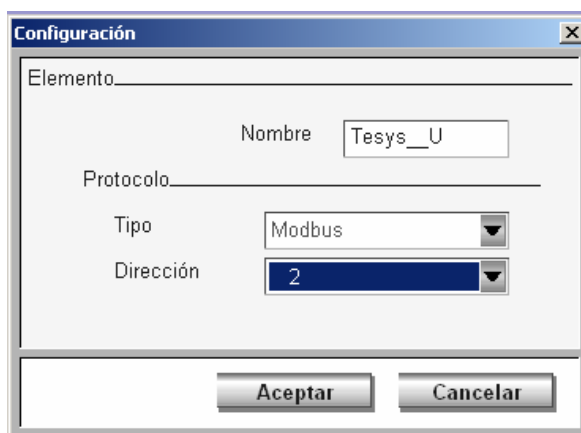
En el cuadro que aparece, Seleccionar **Modbus** en el cuadro Tipo de protocolo, con la dirección como maestro “**Maestro**” y confirmar la configuración del puerto.



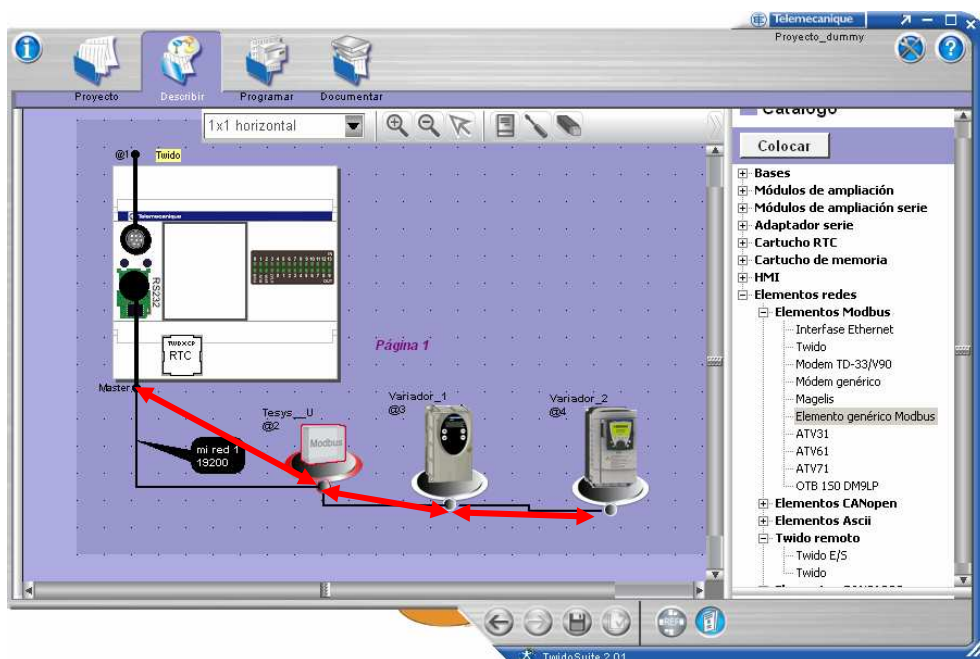
Una vez configurado el puerto del Twido como maestro de Modbus, se tiene que insertar los diferentes esclavos de la red Modbus, con las direcciones que hayamos configurado físicamente en los esclavos. Se seleccionan los dispositivos del catalogo de hardware (si no se encuentra se tiene que seleccionar **elemento genérico modbus**).



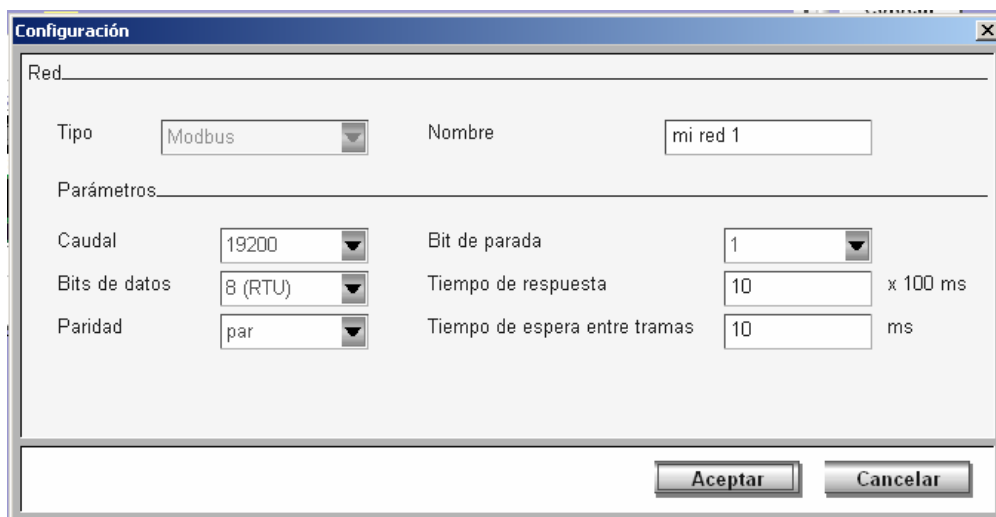
Haciendo doble clic en los esclavos, aparece la ventana de configuración, donde se especifica el “**Protocolo**” Modbus y la “**Dirección**” que dentro de la red Modbus tendrá el esclavo en cuestión, una vez configurado confirmar con “**Aceptar**”.



Cuando se tengan todos los esclavos configurados de la red, unirlos con el puerto del Twido que hace de maestro de la red Modbus.



Ahora tras haber creado la red Modbus, se debe configurar la velocidad de transmisión, así como el protocolo, para ello se pulsará sobre cualquier parte de la línea de bus (color negro) que se ha generado y aparecerá la ventana de configuración del protocolo Modbus.

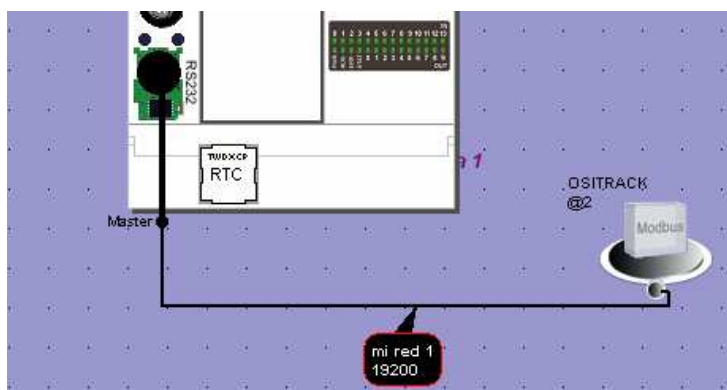


En esta se puede definir la velocidad de transmisión “**Caudal**”, así como el formato del protocolo y el Timeout, generalmente con el que viene predeterminado ya es suficiente (Caudal: 19200, 8 bits de datos, Paridad: Par, 1 bit de parada)

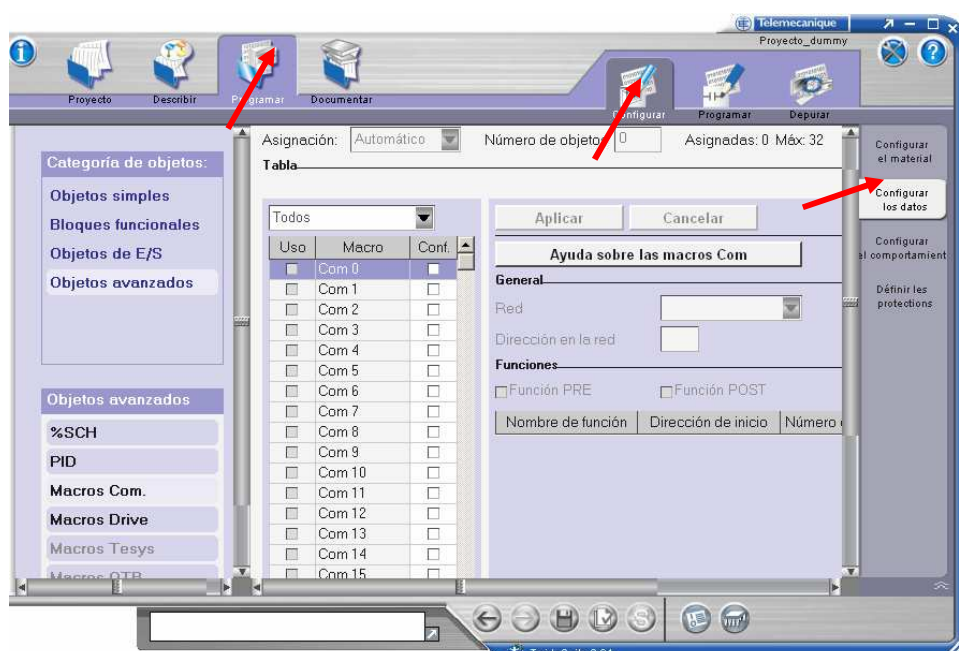
5.2 Macros de comunicaciones:

Se trata de una macro genérica destinada a implementar peticiones de comunicación sobre un puerto Modbus o Modbus TCP/IP. Esta permite configurar hasta 32 instancias. Cada instancia que se configure se ha de vincular a un puerto configurado en Modbus y se le ha de asignar el número de nodo esclavo al que se dirigirán las peticiones realizadas a través de las funciones de la macro.

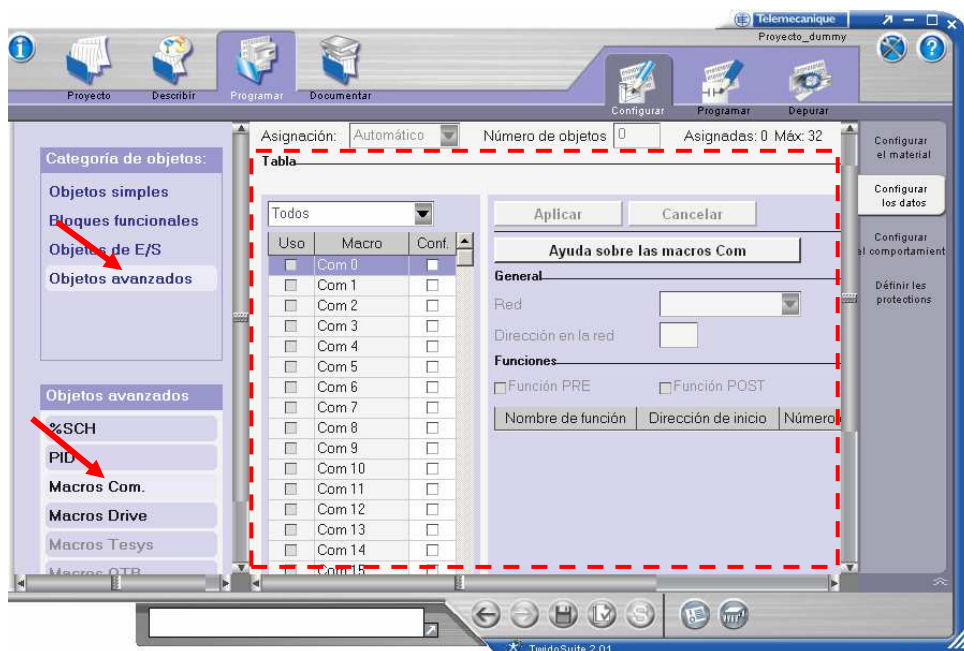
Para una mejor comprensión, se presenta un ejemplo para la programación de la gestión de una antena de radio frecuencia, esta se ha configurado una red Modbus a través del **puerto 2** del Twido que tiene como esclavo genérico de Modbus con dirección **2**.



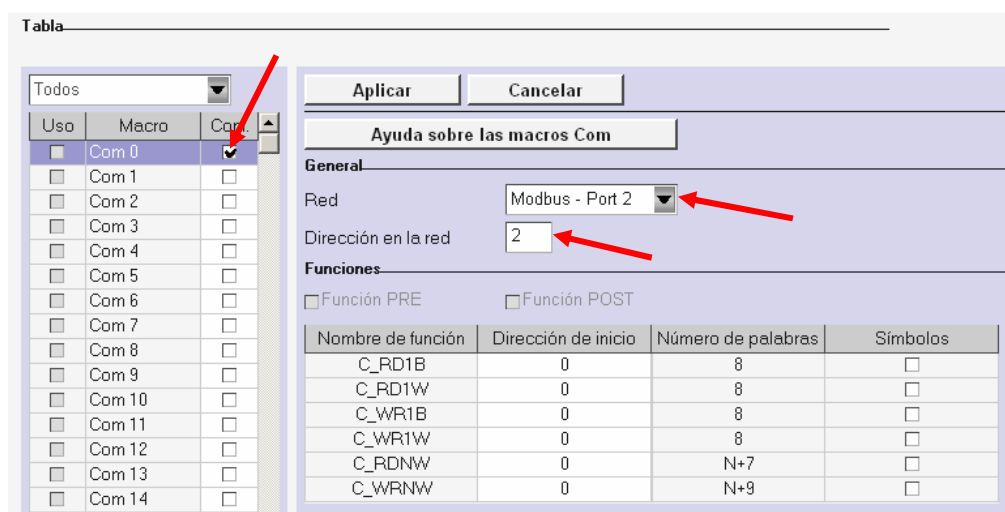
Para la habilitación de la macro se tiene, que ir pestaña general de **“Programa”** → **“Configurar”** → **“Configurar datos”** y aparecerá la ventana de configuración de objetos.



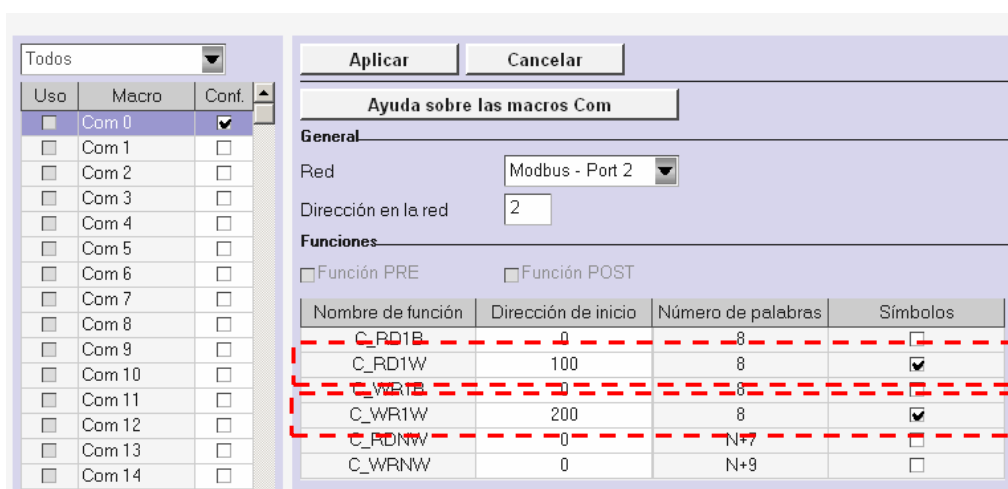
En categoría de objetos se selecciona la opción de “Objetos avanzados” y dentro de objetos avanzados la de “Macros Com.” En el área de trabajo aparece la ventana de configuración de la macro.



Para configurar la macro, seleccionar entre las disponibles el número de macro deseada (en este caso la **0**). En el la casilla de “Red”, se selecciona el puerto donde se ha configurado la red Modbus (**Modbus-Port2**) y en la casilla de “Dirección de red” el número del esclavo que se ha configurado anteriormente en la pestaña “Describir” (esclavo número **2**).



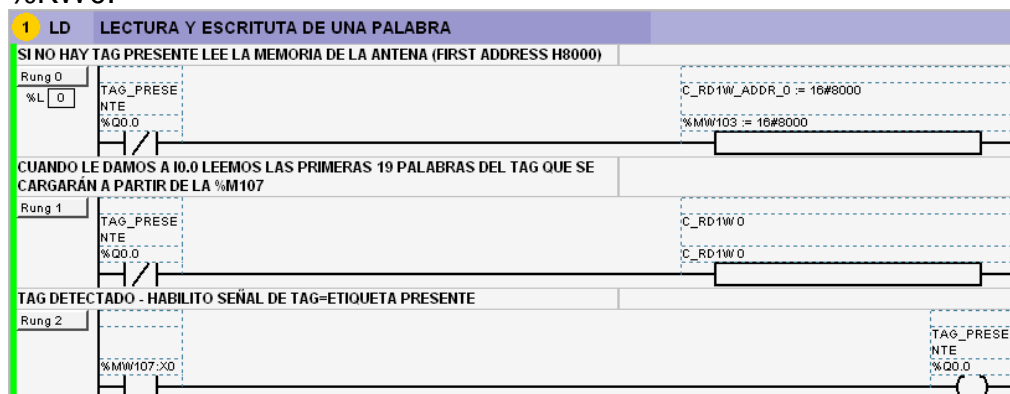
Ahora se tiene que seleccionar la función de comunicación que se quiere realizar sobre el dispositivo (**Lectura o Escritura**) y el volumen de datos (**byte, Word o N Words**). En este caso se seleccionará una función para leer un Word (**C_RD1W**) y otra para escribir un Word (**C_WR1W**). También hay que indicar la dirección de memoria auxiliar donde se van a guardar los datos (**%MW100 para lectura y %MW200 para escritura**) en la columna de “**Dirección de inicio**”, en la columna de “**Número de palabras**” indica el área de memoria utilizada (**8 bits**).

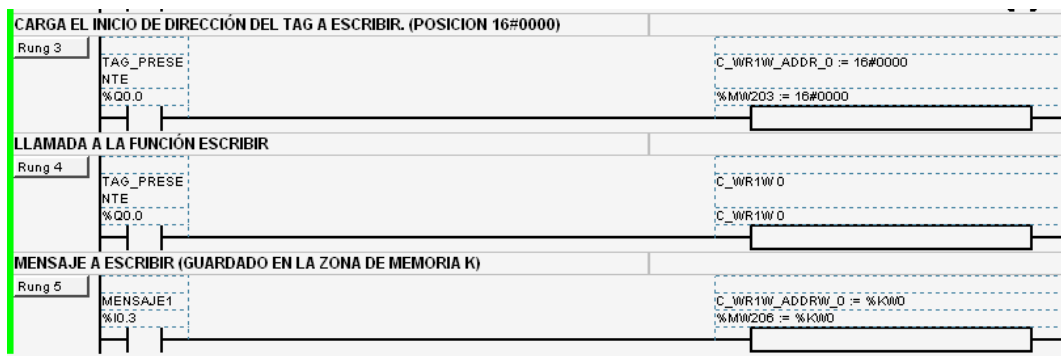


Cuando se haya seleccionado las funciones se confirma la selección, pulsando el botón “**Aplicar**”.

- o *Programa de gestión de la lectura y escritura.*

En el programa se leerá la dirección 8000 decimal (Zona de memoria de la antena) hasta que se pone el bit 0 a 1 que significa que ha detectado una etiqueta. Cuando se detecte que el bit a cambiado, en ese momento se escribirá en la dirección de lectura 00000 decimal los datos guardados en la posición de memoria %KW0.

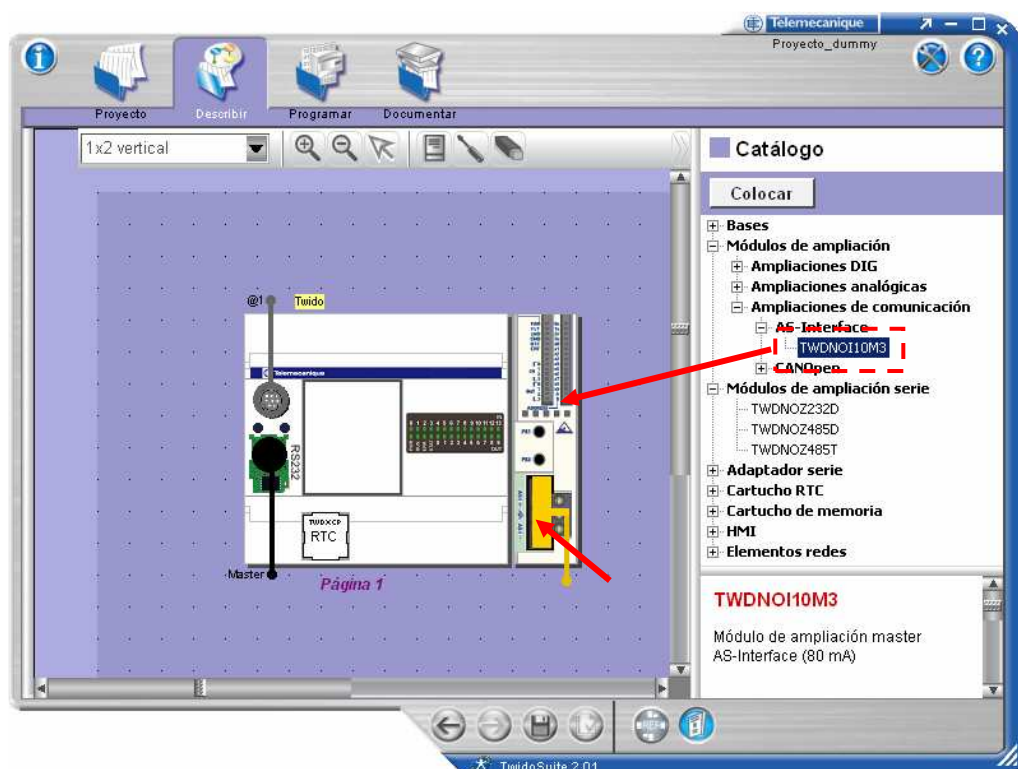




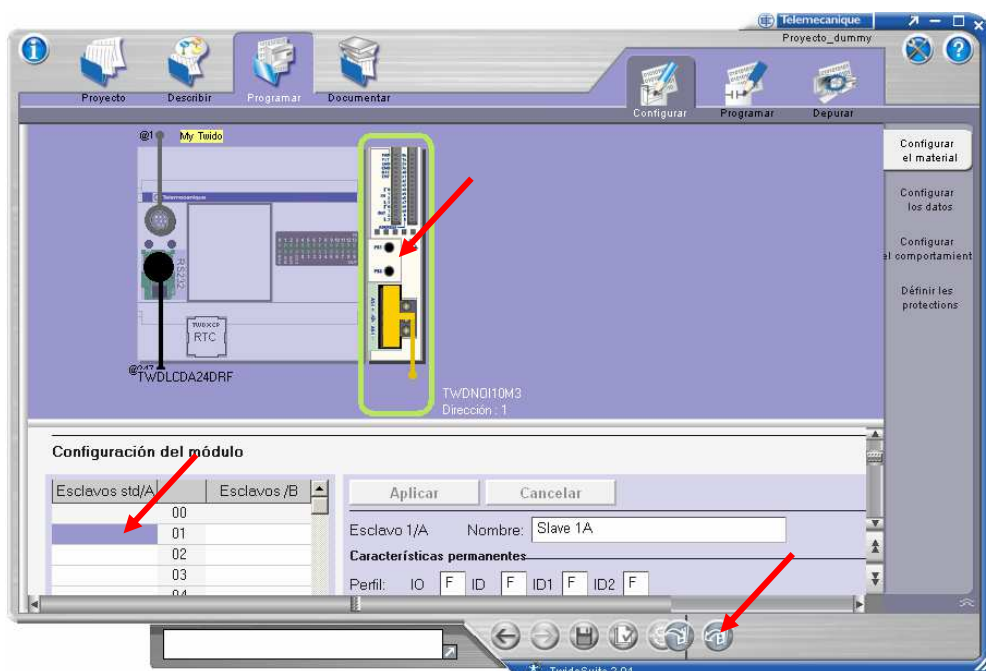
5.3 Configuración de una red ASI:

Para la configuración de una red AS-Interface, con el software TwidoSuite, el usuario tendrá que seguir los siguientes pasos:

Configurar el maestro ASI en la pestaña de “Describir”, se añade desde el catalogo de hardware a la posición en la que se encuentra dentro en la configuración real del equipo. Una vez haya aparecido en la posición deseada de la configuración, seleccionarlo y hacer doble clic sobre el módulo, para ir a la ventana de configuración de la red ASI.

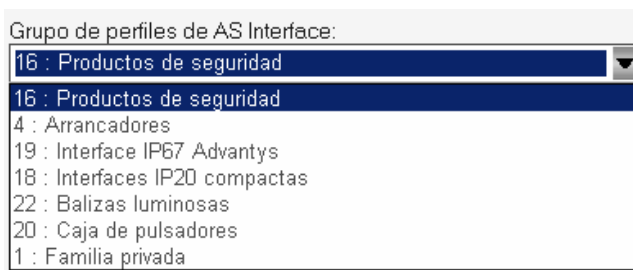


En la ventana de configuración ASI tendremos que insertar en la red los diferentes esclavos ASI que se van a colocar en nuestra red.

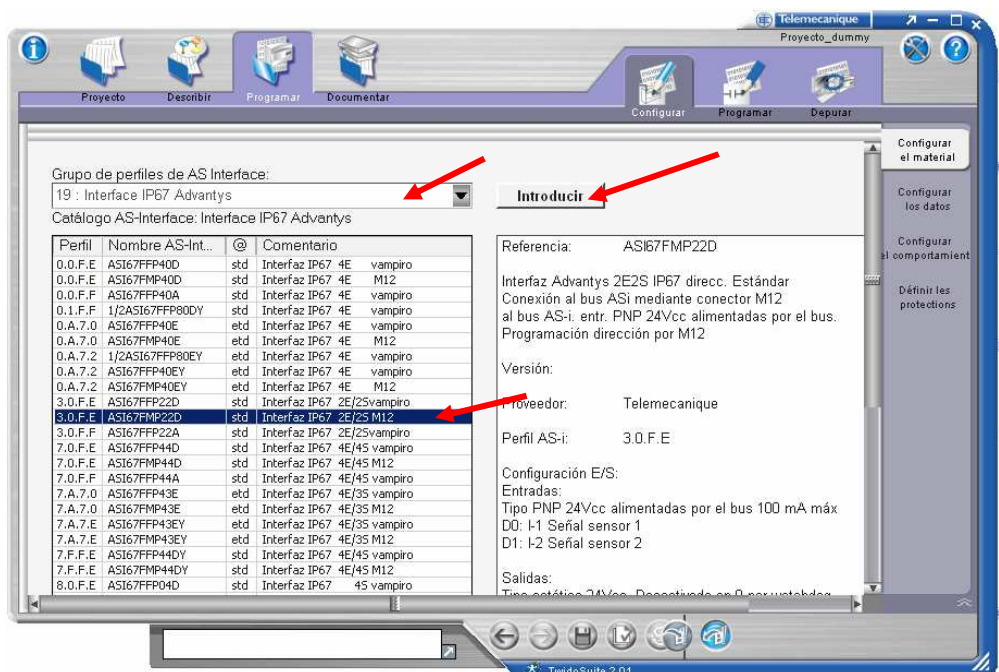


Para la configuración de cada esclavo seleccionar en la tabla, el número que tiene ese esclavo, una vez seleccionado, definiremos el tipo de esclavo, seleccionándolo en el catálogo de esclavos ASI que abriremos haciendo clic sobre el icono **“Introducir a partir del catalogo”** .

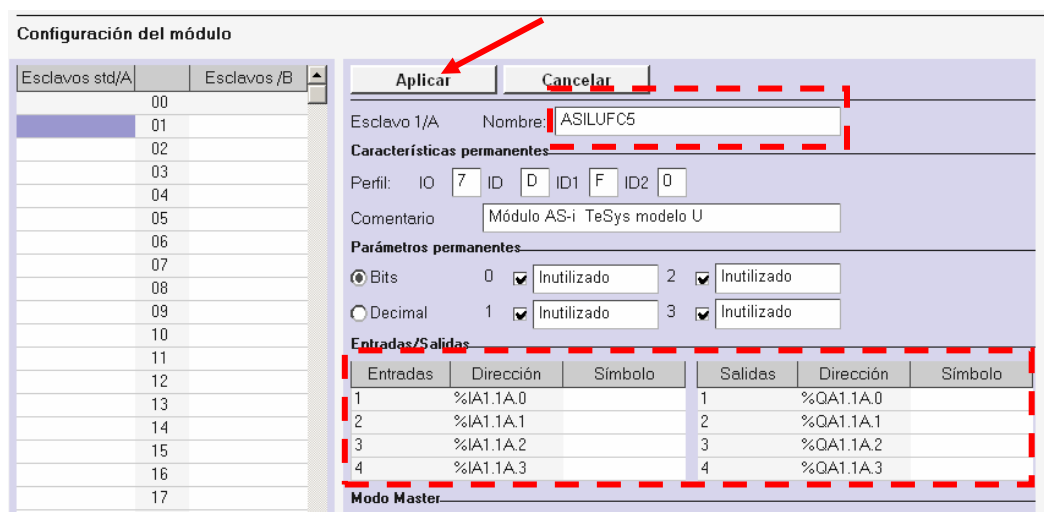
En el catálogo primero se tiene que elegir el grupo de perfil, al cual pertenece el esclavo a configurar, entre la lista de tipos de productos de perfiles (Por ejemplo arrancadores motor, islas advantys, productos de seguridad etc).



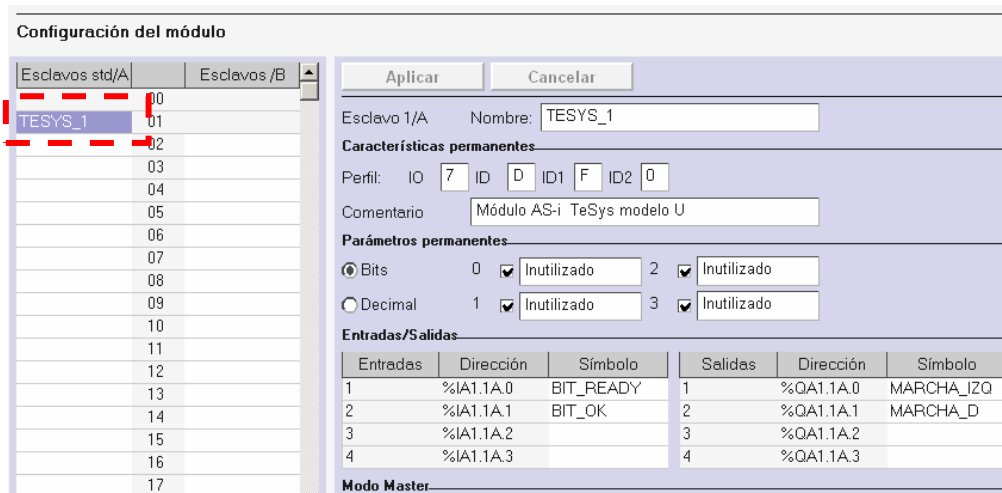
Después de escoger el grupo al que pertenece el esclavo, aparece una lista con la referencia de los diferentes equipos que pertenecen a este grupo, seleccionar dentro de la lista el equipo y pulsar el botón “Introducir”.



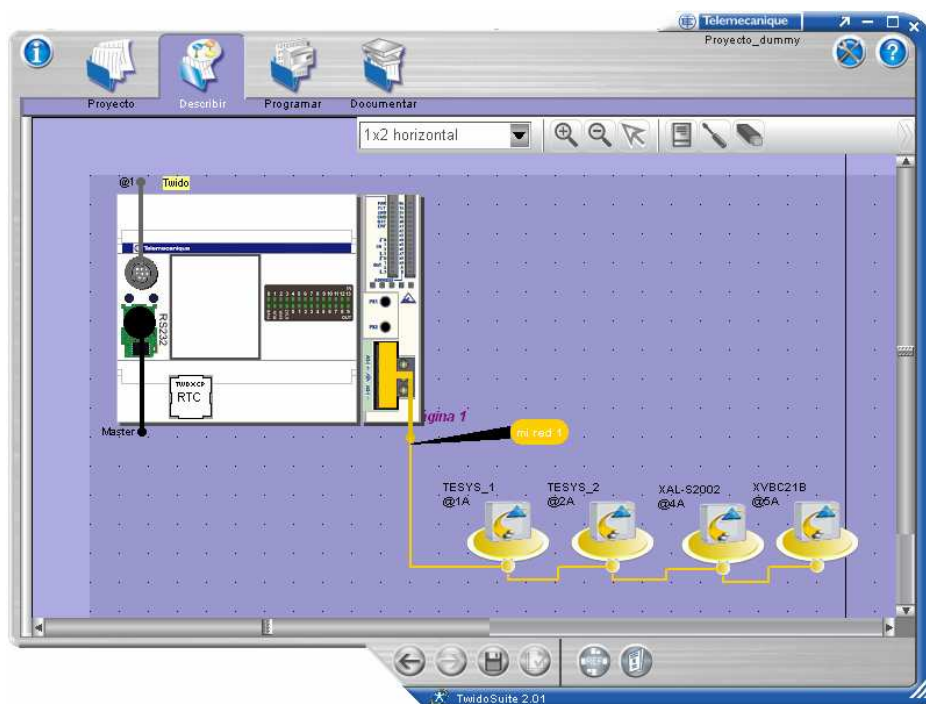
Al pulsar el botón de introducir, veremos que en la ventana de configuración del módulo, aparece la descripción del equipo. En la cual se puede introducir el nombre que se desea para el esclavo. Y en la zona de “Entradas/Salidas” podemos asignar un símbolo a las direcciones ASI de este esclavo.



Cuando se haya acabado de configurarlo, pulsar en el botón “Aplicar”. Se observará que el esclavo configurado aparece en la columna de la izquierda.



Repetir el proceso con los diferentes dispositivos instalados en la red ASI.



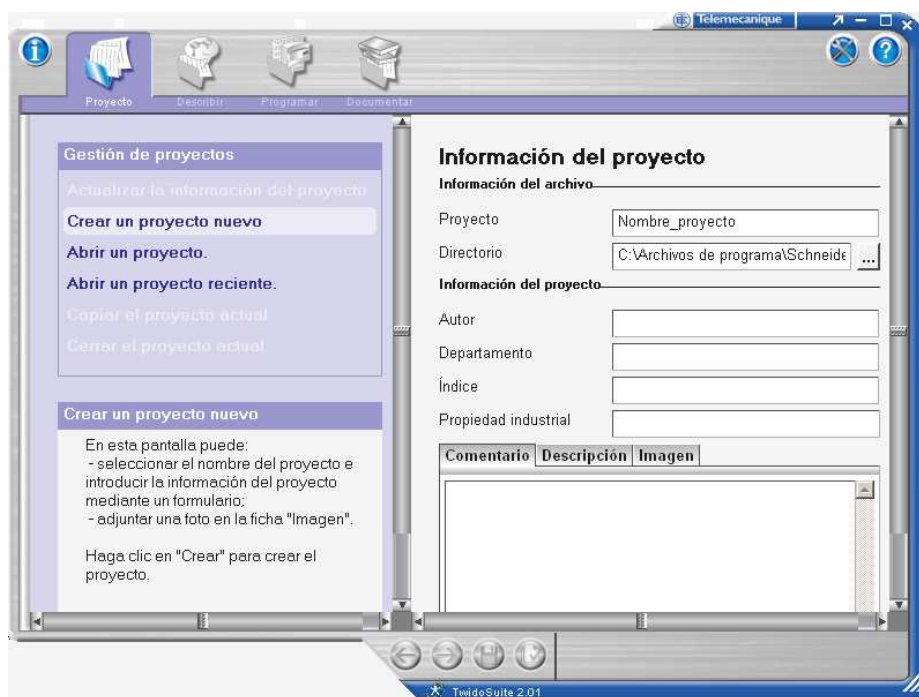
Una vez configurado todos los dispositivos, si pulsamos nuevamente en la pestaña de describir, podemos observar que del modulo maestro ASI aparecen colgada la red ASI, con los esclavos configurados.

5.4 Configuración de una red CanOpen:

El autómatas TWIDO dispone, entre otros, de un módulo de maestro Canopen.

Dentro del software “**Twidosuite**” se tendrá que configurar la red CanOpen con los dispositivos implicados en la red. En este caso tendremos un dispositivo Lexium el cual tendremos que configurar dentro del maestro. Los dispositivos Canopen disponen de un archivo “EDS” el cual se introduce en el maestro con el fin de que este sepa los objetos de comunicación que el esclavo contempla.

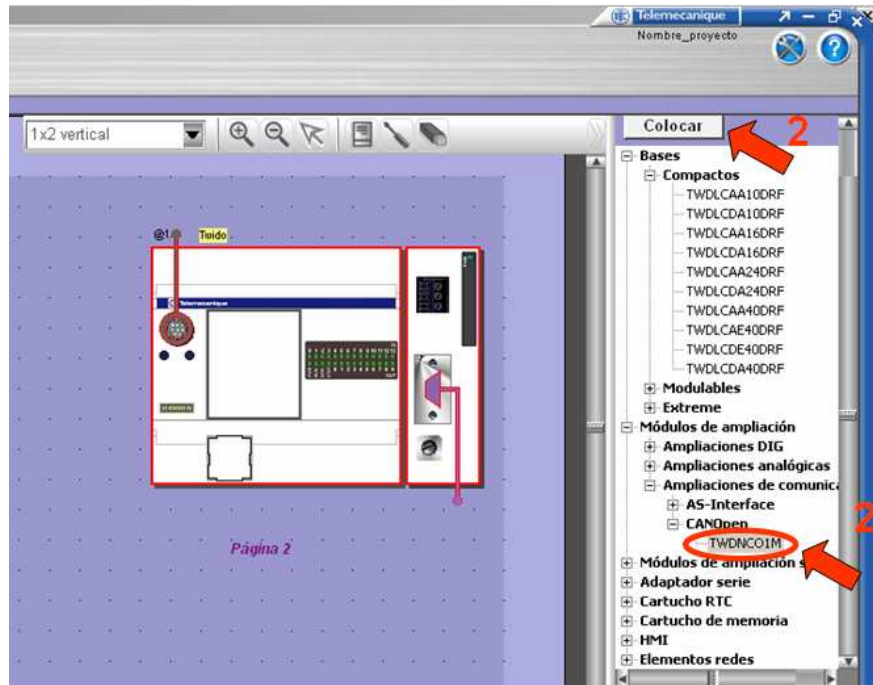
Abrir el programa TwidoSuite y crear un nuevo proyecto.



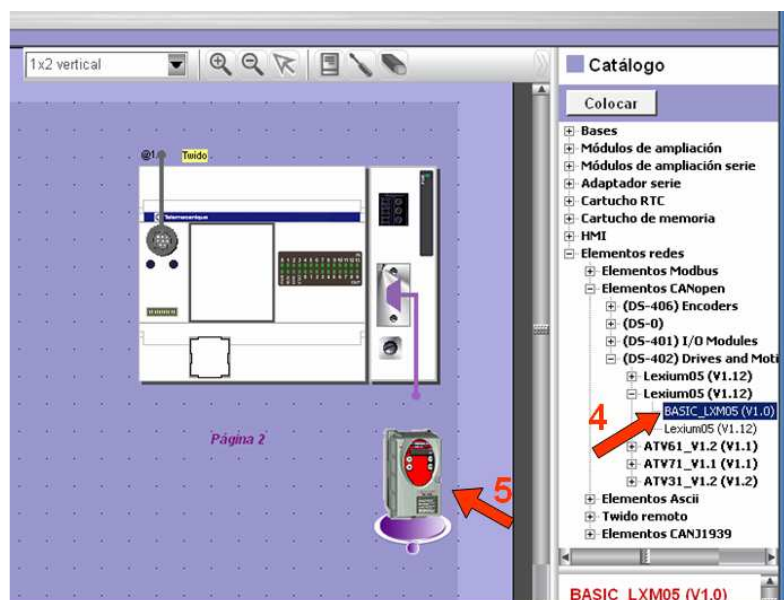
5. Abrir la pestaña Describir.



6. Con “Colocar”, seleccionar el módulo de comunicación CANopen TWDNCO1M.
7. Arrastrar y colocar junto a la base seleccionada.

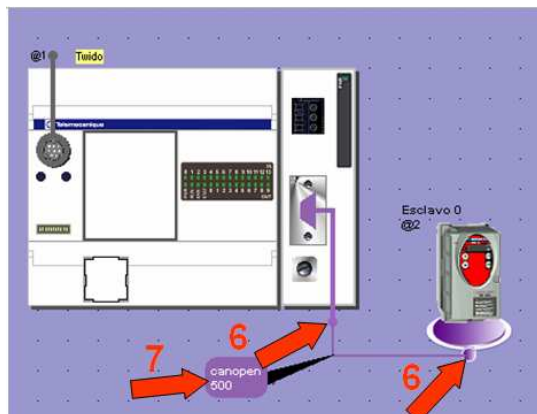


4. Seleccionar el dispositivo **BASIC_LXM05** para poder utilizar las macros de Lexium 05. Arrastrar y colocar.



6. Unir los nodos simplemente haciendo clic en uno de los nodos y con el botón del ratón presionado, unir el otro nodo.

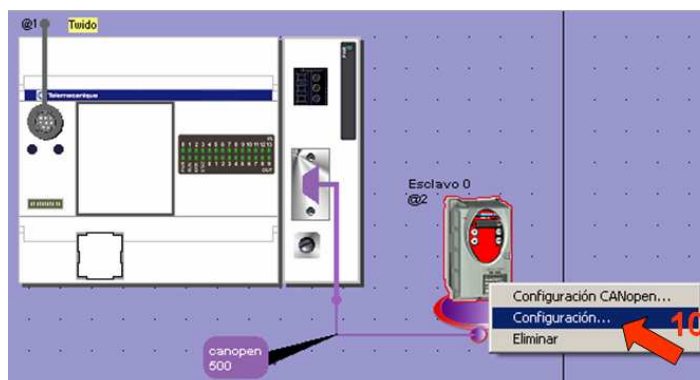
8. Configurar la red. Hacer doble clic sobre “my network”.



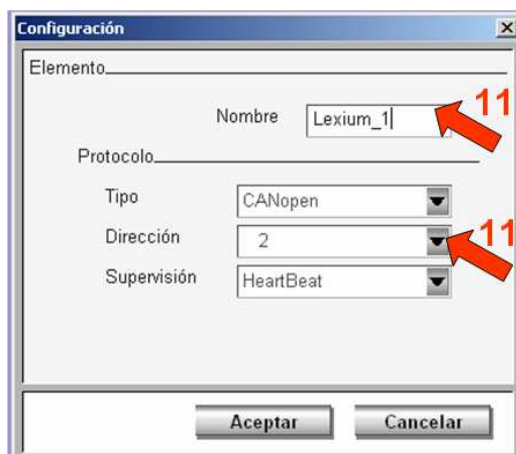
9. Cambiar el nombre de la red y configurar la velocidad, seleccionar la misma que la configurada en el Lexium como **Cobd**. Aceptar con OK.



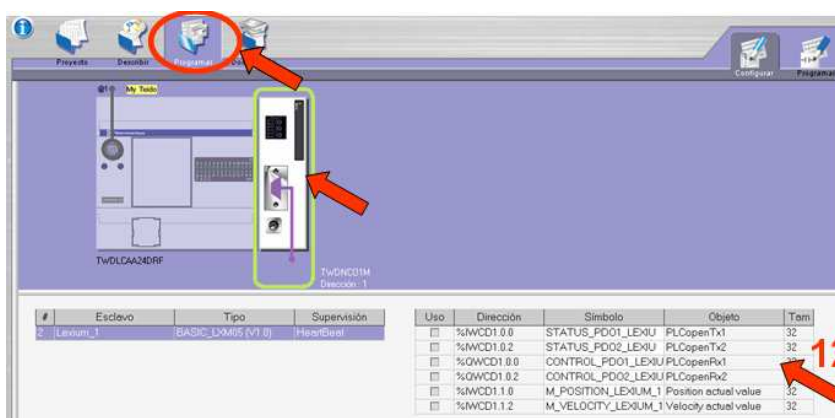
10. Para configurar el Lexium 05, hacemos clic con el botón derecho sobre el Lexium y seleccionamos **Configuración**.



11. Editamos el nombre y la dirección. Esta debe ser igual que la configurada en el Lexium 05 como **CoAd**.



12. En la pestaña Programa podemos ver los PDOs utilizados.



Una vez configurada la red CanOpen con el Lexium 05, ya estaremos listos para realizar un control de posición del Lexium 05 a través del Twido.

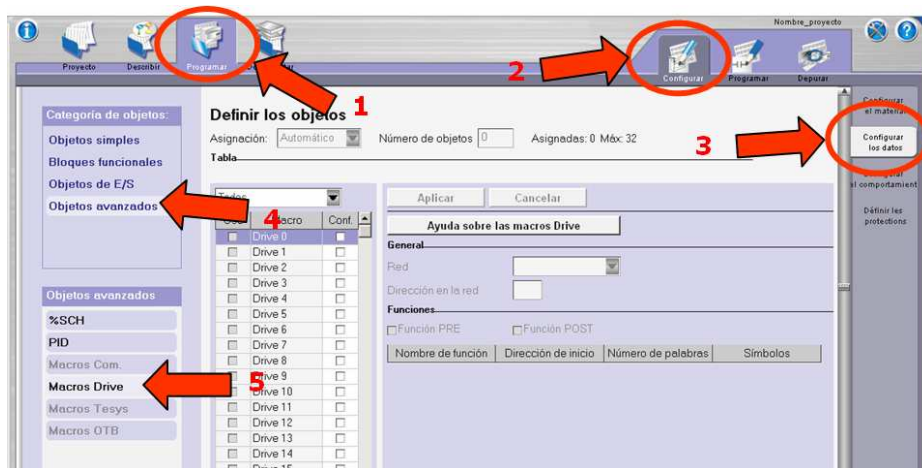
5.5 Macros Drive

El autómatas TWIDO dispone, entre otros, de macros específicas para el control del Lexium 05. Estas macros, como veremos a continuación, se configuran de manera sencilla y simplifica la programación del control de la aplicación.

Gracias a estas macros el programador también no tiene que disponer de unos conocimientos avanzados del protocolo del bus de comuni-

caciones utilizado para el control de la aplicación, ya que la gestión y la creación de la tramas de comunicación se realizan internamente en las macros.

1. Ir a la pestaña “**Programa**” del TwidoSuite.
2. Dentro de la pestaña Programa, en la parte superior derecha de la ventana, seleccionamos la pestaña “**Configurar**”.
3. Una vez se ha entrado en la ventana de configuración, en el lateral derecho activar la opción de “**Configurar los datos**”.
4. Aparecerá en el área principal las diferentes posibilidades de configuración de datos que ofrece el Twido. Las macros para el lexium 05 se encuentran dentro de “**Objetos avanzados**” (Lateral superior izquierdo).
5. Una vez seleccionado las diferentes tipos de macros, seleccionar abajo las “**Macros Drive**” que es la específica para controlar el Lexium 05.



6. Ahora elegiremos el número de macro que vamos a configurar, en la lista de macros que aparecen, vemos que tenemos desde la “**Drive 0**” hasta “**Drive 31**”, elegir el número de macro deseada activándola con el cuadro de activación de la columna de “**Conf.**” de la misma lista.

7. Cuando se ha seleccionado la macro a utilizar, se tendrá que configurar. Primero se seleccionará el tipo de red comunicación (Puede ser que se hayan configurado diferentes redes en el Twido), en **“General”** seleccionar en la opción **“Red”** el bus CanOpen.
8. Después de haber seleccionado el tipo de Red, en la opción **“Dirección de Red”** tendremos que decirle a que número de esclavo de la red vamos a controlar (en este caso como solo tenemos un esclavo Lexium 05, pondremos el número de esclavo que hemos configurado en la red y en el equipo).
9. Configurada el tipo de red y el esclavo al que vamos a controlar a través de la macro, en la zona de **“Funciones”** en la columna **“Símbolos”** activar el cuadro de opción habilitado que es de la fila **“M_MANAGER”**.
10. En la columna de **“Dirección de inicio”** pondremos la dirección de inicio de la memoria que va utilizar la macro (teniendo en cuenta que utiliza 50 palabras).



Asegurarse que tenemos espacios suficientes y que no pisamos otras variables.

11. Para confirmar la configuración de la macro, pulsar el botón **“Aplicar”**.

The screenshot shows the 'Definir los objetos' configuration window. On the left, a list of drives is shown with 'Drive 0' selected (indicated by arrow 6). The main configuration area has several fields: 'Red' set to 'CANopen' (arrow 7), 'Dirección en la red' set to '2' (arrow 8), and a table of functions. In the table, 'M_MANAGER' has its 'Símbolos' checkbox checked (arrow 9) and its 'Dirección de inicio' set to '0' (arrow 10). The 'Aplicar' button is highlighted with a red arrow (arrow 11).

| Nombre de función | Dirección de inicio | Número de palabras | Símbolos |
|-------------------|---------------------|--------------------|-------------------------------------|
| M_GEAR_OUT | 0 | 50 | <input type="checkbox"/> |
| M_MANAGER | 0 | 50 | <input checked="" type="checkbox"/> |
| M_POWER | 0 | 50 | <input type="checkbox"/> |
| M_STOP_AXIS | 0 | 50 | <input type="checkbox"/> |
| M_STOP_AXIS | 0 | 50 | <input type="checkbox"/> |
| M_HOME | 0 | 50 | <input type="checkbox"/> |
| M_RESET | 0 | 50 | <input type="checkbox"/> |
| M_MOVE_VEL | 0 | 50 | <input type="checkbox"/> |
| M_MOVE_ABS | 0 | 50 | <input type="checkbox"/> |
| M_MOVE_REL | 0 | 50 | <input type="checkbox"/> |
| M_SET_TORQUE | 0 | 50 | <input type="checkbox"/> |
| M_GEAR_IN | 0 | 50 | <input type="checkbox"/> |

La utilización de las diferentes funciones de la macro para el control la realizaremos en el programa, teniendo en cuenta que podemos tener configuradas diferentes macros para controlar diferentes equipos.

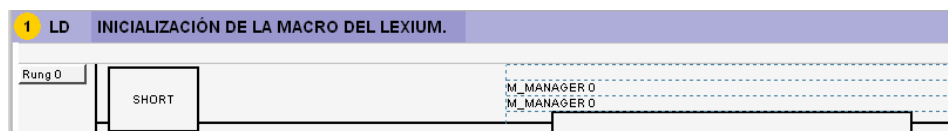
○ **Ejemplo de programación de la Macro Drive:**

El programa de control del autómatá tendrá la siguiente apariencia. Dependiendo de la función utilizada de la macro drive. Las funciones son las siguientes.

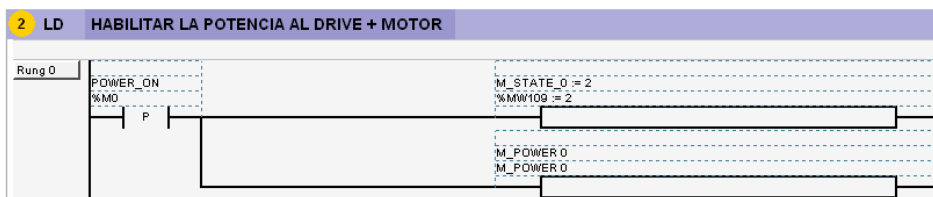
| Macro | Descripción de la función |
|---------------------|---|
| M_POWER | Habilita el servodrive para ejecutar movimientos |
| M_HOME | Ejecuta una búsqueda de origen (homing) |
| M_MOVE_ABS | Ejecuta un movimiento absoluto a la posición de referencia |
| M_MOVE_REL | Ejecuta un movimiento relativo de posición respecto a la posición actual |
| M_STOP_AXIS | Ejecuta una parada con rampa de deceleración |
| M_HALT_AXIS | Ejecuta una pausa del movimiento |
| M_MOVE_VEL | Ejecuta un control de velocidad |
| M_SET_TORQUE | Ejecuta un control de par |
| M_GEAR_IN | Habilita el modo engranaje electrónico (sincronismo maestro/esclavo) |
| M_GEAR_OUT | Deshabilita el modo engranaje electrónico (sincronismo maestro/esclavo) |
| M_RESET | Resetea los errores del servodrive. Se debe corregir la causa del error antes de usarlo |
| M_MANAGER | Esta macro controla el estado del servodrive y lo mantiene operativo |

Para la realización de la práctica realizaremos la programación una por una de las diferentes funciones de la macro, comprobaremos su correcto funcionamiento.

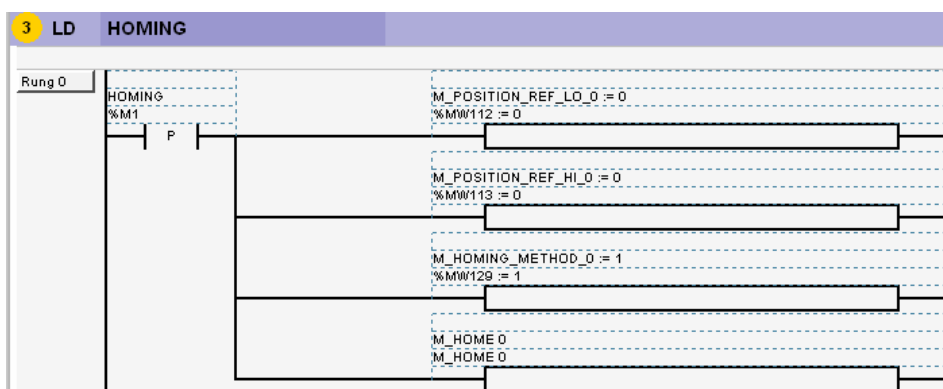
1. Lo primero que se debe colocar es la habilitación de la Macro para ello utilizaremos la función **M_MANAGER**.



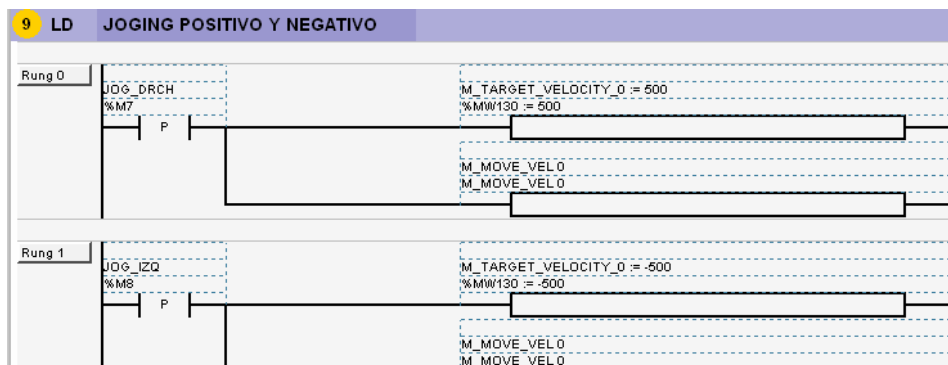
2. Ahora una vez habilitada la macro, lo primero que tenemos que realizar es activar la potencia del Lexium 05 antes de realizar cualquier tipo de movimiento. Para ello tendremos que realizar la llamada a la función **M_POWER**.



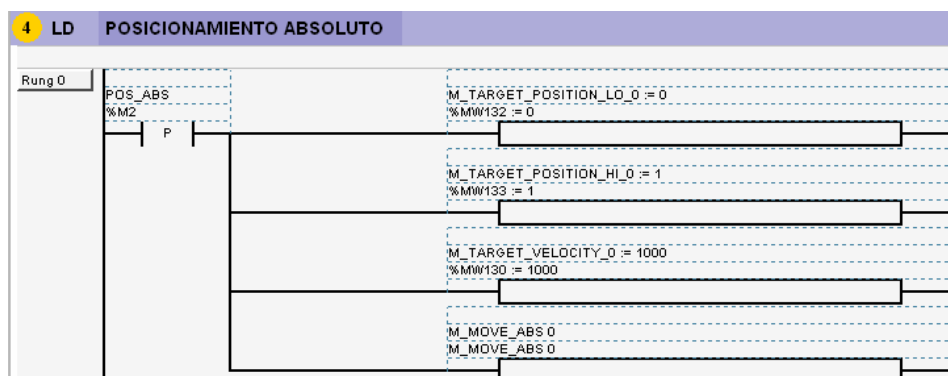
3. Ya tenemos la potencia habilitada en el Drive+Motor, para poder iniciar la aplicación de posicionado. El primer movimiento que realizaremos es el movimiento a referencia o Homing. Este se realiza con la función **M_HOMING**, donde tendremos que decirle que método de homing "**M_HOMING_METHOD**" queremos y el offset de la referencia que deseamos "**M_POSITION_REF_LO**" y "**M_POSITION_REF_HI**". (En este caso el método de homing es el número 1 = Va hasta el final de carrera de Límite negativo. Y el offset de referencia será 0).



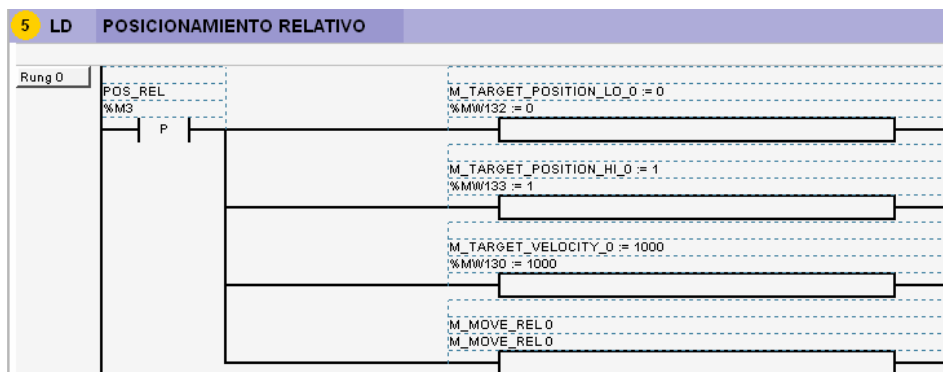
4. Una vez realizado el viaje a referencia ya podremos hacer tareas de posicionado, ya que el Lexium ya sabe donde está la posición 0. Ahora realizaremos un movimiento a velocidad fija (en este caso no realizaremos ningún tipo de posicionado) para ello tendremos que realizar la llamada a la función **M_MOVE_VEL** de la macro. Realizaremos un **JOG** en las dos direcciones del motor a una velocidad constante. Antes de hacer la llamada de la función tendremos que decirle a que velocidad queremos ir en la variable **M_TARGET_VELOCITY**.



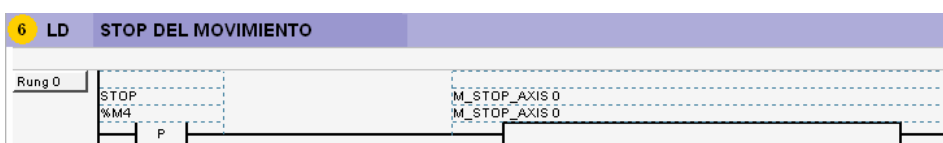
- Después de haber realizado el Jog a velocidad constante, ahora realizaremos un posicionamiento absoluto, es decir le diremos al motor que vaya a una posición en concreto referenciada desde la 0. Este se realiza llamando a la función **M_MOVE_ABS**, donde tendremos que decirle la posición a la que deseamos ir configurando las variables **“M_TARGET_POSITION_LO”**, **“M_TARGET_POSITION_HI”** y a la velocidad que queremos ir **“M_TARGET_VELOCITY”**.



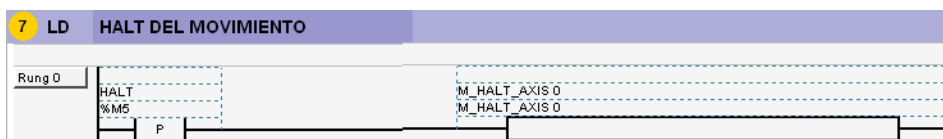
- Para realizar un movimiento relativo (posicionado que realizará desde el punto donde se encuentra en ese momento el motor) tendremos que llamar a la función **M_MOVE_REL** y se tendrá que rellenar exactamente las mismas variables que el movimiento absoluto.



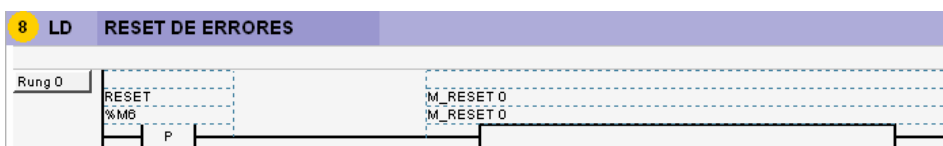
7. Para realizar una parada del movimiento, con rampa de deceleración llamaremos a la función **M_STOP**.



8. En cambio para realizar una pausa en el movimiento tendremos que realizar la llamada a la función **M_HALT**.



9. Si realizando la supervisión del Lexium 05, se detecta que ha entrado en error podemos resetearlo con la función **M_RESET** siempre y cuando la causa que generó el error haya desaparecido.



10. Tendremos unas variables que podremos utilizar para realizar un diagnostico de la aplicación y verificar la secuencia de movimientos.

Para realizar las pruebas pertinentes y comprobar que los movimientos que deseamos en nuestra aplicación son correctos. Nos ayudaremos creando una tabla de variables.

- %M0:** Activar la potencia del Driver+Motor.
- %M1:** Realizar el Homing.

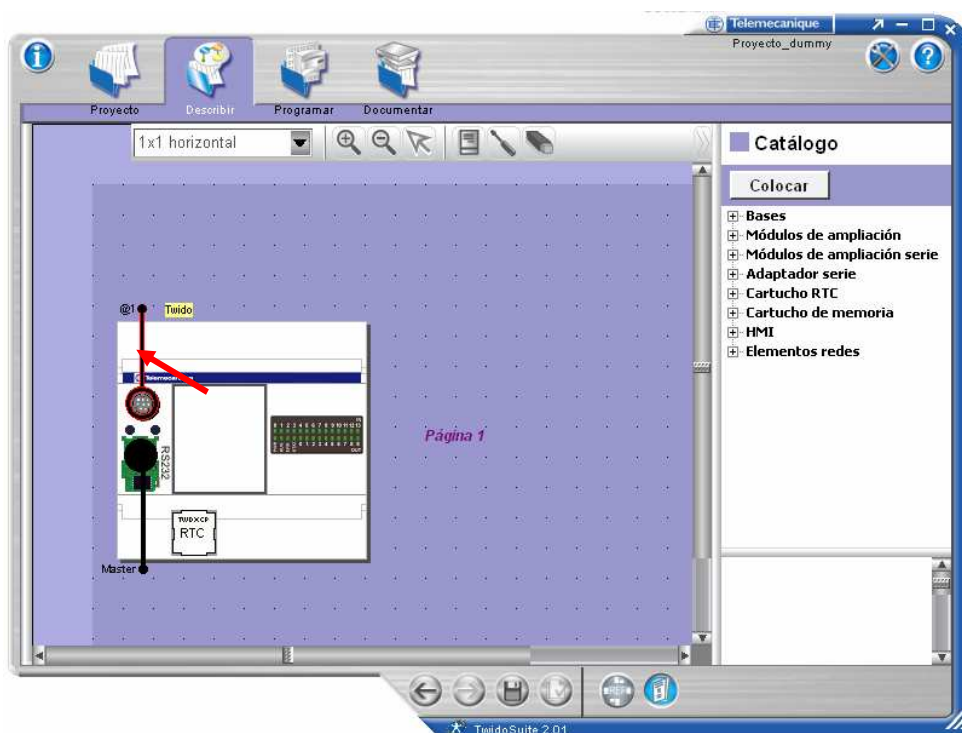
- %M2: Realizar posicionamiento absoluto.
- %M3: Realizar posicionamiento relativo.
- %M4: Stop del movimiento.
- %M5: Pausa del movimiento.
- %M6: Resetear sistema.
- %M7: Movimiento hacia la derecha a velocidad constante.
- %M8: Movimiento hacia la izquierda a velocidad constante.
- %MW130: Velocidad (24 a 6000 rpm).
- %MD132: Posición (dependerá si es absoluta o relativa).
- %MD119: Offset de la referencia.
- %MW112: Método de Homing.

5.6 Implementación del Twido en una red Ethernet:

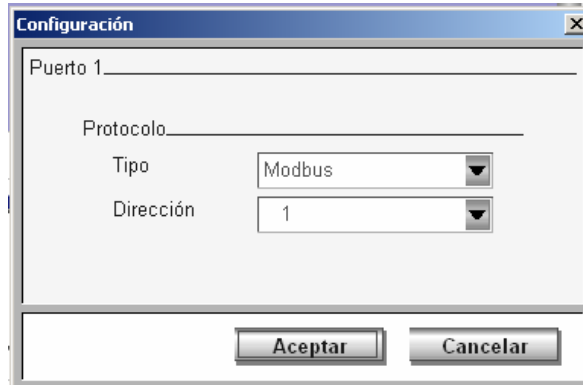
Para incluir el autómatas Twido en una red Ethernet necesitamos configurar el módulo de interface de Ethernet.

- o *Configuración del módulo TwidoPort:*

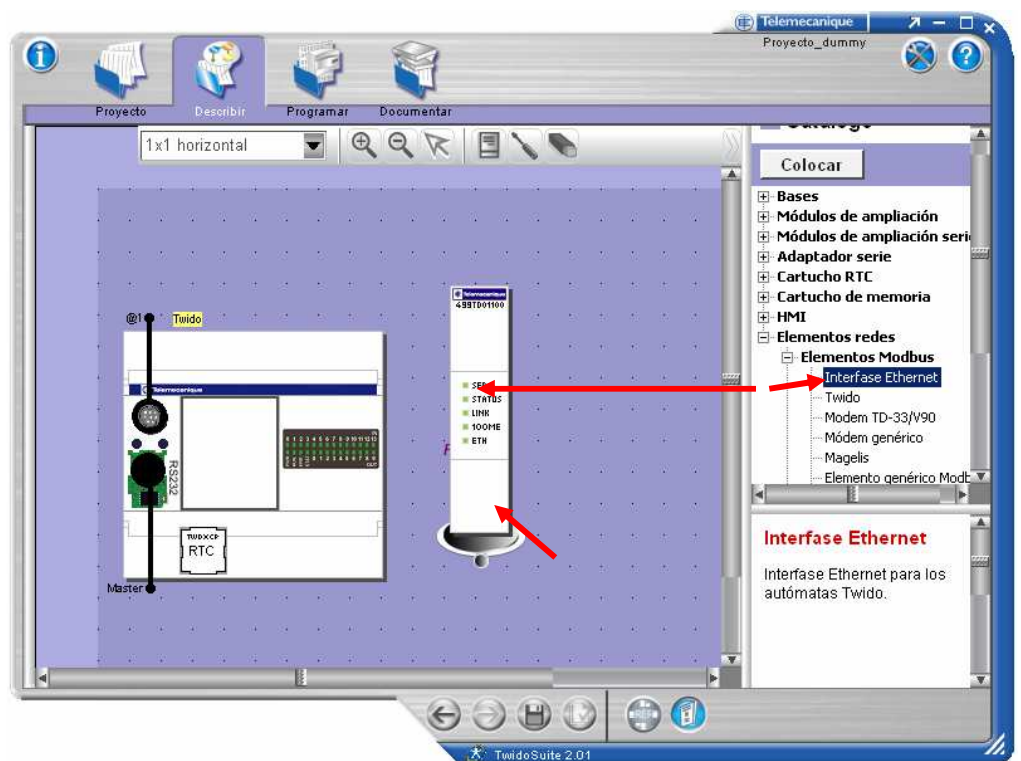
Seleccionar Puerto 1 (o Puerto 2 si está instalado) para configurarlo en la ventana Describir.



Configurar el cuadro de diálogo que aparece, tal como se explica en los pasos siguientes. Seleccionar **Modbus** en el cuadro Tipo de protocolo, con la dirección 1 y confirmar la configuración del puerto.



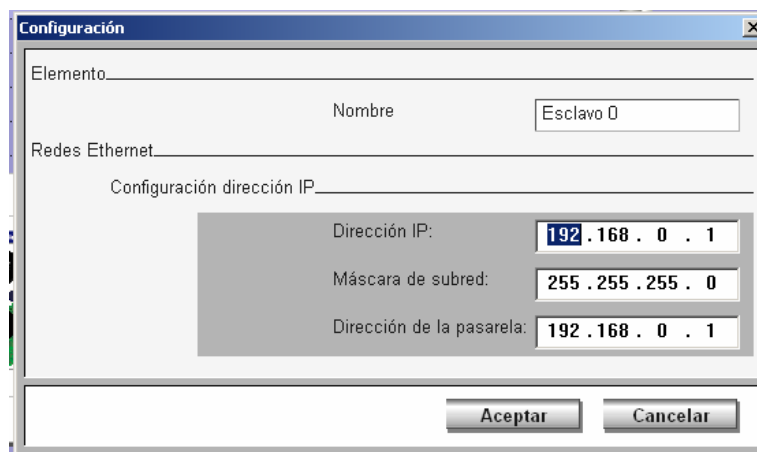
Seleccionar en el catálogo de productos, y elegir y agregar un módulo **499TWD01100** a la descripción de sistema, arrastrándolo de la lista del catálogo al área de configuración.



Para obtener un auto baudío de inicio más rápido, elegir 19200-8-N-1 con una dirección Modbus de Twido de 1.

- Configuración del módulo TwidoPort 499TWD01100:

Seleccionar el módulo TwidoPort 499TWD01100 para configurar los parámetros IP de TwidoPort. Aparece en pantalla la **ventana Configuración de TwidoPort**.



Introducir la **Dirección IP** estática de TwidoPort en formato de notación decimal con comas.

Introducir la **Máscara de subred** válida que el administrador de red ha asignado a TwidoPort. Tenga en cuenta que no se puede dejar este espacio en blanco; debe introducir un valor.



Para una buena comunicación del dispositivo, la máscara de subred configurada es de clase C, que ejecuta la aplicación TwidoSuite y la máscara de subred de TwidoPort deben coincidir.

De forma predeterminada, la aplicación TwidoSuite calcula y muestra una máscara de subred predeterminada basada en la clase de IP que haya introducido en el campo Dirección IP anterior. Los valores de máscara de subred predeterminados, según la categoría de las direcciones IP de red TwidoPort, siguen esta norma:

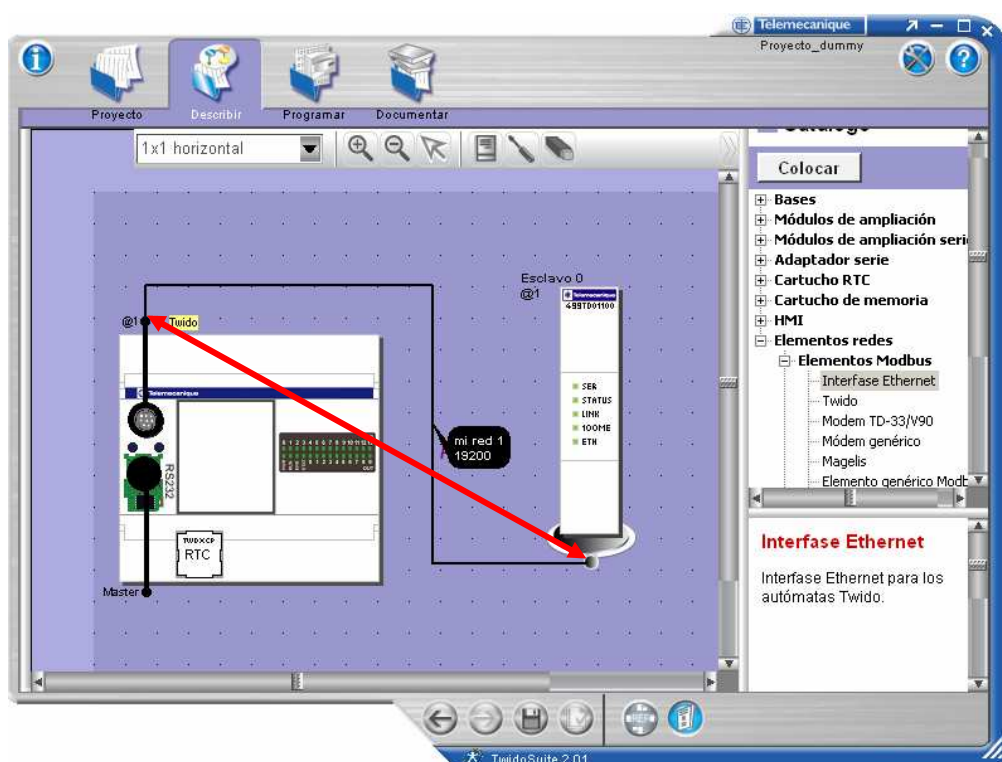
Red de clase A → Máscara de subred predeterminada: 255.0.0.0

Red de clase B → Máscara de subred predeterminada: 255.255.0.0

Red de clase C → Máscara de subred predeterminada: 255.255.255.0

Introducir la dirección IP de la **Pasarela**. En la LAN, la pasarela debe estar en el mismo segmento que TwidoPort. Por lo general, el administrador de red proporcionará esta información. Tener en cuenta que la aplicación no proporciona ningún valor predeterminado y que debe introducirse una dirección de pasarela válida en este campo.

Tras aceptar la configuración del módulo unir el puerto configurado con el módulo Twidoport.



Validar la configuración y transferirla al controlador Twido.


Apagar el controlador Twido y volverlo a encender.



Quando TwidoPort se ha configurado con TwidoSuite, la configuración IP de TwidoPort se almacena en el autómata Twido. De esta manera, el personal de mantenimiento puede intercambiar los TwidoPort sin necesidad de realizar una configuración adicional.

Para hacer uso de la función de instalación rápida, utilice TwidoSuite y actualice el firmware de TwidoSuite a la versión 3.4 o superior.

- Configuración de una conexión Ethernet en TwidoSuite:

Para permitir que el PC que ejecuta TwidoSuite y el controlador Twido se comuniquen mediante la red Ethernet. Ir a la ventana de preferencias a través del icono general de “Preferencias”  que se encuentra en la esquina superior derecha.

Hacer clic en el botón **Agregar** en el cuadro de diálogo **Gestión de conexiones**. Se añade una línea de conexión nueva. La línea nueva muestra ajustes de conexión predeterminados sugeridos. Tendrá que cambiar estos ajustes.

Gestión de conexiones

| Nombre | Tipo de conexión | IP/Phone | Unidad P/Dirección | Velocidad en baudios | Paridad | Bits de parada | Timeout | Tempo de espera de la pausa |
|--------------------|------------------|-------------|--------------------|----------------------|---------|----------------|---------|-----------------------------|
| COM1 | Serie | COM1 | Unidad P | | | 5000 | 20 | |
| COM2 | Serie | COM1 | Unidad P | | | 5000 | 20 | |
| Mi conexión 1 | Serie | 192.168.1.1 | Directo | | | 5000 | 20 | |
| Conexión por módem | Ethernet | xxxxxxxx | | 19200 | Ninguna | 1 | 5000 | 20 |

Para definir un valor nuevo en un campo, seleccionar el campo que desea y, a continuación, pulsar el botón **Modificar**.

En el campo **Nombre**, introducir el nombre descriptivo de la conexión nueva. Un nombre válido puede contener hasta 32 caracteres alfanuméricos.

En el campo **Tipo de conexión**, hacer clic para abrir la lista desplegable que incluye: Serie, Ethernet y USB (si lo hubiera). Seleccionar **Ethernet** si está ajustando una nueva conexión Ethernet entre el PC y el autómata Twido habilitado para Ethernet.

En el campo **IP/Phone**, introducir una **dirección IP** válida que es la información de IP del controlador Twido al que desea conectarse. **Dirección IP:** Introducir la dirección IP estática especificada anteriormente para el autómata Twido.

En una **conexión Ethernet**, el valor de Unidad IP/Dirección es Directo. Este **valor predeterminado debe cambiarse por 1** (o por cualquier dirección que se haya utilizado previamente para el controlador).

Utilizar los ajustes predeterminados en los campos **Timeout** y **Tiempo de espera de la pausa**, a menos que se tengan necesidades de timeout específicas.

Hacer clic en el botón **Aceptar** para guardar los ajustes de la conexión nueva y cerrar el cuadro de diálogo Gestión de conexiones.

Todas las conexiones recién agregadas se adjuntan a la tabla **Preferencias** → **Administración de conexiones** y a la tabla **Programa** → **Depuración** → **Conectar** → **Seleccionar una conexión**.

5.7 Implementación del Remote Link:

El enlace remote Link consiste en un bus diseñado para el intercambio de datos a alta velocidad entre un **Maestro** (dirección cero) y varios esclavos (**direcciones 1 a 7**).

El tipo de datos transferidos dependen de si el controlador remoto está configurado como un controlador peer ó como controlador remoto simple.

En el primer caso cada autómatas tiene su aplicación y los datos que se intercambian son palabras de entradas y salidas del tipo **%INWj.k** y **%QNWj.k** (j=dirección y k=palabra).

En el caso de funcionar como un controlador remoto simple, el Maestro trabaja con entradas y salidas distribuidas.

Una conexión remota debe utilizar un puerto EIA RS485 de tres cables como mínimo. Puede configurarse para utilizar el primero o un segundo puerto opcional, si lo hubiese.



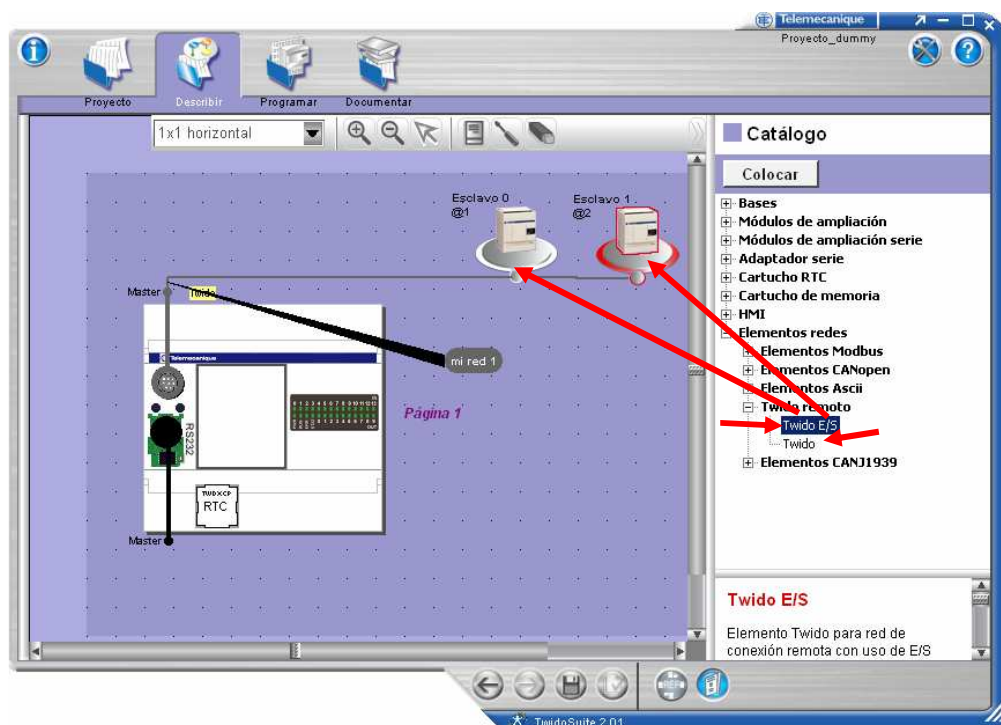
Sólo puede configurarse un puerto de comunicación como conexión remota simultáneamente.

○ *Configuración del autómata maestro*

El autómata maestro se configura mediante TwidoSuite para gestionar una red de conexión remota de hasta siete autómatas remotos. Estos siete autómatas remotos pueden configurarse como **E/S remotas** o **como autómatas Peer**. La dirección del maestro configurado mediante TwidoSuite corresponde a la **dirección 0**.

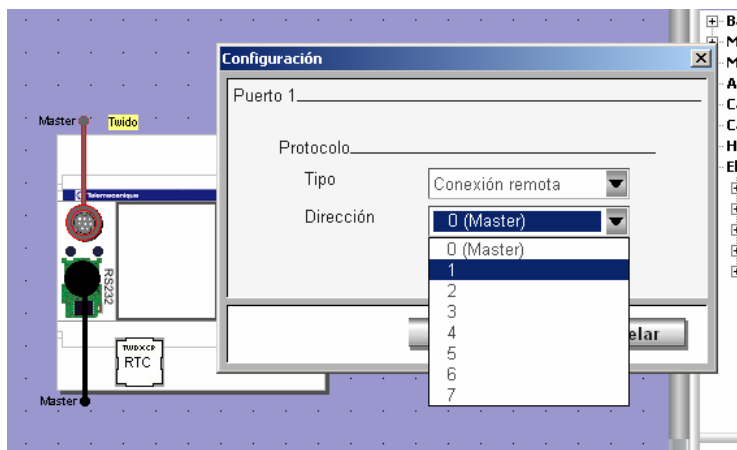
Para configurar un autómata como autómata maestro, debe configurar el **puerto 1 ó 2** con TwidoSuite como conexiones remotas y escoger la dirección **0** (maestro).

A continuación, en la ventana "**Agregar un autómata remoto**", defina los autómatas esclavo, ya sea como E/S remotas o como autómatas Peer, así como sus direcciones.

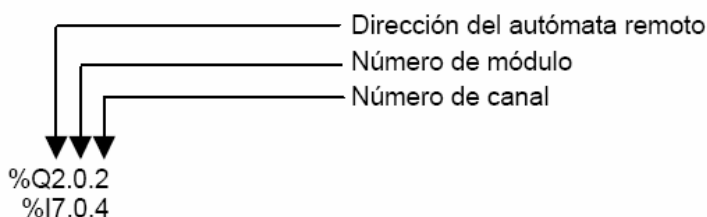


○ *Configuración del autómata esclavo 1 como E/S remota:*

El autómata esclavo se configura mediante TwidoSuite como **E/S remotas**. Para configurar un autómata como autómata maestro, debe configurar el **puerto 1 ó 2** con TwidoSuite como conexiones remotas y escoger la dirección **1**.



En este caso que el esclavo hace de E/S remotas no es necesario crear ningún programa, ya que actuara como entradas salidas distribuidas. Y el maestro accederá a las entradas y salidas directamente.

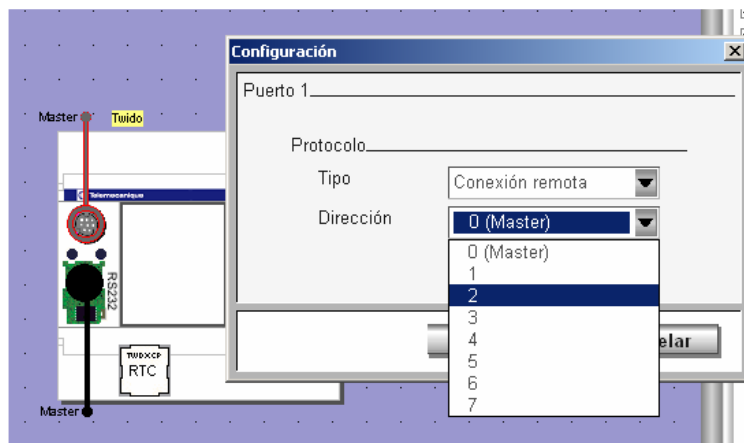


El número de módulo es 0 para las Entradas/Salidas Remotas.

Para comunicarse con la E/S remota, el autómata maestro utiliza la notación de entrada y salida estándar de %I y %Q. Para acceder al tercer bit de salida de la E/S remota configurada en la dirección 1, se utiliza la instrucción %Q1.0.2. De forma similar, para leer el quinto bit de entrada de la E/S remota configurada en la ubicación 7, se utiliza la instrucción %I7.0.4.

o *Configuración del autómata esclavo 2 como Peer:*

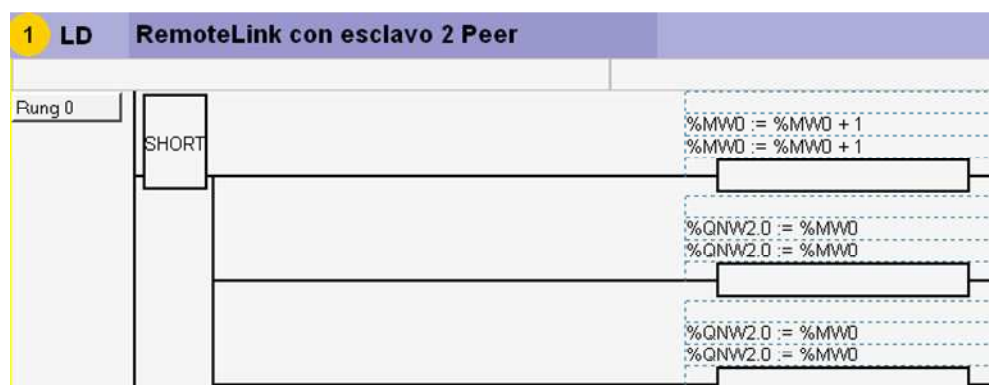
El autómata esclavo se configura mediante TwidoSuite como autómata remoto Peer. Para configurar un autómata como autómata maestro, debe configurar el **puerto 1 ó 2** con TwidoSuite como conexiones remotas y escoger la dirección **2**.



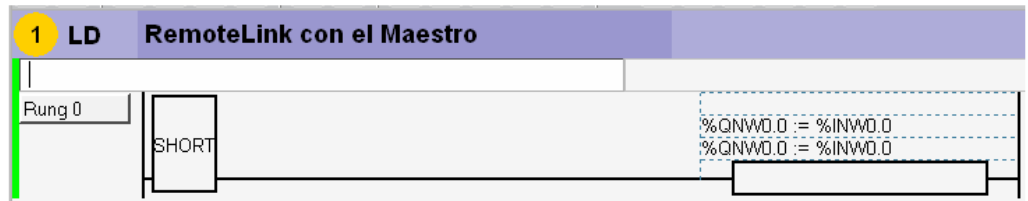
Para comunicarse con los autómatas **Peer**, el maestro utiliza las palabras de red **%INW** y **QNW** para intercambiar datos. Se accede a cada peer de la red mediante su dirección remota "j", a través de las palabras **%INWj.k** y **%QNWj.k**. Cada autómata Peer de la red utiliza de **%INW0.0** a **%INW0.3** y de **%QNW0.0** a **%QNW0.3** para acceder a los datos del **maestro**. Las palabras de red se actualizan de forma automática cuando los autómatas están en modo de ejecución o de detención.

o *Programación en el maestro:*

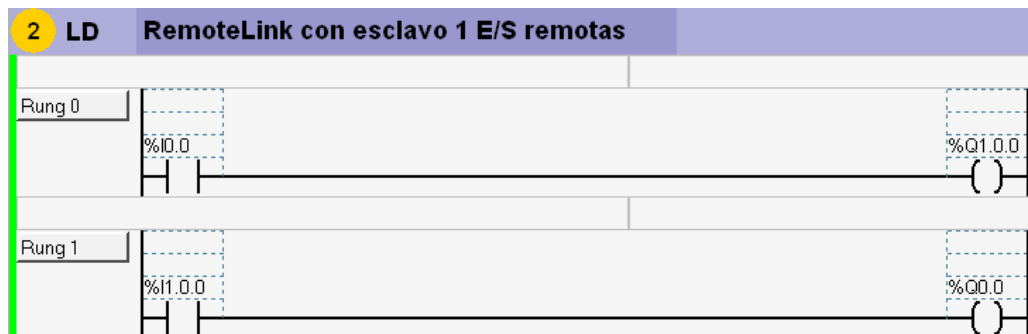
En este ejemplo, la aplicación maestro incrementa una palabra de memoria interna y lo comunica al autómata Peer mediante una sola palabra de red. El autómata Peer toma la palabra recibida del maestro y la reenvía. En el maestro, una palabra de memoria diferente recibe y almacena esta transmisión.



La programación dentro del esclavo Peer para el reenvío de las señales.



Para establecer una comunicación con el autómata de E/S remotas, el maestro envía sus entradas locales a las salidas de E/S remotas. Con el cableado de E/S externa de las E/S remotas, el maestro reenvía y recupera las señales.



6. Glosario

% _____

%I

Según la normativa IEC, %I indica un objeto de lenguaje de tipo entrada binaria.

%IW

Según la norma IEC, %IW indica un objeto de lenguaje de entrada analógica.

%KW

Según la norma IEC, %KW indica un objeto de lenguaje de palabra constante.

%M

Según la normativa IEC, %M indica un objeto de lenguaje de tipo bit de memoria.

%MW

Según la normativa IEC, %MW indica un objeto de lenguaje de tipo palabra de memoria.

%Q

Según la normativa IEC, %Q indica un objeto de lenguaje de tipo salida binaria.

%QW

Según la norma IEC, %QW indica un objeto de lenguaje de salida analógica.

%S

Representa un bit de sistema.

%SW

Representa un registro de palabra de sistema.

10/100 Base-T

El estándar 10/100 Base-T, que es una adaptación del estándar IEEE 802.3 (Ethernet), utiliza conductores de par trenzado con una longitud máxima de segmento de 100 m y termina en un conector RJ-45. Una red 10/100 Base-T es capaz de transmitir datos en redes Ethernet normales (10 Mbit/s) y Fast Ethernet (100 Mbits/s).

A _____

Aplicación

Una aplicación TwidoSuite se compone de un programa, datos de configuración, símbolos y documentación.

Archivo de aplicación

Las aplicaciones Twido se almacenan como archivos de tipo .twd.

ASCII

Código estándar americano para el intercambio de información (del inglés "American Standard Code for Information Interchange"). Protocolo de comunicación que representa caracteres alfanuméricos, incluidos números, letras y algunos caracteres gráficos y de control.

Autómata compacto

Tipo de autómata Twido que proporciona una configuración simple e integrada con ampliación limitada. Modular es el otro tipo de autómata Twido.

Autómata maestro

Autómata Twido configurado para ser el maestro en una red de conexión remota.

Autómata modular

Tipo de autómata Twido que ofrece una configuración flexible con funciones de ampliación. Compacto es el otro tipo de autómata Twido.

Autómata Peer

Autómata Twido configurado para ser el esclavo en una red de conexión remota. Una aplicación puede ejecutarse en la memoria del autómata peer y el programa puede acceder a los datos de E/S locales y de ampliación; sin embargo, los datos de E/S no pueden pasar al autómata maestro. El programa que está ejecutándose

en el autómeta peer pasa información al autómeta maestro mediante palabras de red (%INW y QNW).

Autómeta remoto

Autómeta Twido configurado para comunicarse con un autómeta maestro en una red de conexión remota.

B_____

BIT

Se trata de una unidad binaria para una cantidad de información que puede representar dos valores (o estados) diferentes: 0 ó 1.

Bloque de función

Unidad de programa de entradas y variables organizadas para calcular los valores de las salidas basadas en una función definida como un temporizador o un contador.

Bobina

Elemento del diagrama Ladder que representa una salida del autómeta.

BOOL

BOOL son las siglas del tipo booleano. Éste es el elemento de datos básico en computación. Una variable de tipo BOOL tiene un valor: 0 (FALSE) o 1 (TRUE). Un bit de extracción de palabras de tipo BOOL, por ejemplo: %MW10.4.

BOOTP

Bootstrap Protocol (Protocolo BOOTstrap). Protocolo UDP/IP que permite que un asiento de Internet obtenga los parámetros IP correspondientes basados en su dirección MAC.

BYTE

Un BYTE es un conjunto de ocho bits. Un BYTE puede introducirse en binario o en base 8. El tipo BYTE se codifica en formato de ocho bits que, en formato hexadecimal, tiene un rango de 16#00 a 16#FF.

C_____

CADENA DE CARACTERES

Una variable de tipo STRING es una cadena de caracteres ASCII. La longitud máxima de una cadena de caracteres es de 65.534 caracteres.

CAN

Controller Area Network (Red de área del controlador). El protocolo CAN (ISO 11898) para redes de bus serie está diseñado para la interconexión de dispositivos inteligentes (de varios fabricantes) en sistemas inteligentes para aplicaciones industriales de tiempo real. Los sistemas CAN multimaestro aseguran una alta integridad de datos por medio de la aplicación de la difusión de mensajes y de mecanismos de error avanzados. CAN, diseñado originalmente para utilizarlo en automóviles, se utiliza actualmente en una amplia variedad de entornos industriales de control automático.

CANopen

CANopen es un protocolo de nivel superior que se emplea en redes de automatización. Está basado en la capa de aplicación CAN (CAL) de conformidad con la norma CiA DS 301 (EN 50325-4).

Cartucho de memoria

Cartuchos de memoria de copia de seguridad opcionales que pueden utilizarse para realizar una copia de seguridad y restaurar una aplicación (datos de configuración y programa). Hay dos tamaños disponibles: 32 y 64 KB.

CEM

Electromagnetic Compatibility (Directiva de compatibilidad electromagnética). Los dispositivos que cumplen los requisitos de CEM pueden funcionar dentro de los límites electromagnéticos que estima el sistema sin que se produzcan errores.

Clase de servicio

Las clases de servicios de Transparent Ready permiten identificar los servicios proporcionados por cada dispositivo, como: los servicios de diagnóstico, visualización y control mediante las tecnologías web y servicios de comunicación Ethernet. Las clases de servicios de Transparent Ready simplifican de este modo la elección de productos y garantizan su interoperabilidad dentro de una arquitectura.

COB

Communication Object: unidad de transporte en el bus CAN. Un COB se identifica mediante un identificador único codificado en 11 bits, [0, 2047]. Un COB contiene, al

menos, 8 bytes de datos. La prioridad de transmisión de un COB viene determinada por su identificador; cuanto menor sea éste, mayor prioridad tiene el COB asociado.

CRC

Cyclic Redundancy Checksum: checksum de redundancia cíclica que indica que no se ha "deformado" ningún carácter durante la transmisión de la trama.

Conexión remota

Bus maestro/esclavo de alta velocidad diseñado para transferir una pequeña cantidad de datos entre el autómata maestro y hasta siete autómatas esclavo remotos. Hay dos tipos de autómatas remotos que pueden configurarse para transferir datos a un autómata maestro: autómata peer, que puede transferir datos de la aplicación o autómata remoto de E/S, que puede transferir datos de E/S.

Constantes

Valor configurado que no se puede modificar por el programa que se está ejecutando.

Contacto

Elemento del diagrama Ladder que representa una entrada al autómata.

Contador

Bloque de función utilizado para contar eventos (conteo progresivo o regresivo).

Contadores muy rápidos:

Bloque de función que proporciona un conteo más rápido que el disponible con bloques de función de contadores y contadores rápidos. Un contador muy rápido puede contar a una velocidad de hasta 20 kHz.

Contadores rápidos

Bloque de función que proporciona un conteo progresivo y regresivo más rápido que el disponible en el bloque de función del contador. Un contador rápido puede contar a una velocidad de hasta 5 kHz.

D _____

DINT

DINT es la forma abreviada de «Double INTeGer» (entero doble) (codificado en 32 bits). Los límites inferior y superior figuran a continuación: de - (2 elevado a 31) a (2 elevado a 31) - 1. Ejemplo: -2.147.483.648, 2.147.483.647, 16#FFFFFFFF.

Dirección IP

Internet Protocol Address (Dirección de protocolo de Internet). Esta dirección de 32 bits se asigna a hosts que utilizan TCP/IP.

Dirección MAC

Media Access Control Address (Dirección de control de acceso a medios). Número de 48 bits, exclusivo en una red, que se programa en cada dispositivo o tarjeta de red cuando se fabrica.

Direcciones

Registros internos del autómata utilizados para almacenar valores para variables de programa, constantes, E/S, etc. Las direcciones se identifican con un prefijo con el símbolo de porcentaje (%). Por ejemplo, %I0.1 especifica una dirección de la memoria RAM del autómata que contiene el valor para el canal de entrada 1.

DWORD

DWORD es la forma abreviada de «Double Word» (palabra doble). El tipo DWORD se codifica en un formato de 32 bits.

E_____

Editor de configuración

Ventana especializada de TwidoSuite utilizada para gestionar la configuración de hardware y software.

Editor de Ladder Logic

Ventana de TwidoSuite especializada y utilizada para editar un programa Ladder.

Editor de lista

Editor de programas simple utilizado para crear y editar un programa de lista.

Editor de tablas de animación

Ventana especializada en la aplicación TwidoSuite para ver y crear tablas de animación.

EDS

Archivo de descripción para cada dispositivo CAN (suministrado por el fabricante).

EEPROM

Memoria de sólo lectura programable y que se puede borrar de forma eléctrica. Twido tiene una EEPROM interna y un cartucho de memoria EEPROM externa opcional.

Ejecutar

Comando que hace que el autómatas ejecute un programa de aplicación.

Escalón

Un escalón se introduce entre dos barras potenciales en una cuadrícula y está compuesto por un grupo de elementos gráficos unidos entre sí mediante conexiones horizontales y verticales. Las dimensiones máximas de un escalón son siete filas y once columnas.

Estado del monitor

Estado operativo de TwidoSuite que se muestra en la barra de estado cuando se conecta un PC a un autómatas en modo de protección contra escritura.

Estado inicial

Estado de funcionamiento de TwidoSuite que aparece en la barra de estado cuando se inicia TwidoSuite o no tiene ninguna aplicación abierta.

Estado offline

Estado de funcionamiento de TwidoSuite que se muestra en la barra de estado cuando un PC no está conectado a un autómatas.

Estado online

Estado de funcionamiento de TwidoSuite que se muestra en la barra de estado cuando un PC está conectado al autómatas.

Estados de funcionamiento

Indica el estado de TwidoSuite. Se muestra en la barra de estado. Hay cuatro estados de funcionamiento: inicial, offline, online y supervisar.

F _____

Fechadores

Bloque de función utilizado para programar funciones de fecha y hora con el fin de controlar eventos. Requiere la opción Reloj de tiempo real.

Forzado

Ajustar voluntariamente las entradas y salidas del autómatas en 0 ó 1 aunque los valores reales sean diferentes. Se utiliza para depurar mientras se anima un programa.

Funcionamiento offline

Modo de funcionamiento de TwidoSuite cuando un PC no está conectado al autómatas y la aplicación de la memoria del PC no es la misma que la de la memoria del autómatas.

El usuario crea y desarrolla una aplicación durante el funcionamiento offline.

Funcionamiento online

Modo de funcionamiento de TwidoSuite cuando un PC está conectado al autómatas y la aplicación de la memoria del PC es la misma que la de la memoria del autómatas. El funcionamiento online permite depurar una aplicación.

Funciones de fecha y hora

Permiten el control de eventos por mes, día y hora. Consulte "Fechadores".

G_____

Grafcet

Grafcet permite representar gráficamente y de forma estructurada el funcionamiento de una operación secuencial. Método analítico que divide cualquier sistema de control secuencial en una serie de pasos a los que se asocian acciones, transiciones y condiciones.

H_____

HMI

Human-Machine Interface (Interface hombre-máquina). Interface del operador, generalmente gráfica, para equipos industriales.

HTTP

«Hypertext Transfer Protocol» (protocolo de transferencia de hipertexto)

I_____

IEC 61131-3

Normativa internacional: controles de lógica programables - Apartado 3: lenguajes de programación.

IEEE

Institute of Electrical and Electronics Engineers, Inc (Instituto de Ingeniería Eléctrica y Electrónica). Organismo internacional de normas y de evaluaciones de conformidad para todos los campos de la electrotecnia, incluidas la electricidad y la electrónica.

IL

IL son las siglas de lista de instrucciones (Instruction List). Este lenguaje consiste en una serie de instrucciones básicas. Este lenguaje es muy similar al lenguaje ensamblador utilizado en los procesadores de programa. Cada instrucción está compuesta por un código de instrucción y un operando.

Indicador LED

Light Emitting Diode (Diodo electroluminiscente). Indicador que se enciende cuando pasa por él electricidad. Señala el estado de funcionamiento del módulo de comunicaciones.

Inicializar

Comando que ajusta todos los valores de datos a estados iniciales. El autómata debe estar en modo Detener o Error.

Inicio en frío o reinicio

Inicio del autómata con todos los datos inicializados con los valores predeterminados y el programa iniciado desde el comienzo con todas las variables eliminadas. Todos los parámetros de software y hardware se inicializan. Se puede originar un reinicio en frío cargando una aplicación nueva en la RAM del autómata. Todos los autómatas sin batería de seguridad se activan siempre mediante un inicio en frío.

Instancia

Objeto exclusivo de un programa que pertenece a un tipo específico de bloque de función.

Por ejemplo, en formato de temporizador %T*M*i, i es un número que representa la instancia.

Internet

La interconexión global de redes de comunicación informática basada en TCP/IP.

IP

Protocolo de Internet. Protocolo de capa de red habitual. IP normalmente se usa con TCP.

J _____

K _____

L _____

LD

LD son las siglas de diagrama de contactos (Ladder Diagram). LD es un lenguaje de programación que representa las instrucciones que deben ejecutarse en forma de diagramas gráficos muy similares a los esquemas eléctricos (contactos, bobinas, etc.).

M _____

Máscara de subred

La máscara de subred es una máscara de bits que identifica o determina qué bits, en una dirección IP, corresponden a la dirección de red y cuáles corresponden a las partes de subred de dicha dirección. La máscara de subred se compone de la dirección de red y de los bits reservados para la identificación del trabajo de subred.

Memoria flash

La memoria flash es una memoria no volátil que se puede sobrescribir. Se almacena en una memoria EEPROM especial que se puede borrar y volver a programar.

Modbus

Protocolo de comunicaciones maestro-esclavo que permite a un solo maestro solicitar respuestas de esclavos.

Modo de exploración

Especifica el modo en el que el autómata explora un programa. Existen dos tipos de modos de exploración: normal (cíclico), el autómata explora de forma continua; o periódico, el autómata explora durante el período seleccionado (entre 2 y 150 ms) antes de iniciar otra exploración.

Módulos de E/S de ampliación

Módulos de E/S de ampliación opcionales disponibles para agregar puntos de E/S a un autómata Twido. (No todos los modelos del autómata permiten la ampliación.)

N_____

Navegador de aplicación

Ventana especializada en TwidoSuite que muestra una vista gráfica en forma de árbol de una aplicación. Ofrece una configuración y una visualización correctas de una aplicación.

Nodo

Dispositivo direccionable en una red de comunicaciones.

O_____

P_____

Pasarela

Dispositivo que conecta redes con arquitecturas de redes diferentes y que opera en la capa de aplicación del modelo OSI. Este término puede referirse a un enrutador.

Paso

Un paso Grafcet designa un estado de funcionamiento secuencial de automatización.

PC

Ordenador personal.

PLC

Autómata programable Twido. Existen dos tipos de autómatas: compacto y modular.

PLS

Generación de pulsos. Bloque de función que genera una onda cuadrada con un ciclo de servicio 50% activado y 50% desactivado.

Potenciómetro analógico

Tensión aplicada que puede ajustarse y convertirse en un valor binario para ser utilizado por una aplicación.

Preferencias

Cuadro de diálogo con opciones seleccionables para configurar los editores de programa Ladder y de lista.

Protocolo

Describe los formatos de los mensajes y establece las reglas que usan dos o más dispositivos para comunicarse mediante esos formatos.

Q_____

R_____

RAM

Memoria de acceso aleatorio (del inglés "Random Access Memory"). Las aplicaciones Twido se descargan en una memoria RAM interna y volátil que se va a ejecutar.

Red

Dispositivos interconectados que comparten una ruta de datos y un protocolo comunes para la comunicación.

Reinicio en caliente

Inicio del autómata después de una pérdida de alimentación sin modificar la aplicación. El autómata regresa al estado existente antes de la pérdida de alimentación y completa la exploración en curso. Todos los datos de la aplicación quedan intactos. Esta función sólo está disponible en autómatas modulares.

Reloj de tiempo real

Opción que conservará la hora aunque el autómata no reciba alimentación durante un tiempo determinado.

RTU

Unidad de terminal remoto (del inglés "Remote Terminal Unit"). Protocolo que utiliza ocho bits, empleado para establecer comunicación entre un autómatas y un PC.

S_____

Sección

Módulo de programa perteneciente a una tarea que se puede escribir en el lenguaje elegido por el programador (FBD, LD, ST, IL o SFC). Una tarea puede estar compuesta por distintas secciones, y el orden de ejecución de estas secciones corresponde a su orden de creación. Este orden se puede modificar.

Símbolo

Un símbolo es una cadena con un máximo de 32 caracteres alfanuméricos, de los cuales el primer carácter es alfabético. Permite personalizar un objeto del autómatas para facilitar el mantenimiento de la aplicación.

Subrutina

Módulo de programa perteneciente a una tarea (MAST, FAST) que se puede escribir en el lenguaje elegido por el programador (FBD, LD, ST, o IL). Una subrutina sólo se puede llamar desde una sección o desde otra subrutina que pertenezca a la tarea en la que se declare.

T_____

Tabla de animación

Tabla creada con un editor de lenguaje o una pantalla de funcionamiento. Cuando un PC se conecta al autómatas, muestra las variables del autómatas y permite forzar los valores durante la depuración. Puede guardarse como archivo independiente con la extensión .tat.

Tabla de símbolos

Tabla de los símbolos utilizados en una aplicación. Se muestra en el editor de símbolos.

TCP

Protocolo de control de la transmisión (del inglés "Transmission Control Protocol").

TCP/IP

Conjunto de protocolos formado por el protocolo de control de la transmisión y el protocolo de Internet. Es el conjunto de protocolos de comunicaciones en el que se basa Internet.

Temporizador

Bloque de función utilizado para seleccionar la duración para controlar un evento.

Tipos de trama

Existen dos tipos habituales de trama: Ethernet II y IEEE 802.3.

Trama

Grupo de bits que forman un bloque de información binario. Las tramas contienen información o datos de control de la red. El tamaño y la composición de una trama están determinados por la tecnología de red utilizada.

Twido

Línea de autómatas de Schneider Electric compuesta por dos tipos de autómatas (compacto y modular), módulos de ampliación para agregar puntos de E/S y opciones como reloj de tiempo real, comunicaciones, monitor de operación y cartuchos de memoria de copia de seguridad.

TwidoSuite

Software de desarrollo gráfico de Windows de 32 bits para configurar y programar autómatas Twido.

U _____

USB

Universal Serial Bus (Bus de serie universal). Un USB es una interface de hardware casi universal para la conexión de dispositivos periféricos.

V _____

Variable

Entidad de memoria del tipo BOOL, WORD, DWORD, etc., cuyos contenidos se pueden modificar desde el programa durante su ejecución.

Visualizador de errores de programa

Ventana de TwidoSuite especializada utilizada para ver errores de programa y advertencias.

Visualizador de referencias cruzadas

Ventana especializada en la aplicación TwidoSuite para ver referencias cruzadas.

W_____

WORD

El tipo WORD se codifica en formato de 16 bits y se utiliza para procesar cadenas de bits.

X_____

Y_____

Z_____

Notas

A series of 30 horizontal lines for taking notes.

Notas

