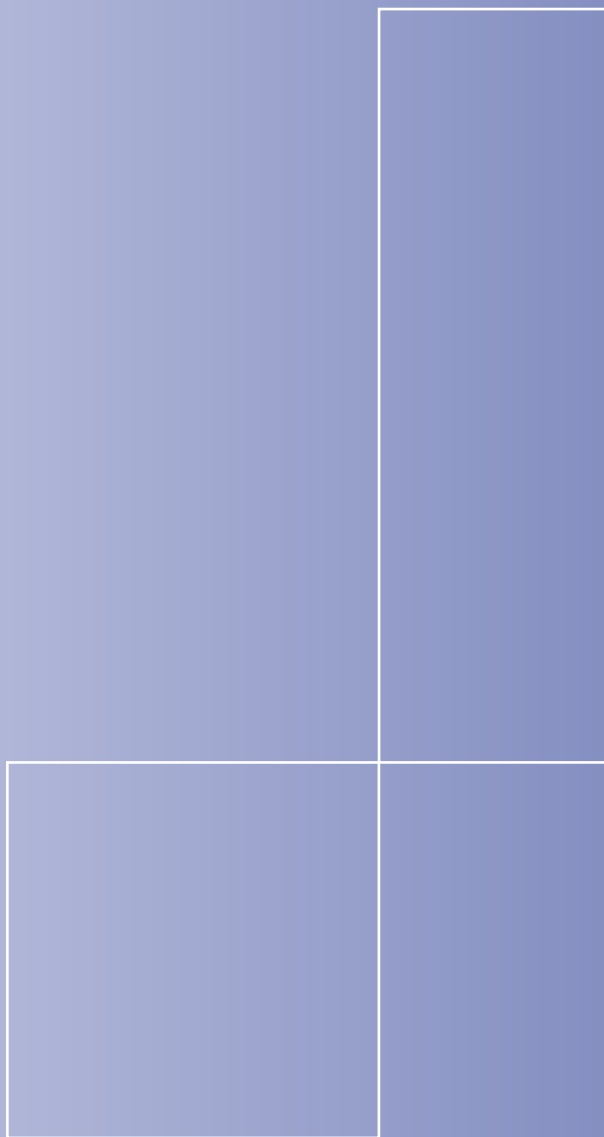


Introducción a STEP 7 Lite V3.0

SIMATIC
STEP 7 Lite V3.0
Edición 04/2004



simatic
STEP 7 Lite

SIEMENS

SIEMENS

Software SIMATIC

Introducción a STEP 7 Lite V3.0

Getting Started

04/2004

A5E00293892-01

Consignas de seguridad

Este manual contiene las informaciones necesarias para la seguridad personal así como para la prevención de daños materiales. Las informaciones están puestas de relieve mediante señales de precaución. Las señales que figuran a continuación representan distintos grados de peligro:



Peligro

Significa que, si no se adoptan medidas preventivas adecuadas, **se producirá** la muerte o bien lesiones corporales graves o daños materiales considerables,



Advertencia

Significa que, si no se adoptan medidas preventivas adecuadas, **puede producirse** la muerte o bien lesiones corporales graves o daños materiales considerables.



Precaución

Significa que, si no se adoptan medidas preventivas adecuadas, **pueden producirse** lesiones corporales o daños materiales.

Precaución

Significa que, si no se adoptan medidas preventivas adecuadas, pueden producirse daños materiales.

Atención

Se trata de una información importante sobre el producto o sobre una parte determinada del manual sobre la que se desea llamar particularmente la atención.

Personal cualificado

La puesta en servicio y el funcionamiento de los equipos sólo está autorizado a **personas cualificadas**. En el sentido del manual se trata de personas que disponen de los conocimientos técnicos y de la autorización necesaria para poner en funcionamiento, conectar a tierra y marcar los equipos, sistemas y circuitos de acuerdo con los estándares de seguridad.

Uso conforme

Considere lo siguiente:



Advertencia

El equipo y los componentes del sistema sólo se podrán utilizar para los casos de aplicación previstos en el catálogo y en la descripción técnica, y sólo con los equipos y componentes de proveniencia tercera recomendados y homologados por Siemens. El funcionamiento correcto y seguro del producto presupone un transporte, almacenamiento, instalación y montaje conforme a las prácticas de la buena ingeniería, así como un manejo y mantenimiento rigurosos.

Marcas

SIMATIC®, SIMATIC HMI® y SIMATIC NET® son marcas registradas de Siemens AG.

Los restantes nombres y designaciones que aparecen en este documento pueden ser marcas cuya utilización por terceros para sus propios fines puede violar los derechos de sus propietarios.

Copyright © Siemens AG 2004 All rights reserved

La divulgación y reproducción de este documento, así como el uso y la comunicación de su contenido no están autorizados, a no ser que se obtenga el consentimiento expreso. Los infractores quedan obligados a la indemnización de los daños. Todos los derechos reservados, en particular para el caso de concesión de patentes o de modelos de utilidad.

Siemens AG
Bereich Automation and Drives
Geschäftsgebiet Industrial Automation Systems
Postfach 4848, D-90327 Nürnberg

Siemens Aktiengesellschaft

Exención de responsabilidad

Hemos probado si el contenido de esta documentación se corresponde con el hardware y software descritos en ella. Sin embargo, no podemos excluir que existan diferencias, de manera que no nos responsabilizamos de las posibles divergencias. El contenido de esta documentación está sometido a revisiones con regularidad. En caso necesario se incluyen las correcciones en la siguiente edición. Agradecemos sugerencias.

© Siemens AG 2004
Sujeto a cambios técnicos.

A5E00293892

Bienvenido a STEP 7 Lite

... el software SIMATIC que permite crear programas PLC en KOP, FUP o AWL para los sistemas SIMATIC S7-300 (incl. SIMATIC C7), ET 200S y ET 200X. STEP 7 Lite ha sido concebido para usuarios que trabajen con proyectos no conectados a red así como para principiantes de SIMATIC.

Si quiere utilizar sistemas SIMATIC S7-400, periferia descentralizada, tarjetas de comunicación CP, módulos de función FM o sistemas que integren más de una CPU, deberá utilizar el software estándar STEP 7 o bien STEP 7 Professional.

Información sobre STEP 7 Lite

Si bien STEP 7 Lite es el software para principiantes, también lo es para expertos que programan fundamentalmente para la gama media de potencia. Los programas creados con STEP 7 Lite pueden reutilizarse en STEP 7 mediante funciones de exportación e importación. A diferencia de STEP 7, con STEP 7 Lite hemos apostado por una tecnología más moderna en lo que al interface de usuario se refiere. Funciones de explorador ampliadas, más transparencia en la representación del proyecto y la filosofía de manejo que es habitual en Windows le ayudarán a aprender y utilizar rápidamente el software SIMATIC.

Información sobre el presente Getting Started

Aquí aprenderá las nociones básicas de STEP 7 Lite. Le enseñaremos las pantallas y secuencias de comandos más importantes mediante ejercicios prácticos. Los ejercicios están estructurados de forma que pueda empezar con cualquier capítulo. Las descripciones y los pasos que deberá ejecutar o que por lo menos debería conocer están marcados en **rojo**. Los enlaces con otros apartados aparecen marcados en **azul**.

0.3

Requisitos para trabajar con el Getting Started

Para poder llevar a cabo los ejercicios prácticos de STEP 7 Lite que contiene este Getting Started, necesitará

- una unidad de programación SIMATIC o un PC,
- el paquete de software STEP 7 Lite y el disquete de autorización,
- un sistema de automatización SIMATIC S7-300.

Consulte asimismo la tabla que contiene los números de referencia en el capítulo 1.

Documentación adicional

- El manual electrónico "Programar con STEP 7 Lite" se encuentra en el CD-ROM o también bajo el menú **Inicio > Simatic > Documentación** después de instalar STEP 7 Lite.

Esperamos que nuestro producto sea de su entera satisfacción

SIEMENS AG

Los proyectos de ejemplo del Getting Started

Tras instalar el software STEP 7 Lite encontrará los siguiente programas de ejemplo en el directorio <Unidad de disco>:\Siemens\S7lite\Examples\Español, salvo que haya indicado otro directorio.

El presente manual Getting Started explica los siguientes ejemplos.

- **primeros_pasos_awl.k7p**
- **primeros_pasos_fup.k7p**
- **primeros_pasos_kop.k7p**

Los ejemplos son idénticos; se distinguen únicamente por el lenguaje en el que están programados.

1ª parte: Primeros pasos con STEP 7 Lite – Nociones básicas

Presentación e instalación	1
¿Qué aprenderé?	1.2
Interacción de hardware y software	1.4
Guía de STEP 7 Lite	1.6
Instalar STEP 7 Lite	1.8
Inicio y manejo	2
Abrir un proyecto de ejemplo	2.2
Manejo de proyectos	2.6
Llamar las funciones de ayuda	2.8

2ª parte: Programar una solución de automatización con STEP 7 Lite

Ejecución de la tarea planteada	3
Planteamiento de la tarea - Banco de prueba de motores	3.2
Subdividir el proceso	3.4
Configuración de los módulos	4
¿En qué consiste la configuración?	4.2
Crear un proyecto	4.4
Trabajar en la vista Configuración HW	4.6
Parametrizar los módulos	4.12
Guardar datos de configuración	4.14
Transferir la configuración hardware a la CPU	4.16
Creación de la tabla de símbolos	5
Programación absoluta	5.2
Programación simbólica	5.4

Primeros pasos de programación	6
Elegir KOP, FUP o AWL	6.2
Trabajar con el editor de bloques	6.4
Programar el OB 1 en KOP	6.6
Programar el OB 1 en AWL	6.12
Programar el OB 1 en FUP	6.18
Referencias cruzadas	6.24
 Uso de bloques de función	 7
Crear y abrir un bloque de función (FB)	7.2
Programar FB en KOP	7.6
Programar FB en AWL	7.8
Programar FB en FUP	7.10
Crear bloques de datos de instancia y modificar valores actuales	7.12
Programar llamadas de bloque en KOP	7.14
Programar llamadas de bloque en AWL	7.16
Programar llamada de bloque en FUP	7.18
 Uso de funciones	 8
Crear y abrir una función (FC)	8.2
Programar una función	8.6
Llamar una función en el OB 1	8.8
 Uso de bloques de datos globales	 9
Crear y abrir un bloque de datos globales (DB)	9.2
Programar variables en el DB	9.4

3ª parte: Transferencia, test y diagnóstico

Transferir el programa a la CPU 10

 Establecer una conexión online y conectarse online. 10.2

 Borrado total de la CPU y transferencia del programa 10.6

Comprobar el programa. 11

 Comprobar el estado del programa (Observar) 11.2

 Observar y forzar variables 11.6

Diagnóstico de errores 12

 Diagnóstico HW a simple vista 12.2

 Estado de los módulos e historial de errores 12.5

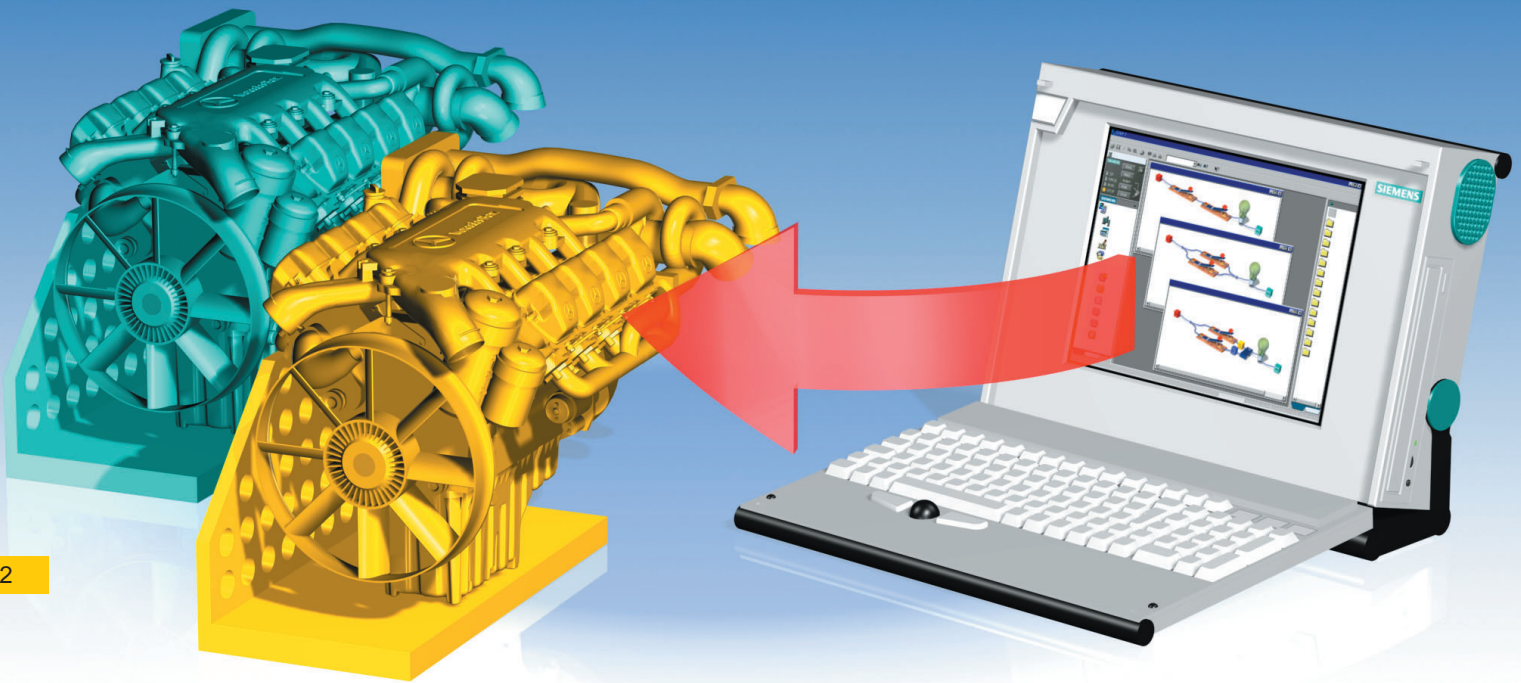
Índice alfabético. 13

1

Presentación e instalación



¿Qué aprenderé?

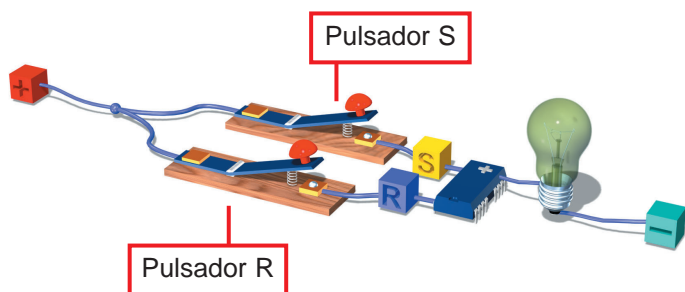
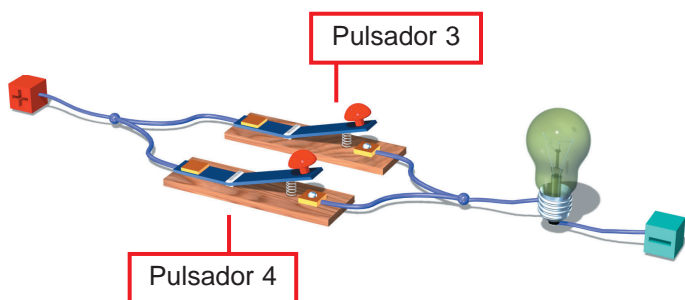
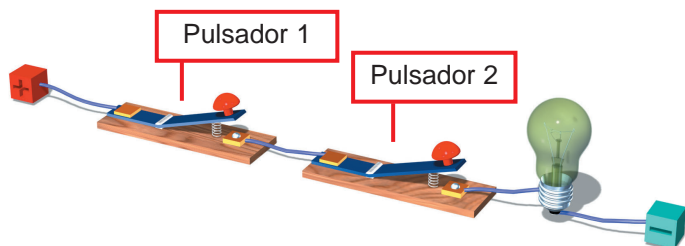


1.2

Con este manual aprenderá mediante algunos ejercicios prácticos, lo sencillo que resulta programar en los lenguajes de programación KOP (esquema de contactos), FUP (diagrama de funciones) y AWL (lista de instrucciones) con STEP 7 Lite.

Para empezar, crearemos un proyecto con el nombre "Getting Started". En este proyecto, crearemos un programa PLC con las funciones de lógica binaria más sencillas: Y, O y ELEMENTO DE MEMORIA.

A continuación, configuraremos el proyecto en un programa PLC para un banco de prueba de motores.



Know-how básico

Nuestros ejemplos de programación se basan en tres combinaciones lógicas binarias fundamentales:

Conexión en serie

La primera combinación lógica binaria, que programaremos más adelante, es la función Y. La función Y se puede ilustrar con dos pulsadores en un circuito eléctrico.

Si se pulsan los pulsadores 1 y 2, se enciende la bombilla.

Conexión en paralelo

La segunda combinación lógica binaria es la función O. La función O también se puede representar en un circuito eléctrico.

Si se pulsa el pulsador 3 o el 4, se enciende la bombilla.

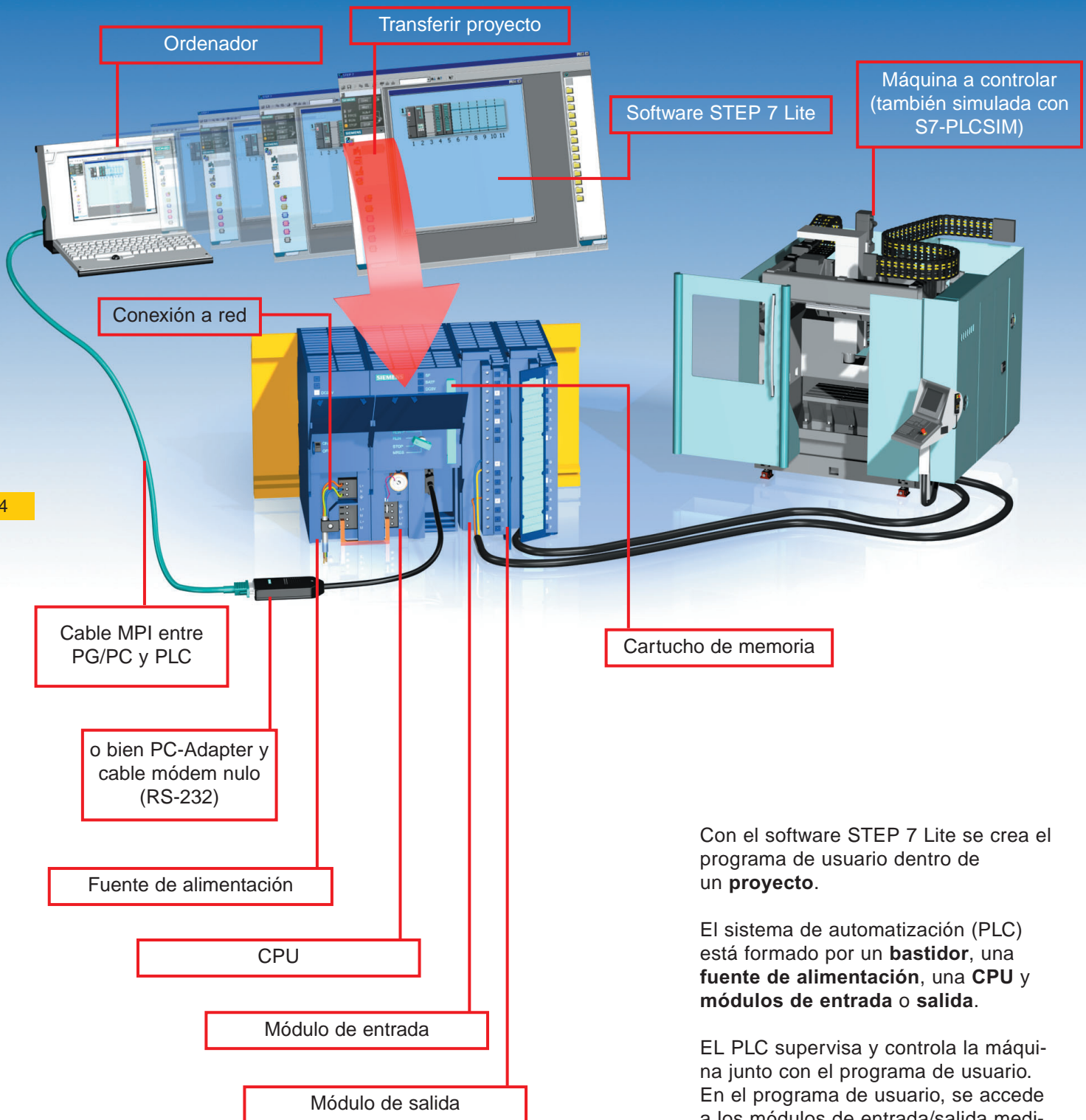
Elemento de memoria (Flipflop de desactivación/activación)

La tercera combinación lógica binaria es el elemento de memoria. Reacciona en un circuito eléctrico ante determinados estados de tensión y los transmite.

Si se pulsa el pulsador S, se enciende la bombilla **hasta que** se pulse el pulsador R.

Estas tres combinaciones lógicas se van a configurar en un ejemplo concreto: un banco de prueba de motores. Durante este proceso, se familiarizará con los siguientes elementos de programa de STEP 7 Lite: bloques de organización, bloques de función, bloques de datos de instancia, funciones y bloques de datos globales.

Interacción de hardware y software



Con el software STEP 7 Lite se crea el programa de usuario dentro de un **proyecto**.

El sistema de automatización (PLC) está formado por un **bastidor**, una **fuentes de alimentación**, una **CPU** y **módulos de entrada o salida**.

EL PLC supervisa y controla la máquina junto con el programa de usuario. En el programa de usuario, se accede a los módulos de entrada/salida mediante las direcciones de E/S.

Equipo PLC

Referencias:

Fuente de alimentación (PS 307 2A)	6ES7307-1BA00-0AA0
Módulo central (CPU 315)	6ES7315-1AF03-0AB0
Entrada digital (SM32DI 16xDC24V)	6ES7321-1BH02-0AA0
Salida digital (SM322 DO 16xDC24V/0,5A)	6ES7322-1BH01-0AA0
Pila de respaldo (Li) 3,4V	6ES7971-1AA00-0AA0
Perfil soporte 480 mm	6ES7390-1AE80-0AA0

Ordenador

Unidad de programación SIMATIC Power PG, Field PG o PC convencional con CP 5611	www.ad.siemens.de/ simatic-pg
Sistema operativo Windows 2000 Windows XP Home o Professional Edition Internet-Explorer 6.0 o superior	O BIEN

Software

Software STEP 7 Lite (Floating License)	6ES7810-3CC07-0YA5
--	--------------------

Documentación

Introducción a STEP 7 Lite V3.0

Paquete opcional

Software de simulación S7-PLCSIM (Floating License)	6ES7841-0CC04-0YA5
Software de simulación S7-PLCSIM (Upgrade)	6ES7841-0CC04-0YE5

Componentes de la lista de verificación

Si desea crear el proyecto de ejemplo descrito, necesitará los siguientes componentes.

1

Con STEP 7 Lite se pueden programar componentes de las series S7-300, ET 200S y ET 200X. Los módulos utilizados en el proyecto de ejemplo aparecen entre paréntesis. No obstante, también se pueden utilizar otros módulos de las series mencionadas.

2

Recomendamos el uso de nuestras PG SIMATIC, que están a la altura de los exigentes criterios del entorno industrial. Para los PC convencionales necesitará un cable de interface adicional. En las PG SIMATIC, este interface ya está integrado.

3

En el archivo STEP7Lite\Disk1\Léame.WRI del CD encontrará indicaciones para la instalación.

4

Además del manual "Primeros pasos con STEP 7 Lite", en el CD del software se incluye también el manual electrónico "Programar con STEP 7 Lite" y la ayuda en pantalla.

5

El S7-PLCSIM simula un sistema de automatización conectado. Este software de simulación le resultará muy útil si desea comprobar la viabilidad de un programa y no dispone de ningún hardware in situ.

Guía de STEP 7 Lite

Configurar

Diseñar la tarea de automatización
Capítulo 3



Crear el proyecto

Capítulo 4



Configurar el hardware

Capítulo 4



Crear el programa

Capítulos 5 – 9



Transferir el programa a la CPU

Capítulo 10



Probar el programa

Capítulo 11

El proyecto es el elemento central de STEP 7 Lite. Dentro del proyecto se resuelven todas las tareas de automatización: desde la configuración del hardware, hasta la comprobación de la viabilidad del programa.



Para los programas de gran tamaño, con muchas entradas y salidas, es recomendable configurar primero el hardware. Esto tiene la ventaja de que STEP 7 Lite muestra las direcciones disponibles en la configuración de hardware.

Si primero crea el programa, luego tendrá que averiguar por sí mismo las direcciones correspondientes en función de los componentes seleccionados y no podrá llamarlas a través de STEP 7 Lite.

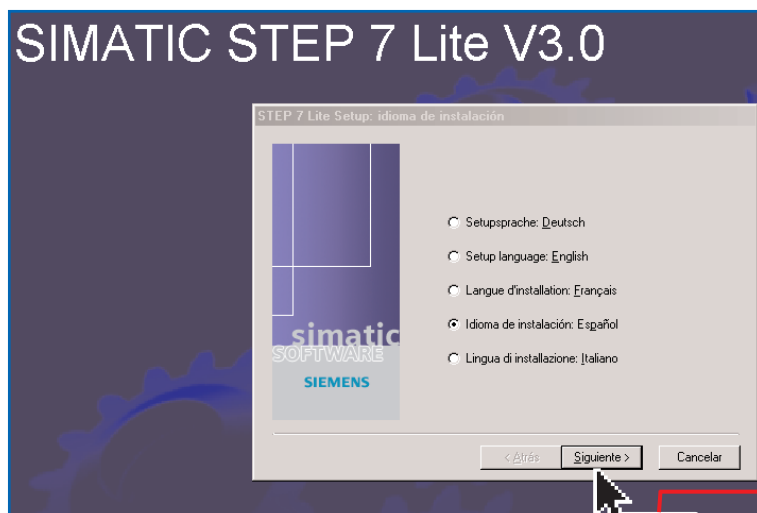
La configuración de hardware no sólo permite determinar direcciones, sino también modificar los parámetros y las propiedades de los módulos.

Puesto que en "Getting Started" sólo se necesitan unas pocas entradas y salidas, puede obviar la configuración de hardware y pasar directamente a la programación si lo considera más importante.

Instalar STEP 7 Lite



1.8



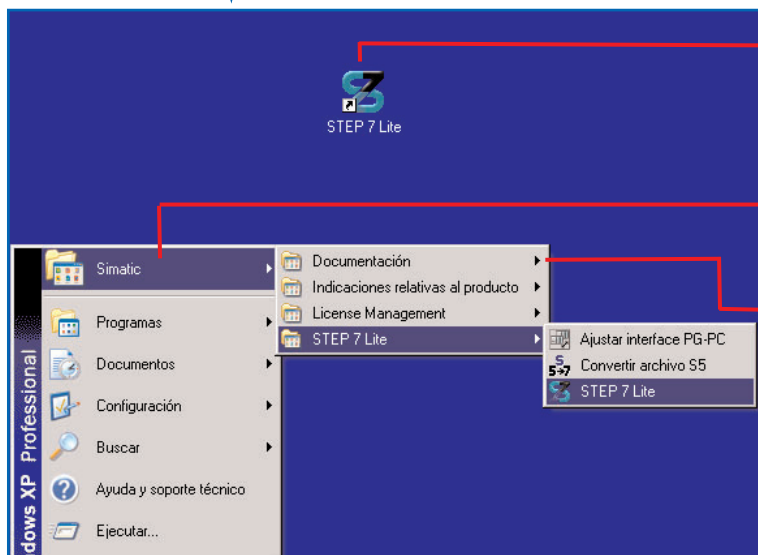
Para la instalación necesitará

- el CD de STEP 7 Lite que contiene además las instrucciones de instalación en el archivo **STEP7Lite\Disk1\Leame.WRI**,
- y
- la clave de la licencia o License Key para poder utilizar el software.

1 Introduzca el CD de STEP 7 Lite. El programa de instalación arrancará automáticamente o con el comando **<Unidad>:\setup.exe**.

Siga las instrucciones que van apareciendo en la pantalla.

El escritorio tras instalar



2

Introduzca el soporte de datos que contiene la clave de la licencia cuando el programa lo solicite.

Siga las instrucciones de instalación de la clave de la licencia.

Retire el soporte de datos antes de reiniciar el equipo.

3

Una vez instalado, STEP 7 Lite aparecerá en el escritorio y en el menú Inicio.

4

Todo paquete de software SIMATIC que instale, se encontrará en la carpeta SIMATIC.

La documentación de STEP 7 Lite que se puede imprimir se encuentra bajo **Simatic > Documentación**.

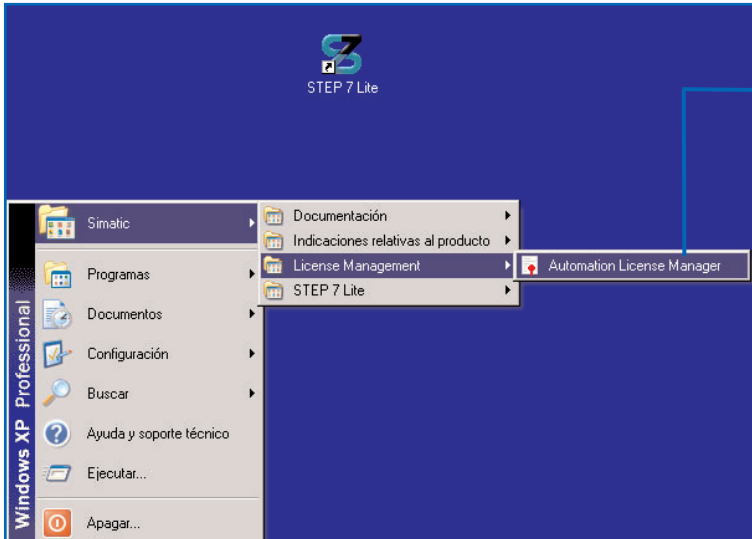
Transferir la clave de la licencia



Si no se ha instalado ninguna clave de licencia válida para STEP 7 Lite, se utiliza una clave de tipo Trial License suministrada e instalada de forma estandarizada con STEP 7 Lite. Con dicha clave de licencia se podrá utilizar STEP 7 Lite únicamente durante los siguientes 14 días. Al iniciar por primera vez STEP 7 sin una clave de licencia válida, se activa la Trial License.

Si desea transferir la clave de la licencia de un ordenador a otro, proceda de la manera siguiente:

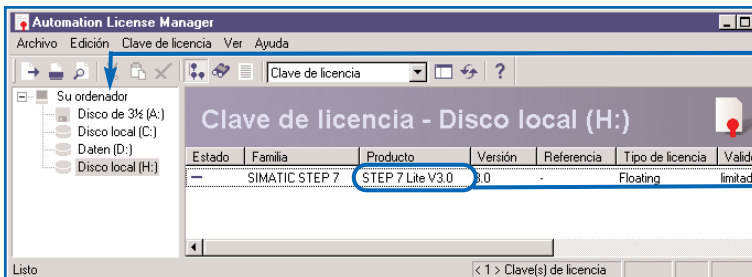
5 Inicie el **Automation License Manager**.



Interface de usuario del
Automation License Manager

6 Abra la unidad en la que se encuentra la clave de la licencia que desea transferir.

7 Seleccione la clave y elija el comando de menú **Clave de licencia > Transferir**.



Elija en el cuadro de diálogo que aparece a continuación en el equipo destino la unidad a la que desea transferir la clave..

2

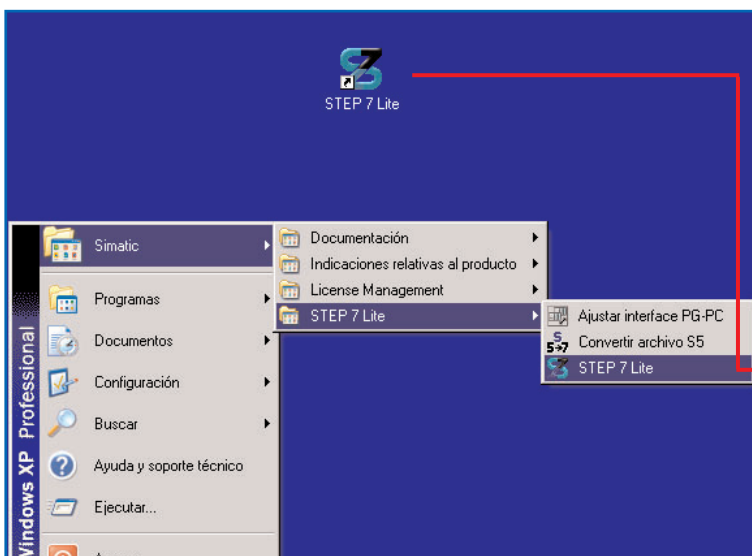
Inicio y manejo



Abrir un proyecto de ejemplo



2.2

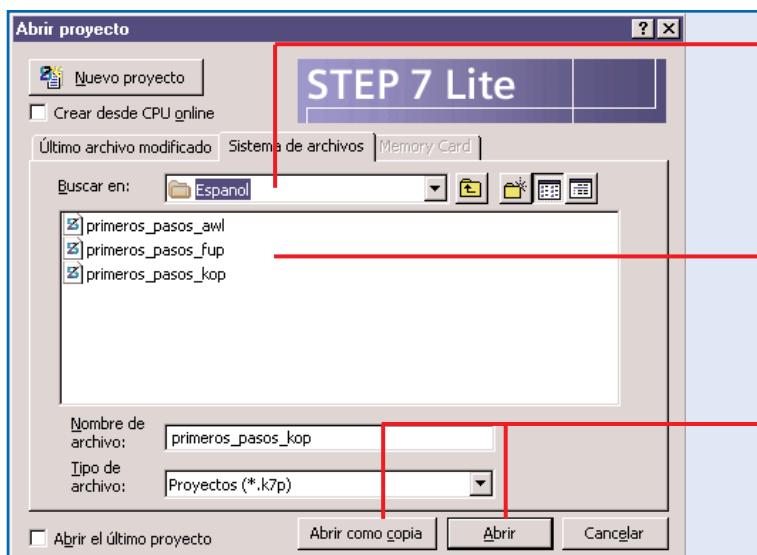


Abrir un proyecto de ejemplo KOP

STEP 7 Lite ya está instalado en el equipo.

En este capítulo aprenderá los aspectos más importantes del interface de usuario.

1 Inicie STEP 7 Lite a través del menú Inicio o haciendo clic en el icono que aparece en el escritorio.



2 Abra el directorio que contiene los ejemplos de programación

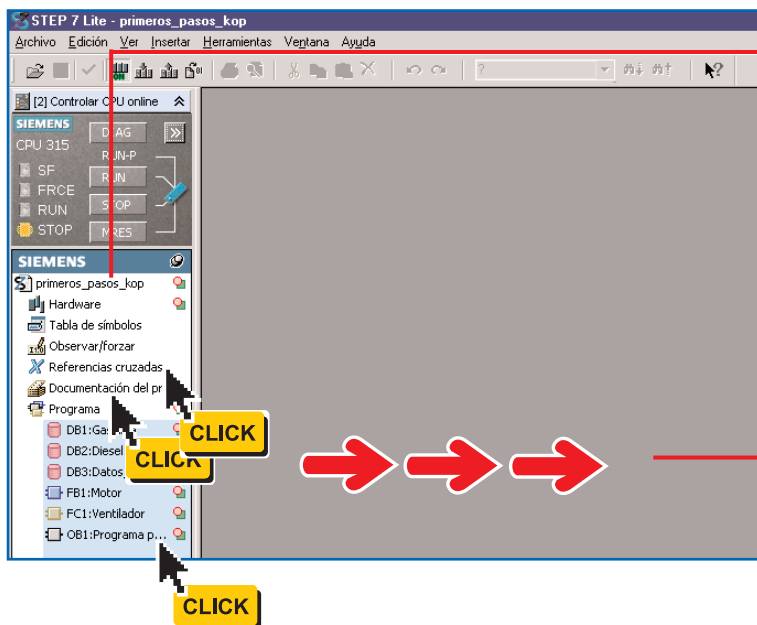
<Unidad>:\Siemens\S7Lite
\Examples\Español ...

3 Seleccione en él el ejemplo del lenguaje KOP.

primeros_pasos_kop.k7p

4 Abra el proyecto.
Si no quiere correr el riesgo de modificar el proyecto por equivocación, haga una copia del mismo y ábrala.

Se abre el proyecto seleccionado



5 En la ventana del proyecto aparecerá el proyecto "primeros_pasos_kop".

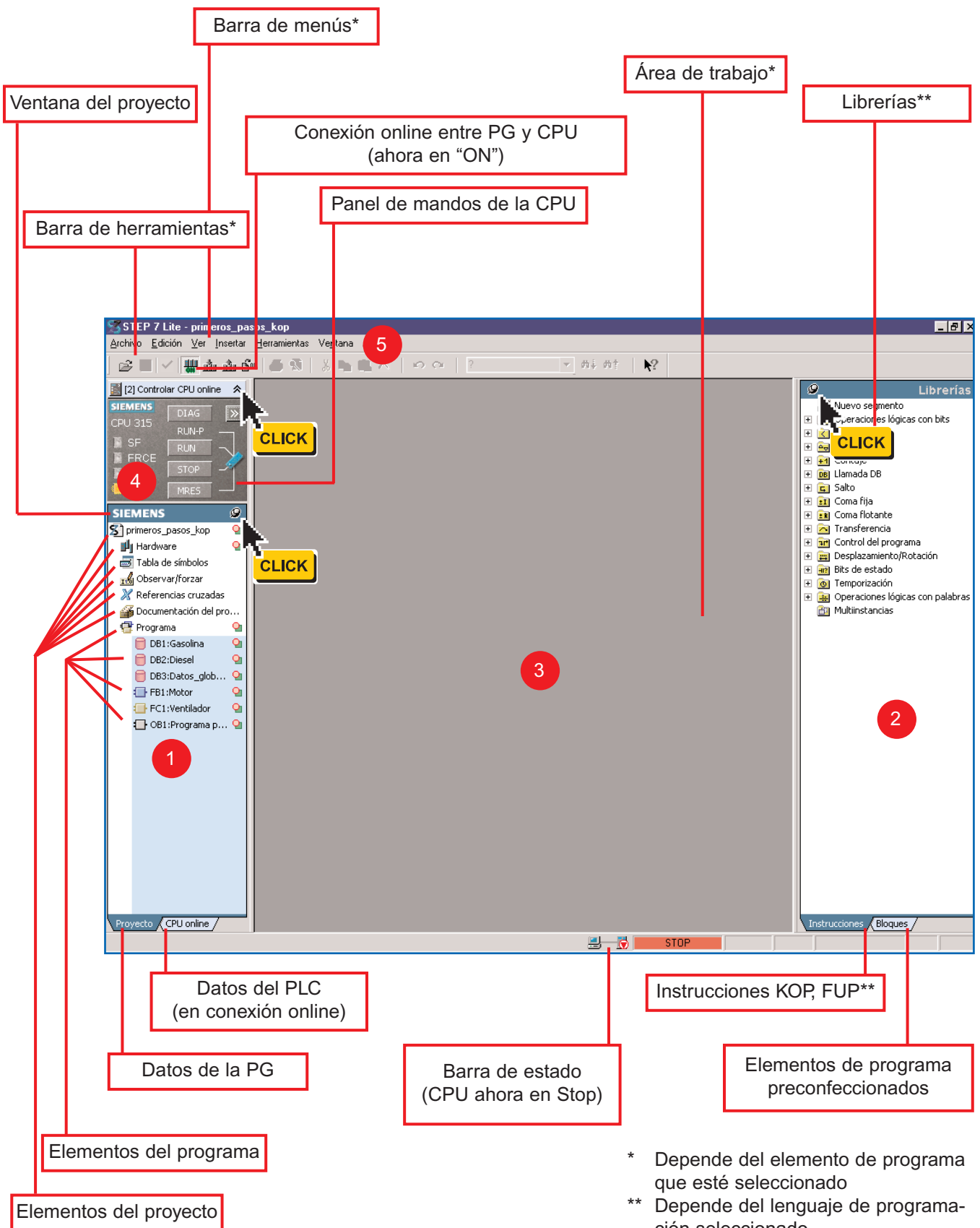
La ventana de proyecto que aparece en la parte izquierda constituye el medio principal para moverse por STEP 7 Lite.

Desde aquí puede abrir todas las vistas de STEP7 Lite a través de los elementos de proyecto "Hardware", "Tabla de símbolos", etc.

6 Haga doble clic en uno de los elementos. En la zona de trabajo se abrirán las distintas vistas, a las que se adaptará la barra de menús que aparece en la parte superior.



Si hace doble clic sobre los elementos de proyecto, se irán abriendo una a una todas las vistas de STEP 7 Lite. No olvide cerrar de vez en cuando las ventanas que no necesite para tener siempre una visión de conjunto.



* Depende del elemento de programa que esté seleccionado

** Depende del lenguaje de programación seleccionado

El interface de usuario

El interface de usuario se compone de cinco áreas:

- 1 **Ventana de proyecto**
Cada vez que se crea un proyecto, también se crean todos los elementos de proyecto que pueden resultar necesarios.
- 2 **Librerías**
En “Bloques” encontrará los bloques suministrados. En “Instrucciones” encontrará las instrucciones KOP y FUP que se pueden utilizar en los bloques.
- 3 **Zona de trabajo**
En esta zona se abren las vistas en las que puede procesar los elementos del proyecto.
- 4 **Panel de mandos de la CPU**
Representa la parte frontal de la CPU con sus indicadores y sus elementos de mando. En ella también puede cambiar el estado operativo.
- 5 **Barra de menús**
Contiene todos los menús disponibles en STEP 7 Lite, como el comando de menú **Ver > KOP** para cambiar el lenguaje de programación cuando hay abierto un bloque.



Si hace clic en el indicador de expansión podrá visualizar u ocultar el panel de mandos de la CPU. Si hace clic en la chincheta, la vista del proyecto y de las librerías quedará fija y viceversa. Si la vista no está fija, al desplazar el puntero del ratón en el margen de la zona de trabajo podrá ampliarla o reducirla.

Manejo de proyectos

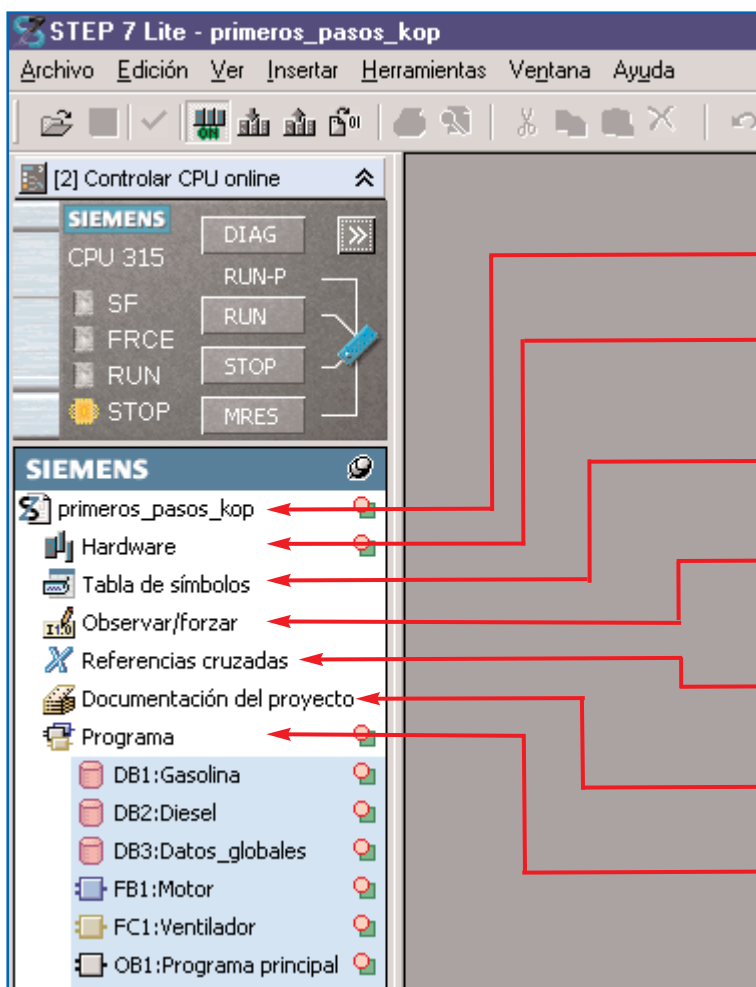


2.6

¿Qué es un proyecto?

Los datos de un proyecto STEP 7 Lite abarcan todos los datos de una SIMATIC S7-300, C7 o de un sistema de periferia descentralizada modular ET 200X o ET 200S (stand alone).

Los proyectos se utilizan para guardar ordenadamente los datos que se generan al crear una solución de automatización.



Gestión del proyecto

Cada elemento de un proyecto está sujeto a un proceso distinto:

- 1 Crear y guardar el proyecto
- 2 Configurar el hardware, parametrizar módulos y diagnosticar fallos de hardware
- 3 Definir símbolos para la programación simbólica
- 4 Observar y forzar los operandos en la CPU para comprobar el funcionamiento del programa
- 5 Evaluar los operandos utilizados y la estructura del programa
- 6 Confeccionar la documentación del proyecto de forma personalizada
- 7 Crear un programa de usuario compuesto por bloques para el PLC.

2.7

Gestión de archivos

Los proyectos se guardan con su nombre y el formato de datos ... **.k7p**.

En STEP 7 Lite sólo es posible abrir archivos con la extensión **.k7p**.



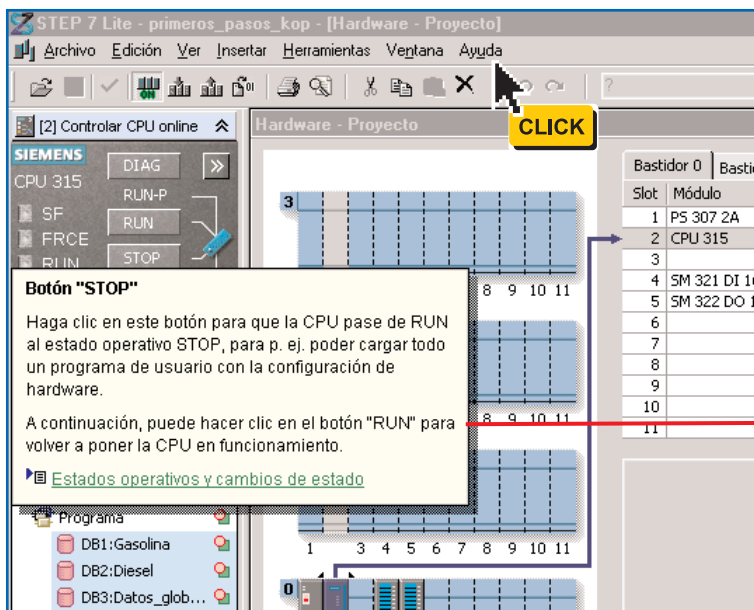
Peligro de accidente – Cuando hay una conexión online entre la unidad de programación y la CPU se pueden provocar, por ejemplo, movimientos indeseados en la instalación al tocar el panel de mandos de la CPU.

Por consiguiente, asegúrese de que nadie corre peligro antes de seleccionar "RUN" .

Llamar las funciones de ayuda



2.8

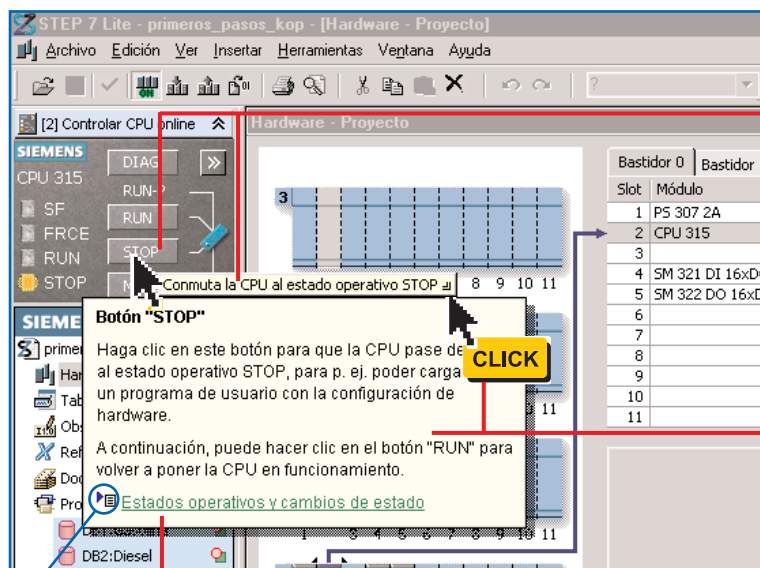


Ayuda de Windows

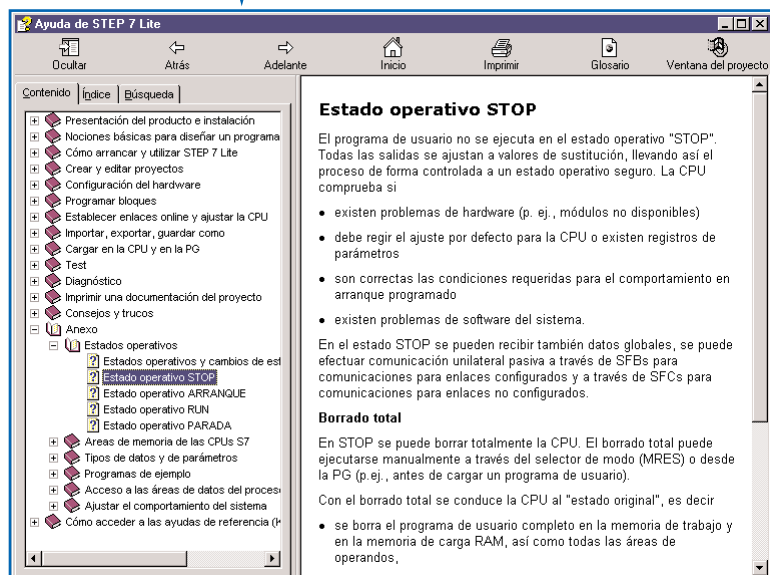
Si ya ha trabajado con programas de Microsoft, el funcionamiento de la ayuda de STEP 7 Lite le resultará familiar.

- 1 Pulse **F1**:
La ayuda básica se abrirá automáticamente.
- 2 Pulse **Mayús + F1**:
A continuación, haga clic con el puntero del ratón en forma de interrogación en cualquier botón. Aparecerá la ayuda directa de ese botón.

También puede acceder a estos dos tipos de ayuda con **Ayuda** en la barra de menús.



Se abre la ayuda básica.



Tres tipos de ayuda

Ayuda breve

Sitúe el cursor sin hacer clic, por ejemplo, sobre el botón **STOP**.

Algunos segundos después aparecerá una ayuda breve - con texto y links - siempre que haya conexión online.

Ayuda contextual

Haga clic sobre las pequeñas flechas que aparecen junto al texto.

Ayuda básica

Haga clic en el link. Aparecerá una ventana con ayuda básica detallada sobre el tema seleccionado.

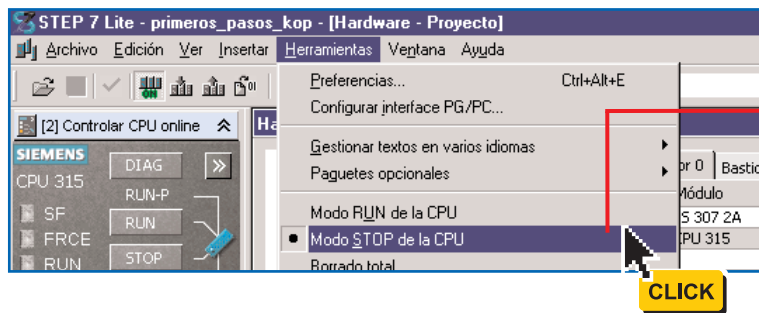
Nota:

Los distintos símbolos de los links indican qué tipo de ayuda se abrirá en la ayuda básica.

Hoja = Información contextual

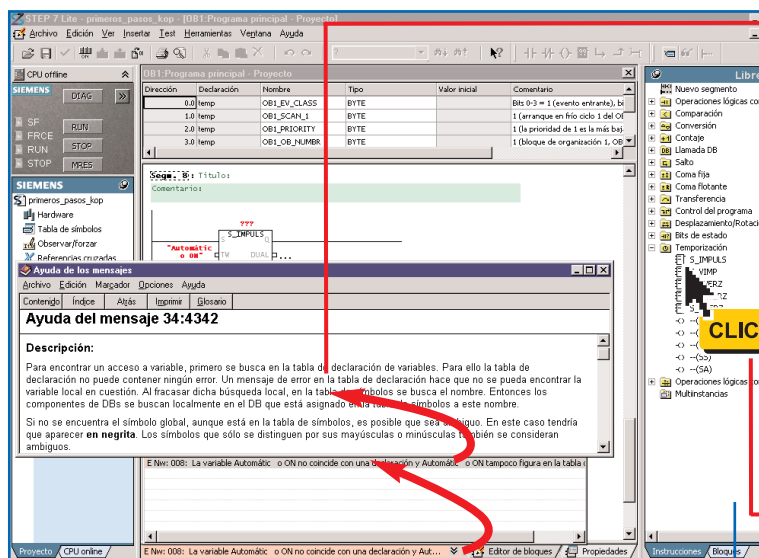
Lista = Instrucciones

Inicio y manejo



Más ayuda

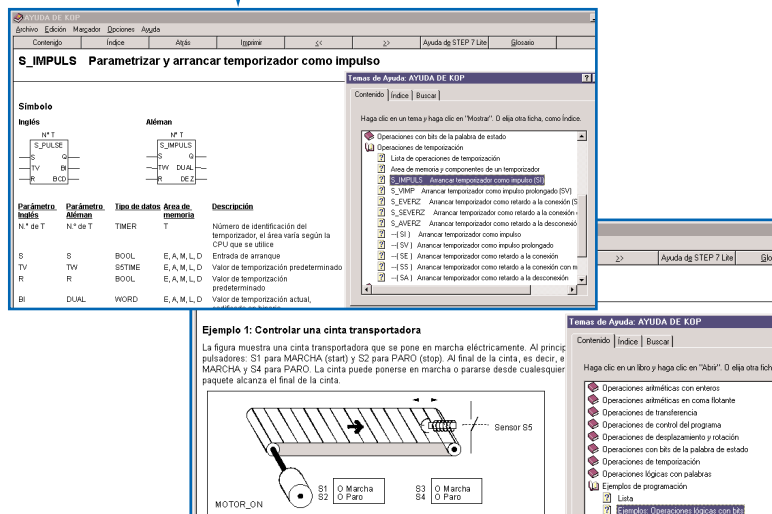
Ayuda de los comandos de menú
Pulse **MAYÚS + F1**. Haga clic en un comando de un menú desplegable. Aparecerá la ayuda del comando.



Ayuda de los mensajes de error
También puede abrir la ayuda de los mensajes de error que aparecen en la barra de estado. Seleccione el mensaje de error y abra la **Ayuda** haciendo clic con el botón derecho del ratón.

Ayuda de referencia
Pulse **MAYÚS + F1** y haga clic, por ejemplo, en **S_IMPULS**. La ayuda de referencia se abrirá en otra ventana.

Se abre la ayuda de referencia



En la ayuda de referencia obtendrá ayuda sobre la instrucción o bloque seleccionado.

Desde aquí también es posible, por ejemplo, abrir ejemplos de programación.

Si ya se ha familiarizado con el manejo de STEP 7 Lite, cierre la copia del proyecto de ejemplo.

3

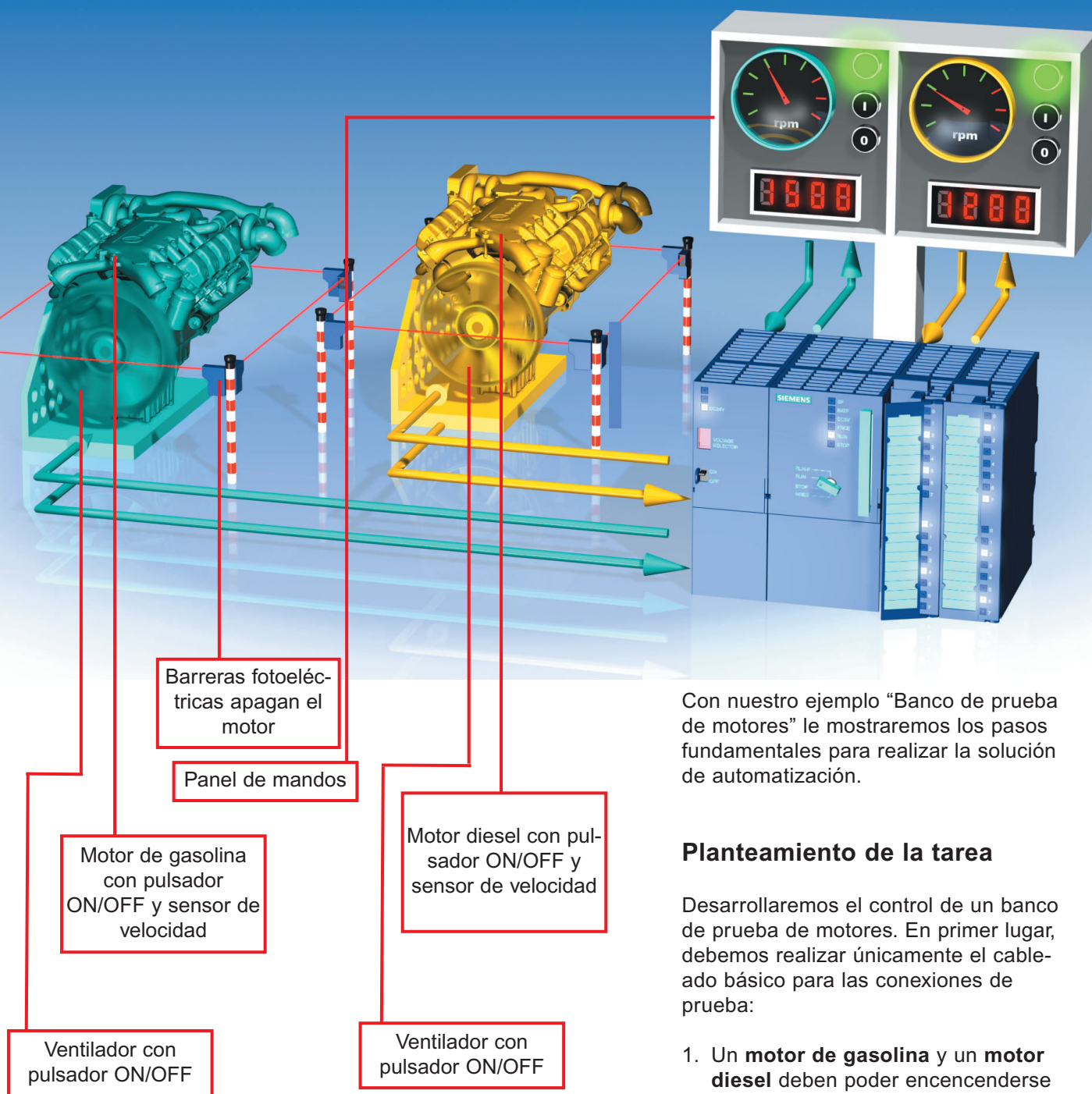
Ejecución de la tarea planteada



Planteamiento de la tarea

Banco de prueba de motores

3.2



Con nuestro ejemplo “Banco de prueba de motores” le mostraremos los pasos fundamentales para realizar la solución de automatización.

Planteamiento de la tarea

Desarrollaremos el control de un banco de prueba de motores. En primer lugar, debemos realizar únicamente el cableado básico para las conexiones de prueba:

1. Un **motor de gasolina** y un **motor diesel** deben poder encenderse y apagarse por separado en el banco de prueba.

2. Una barrera fotoeléctrica alrededor de cada motor garantiza la seguridad en las zonas de peligro y ejecuta el paro de emergencia independientemente del ejemplo programado.
3. Al mismo tiempo que se enciende y se apaga el motor, se conecta y desconecta un **ventilador** eléctrico.
4. Después de pararse el motor, el ventilador debe seguir funcionando durante 4 segundos.
5. El usuario debe recibir una señal en el momento en que los motores alcancen las revoluciones por minuto que se indican a continuación:

Motor de gasolina = 1.500 U/min
Motor diesel = 1.200 U/min

Solución

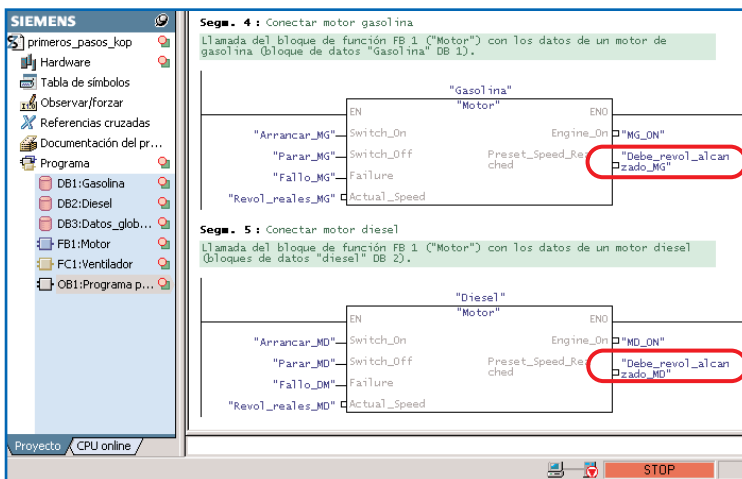
La solución: en los programas de ejemplo encontrará en **OB 1** la señal “Debe_revol_alcanzado” en:

- **Segmento 4** para el motor de gasolina y en
- **Segmento 5** para el motor diesel.

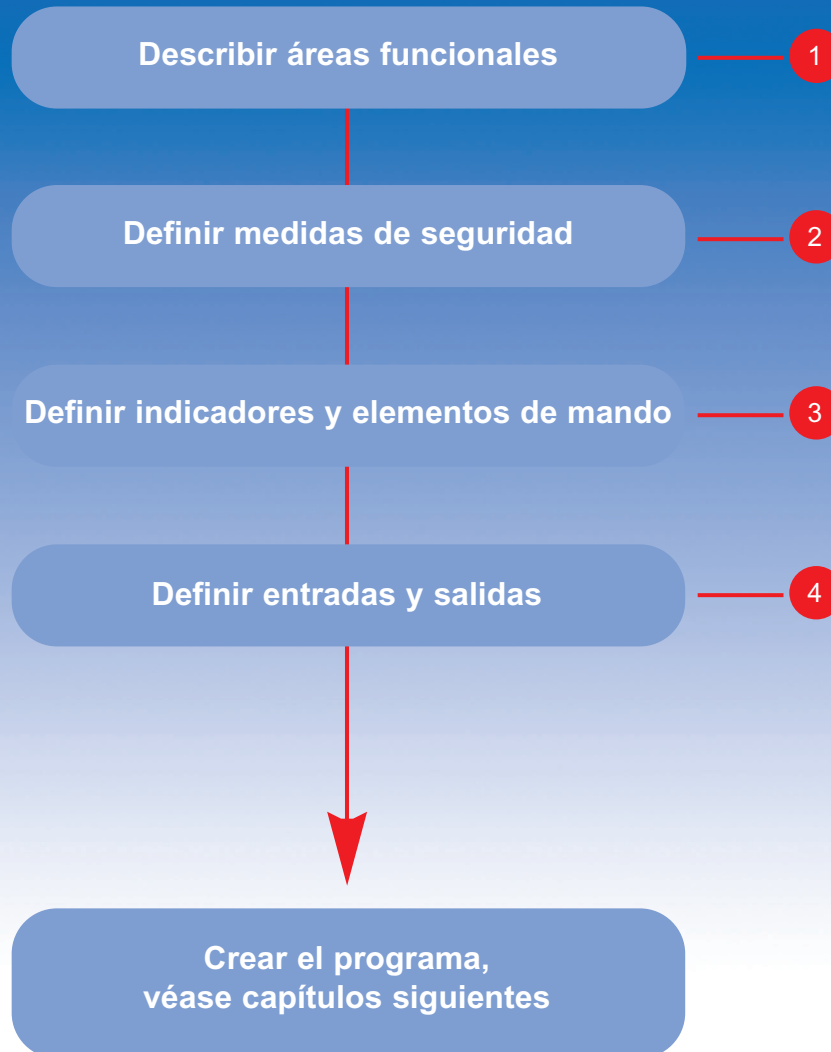
Encontrará más información a este respecto en el capítulo 7, apartado “Programar llamadas de bloque”.

Con la señal “Debe_revol_alcanzado” podría ahora iniciarse un proceso de prueba, p. ej.:

- Iniciar una comparación de las emisiones de gas, o
 - Iniciar una medición de la estabilidad de la velocidad.
- Sin embargo, esto no formaría parte de nuestro programa de ejemplo.



Subdividir el proceso



Antes de comenzar a programar, subdivide el proceso.

El proceso básico que se debe seguir para cualquier proyecto aparece representado arriba.

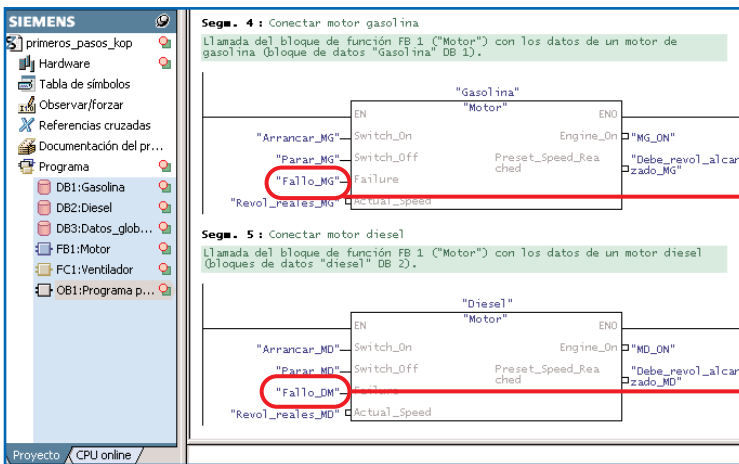
No obstante, cada uno de los pasos puede a su vez dividirse. Cuanto mayor sea la división del proceso, más fácil será la estructuración del programa de usuario.

Área funcional	Equipos correspondientes
Área funcional A	= Motor gasolina = Sensor de velocidad = Ventilador
Área funcional B	= Motor diesel = Sensor velocidad = Ventilador

1

Descripción de las áreas funcionales:

- Divida el proceso en grupos funcionales.
- Defina los elementos que deban controlar este área.
- Defina las entradas y salidas eléctricas, mecánicas y lógicas de cada tarea.
- Defina los bloqueos y la interdependencia entre las tareas.



2

Definición de los criterios de seguridad:

En nuestro ejemplo se trata de un paro de emergencia cuya programación siempre será más compleja en las aplicaciones reales.

3.5

Definir los indicadores y elementos de manejo:

3

Cada proceso requiere un sistema de manejo y visualización que permita al personal intervenir en el proceso.

Estado	Símbolo	Dirección	Tipo de dato	Comentario
	Arrancar_MD	E 1.4	BOOL	Arrancar el motor diesel
	Arrancar_MG	E 1.0	BOOL	Arrancar motor de gasolina
	Automatico ON	E 0.5	BOOL	para la función de memoriza
	Conectar_ventilador_MD	A 5.6	BOOL	Comando para conectar ven
	Conectar_ventilador_MG	A 5.2	BOOL	Comando para conectar ven
	Datos_globales	DB 3	DB 3	Bloque de datos globales
	Debe_revol_alcanzado_MD	A 5.5	BOOL	Indicador motor diesel alcan
	Debe_revol_alcanzado_MG	A 5.1	BOOL	Indicador motor gasolina alc
	Diesel	DB 2	FB 1	Datos del motor diesel
	Fallo_DM	E 1.6	BOOL	Fallo del motor diesel
	Fallo_MG	E 1.2	BOOL	Fallo del motor de gasolina
	Gasolina	DB 1	FB 1	Datos del motor de gasolina
	Lámpara roja	A 4.1	BOOL	Resultado de la consulta O
	Lámpara verde	A 4.0	BOOL	Resultado de la consulta Y
	Manual ON	E 0.6	BOOL	para la función de memoriza
	MD_ON	A 5.4	BOOL	Comando para arrancar el m
	MG_ON	A 5.0	BOOL	Comando para arrancar el m
	Modo automatico	A 4.2	BOOL	Salida con memorización
	Motor	FB 1	FB 1	Control del motor
	Parar_MD	E 1.5	BOOL	Parar motor diesel
	Parar_MG	E 1.1	BOOL	Parar el motor de gasolina

4

Definir las entradas y salidas:

Por pequeño que sea nuestro ejemplo, siguen siendo necesarias tres Entradas y Salidas físicas en el motor de gasolina (MG).

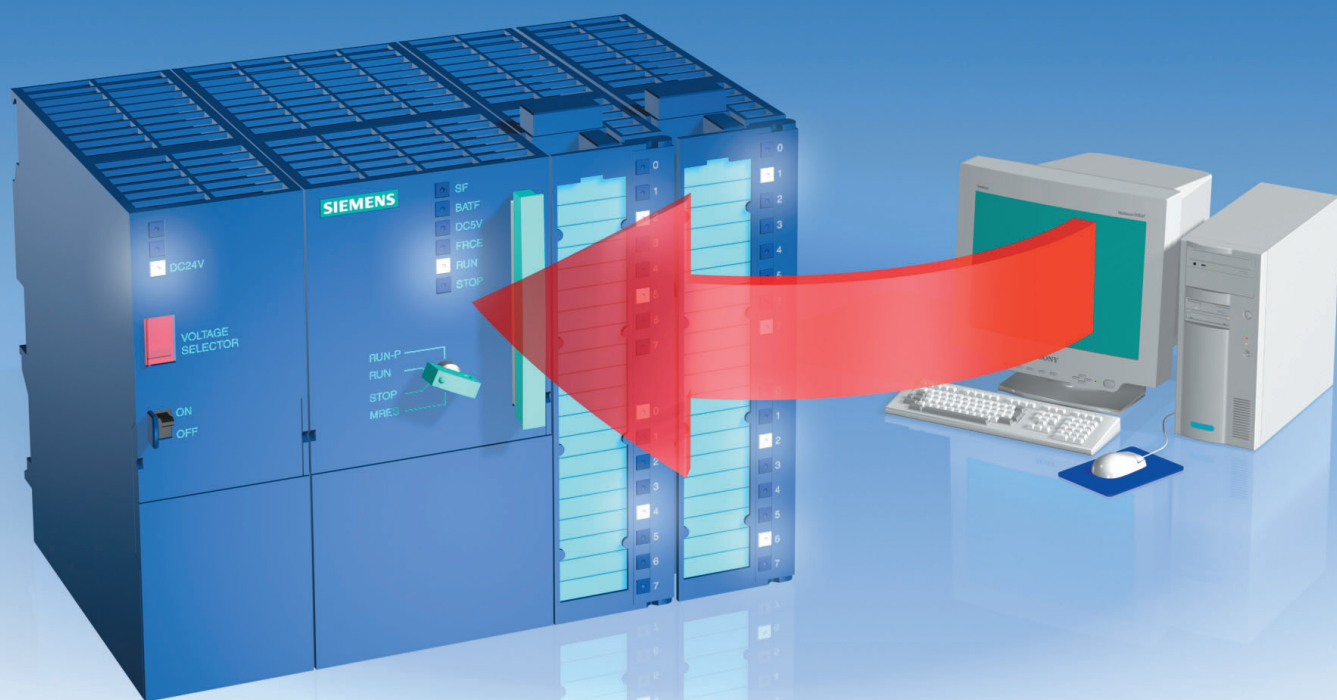
Para obtener una visión de conjunto de todas las entradas y salidas, consulte la tabla de símbolos del capítulo 5.



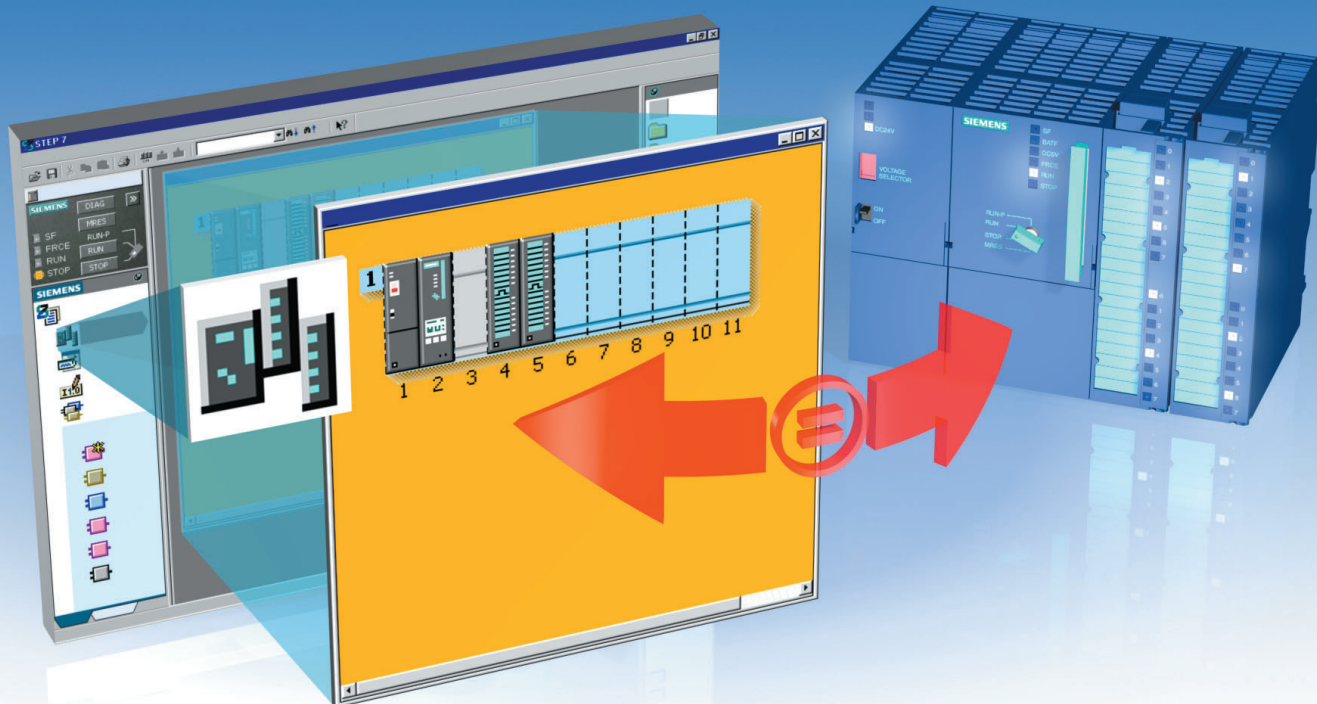
Recomendamos a los principiantes nuestros cursos de formación de SIEMENS, en los que aprenderán a automatizar procesos con SIMATIC mediante ejemplos.

4

Configuración de los módulos



¿En qué consiste la configuración?

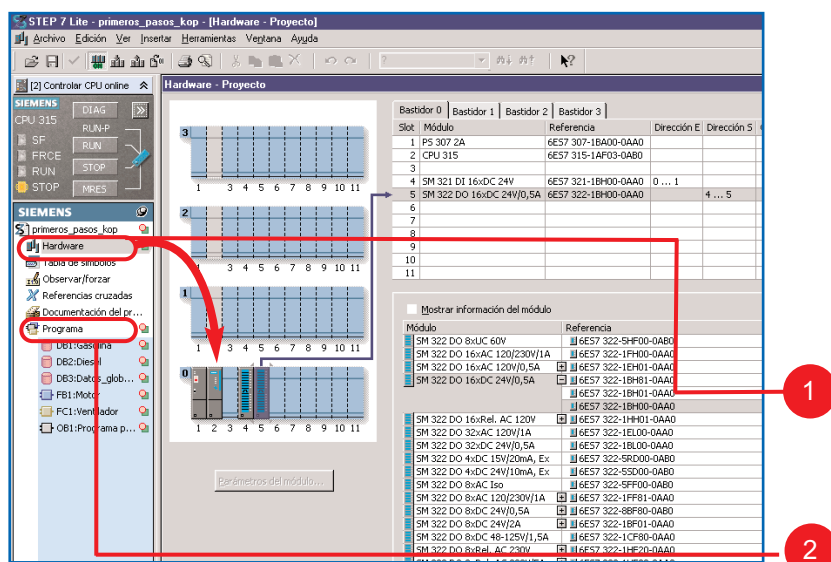


4.2

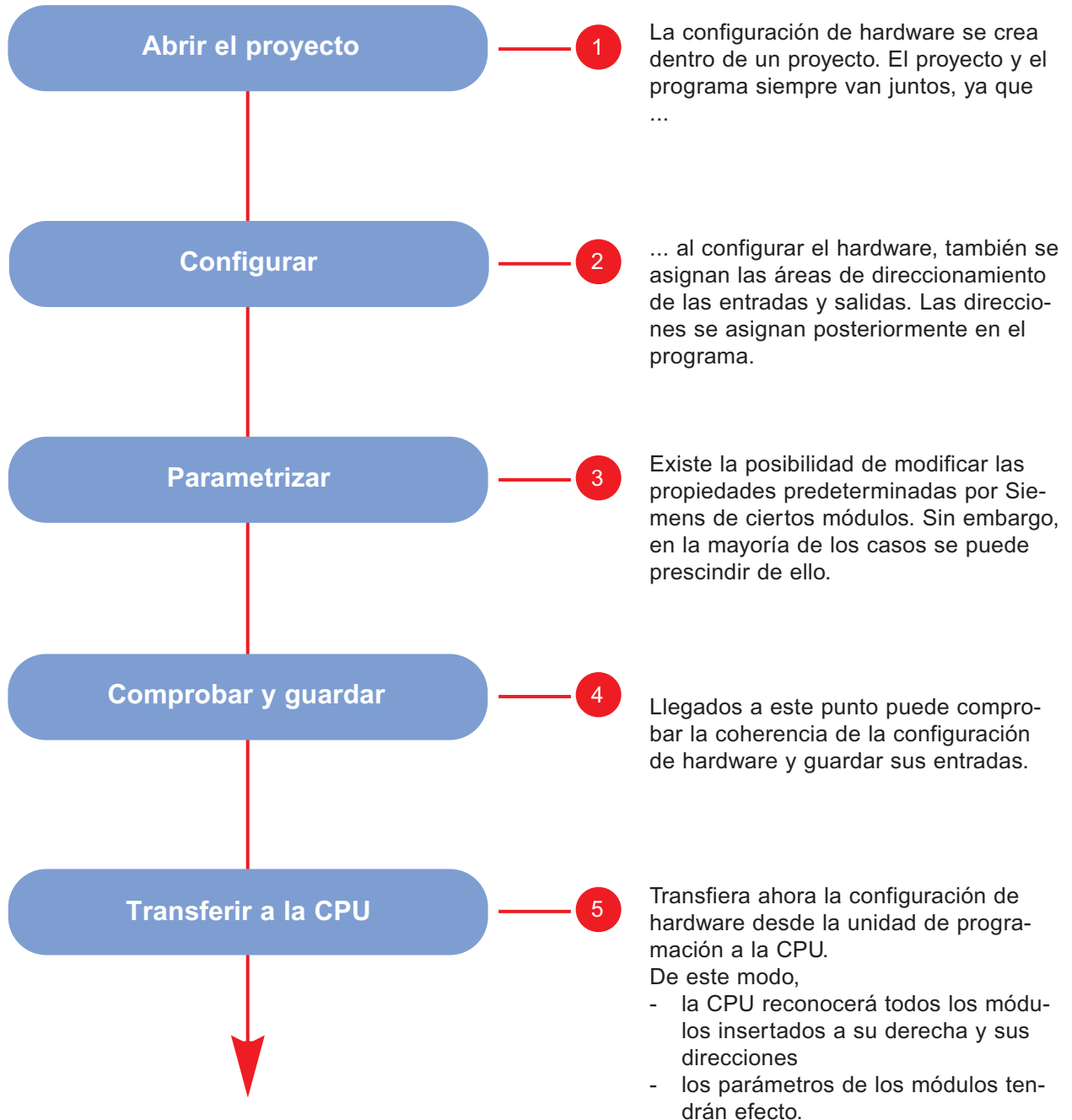
La configuración del hardware consiste en reproducir la estructura real del PLC en la unidad de programación. Esta configuración se puede copiar después en otros proyectos de STEP 7 Lite, modificarla en caso necesario y transferirla a otros equipos. Durante el arranque del sistema de automatización, la CPU compara la configuración teórica creada en STEP7 Lite con la configuración física (real) de la instalación. Así es posible detectar y señalar inmediatamente los posibles fallos.

En la ventana del proyecto, el elemento Hardware muestra gráficamente un bastidor, en el que deberá indicar todos los módulos integrados en el PLC.

Posteriormente editará en el elemento Programa el programa de usuario para esta configuración de hardware.



La configuración hardware a simple vista



Crear un proyecto



4.4

Su proyecto "Getting Started" deberá contener la configuración hardware que corresponda a su sistema de automatización y no la configuración de nuestros ejemplos.

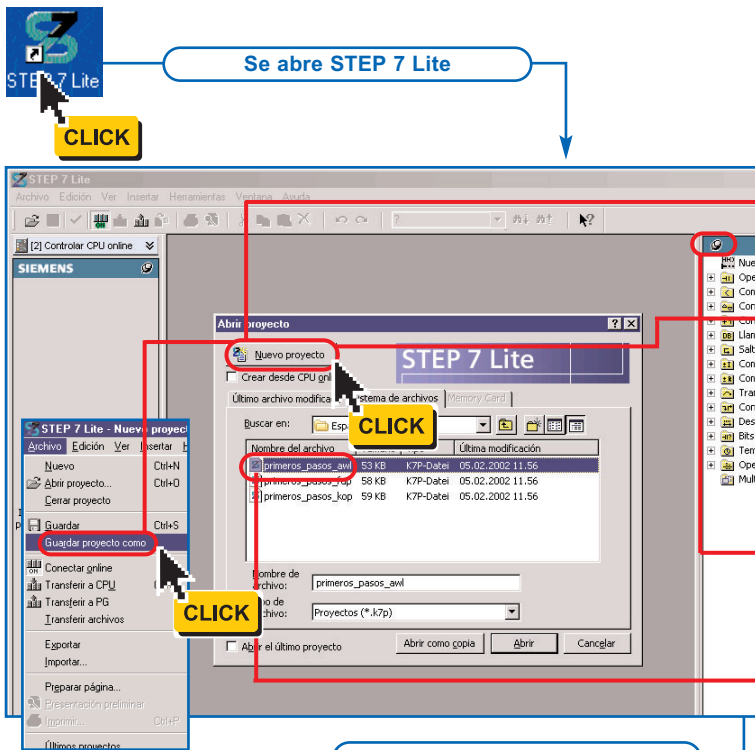
1 En este capítulo creará un proyecto nuevo con el título "Getting Started". Aquí aparecerá en el nivel superior.

En los capítulos sucesivos iremos ampliando este proyecto.



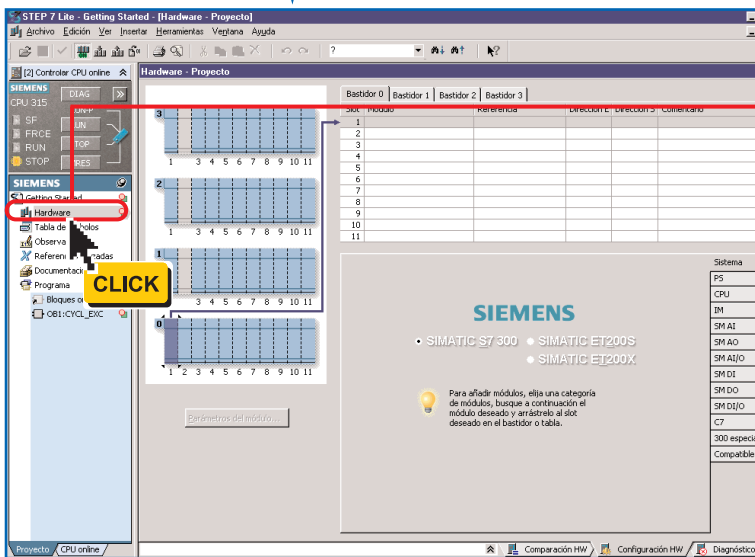
Al configurar el hardware puede guiarse por los proyectos de ejemplo que ya están instalados. Encontrará estos ejemplos en:
Unidad:\Siemens\S7Lite\Examples\Español...

La configuración hardware es idéntica en los tres proyectos suministrados.



Crear un nuevo proyecto

- 1 Haga doble clic para iniciar STEP 7 Lite.
- 2 Abra un proyecto nuevo.
- 3 Guarde el nuevo proyecto con el nombre
“Getting Started”
- 4 Haga clic en la chincheta para ocultar las librerías.
- 5 Si desea utilizar a modo de guía una de nuestras configuraciones, abra una segunda instancia de STEP 7 Lite con uno de los proyectos de ejemplo.



- 6 Haga doble clic en Hardware. En el área de trabajo aparecerá la ventana de configuración de hardware.



Las configuraciones de hardware se pueden copiar de un proyecto a otro. Si, por ejemplo, quiere pasar inmediatamente al capítulo 5, abra uno de los proyectos de ejemplo suministrados y copie el elemento **Hardware** en su proyecto “Getting Started” (consulte el paso 5).

Trabajar en la vista Configuración HW

Representación gráfica de la configuración hardware

Tabla de configuración

Catálogo de hardware

Elemento Hardware

Parametrización de módulos

Selección del sistema SIMATIC

Pictogramas de incoherencias

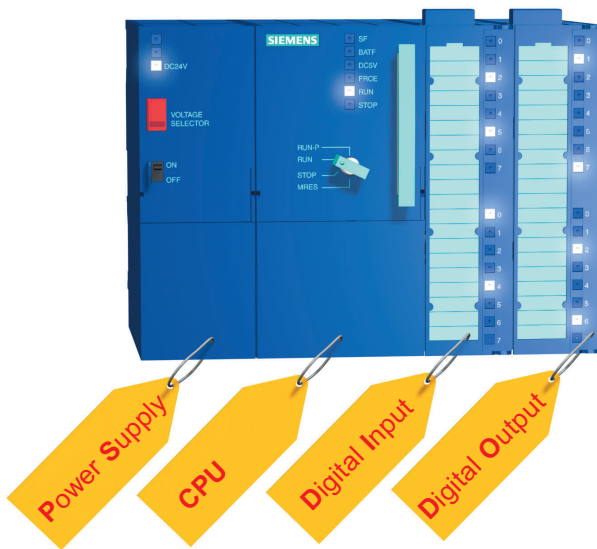
Ver diferencias entre equipo PLC y CPU (sólo si existe una conexión online con la CPU)

Ver qué problemas presentan los módulos (sólo si existe una conexión online con la CPU)

Información general

Si ha hecho doble clic para abrir el elemento Hardware en la ventana de proyecto, en la zona de trabajo de la derecha aparecerá la vista "Hardware".

Seleccione en el catálogo de hardware los módulos que estén disponibles en su equipo PLC.



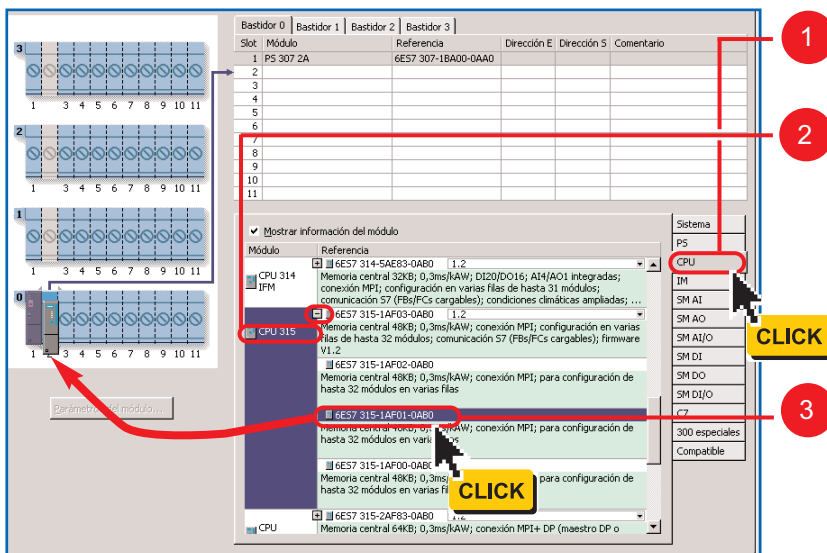
Configurar el hardware

Los programas de ejemplo utilizan los siguientes módulos:

- 1 **Power Supply** = Fuente de alimentación
- 2 **CPU** = Módulo PLC
- 3 **Digital Input** = Módulo de entradas digitales
- 4 **Digital Output** = Módulo de salidas digitales

Los números de pedido (referencias) aparecen impresos en la parte frontal de los módulos.

Configure los módulos tal y como se describe a continuación.



1 Haga clic en **CPU**.

2 Mueva el cursor hasta **CPU 315** y haga clic en el signo "+". Aparecerán todas las versiones anteriores de la CPU 315.

3 Utilice la función Arrastrar y soltar para desplazar la CPU hasta el bastidor. De acuerdo con las reglas de ocupación de slots, las posiciones no admitidas se bloquearán y mostrarán una señal de prohibición.

Se abre otra ventana

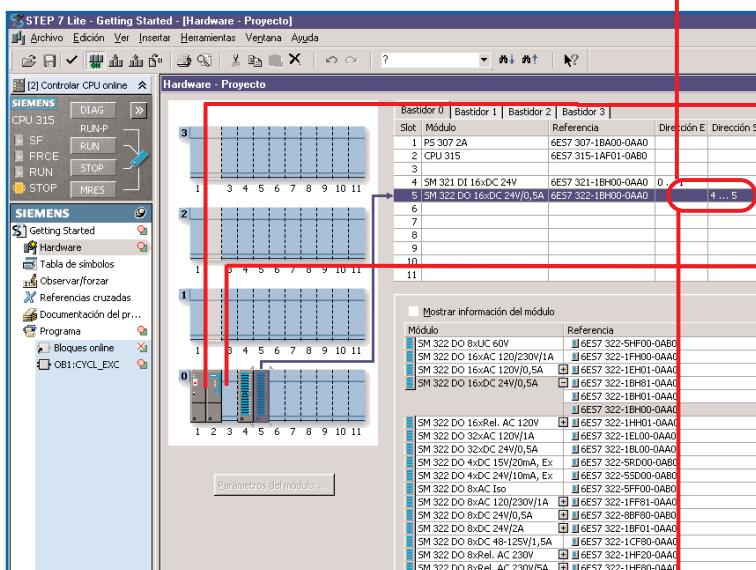
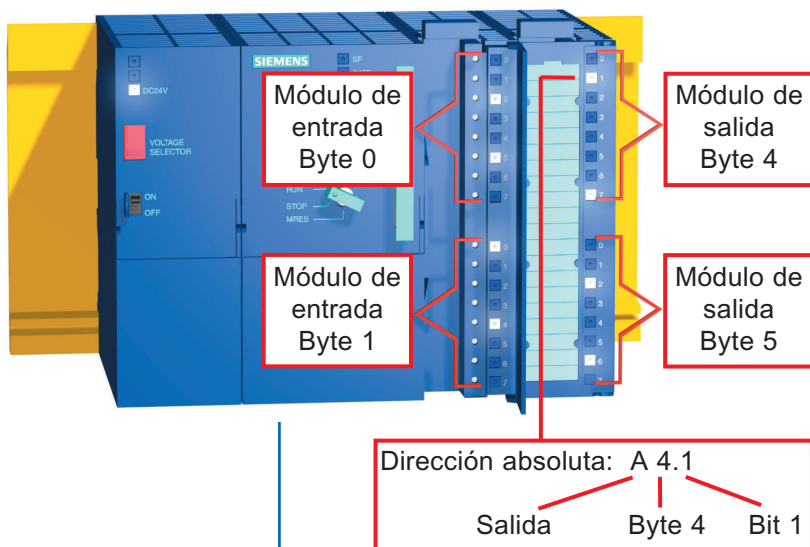


Siga este mismo método con los demás módulos.

Para continuar con el proceso, pase a "Parametrizar los módulos" en la página 4.12. No obstante, si desea conocer más detalles, pase a la página siguiente.

Si más tarde desea transferir la configuración hardware a su CPU, lógicamente deberá configurar el hardware del que dispone realmente y no el de nuestro proyecto de ejemplo.

Configuración de los módulos



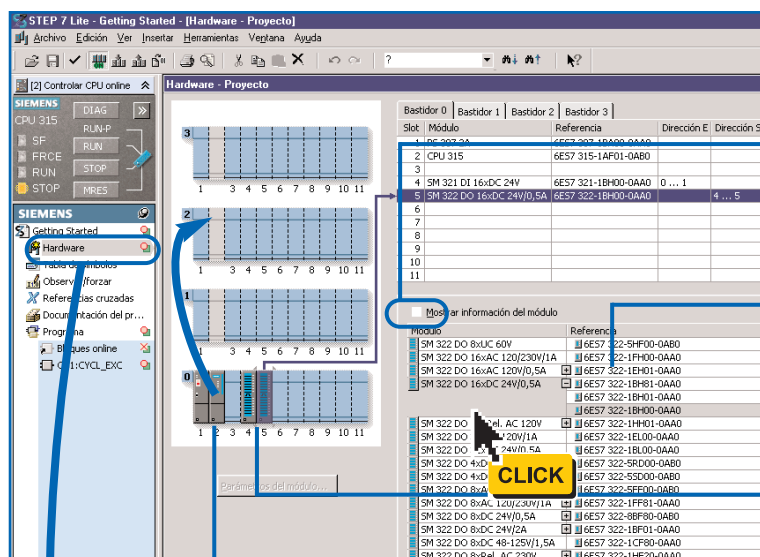
Resultado de la configuración

4 De izquierda a derecha, deberá conectar: la fuente de alimentación, la CPU, el módulo de entrada y el módulo de salida.

5 Inserte los módulos sin dejar slots vacíos; de lo contrario, los módulos que sigan al espacio vacío no recibirán corriente a través del bus posterior.

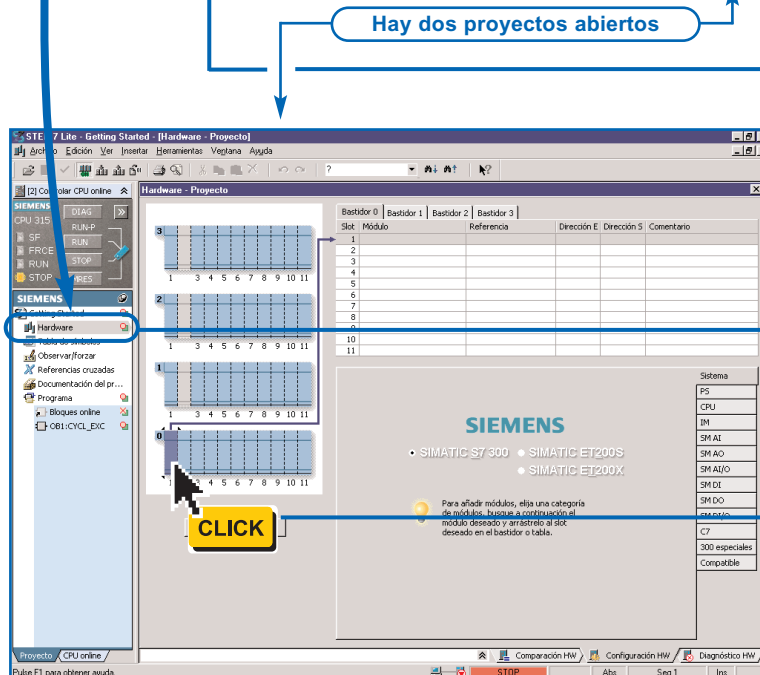
Excepción en STEP 7 Lite: El slot 3 está reservado para el módulo interfase (IM), con el que podrá conectar los bastidores superiores. Puede dejar este espacio libre cuando solamente el bastidor inferior contenga módulos.

6 En la **tabla de configuración**, en las columnas **Dirección E** y **Dirección S** aparecen automáticamente los bytes asignados como direcciones. Utilice estos bytes al indicar las direcciones en su programa.

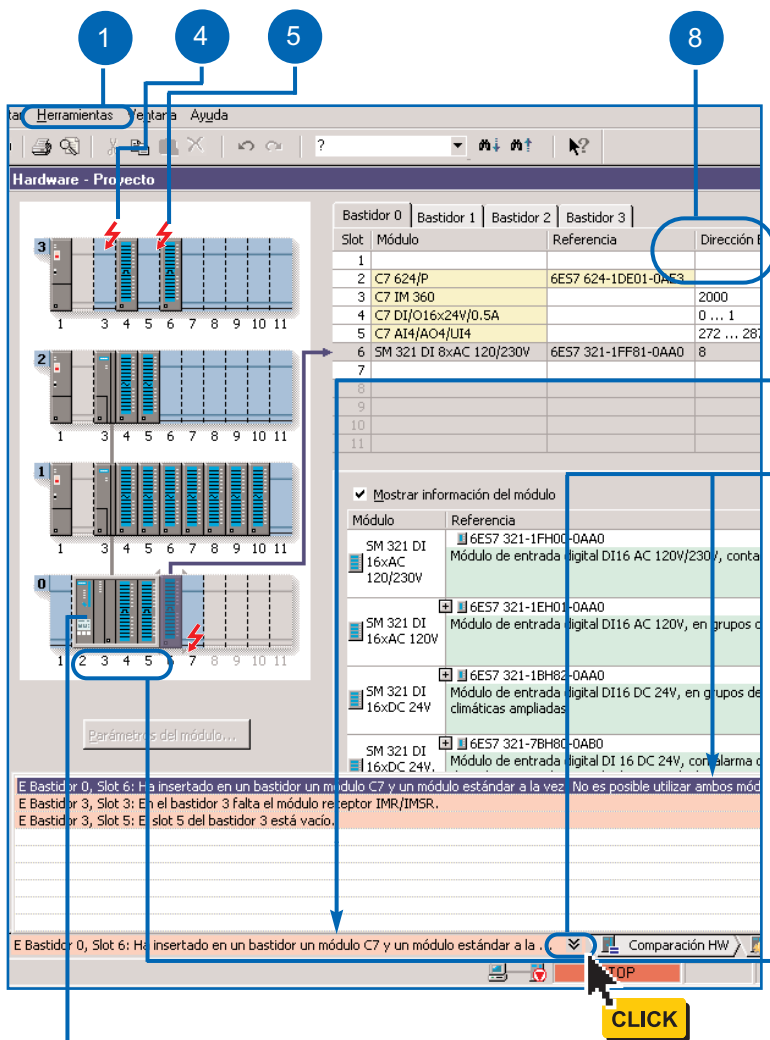


Consejos para insertar los módulos

- 1 Oculte la **información del módulo** si quiere tener una visión de conjunto, p. ej. cuando únicamente busque el nº de pedido (ref.).
- 2 Inserte los módulos, por ejemplo, haciendo clic con el botón derecho del ratón en el módulo y seleccionando **Insertar módulo** en el menú emergente.
- 3 Entre dos módulos pueden insertarse otros. Los módulos se desplazarán hacia la derecha para ocupar los slots contiguos.
- 4 Dentro del bastidor puede seleccionar uno o varios módulos con la tecla Mayús y moverlos o copiarlos con la función Arrastrar y soltar.
- 5 Abra otro proyecto en una segunda instancia de STEP 7 Lite y copie, por ejemplo, toda la configuración hardware de un proyecto a otro con Arrastrar y soltar.
- 6 Para borrar un módulo, abra el menú contextual con el botón derecho del ratón y elija el comando correspondiente.



Pruebe todas las funciones que se utilizan en otras aplicaciones de Windows. STEP 7 Lite ofrece muchas funciones típicas de Windows, como los menús contextuales, Arrastrar y soltar, combinaciones de teclas, etc.



Ejemplo de una configuración completa

A modo de ejemplo hemos creado una configuración de gran tamaño y hemos incluido un par de errores.

1 Para localizar el error, elija: **Herramientas > Comprobar coherencia.**

2 Aparecerán todos los errores de configuración que haya.

3 Si hay varios errores, aparecerán cuando haga clic con el botón izquierdo del ratón en el indicador de expansión.

Errores existentes:

4 En el bastidor 3 falta el módulo interfase (IM). Por esta razón, no hay conexión con el bastidor 3. En los bastidores 1 y 2 sí aparece este módulo interfase.

5 No está permitido dejar slots vacíos.

6 Se trata de un sistema C7 completo (aparece sobre fondo amarillo en la tabla de configuración). El módulo que hay a su derecha no es compatible.

Cuando haya eliminado todos los errores, vuelva a comprobar la coherencia.

Por lo demás:

7 En STEP 7 Lite sólo se admite una CPU por proyecto. Ésta se encuentra siempre en el bastidor 0. Los slots por encima del suyo quedarán vacíos.

8 Observe que, en la tabla de configuración, las áreas de direccionamiento se cuentan en orden ascendente.



Ayuda en pantalla: F1

- En la ayuda de STEP 7 Lite, seleccione **Contenido > Configurar el hardware > Configurar los módulos**. Aparecerá una ventana con las reglas básicas para configurar módulos.
- En **Índice > Reglas de slots** encontrará las reglas de ocupación de los slots.

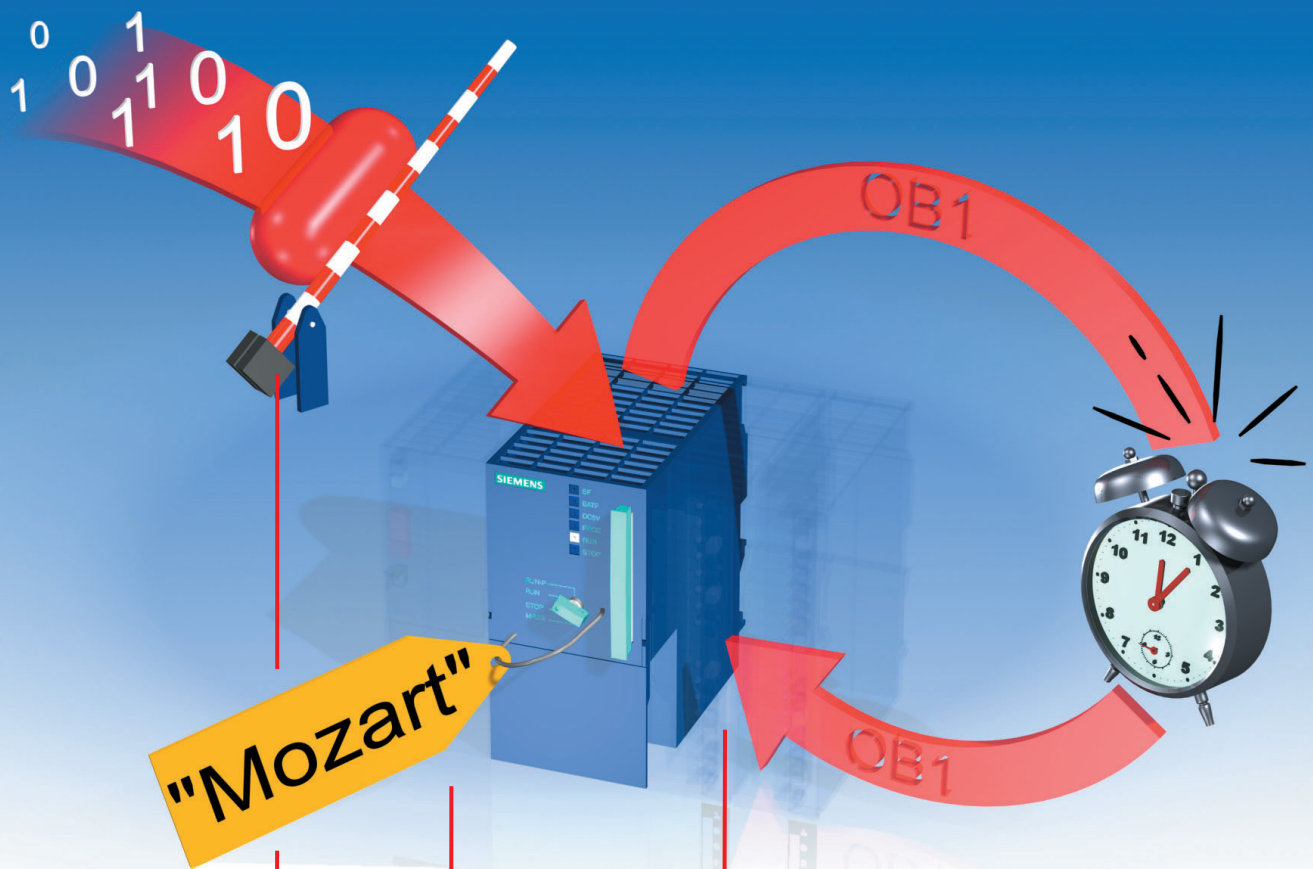
Navegar por el catálogo de hardware con la referencia

El número de referencia (o número MLFB) es el número de pedido de Siemens.

Si conoce el número de referencia de los módulos que quiere extraer del catálogo de hardware y ha seleccionado el catálogo, escriba la referencia en el campo de entrada “Buscar” de la barra de herramientas, Pulse **Entrar** y, a continuación, aparecerá el módulo correspondiente.

4.11

Parametrizar los módulos



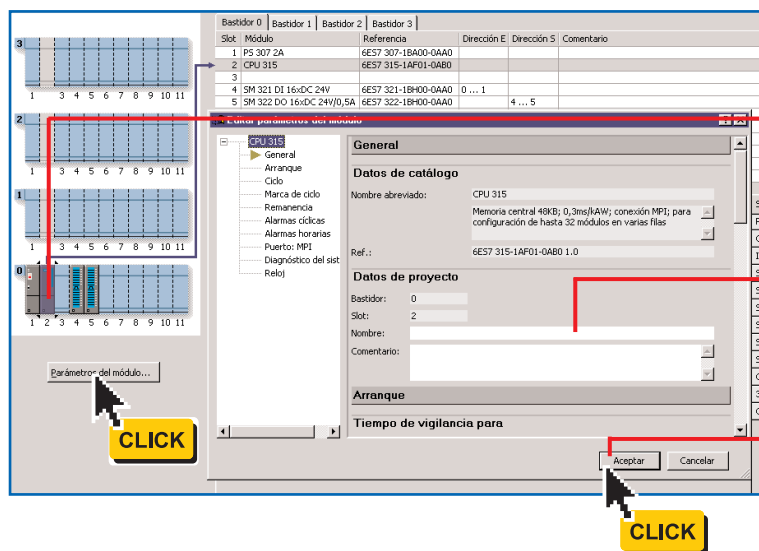
4.12

Qué significa parametrizar

Es posible modificar el comportamiento de la CPU y de ciertos módulos analógicos y digitales. Este ajuste se denomina "Parametrización".

Ejemplos de parametrización de una CPU:

- 1 – Puede interrumpir la ejecución cíclica del programa en la CPU con una alarma cíclica.
- 2 – Déle un nombre a la CPU. En este caso la llamaremos "Mozart".
- 3 – También puede proteger la CPU con una contraseña contra accesos de lectura/escritura a través de MPI.



Parametrización de la CPU 315

- 1 Seleccione la CPU 315 y haga clic en el botón **Parámetros del módulo**.
- 2 En el campo **Nombre** de la ventana **Editar parámetros del módulo**, escriba el nombre "Mozart".
- 3 Confirme las entradas con **Aceptar**. La ventana se cerrará.



Todos los parámetros vienen ajustados de fábrica y son válidos para casi todas las funciones estándar.

Si algo falla tras haber modificado algún parámetro, no se preocupe, en el catálogo de hardware encontrará los parámetros estándar de todos los módulos.

Ayuda en pantalla: F1

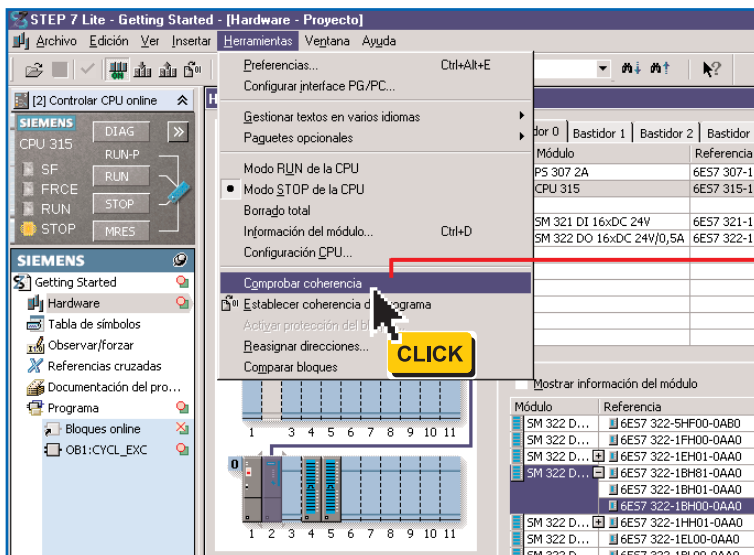
Los parámetros de las CPU suelen estar relacionados con los bloques de organización.

Por esta razón, en **Índice**, bajo **Alarma cíclica**, encontrará la explicación de los **Bloques de organización de alarma cíclica (OB 30 a OB 38)**.

Guardar datos de configuración



4.14



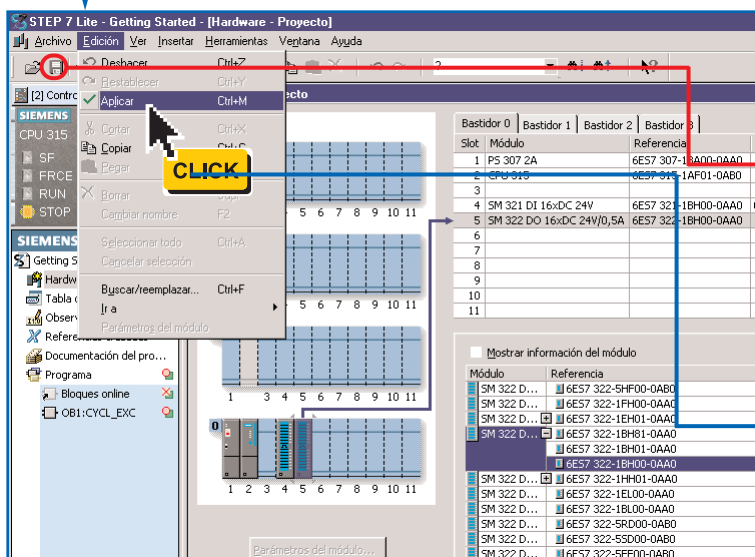
Comprobar los datos de configuración

Antes de guardar la configuración, deberá comprobar si ha introducido algún error.

Ejecute el comando de menú **Herramientas > Comprobar coherencia**. Se comprobará si se pueden crear datos de configuración a partir de los datos que ha introducido.

Confirme el mensaje “La configuración no contiene errores.” con **Aceptar**.

Continuar sin errores



Guardar los datos de configuración

Elija el menú **Archivo > Guardar** o haga clic en el icono del disquete en la barra de herramientas.

Además de la configuración de hardware, se guardarán todos los elementos del proyecto.

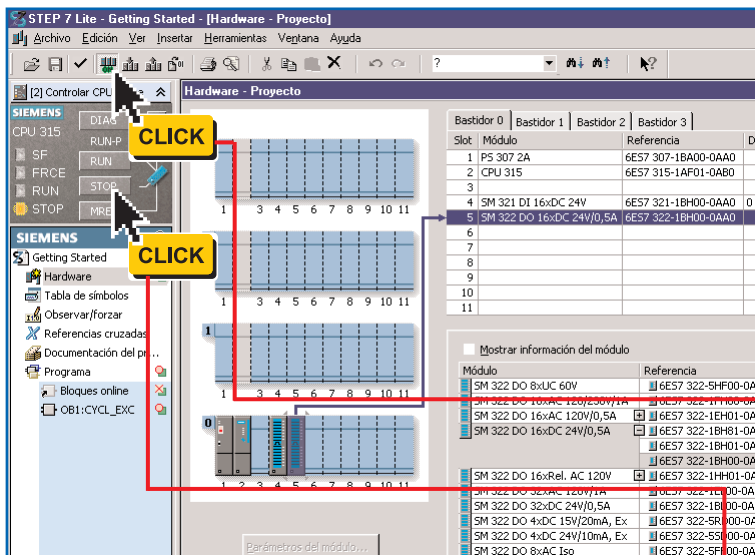
Seleccione el comando de menú **Edición > Aplicar** para guardar los datos de configuración en un archivo temporal (se guardará el contenido de la ventana activa).

Recomendamos guardar los datos de este modo en caso de que necesite recuperar la configuración anterior. Si ha ido guardando los datos de forma temporal y cierra el proyecto, aparecerá un mensaje preguntando si desea guardar los cambios.

Transferir la configuración hardware a la CPU



4.16



Listo para la transferencia

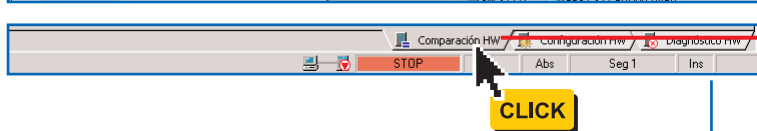
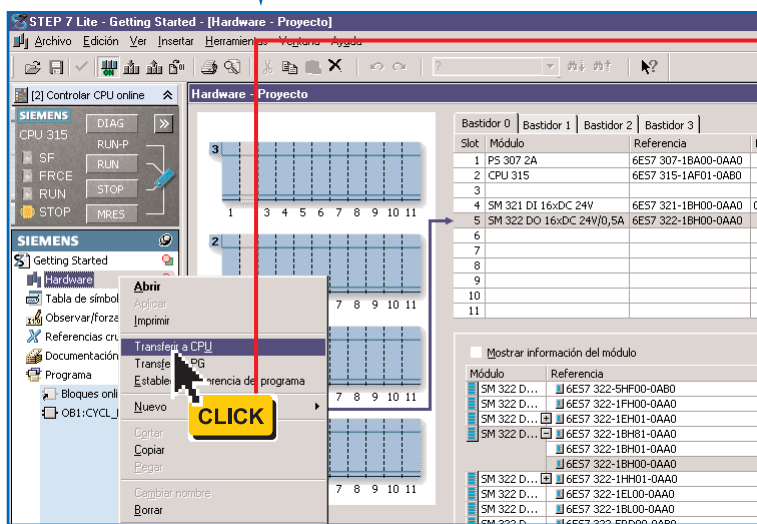
Preparativos y transferencia

En el proceso de transferencia se envían todos los datos de configuración a la CPU.

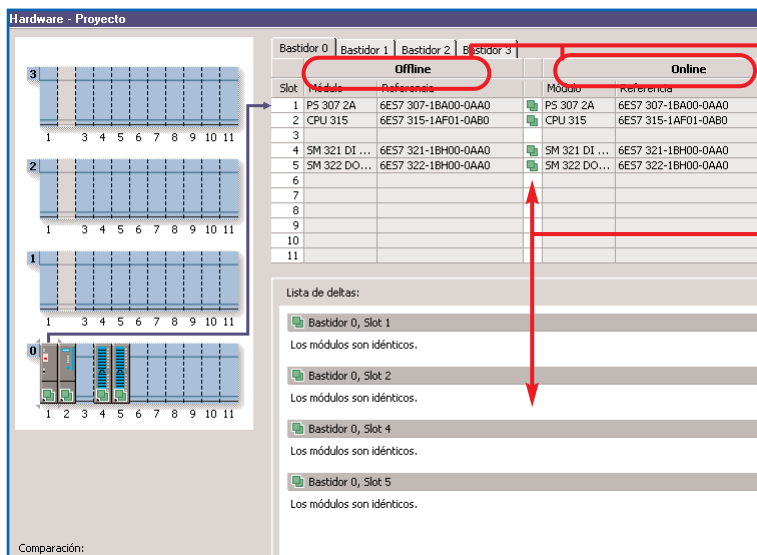
Sin embargo, antes deberá “conectar online” la CPU y la unidad de programación. Para más información consulte el capítulo 10.

Pasos principales para “conectar online”:

- 1 Después de enchufar el cable y de efectuar el borrado total de la CPU, haga clic en **Conectar online**.
- 2 Vaya al panel de mandos de la CPU y conmutela a **STOP**. En la barra de estado se iluminará la palabra STOP en rojo.



Se abre la vista "Comparación HW"



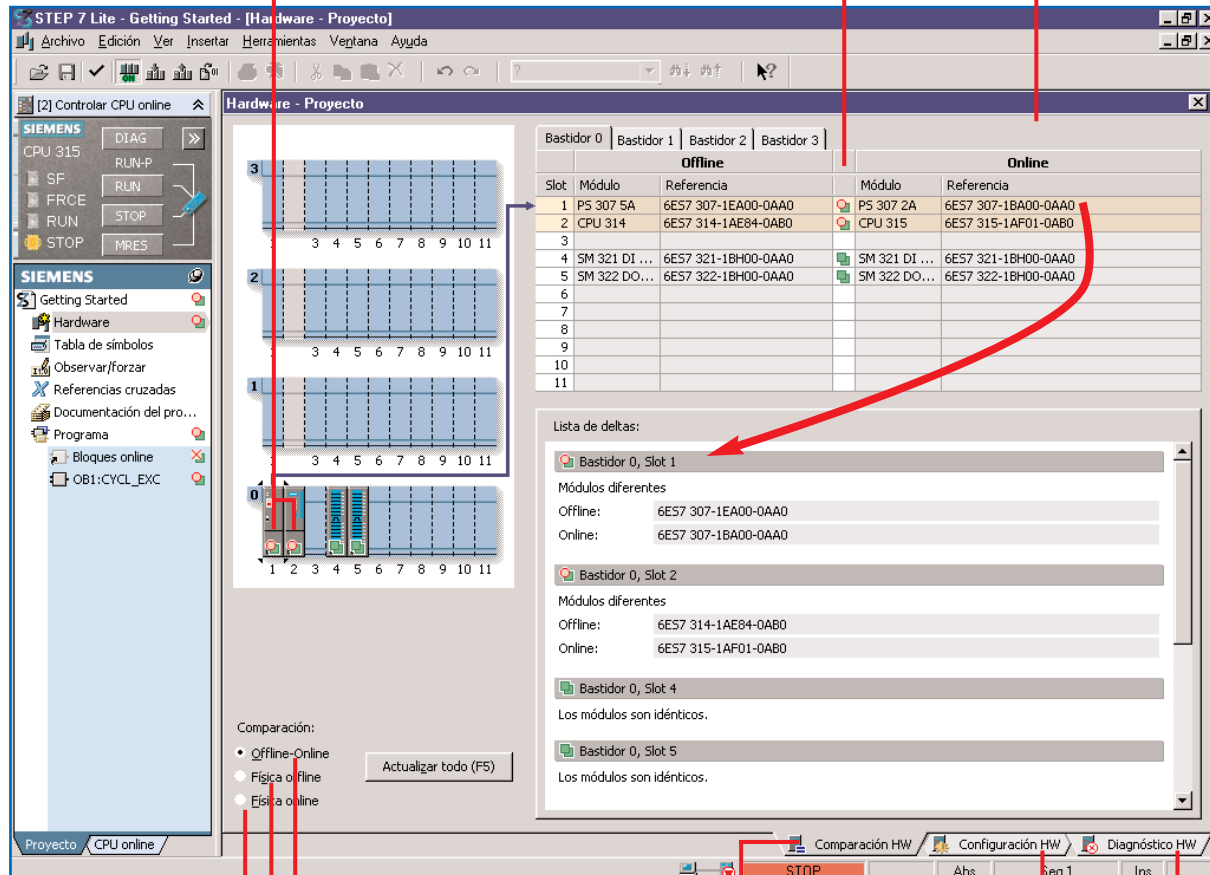
Más detalles sobre la pantalla



Para volver a transferir la configuración hardware de una CPU a la unidad de programación, utilice el comando **Transferir a PG**. Este es el comando que utiliza típicamente el técnico de servicio cuando trabaja con la unidad de programación en un armario eléctrico para analizar posibles errores.

Tabla de comparación
Contraposición Online/Offline/HW real

Pictograma de diferencias
entre módulos



Información general

1 En la ficha **Configuración HW**, indique su hardware como se ha descrito anteriormente.

2 Si surgen problemas, compare en la ficha **Comparación HW** si los datos de configuración coinciden. Más adelante continuaremos profundizando en el tema.

3 Si sigue habiendo problemas, es posible que los módulos estén defectuosos. Podrá comprobarlo en la ficha **Diagnóstico HW**. Consulte el capítulo "12 Diagnósticos de error".

... muestra las diferencias entre la configuración realizada en la ficha Proyecto de la PG (Offline) y la configuración que ha sido transferida a la CPU

... muestra las diferencias entre la configuración realizada y el hardware realmente insertado

... muestra las diferencias entre la configuración transferida y la configuración real

Detección de errores:

Ha transferido la configuración a la CPU y ha abierto la ficha **Comparación HW**.

En la ventana del proyecto aparecerá un **pictograma de conjunto** detrás de **Hardware**. Esto significa que uno o varios módulos no coinciden.

En los módulos del **Bastidor** aparecen pictogramas que muestran el estado de estos módulos.

Comparación: Offline - Online

Online: Configuración transferida a la CPU.

Offline: Configuración en la unidad de programación.

Si hace clic en la ficha **Comparación HW**, de forma estándar aparecerá el botón de opción **Comparar: Offline-Online**. En la **lista Delta** se indicarán las diferencias detectadas en la configuración y en la parametrización.

4.19

Comparación: Offline - Hardware real Comparación: Online - Hardware real

Hardware real: La configuración que una CPU reconoce por sí misma sin que se haya transferido a ella una configuración se denomina Hardware real.

Si elige el botón de opción correspondiente, podrá comparar la configuración online y offline del hardware real.

Símbolos

He aquí los símbolos más importantes de la configuración hardware.



El módulo del proyecto no coincide con el módulo de la CPU online.



El módulo realmente insertado coincide con el módulo configurado, pero sus parámetros son distintos.



El módulo está configurado, pero no está disponible online.



Simboliza un “módulo probablemente idéntico”. El tipo de módulo realmente insertado coincide con el del módulo configurado, pero no se puede determinar si el número de pedido (ref.) también coincide.



Estado operativo RUN



Estado operativo STOP



Estado operativo PARADA



Fallo



Símbolos (iconos)

La ayuda rápida de los símbolos muestra información adicional.

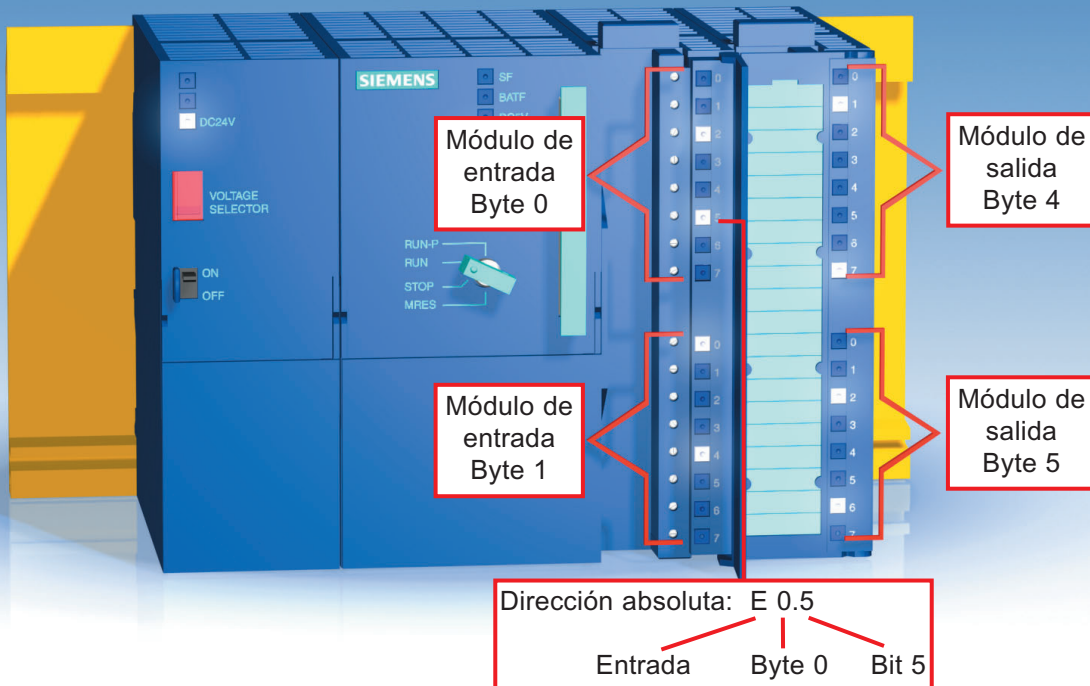
En el menú **F1 > Índice > Símbolos** encontrará un resumen de los símbolos que pueden aparecer en la ventana del proyecto, en el bastidor y en la tabla de comparación.

5

Creación de la tabla de símbolos



Programación absoluta



5.2

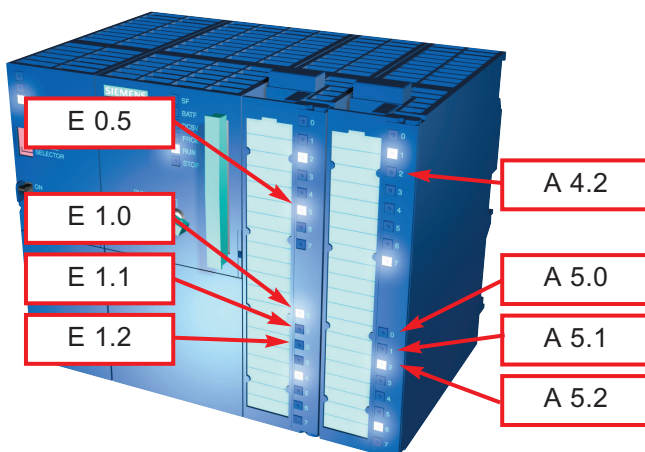
Asignación de direcciones

En el capítulo 4 se describe cómo se asignan automáticamente las direcciones absolutas al configurar el hardware. Recordemos:

A cada entrada y salida se le asigna una dirección absoluta en el sistema de automatización.

La dirección absoluta puede sustituirse por un nombre simbólico libremente definible, como por ejemplo A 4.2: AUTOMATICO ON.

La asignación de los símbolos es independiente de los lenguajes de programación KOP, FUP y AWL.



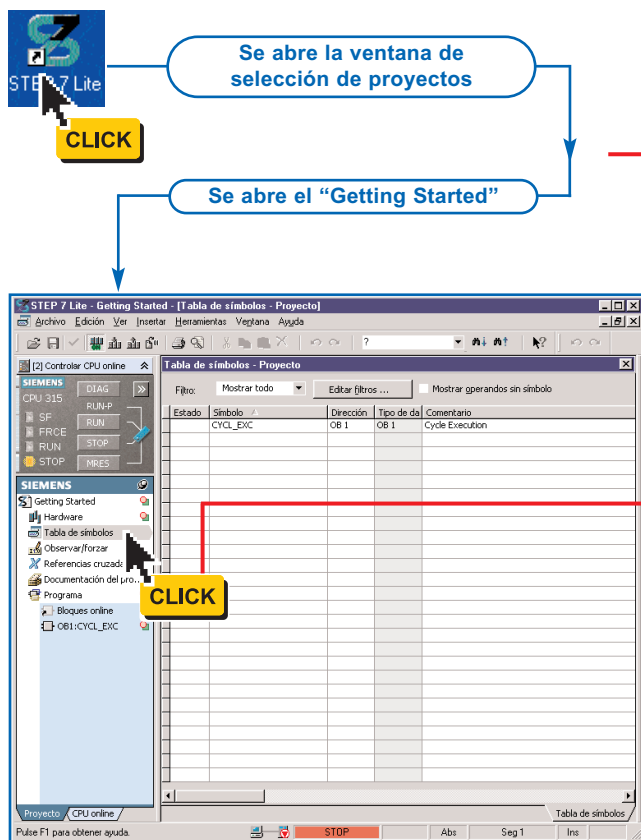


Tabla de símbolos y direcciones absolutas

Abra STEP 7 Lite y, en la ventana **Abrir proyecto**, haga doble clic en el proyecto "Getting Started.k7p", creado en el capítulo 4.

De momento, el proyecto se compone exclusivamente de los elementos de proyecto predeterminados y del elemento de programa **OB 1**.

Haga doble clic sobre el elemento **Tabla de símbolos** en la ventana del proyecto.

En estos momentos, la tabla de símbolos contiene solamente el bloque de organización predefinido OB 1.

Si desea utilizar direcciones absolutas, en su programa, no necesitará realizar ninguna entrada más. Simplemente vuelva a cerrar la ventana.

En nuestro proyecto de ejemplo utilizaremos direcciones simbólicas. Para ello, siga las instrucciones que aparecen en las páginas siguientes.



Utilice direcciones absolutas solamente cuando su programa STEP 7 Lite deba direccionar un número muy reducido de entradas y salidas.

Programación simbólica

Filtrar la tabla (para ver, p. ej., sólo salidas)

Cambiar el orden de aparición haciendo clic en la cabecera

STEP 7 Lite - primeros_pasos_kop - [Tabla de símbolos - Proyecto]

Archivo Edición Ver Insertar Herramientas Ventana Ayuda

CPU offline

SIEMENS

DIAG >>

SF RUN

FRCE STOP

RUN MRES

STOP

SIEMENS

primeros_pasos_kop

Hardware

Tabla de símbolos

Observar/forzar

Referencias cruzadas

Documentación del proyecto

DB3:Datos_globales

FB1:Motor

FC1:Ventilador

OB1:Programa prin...

Tabla de símbolos - Proyecto

Filtro: Mostrar todo Editar filtros ...

Mostrar operandos sin símbolo

Estado	Símbolo	Dirección	Tipo de da	Comentario
	Arrancar_MD	E 1.4	BOOL	Arrancar el motor diesel
	Arrancar_MG	E 1.0	BOOL	Arrancar motor de gasolina
	Automático ON	E 0.5	BOOL	para la función de memorización (activar)
	Conectar_ventilador_MD	A 5.6	BOOL	Comando para conectar ventilador motor diesel
	Conectar_ventilador_MG	A 5.2	BOOL	Comando para conectar ventilador motor gasolina
	Datos_globales	DB 3	DB 3	Bloque de datos globales
	Debe_revol_alcanzado_MD	A 5.5	BOOL	Indicador motor diesel alcanza núm. revol. teórico""
	Debe_revol_alcanzado_MG	A 5.1	BOOL	Indicador motor gasolina alcanza núm. revol. teórico""
	Diesel	DB 2	FB 1	Datos del motor diesel
	Fallo_DM	E 1.6	BOOL	Fallo del motor diesel
	Fallo_MG	E 1.2	BOOL	Fallo del motor de gasolina
	Gasolina	DB 1	FB 1	Datos del motor de gasolina
	Lámpara roja	A 4.1	BOOL	Resultado de la consulta O
	Lámpara verde	A 4.0	BOOL	Resultado de la consulta Y
	Manual ON	E 0.6	BOOL	para la función de memorización (desactivar)
	MD_ON	A 5.4	BOOL	Comando para arrancar el motor diesel
	MG_ON	A 5.0	BOOL	Comando para arrancar el motor de gasolina
	Modo automático	A 4.2	BOOL	Salida con memorización
	Motor	FB 1	FB 1	Control del motor
	Parar_MD	E 1.5	BOOL	Parar motor diesel
	Parar_MG	E 1.1	BOOL	Parar el motor de gasolina
	Programa principal	OB 1	OB 1	Este bloque contiene el programa de usuario
	Pulsador 1	E 0.1	BOOL	para la consulta Y
	Pulsador 2	E 0.2	BOOL	para la consulta Y

Aquí puede introducir comentarios extensos

Copiar la tabla de símbolos de un proyecto a otro

Símbolos

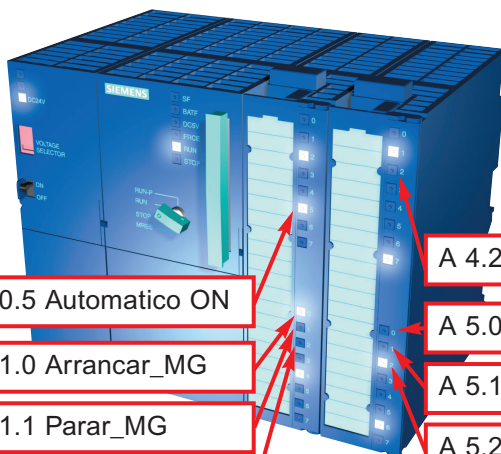
Direcciones absolutas

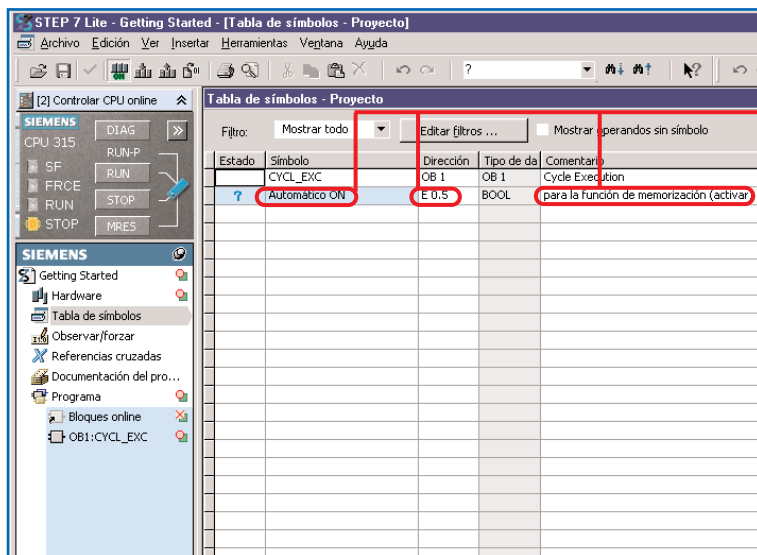
La tabla de símbolos

En la tabla de símbolos se asigna un nombre simbólico y un tipo de datos a toda dirección absoluta que se direcciona posteriormente en el programa, p. ej. a la entrada "E 0.5" se le asignará el símbolo "AUTOMATICO ON".

Los nombres simbólicos aquí definidos son válidos para todo el proyecto y se denominan símbolos globales.

Gracias a la programación simbólica el programa resulta más fácil de entender.





Rellenar la tabla de símbolos

Para la dirección “E 0.5”, introduzca en la columna **Símbolo** “Automático ON”. En la columna **Comentario**, escriba el comentario tal y como se indica a la izquierda.

Al realizar las entradas:
Entrar = Una fila más abajo
Ctrl + z = Deshacer

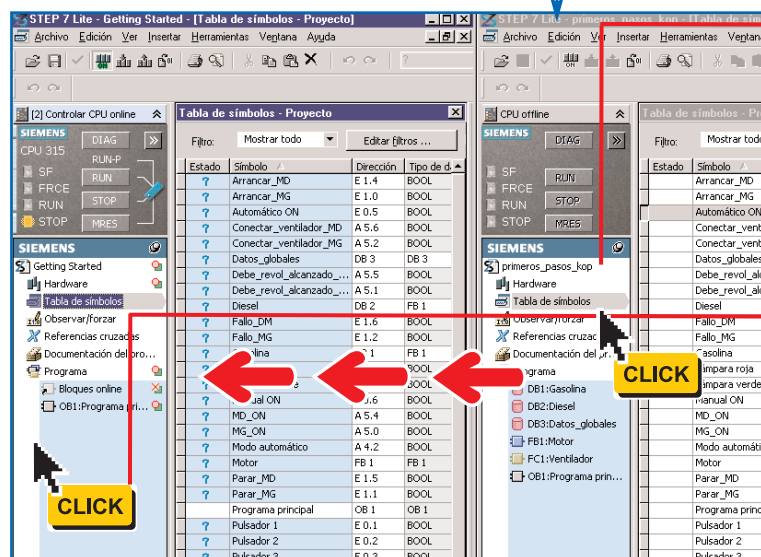
Guarde las entradas con **Archivo > Guardar**.

Copiar la tabla de símbolos

Como el proyecto “Getting Started” contiene gran cantidad de símbolos, copiaremos la tabla de símbolos de uno de los proyectos de ejemplo suministrados.



Se abre una segunda instancia de STEP 7 Lite



Abra en una segunda instancia de STEP 7 Lite el proyecto “primeros_pasos_kop.k7p”.

Haga clic con el botón derecho del ratón en la **tabla de símbolos** del proyecto “primeros_pasos_kop” y seleccione **Copiar** en el menú emergente.

En el proyecto “Getting Started”, haga clic con el botón derecho del ratón en la ventana del proyecto y seleccione **Pegar** en el menú emergente. Aparecerá un mensaje preguntando si desea sobrescribir la tabla existente. Confirme la pregunta con “Aceptar”.

Cierre el proyecto “primeros_pasos_kop”.
 Guarde el proyecto “Getting Started” con **Archivo > Guardar**.

Tipos de datos

Los tipos de datos determinan el tipo de señal que debe procesar la CPU.

STEP 7 Lite utiliza, entre otros, los tipos de datos que figuran a la izquierda.

BOOL
BYTE
WORD
DWORD

- Los datos de este tipo son combinaciones de bits. De 1 bit (tipo BOOL) a 32 bits (DWORD).

CHAR

- Los datos de este tipo ocupan exactamente 1 carácter del juego de caracteres ASCII.

INT
DINT
REAL

- Los datos de este tipo sirven para procesar valores numéricos (p. ej. para calcular expresiones aritméticas).

S5TIME
TIME
DATE
TIME_OF_DAY

- Los datos de este tipo representan los distintos valores de fecha y hora dentro de STEP 7 Lite (p. ej. para ajustar la fecha o para indicar la hora).



Para más información sobre los tipos de datos (p. ej. los rangos de valores admitidos y ejemplos de aplicación) haga clic sobre un tipo de datos y vaya a **Introducción a los tipos de datos** y a **los tipos de parámetros**.

6

Primeros pasos de programación



Elegir KOP, FUP o AWL



6.2

Páginas 6.6 a 6.11

El esquema de contactos (KOP) es especialmente apropiado para usuarios del ramo de la electricidad y electrónica.

Páginas 6.12 a 6.17

La lista de instrucciones (AWL) es especialmente apropiada para usuarios del ramo de la informática.

Páginas 6.18 a 6.23

El diagrama de funciones (FUP) es especialmente apropiado para usuarios acostumbrados a trabajar con esquemas de circuitos.

Ya sea con KOP, FUP o AWL, en STEP 7 Lite creará su programa con un solo interface de programación, el editor de bloques. El interface se ajustará al lenguaje de programación elegido.

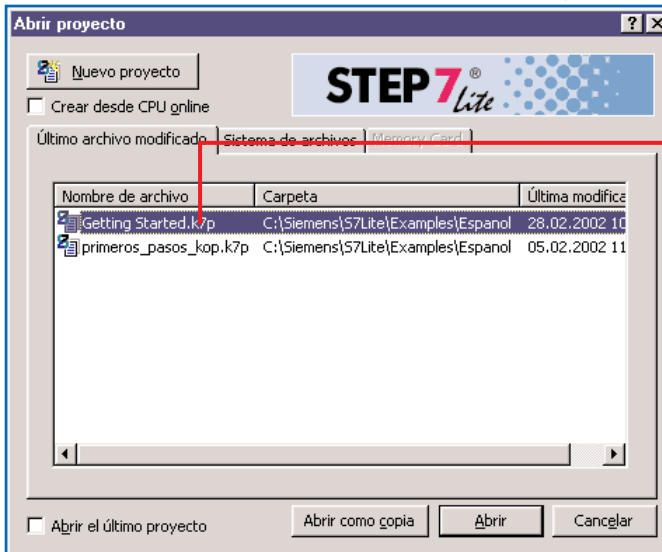
Por ejemplo, si elige programar con KOP, consulte las páginas 6.6 a 6.11.



Se abre el cuadro de diálogo de selección de proyectos.

Abrir el OB 1.

Abra STEP 7 Lite.

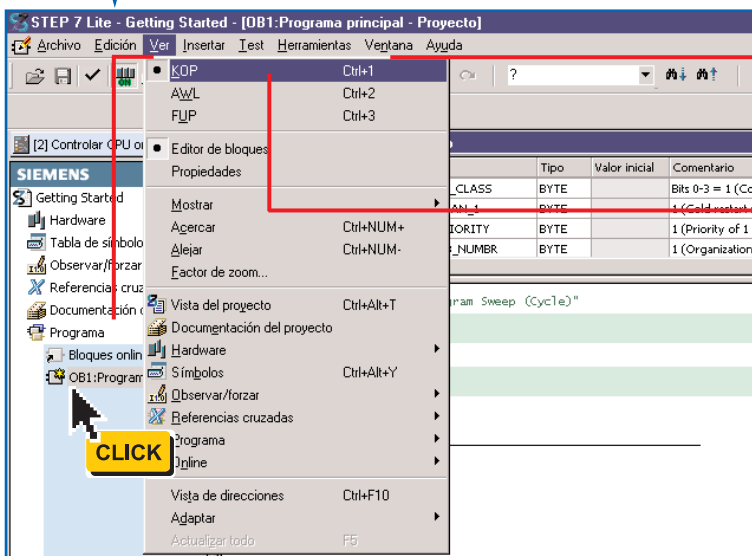


En el cuadro de diálogo **Abrir proyecto**, haga doble clic en el proyecto "Getting Started.k7p".

Se trata del proyecto creado en el capítulo 4 y cuya tabla de símbolos ya rellenó en el capítulo 5.

Si no es el caso, abra simplemente un "Nuevo proyecto" y copie la tabla de símbolos de uno de los proyectos de ejemplo suministrados.

Se abre STEP 7 Lite



Haga doble clic en el **OB 1**. En el área de trabajo se abrirá el editor de bloques.

Haga clic en **Ver**. En este menú puede comprobar si está seleccionado el lenguaje **KOP**, **FUP** o **AWL**. Aquí también puede cambiar de vista.

Nota:

Ciertas instrucciones no se pueden visualizar en todos los lenguajes de programación. Éstas aparecen siempre en AWL.

Para más información sobre las distintas operaciones KOP, FUP y AWL consulte **F1 > Contenido > Ayudas de referencia**.

Puede trabajar en el editor de bloques.

Trabajar con el editor de bloques

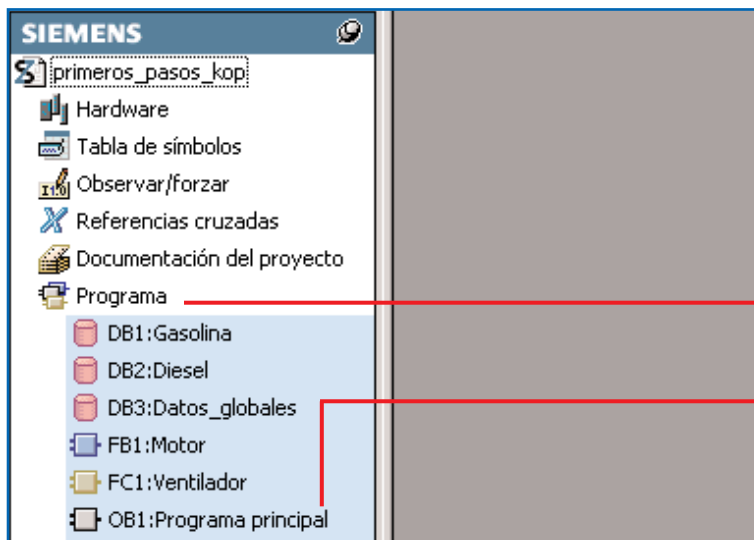
The screenshot shows the STEP 7 Ladder Editor interface. The main window displays a ladder logic diagram for a program named 'OB1: Programa principal - Proyecto'. The interface includes a menu bar, a toolbar, and several panels. The left panel shows the project tree with 'OB1: Programa principal' selected. The top panel shows the variable declaration table. The right panel shows the library of logic blocks. The bottom panel shows the ladder logic diagram with a 'STOP' block.

Labels and their corresponding functions:

- Cambiar el lenguaje de programación**: Points to the 'Editor de bloques' button in the bottom toolbar.
- Las operaciones más importantes de KOP y FUP**: Points to the 'Librerías' panel on the right, which contains various logic blocks.
- Insertar otro segmento**: Points to the 'Nuevo segmento' button in the 'Librerías' panel.
- Mostrar y ocultar el panel de mandos de la CPU**: Points to the 'CPU online' button in the left panel.
- Tabla de declaración de variables**: Points to the 'Declaración' table at the top of the main window.
- Cambiar entre programación simbólica y absoluta**: Points to the 'Simbólico' and 'Absoluto' buttons in the bottom toolbar.
- Elemento de proyecto "Programa"**: Points to the 'Programa' folder in the left panel.
- Elemento de programa "OB 1"**: Points to the 'OB1: Programa principal' folder in the left panel.
- Título y comentario del segmento**: Points to the 'Título' and 'Comentarios' fields in the main window.
- Segmento para introducir el programa**: Points to the 'STOP' block in the ladder logic diagram.
- Definir propiedades del bloque: p. ej., cambiar un nombre simbólico**: Points to the 'Propiedades' button in the bottom toolbar.
- Todas las operaciones para KOP y FUP**: Points to the 'Librerías' panel on the right.
- Un clic con el signo de interrogación abre la ayuda de referencia de la operación**: Points to the question mark icon in the toolbar.

En el editor de bloques se programan todos los bloques.

A modo de ejemplo, aquí aparece representada la vista del lenguaje de programación KOP.



Elemento de proyecto “Programa”

En STEP 7 Lite, el programa de usuario se divide en bloques. Ello facilita la programación de programas muy extensos.

Debajo del elemento de proyecto **Programa** aparecen estos bloques.

Si se trata de un proyecto recién creado, sólo aparecerá el **OB 1** que STEP 7 Lite crea automáticamente. Posteriormente puede insertar otros bloques, p.ej.:

OB = Bloque de organización
DB = Bloque de datos
FB = Bloque de función
FC = Función

El bloque de organización OB 1 es el interface con el sistema operativo de la CPU y contiene el programa principal. El OB 1 suele contener las llamadas a otros bloques y transferir los parámetros necesarios para controlar el proceso.

6.5



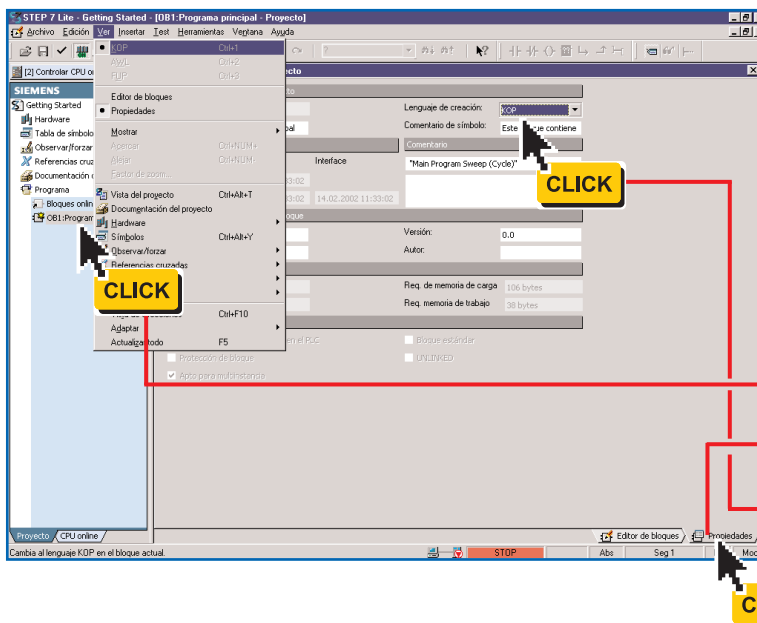
Con el comando de menú contextual **Cambiar nombre** puede cambiar el nombre de un bloque.

Como principiante, quizá necesite más información para trabajar con bloques. Para ello, haga clic a la izquierda en la ventana del proyecto y, a continuación, en **F1 > Índice > Jerarquía de llamada en el programa de usuario**.

Programar el OB1 en KOP



6.6



En este apartado programaremos una conexión en serie, una conexión en paralelo y la función de memoria Activar y Desactivar en KOP (esquema de contactos).

Elija el lenguaje de programación con el que vaya a programar el OB 1 y con el que lo abrirá posteriormente:

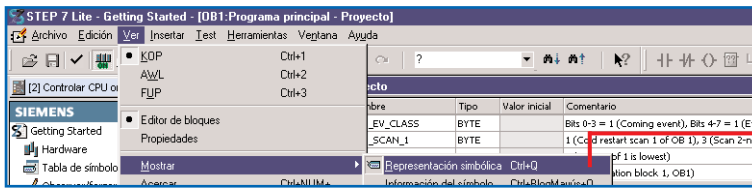
- 1
- 2
- 3

Haga doble clic en el **OB 1**.

Haga clic en **Propiedades**.

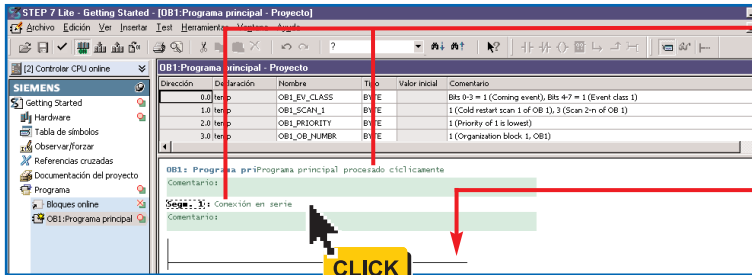
Seleccione **KOP**. El OB 1 se abrirá a partir de ahora en KOP.

Salga del diálogo **Propiedades**. En el menú **Ver** ahora también aparecerá seleccionado **KOP**.



Programar una conexión en serie con KOP

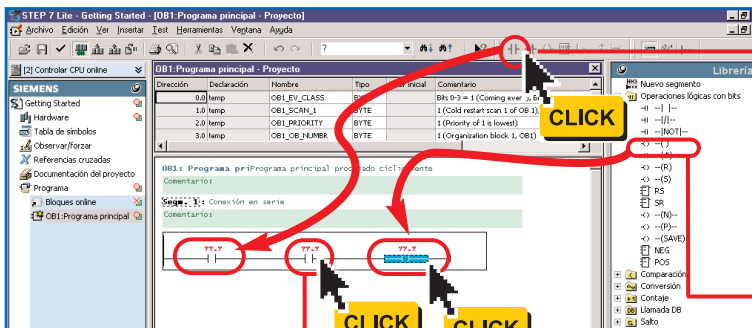
En el menú **Ver**, seleccione la representación simbólica.



En el **OB 1**, indique “Programa principal procesado cíclicamente”. En el **Segmento 1**, escriba “Conexión en serie”.

Marque el circuito vacío haciendo clic sobre él.

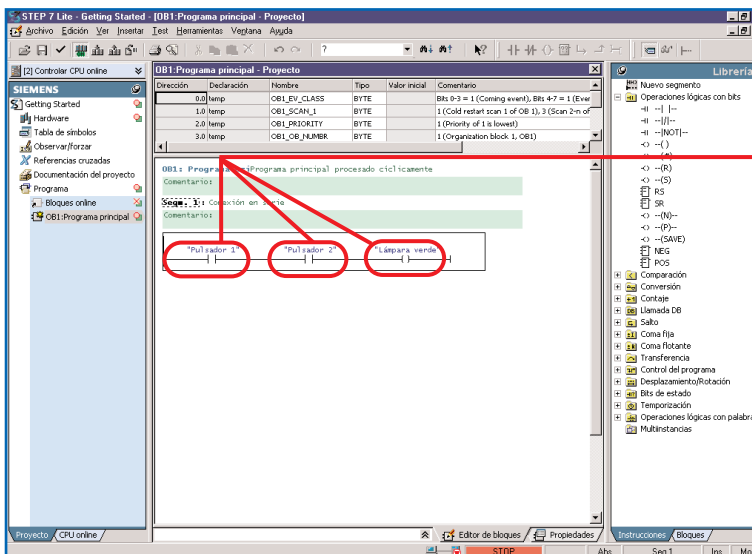
Inserte tres elementos de programa de distinto tipo:



Haga clic en el símbolo del contacto normalmente abierto. Se insertará inmediatamente.

Haga clic con el botón derecho del ratón en el circuito y elija Contacto normalmente abierto en el menú emergente.

Arrastre la bobina hasta el circuito.



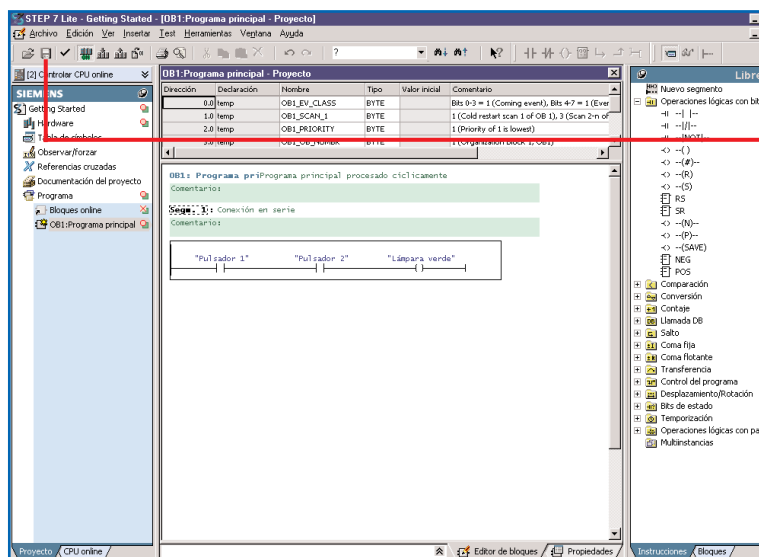
En la conexión en serie todavía falta direccionar los contactos normalmente abiertos y las bobinas:

Haga clic en ??? e introduzca el nombre simbólico “Pulsador 1” (entre comillas). También puede seleccionar un nombre de la lista de selección de símbolos que aparecerá al hacer clic en ???.

Confirme con **Entrar**.

Para el segundo contacto normalmente abierto, indique el nombre simbólico “Pulsador 2”.

En la bobina, indique el nombre “Lámpara verde”.



La conexión en serie quedará programada completamente.

Si no queda ningún símbolo más marcado en rojo, ahora podrá guardar las entradas haciendo clic en el icono del disquete.

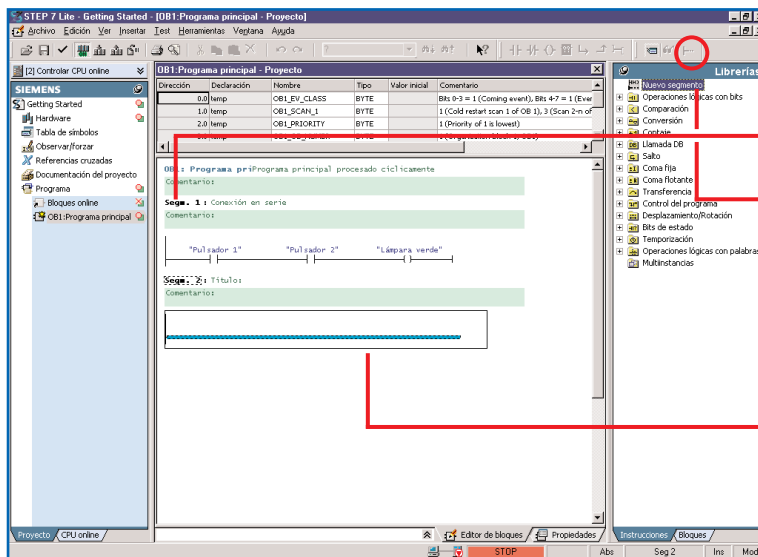
No se guardarán únicamente las entradas del OB 1, sino todos los elementos del proyecto.

13 Seleccione el comando de menú **Edición > Aplicar** para guardar los datos de configuración en un archivo temporal (se guardará el contenido de la ventana activa). Recomendamos guardar los datos de este modo en caso de que necesite recuperar la configuración anterior. Si ha ido guardando los datos de forma temporal y cierra el proyecto, aparecerá un mensaje preguntando si desea guardar los cambios.



Los símbolos aparecerán marcados en rojo cuando, por ejemplo, el símbolo no se encuentre en la tabla de símbolos o haya un error de sintaxis.

No será posible guardar los datos y en la parte inferior de la ventana del editor aparecerá un mensaje de error con indicaciones sobre el procedimiento correcto que se debe seguir.



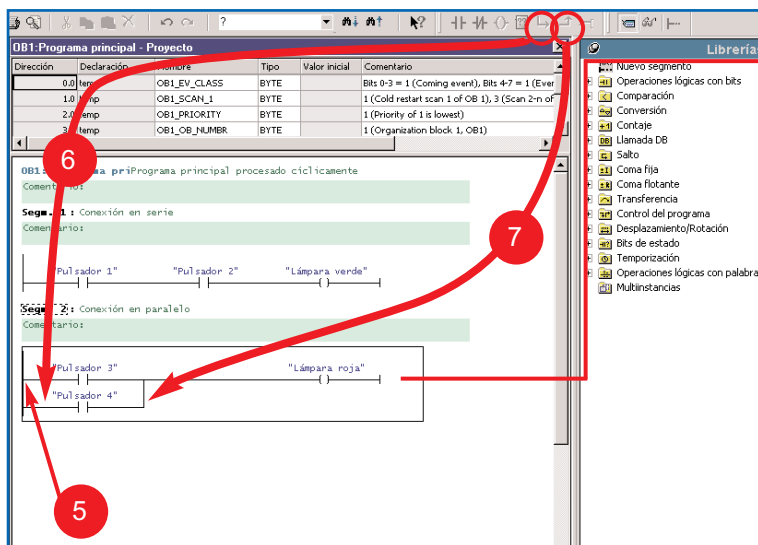
Programar una conexión en paralelo con KOP

1 Seleccione el segmento 1.

2 Inserte un nuevo segmento.

Puede hacerlo a través de un icono de la barra de herramientas, con el botón derecho del ratón o con CTRL R.

3 Vuelva a seleccionar el circuito.



4 Inserte un contacto normalmente abierto y una bobina. Aquí los llamaremos "Pulsador 3" y "Lámpara roja".

5 Seleccione la barra de alimentación izquierda.

6 Inserte una rama paralela. Coloque en ella otro contacto normalmente abierto.

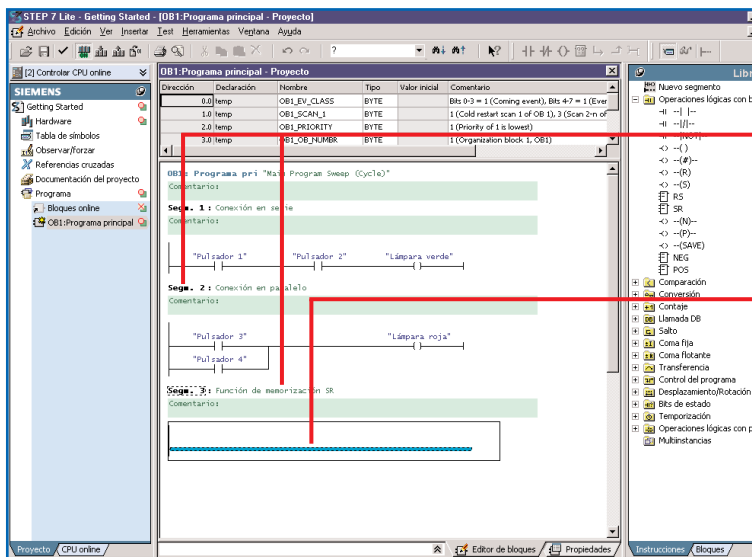
7 Cierre la rama con el símbolo correspondiente o arrastrando la flecha doble que aparece después de insertar el contacto.

8 En la rama paralela sólo falta el direccionamiento. Indique su nombre como en la figura. Guarde los datos.



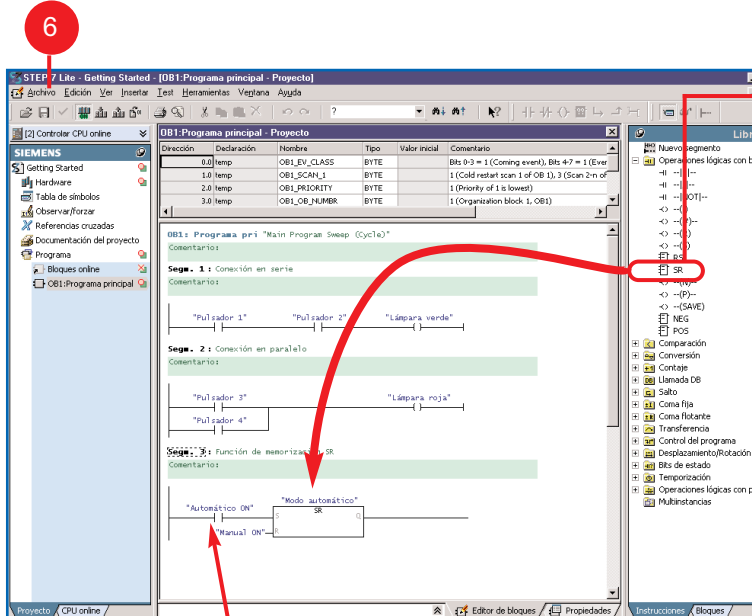
Asigne nombres sencillos y autoexplicativos a los segmentos. De esta forma, la búsqueda resultará más fácil al desplazarse con la barra de desplazamiento derecha por distintos programas. Al desplazarse aparecerán los nombres.

Programar una función de memoria con KOP



1 Seleccione el segmento 2, inserte un nuevo segmento y déle el nombre "Función de memoria".

2 Vuelva a seleccionar el circuito.



3 En la vista de instrucciones, mueva el cursor hasta **Operaciones lógicas con bits** y seleccione **Elemento SR** para insertar el segmento.

4 Inserte un contacto normalmente abierto delante de la entrada S (Activar).

5 Asigne los siguientes nombres simbólicos:

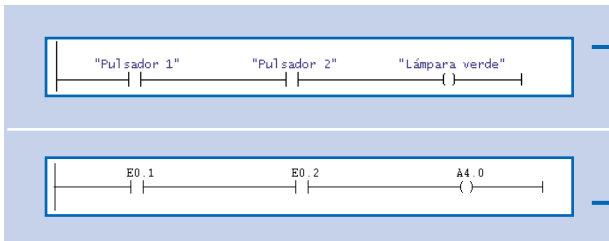
- contacto normalmente cerrado superior "Automático ON",
- entrada R (Desactivar) "Manual ON",
- elemento SR "Modo automático".

6 Guarde las entradas con **Archivo > Guardar**.

Personalizar el interface de programación

Utilice comandos de menú para configurar el interface de programación a su voluntad en STEP 7 Lite.

Menú **Ver** - Ejemplos:



1

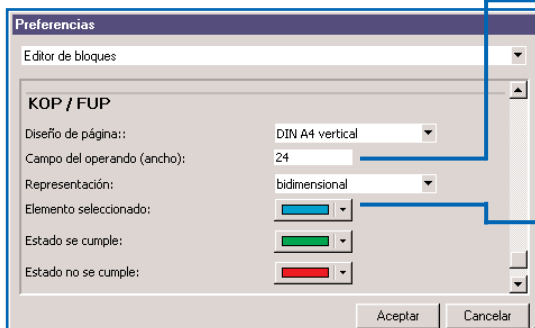
Direccionamiento simbólico en KOP:
Active **Ver > Mostrar > Representación simbólica**

2

Direccionamiento absoluto en KOP:
Desactive **Ver > Mostrar > Representación simbólica**

Cambio de lenguaje de programación
Ver > KOP/FUP/AWL

Comando de menú **Herramientas > Preferencias** - Ejemplos:



3

Ajustar un cambio de línea en el direccionamiento simbólico entre el carácter 10° y el 24°:
Herramientas > Preferencias > KOP/FUP > Campo del operando (ancho)

4

Cambiar los colores en el circuito:
Herramientas > Preferencias > KOP/FUP > Elemento seleccionado

7

Cierre el bloque con el símbolo **Cerrar** de la ventana.

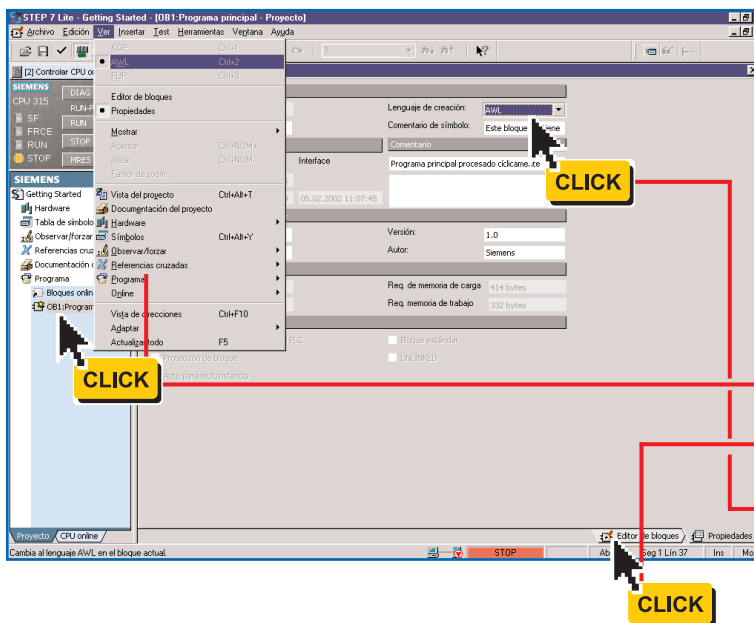


El comando de menú **Herramientas > Preferencias** ofrece muchas opciones para modificar la representación de STEP 7 Lite mediante colores, tipos de letra, campos de operandos, etc.

Programar el OB1 en AWL



6.12



A continuación programaremos una instrucción Y (U), una instrucción O (O) y la instrucción de memoria Activar y Desactivar en AWL (lista de instrucciones).

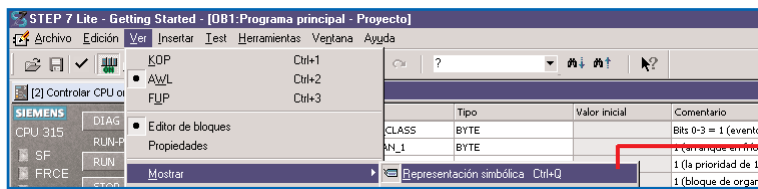
Elija el lenguaje de programación, en el que vaya a programar el OB 1 y con el que lo abrirá posteriormente:

Haga doble clic en el **OB 1**.

Haga clic en **Propiedades**.

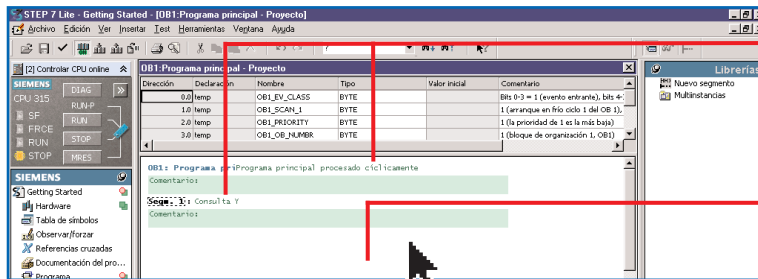
Seleccione **AWL**. El OB 1 se abrirá a partir de ahora en AWL.

Salga del diálogo **Propiedades**. En el menú **Ver** ahora aparecerá seleccionado **AWL**.



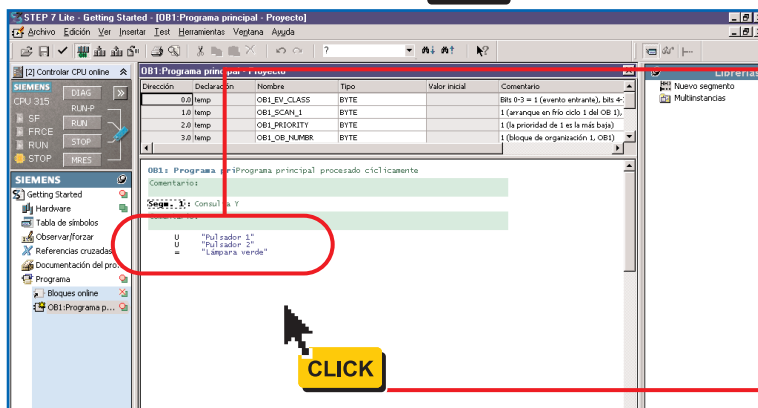
Programar una instrucción U (Y) en AWL

En el menú **Ver**, seleccione la representación simbólica.



En el **OB 1**, introduzca “Programa principal procesado cíclicamente”. En el **Segmento 1**, escriba “Consulta Y”.

Haga clic en el área de entrada.



En la primera fila de programación, escriba “U”, a continuación deje un espacio en blanco y finalmente, el símbolo “Pulsador 1” (entre comillas).

Salga de la fila pulsando Entrar. El cursor saltará a la nueva fila.

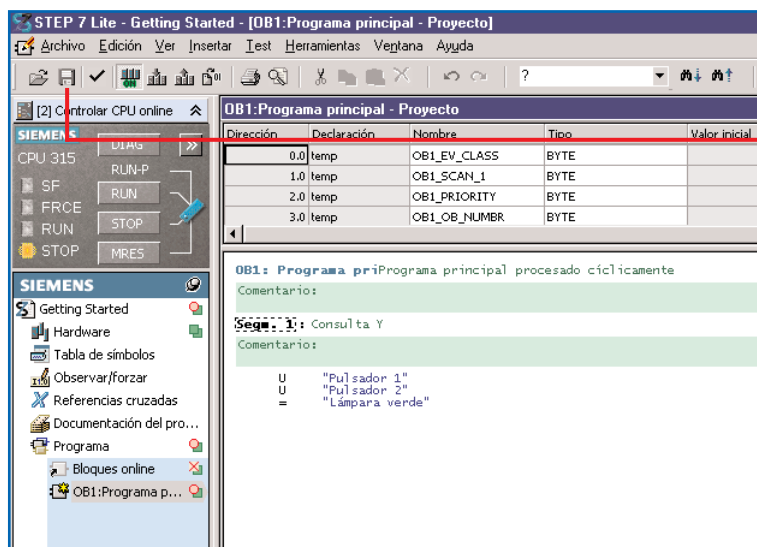
En la siguiente fila, vuelva a escribir “U” y, esta vez, haga clic con el botón derecho del ratón en el área de entrada.

En el menú emergente que aparece al pulsar la tecla derecha del ratón, seleccione **Insertar símbolo** y elija en la lista “Pulsador 2”.

En la siguiente fila introduzca el signo “=” y, a continuación, “Lámpara verde” con el teclado o el menú emergente.



En una fila no tiene que empezar necesariamente por el principio. En cuanto pulse Entrar, STEP 7 Lite ordenará las instrucciones una bajo la otra, formando columnas.



La consulta U (Y) quedará programada.

11 Cuando no quede ningún texto marcado en rojo, podrá guardar las entradas haciendo clic en el icono del disquete.

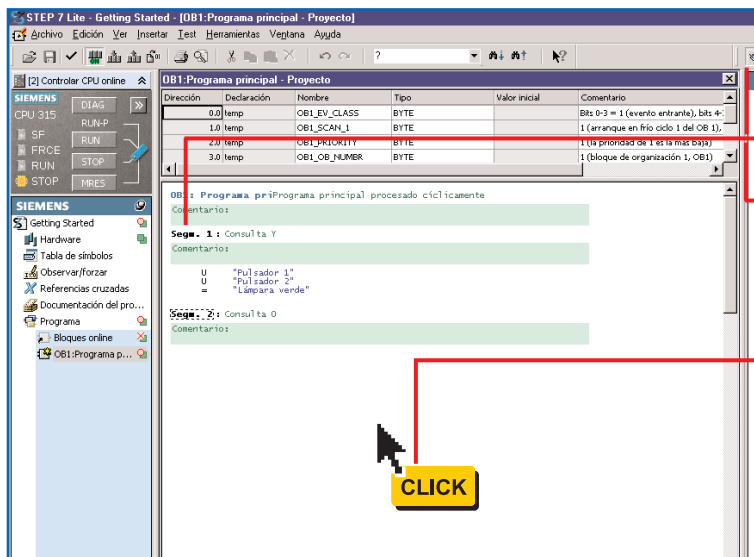
No se guardarán únicamente las entradas del OB 1, sino todos los elementos del proyecto.

12 Seleccione el comando de menú **Edición > Aplicar** para guardar los datos de configuración en un archivo temporal (se guardará el contenido de la ventana activa). Recomendamos guardar los datos de este modo en caso de que necesite recuperar la configuración anterior. Si ha ido guardando los datos de forma temporal y cierra el proyecto, aparecerá un mensaje preguntando si desea guardar los cambios.



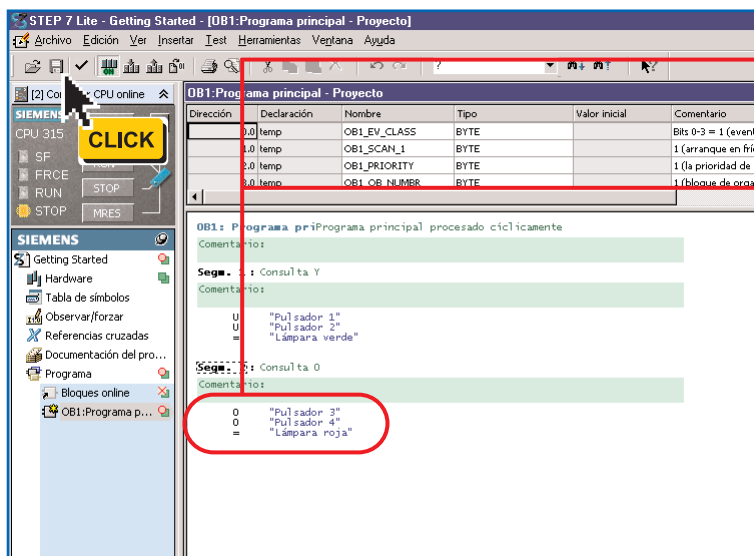
Los símbolos aparecerán marcados en rojo cuando, por ejemplo, el símbolo no se encuentre en la tabla de símbolos o haya un error de sintaxis.

En tal caso, no podrá guardar los datos y en la parte inferior de la ventana del editor aparecerá un mensaje de error con indicaciones sobre el procedimiento correcto que se debe seguir.



Programar una instrucción O en AWL

- 1 Seleccione el segmento 1.
- 2 Inserte un nuevo segmento. Puede hacerlo a través del icono en la barra de herramientas, con el botón derecho del ratón o con CTRL R.
- 3 Vuelva a hacer clic en el área de entrada.

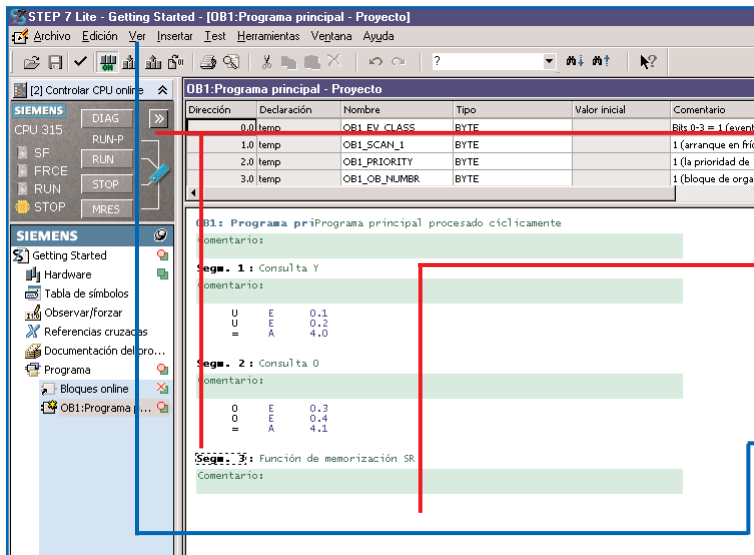


- 4 Al igual que hicimos con AND, escriba "O" y el símbolo "Pulsador 3".
- 5 Complete la consulta O y guarde las entradas.

6.15

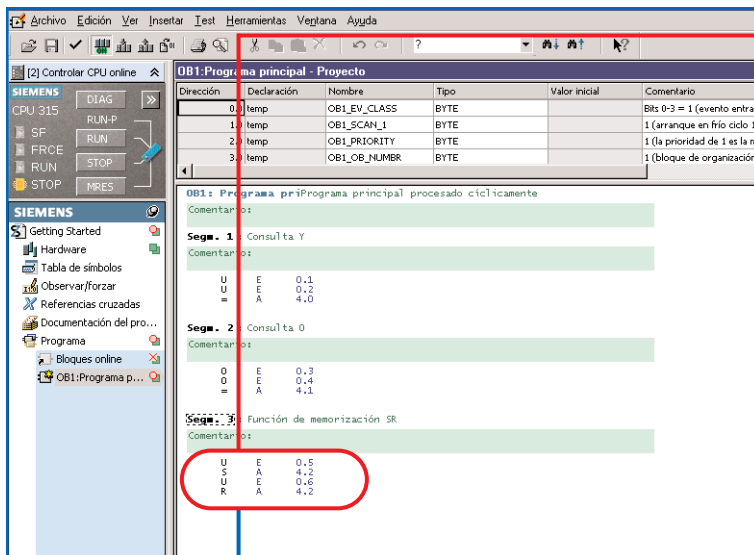


Asigne nombres sencillos y autoexplicativos a los segmentos. De esta forma, la búsqueda resultará más fácil al desplazarse con las barras de desplazamiento por distintos programas. Al desplazarse aparecerán los nombres.



Programar una instrucción de memoria en AWL

- 1 Seleccione el segmento 2 e inserte un nuevo segmento.
- 2 Vuelva a hacer clic en el área de entrada.
- 3 Con el comando de menú **Ver > Mostrar > Representación simbólica** podrá pasar de la representación simbólica a la absoluta, tal y como se muestra a la izquierda.



- 4 En la primera fila, escriba la instrucción U con el nombre simbólico "Automático ON". Complete la instrucción de memoria tal y como sigue:

Simbólica:

U "Automático ON"
S "Modo automático"
U "Manual ON"
R "Modo automático"

- 5 Guarde estas entradas con el comando **Archivo > Guardar**.

- 6 Si en el paso 3 cambió a la representación absoluta, escriba lo siguiente:

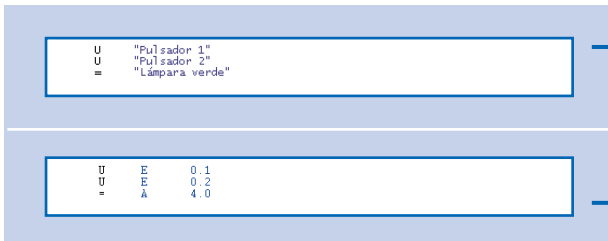
Absoluta:

U E 0.5
S A 4.2
U E 0.6
R A 4.2

Personalizar el interface de programación

Utilice comandos de menú para configurar el interface de programación a su voluntad en STEP 7 Lite.

Menú **Ver** - Ejemplos:



1

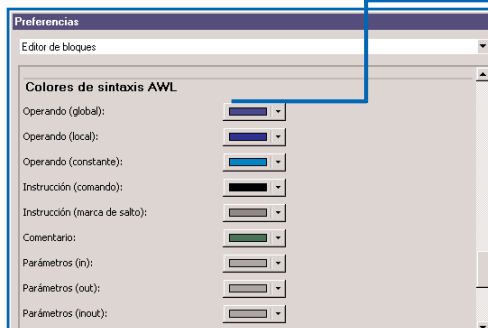
Direccionamiento simbólico en AWL:
Active **Ver > Mostrar > Representación simbólica**.

2

Direccionamiento absoluto en AWL:
Desactive **Ver > Mostrar > Representación simbólica**.

Cambio de lenguaje de programación
Ver > KOP/FUP/AWL

Comando de menú **Herramientas > Preferencias** - Ejemplo:



3

Cambiar los colores de las instrucciones:
Herramientas > Preferencias > AWL > Colores de sintaxis AWL

7

Cierre el bloque con el botón **Cerrar** de la ventana.

6.17

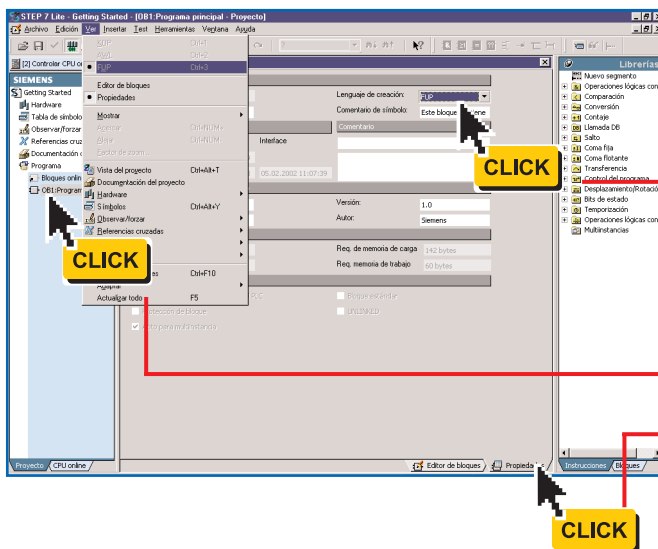


El comando de menú **Herramientas > Preferencias** ofrece muchas opciones para modificar la representación de STEP 7 Lite mediante colores, tipos de letra, etc.

Programar el OB 1 en FUP



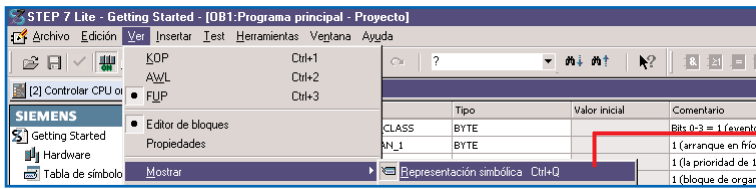
6.18



A continuación programaremos una función Y (U), una función O (O) y una función de memoria en FUP (diagrama de funciones).

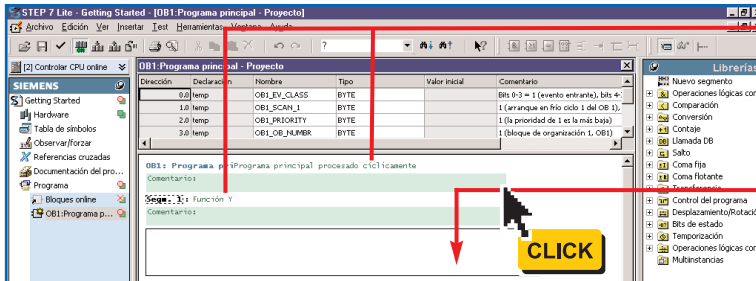
Elija el lenguaje de programación en el que vaya a programar el OB 1 y con el que lo abrirá posteriormente:

- 1 Haga doble clic en el **OB 1**.
- 2 Haga clic en **Propiedades**.
- 3 Seleccione **FUP**. El OB 1 se abrirá a partir de ahora en FUP.
- 4 Salga del diálogo **Propiedades**. En el menú **Ver** ahora también aparecerá seleccionado **FUP**.



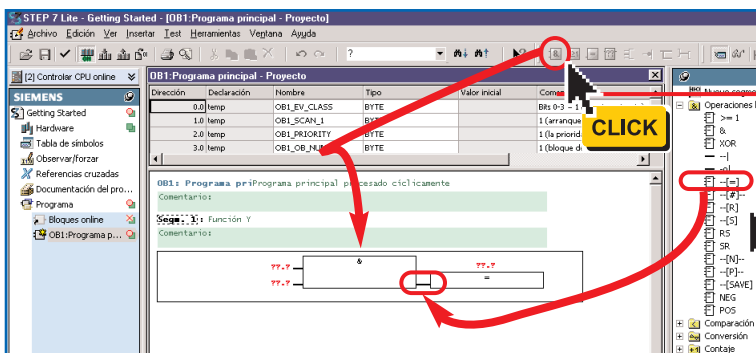
Programar una función U (Y) en FUP

En el menú **Ver**, seleccione la representación simbólica.



En el **OB 1**, indique “Programa principal procesado cíclicamente”. En el **Segmento 1**, escriba “Función Y”.

Haga clic en el área de entrada.

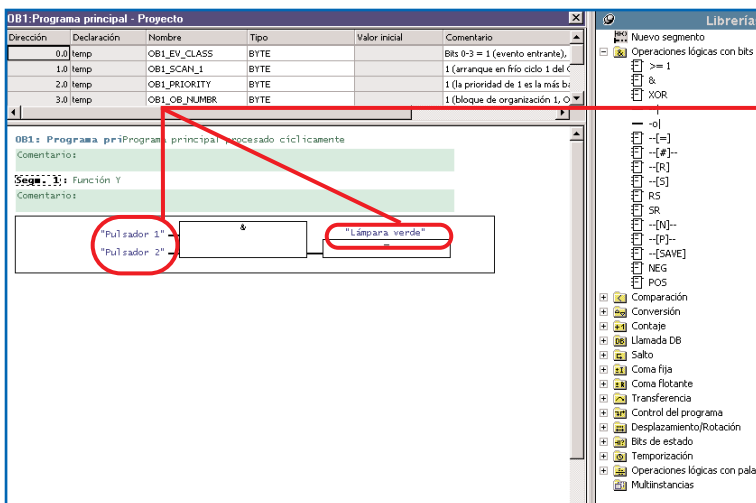


Inserte dos elementos de programa de distinto tipo:

Haga clic en el símbolo de la puerta Y. Se insertará inmediatamente.

Mueva la asignación mediante Arrastrar y soltar hasta la zona enmarcada en el gráfico. Si tiene problemas para encontrarla, la asignación se encuentra siempre bajo el cuadro Y.

Si no desea mover la asignación con Arrastrar y soltar, marque la zona rodeada y haga doble clic en la asignación.

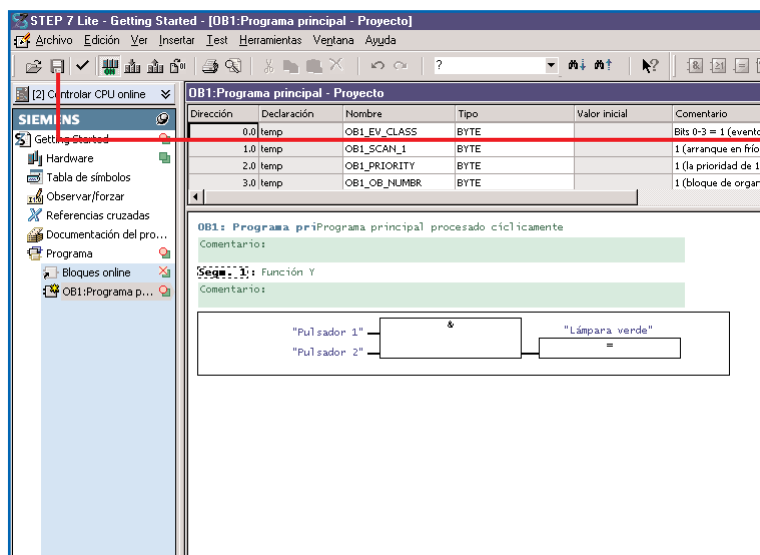


En la función Y todavía falta el direccionamiento:

Haga clic en ??? e indique el nombre simbólico “Pulsador 1” (entre comillas). También puede hacer clic en los signos de interrogación, seleccionar con el botón derecho del ratón **Insertar símbolo** e insertar el nombre desde la lista.

Para la segunda entrada del cuadro Y, indique el nombre “Pulsador 2”.

En la asignación, indique el nombre “Lámpara verde”.



La función Y (U) quedará programada.

Cuando ya no quede ningún símbolo marcado en rojo, puede guardar las entradas haciendo clic en el icono del disquete.

No se guardarán únicamente las entradas del OB 1, sino todos los elementos del proyecto.

Seleccione el comando de menú **Edición > Aplicar** para guardar los datos de configuración en un archivo temporal (se guardará el contenido de la ventana activa).

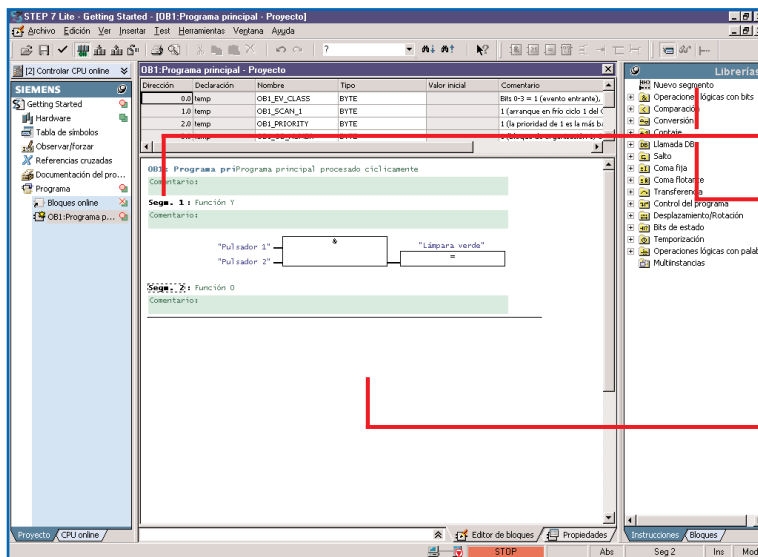
Recomendamos guardar los datos de este modo en caso de que necesite recuperar la configuración anterior.

Si ha ido guardando los datos de forma temporal y cierra el proyecto, aparecerá un mensaje preguntando si desea guardar los cambios.



Los símbolos aparecerán marcados en rojo cuando, por ejemplo, el símbolo no se encuentre en la tabla de símbolos o haya un error de sintaxis.

No será posible guardar los datos y en la parte inferior de la ventana del editor aparecerá un mensaje de error con indicaciones sobre el procedimiento correcto que se debe seguir.



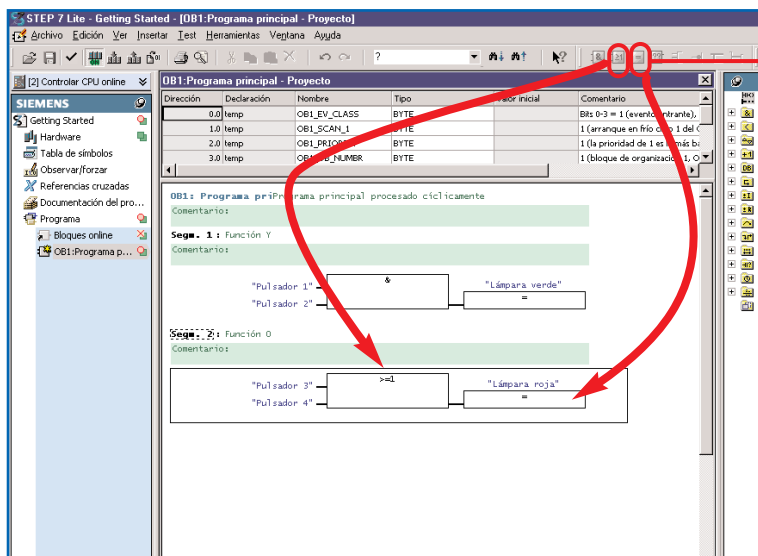
Programar una función O en FUP

1 Seleccione el segmento 1.

2 Inserte un nuevo segmento.

Puede hacerlo a través del icono de la barra de herramientas, con el botón derecho del ratón o con CTRL R.

3 Vuelva a seleccionar el área de entrada.



4 Inserte la función O y una asignación.

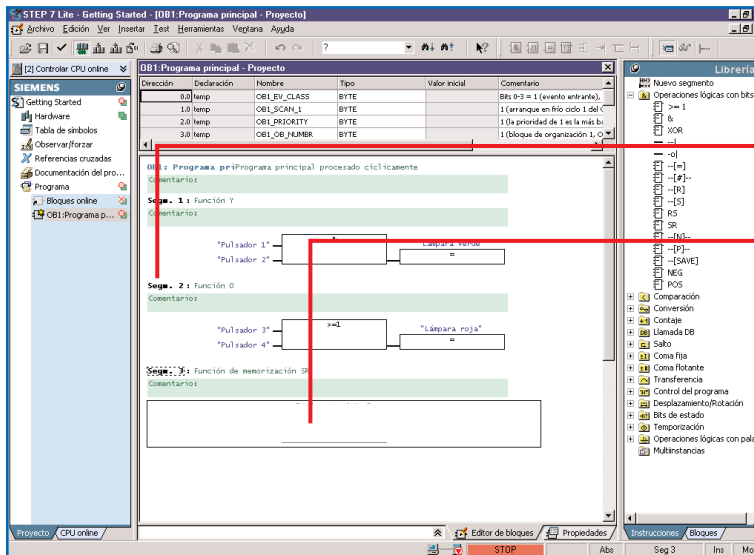
5 Ahora sólo falta indicar el direccionamiento. Escriba los nombres tal y como se indica en la figura de la izquierda. Guarde los datos.

6.21



Asigne nombres sencillos y autoexplicativos a los segmentos. De esta forma, la búsqueda resultará más fácil al desplazarse con la barra de desplazamiento derecha por distintos programas. Al desplazarse aparecerán los nombres.

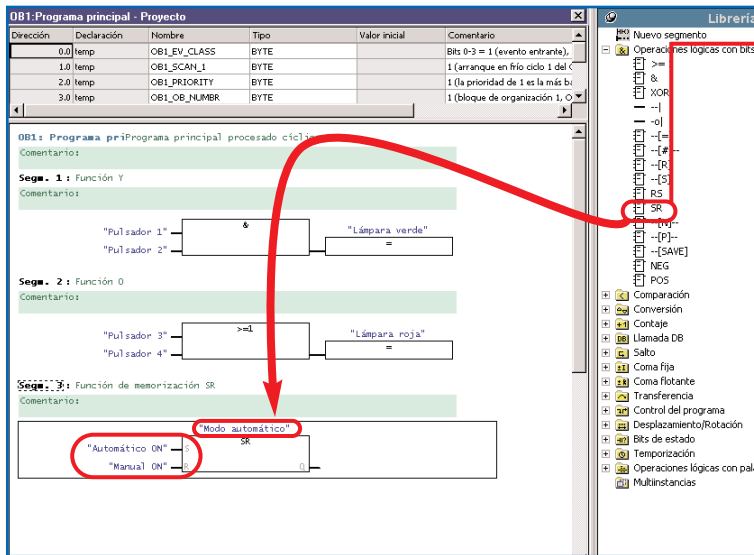
Primeros pasos de programación



Programar una función de memoria en FUP

1 Seleccione el segmento 2 e inserte un nuevo segmento.

2 Vuelva a seleccionar el área de entrada.



3 En la vista de instrucciones, acceda a través de las **Operaciones lógicas** hasta el **Elemento SR**.

Inserte el elemento.

4 Asigne los siguientes nombres simbólicos:

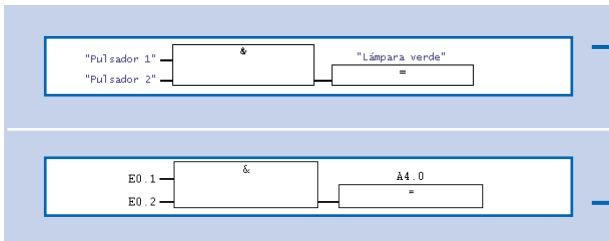
- Activar “Automático ON”,
- Desactivar “Manual ON”,
- Flipflop de activación/desactivación “Modo automático”.

5 Guarde las entradas con **Archivo > Guardar**.

Personalizar el interface de programación

Utilice comandos de menú para configurar el interface de programación a su voluntad en STEP 7 Lite.

Menú **Ver** – Ejemplos:



1

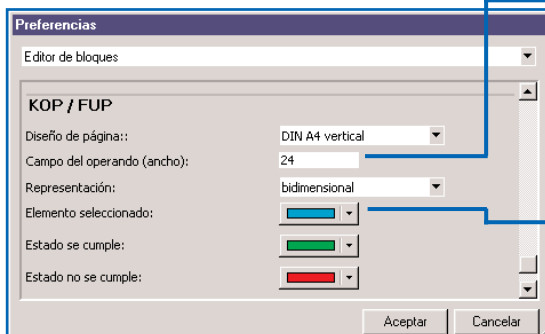
Direccionamiento simbólico en FUP:
Active **Ver > Mostrar > Representación simbólica**

2

Direccionamiento absoluto en KOP:
Desactive **Ver > Mostrar > Representación simbólica**

Cambiar el lenguaje de programación
Ver > KOP/FUP/AWL

Menú **Herramientas > Preferencias** – Ejemplos:



3

Ajustar un cambio de línea en el direccionamiento simbólico entre el carácter 10° y el 24°:

Herramientas > Preferencias > KOP/FUP > > Campo del operando (ancho)

4

Cambiar los colores:
Herramientas > Preferencias > KOP/FUP > > Elemento seleccionado

6

Cierre el bloque con el botón **Cerrar** de la ventana.

6.23



El menú **Herramientas** ofrece muchas opciones para cambiar la representación de STEP 7 Lite mediante colores, tipos de letra, campos de operandos, etc.

Referencias cruzadas

Saltar al punto de aplicación

Filtrar operandos

Definir filtros

Mostrar las referencias cruzadas de un operando

Mostrar la jerarquía de llamada de los bloques

Mostrar los bits, bytes, temporizadores y contadores utilizados

Listar los operandos utilizados y saltar al punto de aplicación

CLICK

Operand	Símbolo	Bloque	Símbolo del bloque	Segmento	Línea	Acceso	Lenguaje	Operación
A4.0	Lámpara verde	OB1	Programa principal	1		W	KOP	=
A4.1	Lámpara roja	OB1	Programa principal	2		W	KOP	=
A4.2	Modo automático	OB1	Programa principal	3		W	KOP	5
A4.2	Modo automático	OB1	Programa principal	3		W	KOP	R
E0.1	Pulsador 1	OB1	Programa principal	1		R	KOP	U
E0.2	Pulsador 2	OB1	Programa principal	1		R	KOP	U
E0.3	Pulsador 3	OB1	Programa principal	2		R	KOP	O
E0.4	Pulsador 4	OB1	Programa principal	2		R	KOP	O
E0.5	Automático ON	OB1	Programa principal	3		R	KOP	U
E0.6	Manual ON	OB1	Programa principal	3		R	KOP	U

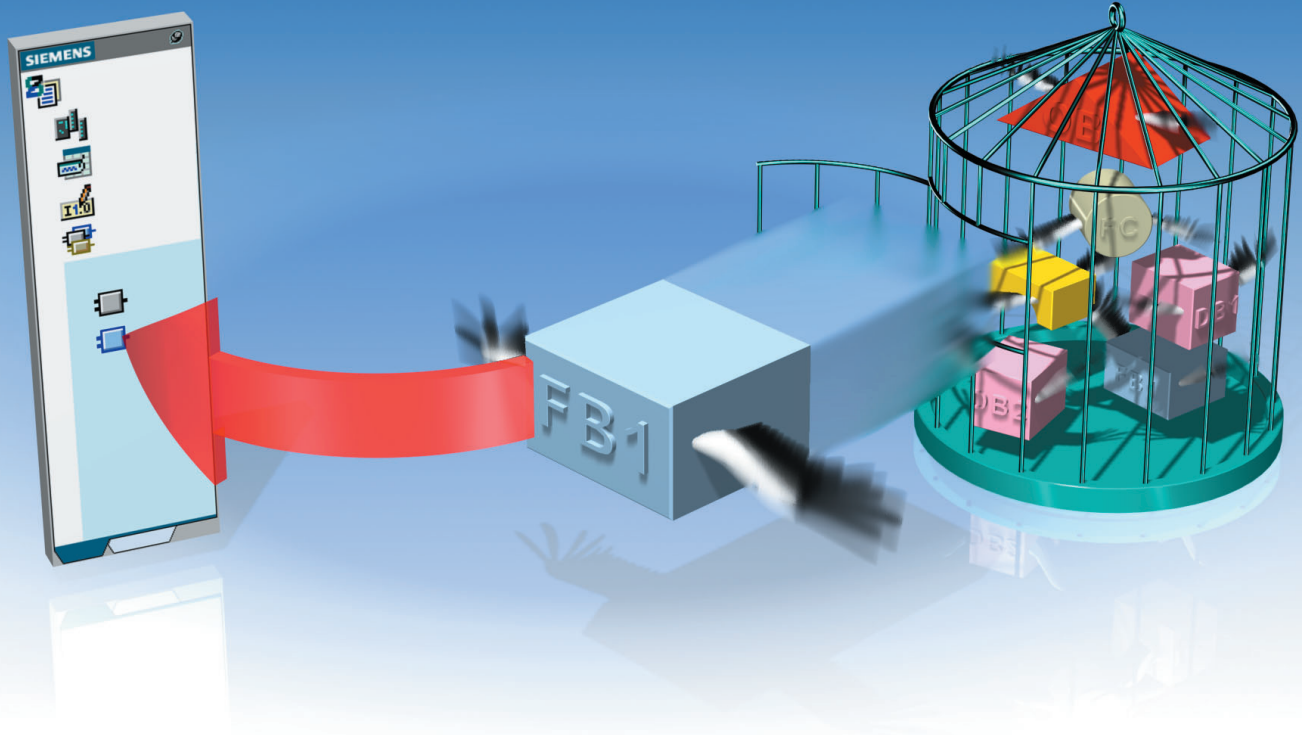
Utilice las vistas "Referencias cruzadas", "Operandos utilizados" y "Estructura del programa" para obtener una visión de conjunto del uso de los operandos, áreas de memoria, bloques etc. Para acceder a las referencias cruzadas haga clic en el símbolo "Referencias cruzadas" representado en la ventana del proyecto.

7

Uso de bloques de función



Crear y abrir bloques de función (FB)



7.2

Se suelen utilizar bloques de función para programar una función cuando hay que guardar resultados intermedios, ajustes y estados operativos hasta la siguiente llamada.

Por este motivo, se denominan también “bloques con memoria”.

En el proyecto de ejemplo, vamos a programar el bloque de función FB 1 con el nombre simbólico “Motor”.

Utilice el lenguaje de programación en el que haya programado el OB 1.

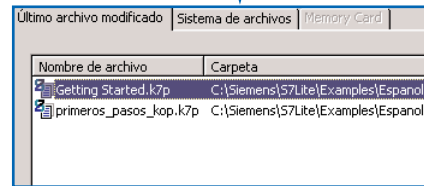
Si quiere continuar este capítulo, copie la tabla de símbolos en el proyecto “Getting Started” (consulte la página 5.5).



Se abre el cuadro de diálogo de selección de proyectos

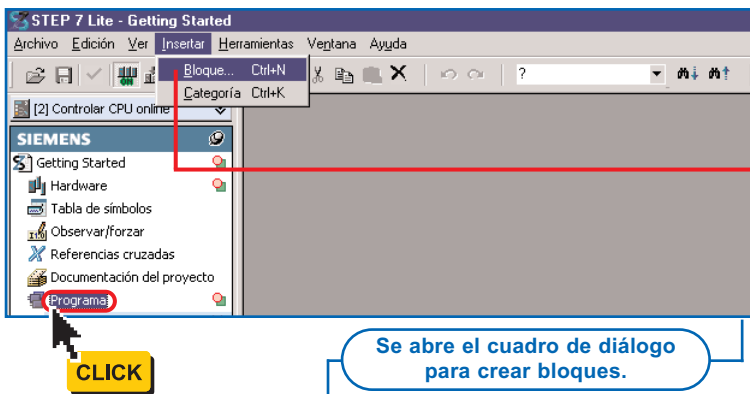
1

Si es necesario, abra STEP 7 Lite.



2

En el cuadro de diálogo **Abrir proyecto**, haga doble clic en el proyecto "Getting started".



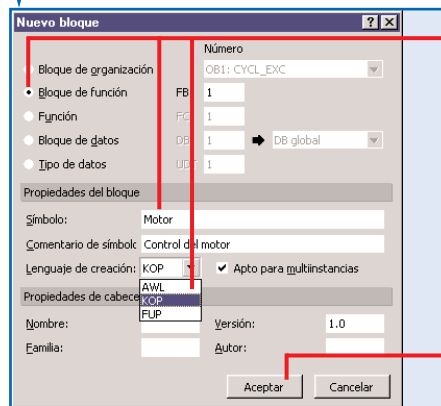
3

En la ventana del proyecto, haga clic en Programa.

4

Seleccione el comando de menú **Insertar > Bloque** o abra el menú emergente mediante el botón derecho del ratón y seleccione el comando **Nuevo > Bloque**.

Se abre el cuadro de diálogo para crear bloques.



5

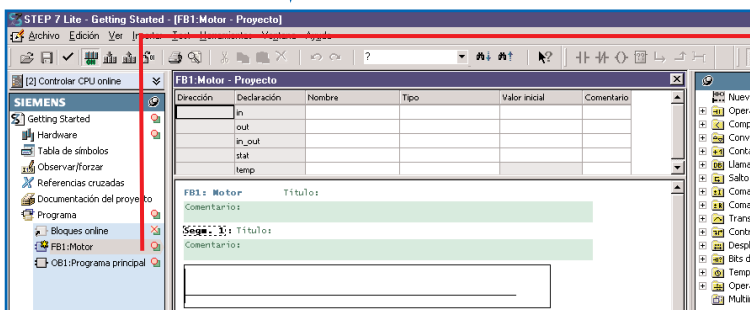
Seleccione la opción **Bloque de función**.

En el campo **Lenguaje de creación**, seleccione el lenguaje en el que desee programar el bloque.

6

Confirme los datos con **Aceptar**.

Se inserta el bloque y se abre inmediatamente.



7

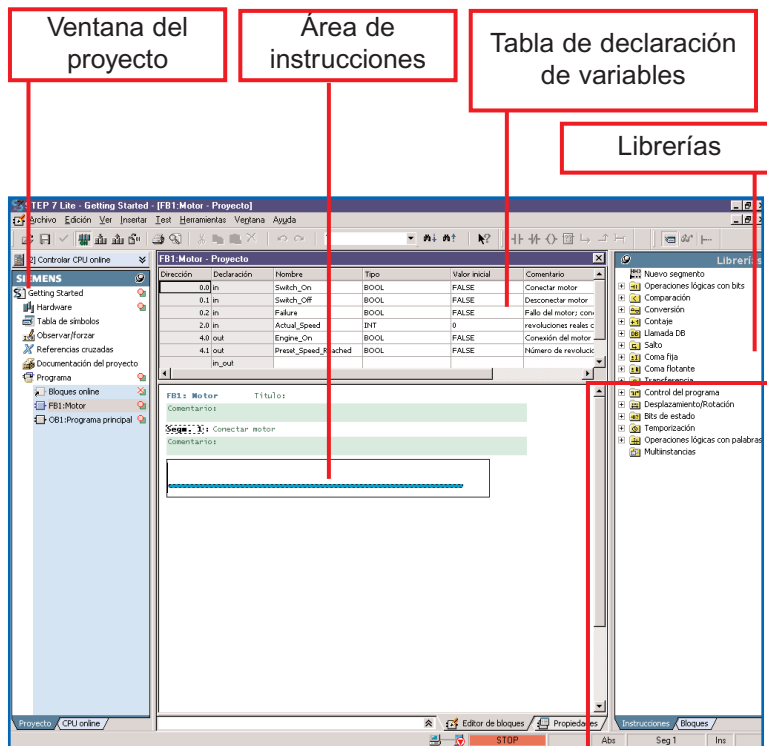
El bloque nuevo se inserta en la ventana de proyectos y se abre inmediatamente.

Completar la tabla de declaración de variables

Aquí aprenderá cómo programar un bloque de función para controlar y supervisar un motor diesel y un motor de gasolina con un bloque de datos respectivamente.

Todas las señales “específicas del motor” se transfieren como parámetros de bloque entre el bloque de organización y el bloque de función y, por tanto, se deben incluir en la tabla de declaración de variables como parámetros de entrada y salida (declaración “in” y “out”). De este modo se define el “interface” para la llamada del FB en el programa.

Antes de comenzar a programar en el área de instrucciones, complete la tabla de declaración de variables del FB.



Dirección	Declaración	Nombre	Tipo	Valor inicial	Comentario
0.0	in	Switch_On	BOOL	FALSE	Conectar motor
0.1	in	Switch_Off	BOOL	FALSE	Desconectar motor
0.2	in	Failure	BOOL	FALSE	Fallo del motor; conduce a una desconexión
2.0	in	Actual_Speed	INT	0	revoluciones reales del motor
4.0	out	Engine_On	BOOL	FALSE	Conexión del motor
4.1	out	Preset_Speed_Reached	BOOL	FALSE	Número de revoluciones alcanza
	in_out				
6.0	stat	Preset_Speed	INT	1500	Número de revoluciones solicitado
	temp				

Introduzca las variables de la figura en la tabla de declaración de variables.

Para ello, haga clic en una celda y acepte el nombre y el comentario como muestra la figura.

Seleccione el **tipo** de datos en el menú emergente **Datos simples** (botón derecho del ratón).

Con la tecla Entrar se mueve el cursor a la columna siguiente o se inserta una fila nueva.



1. Editar la tabla de declaración de variables

Para indicar el nombre de los parámetros de bloque en la tabla de declaración de variables sólo se admiten letras, números y guiones bajos.

2. ¿Necesita ayuda para editar la tabla de declaración de variables?

Pulse **F1 > Contenido > Programar bloques > Crear bloques lógicos y Editar la tabla de declaración de variables** para obtener más información.

3. Consejos para el capítulo siguiente

En los capítulos siguientes programaremos un proceso de conexión y desconexión y una vigilancia de la velocidad de giro.

¿Cuándo se conecta y se desconecta el motor?

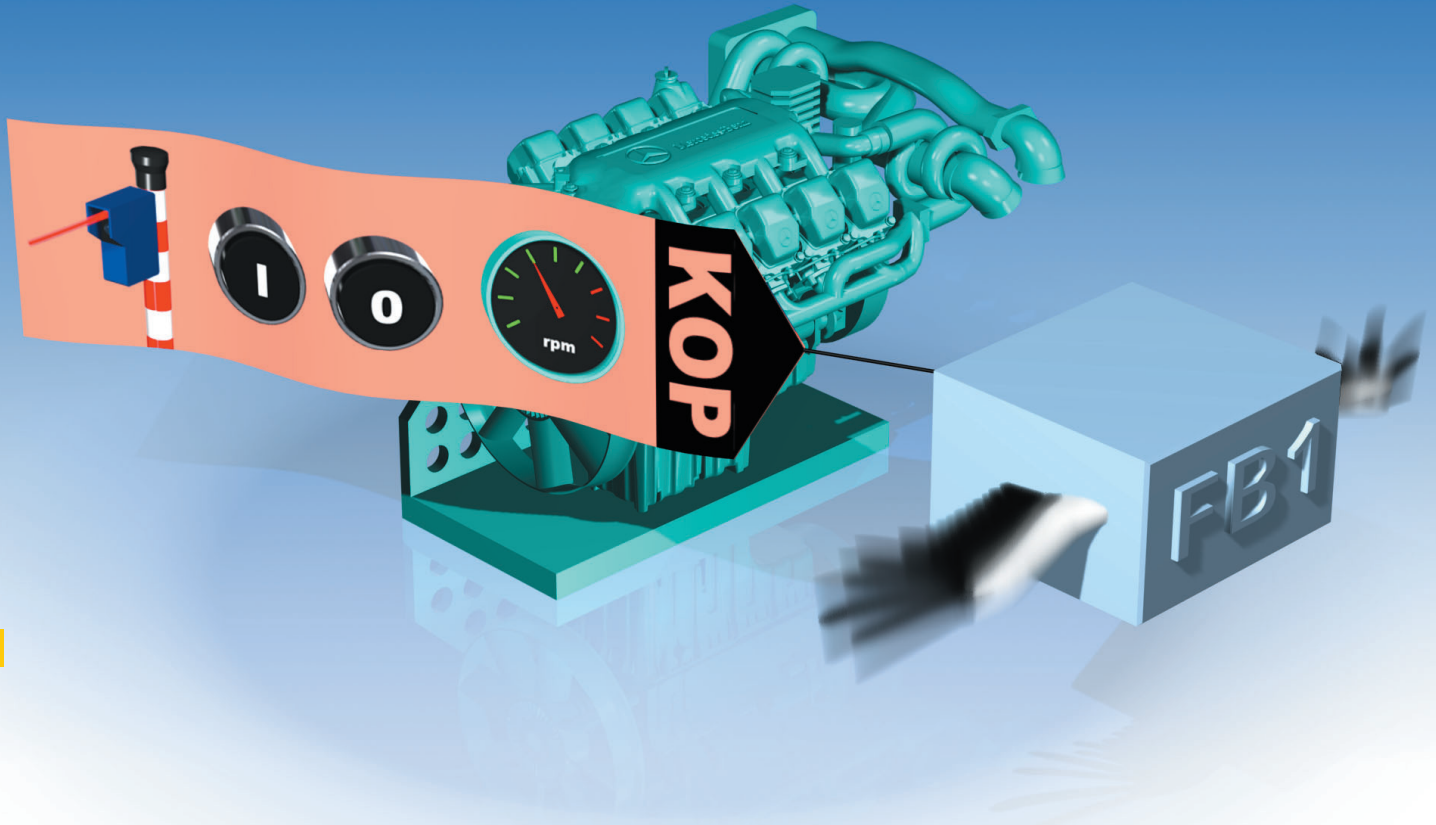
- El motor arranca cuando la variable #Switch_ON toma el estado de señal "1" y la variable "Modo automático" toma el estado de señal "0".
- El motor se detiene cuando la variable #Switch_OFF toma el estado de señal "1" o la variable #Failure toma el estado de señal "0".

¿Cómo supervisa el comparador la velocidad del motor?

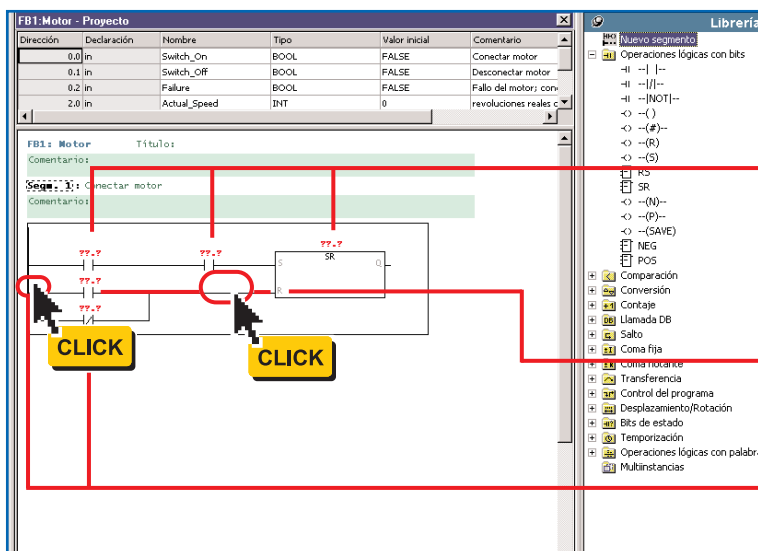
- El comparador compara las variables #Actual_Speed y #Preset_Speed y asigna el resultado a la variable #Preset_Speed_Reached (estado de señal 1).

Dependiendo del lenguaje de programación seleccionado para el OB 1, consulte las siguientes páginas:
para KOP: páginas 7.6 - 7.7
para AWL: páginas 7.8 - 7.9
para FUP: páginas 7.10 - 7.11.

Programar FB en KOP

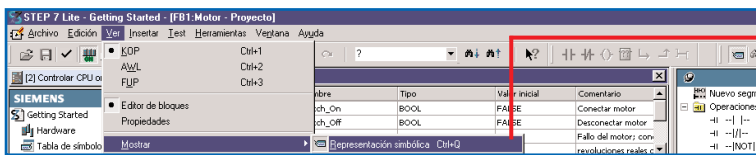


7.6

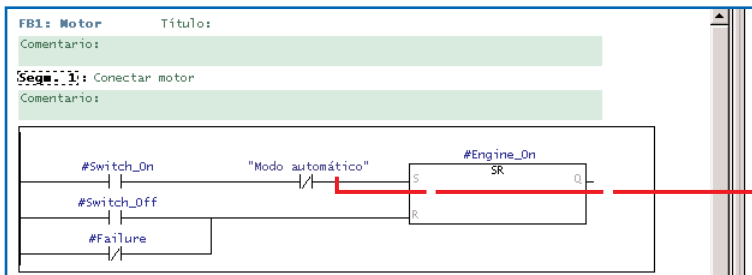


Programar el proceso de conexión y desconexión del motor

- 1 En el segmento 1, inserte un contacto normalmente abierto, un contacto normalmente cerrado y un elemento SR en serie mediante **Librerías > Instrucciones**.
- 2 Marque después el circuito anterior a la entrada R e inserte otro contacto normalmente abierto.
- 3 Marque la barra de alimentación izquierda anterior al contacto normalmente cerrado paralelo al contacto normalmente abierto.



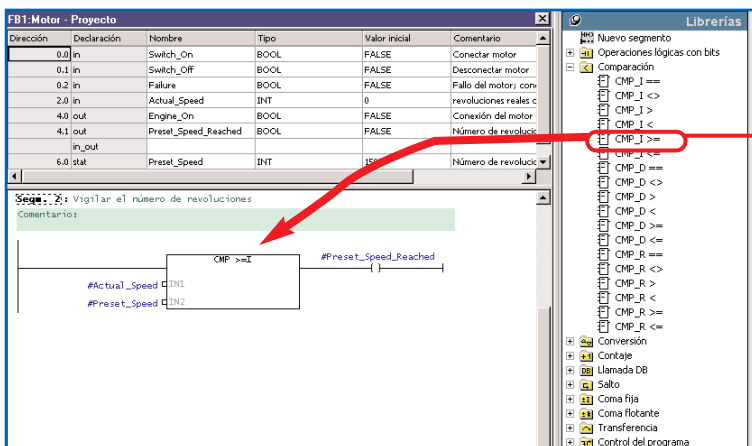
4 Compruebe si está activada la representación simbólica.



5 Marque todos los ??? e introduzca los nombres correspondientes de la tabla de declaración de variables (# se asigna automáticamente).

6 Introduzca el nombre simbólico “Modo automático” para el contacto normalmente cerrado de la conexión en serie.

Programar la vigilancia de la velocidad de giro



7 Inserte un segmento nuevo y marque el circuito.

8 Navegue por la vista de instrucciones hasta el comparador e inserte **CMP>=I**. Además, inserte una bobina en el circuito.

9 Marque de nuevo los signos de interrogación e indique para la bobina y el comparador los nombres que figuran en la tabla de declaración de variables.

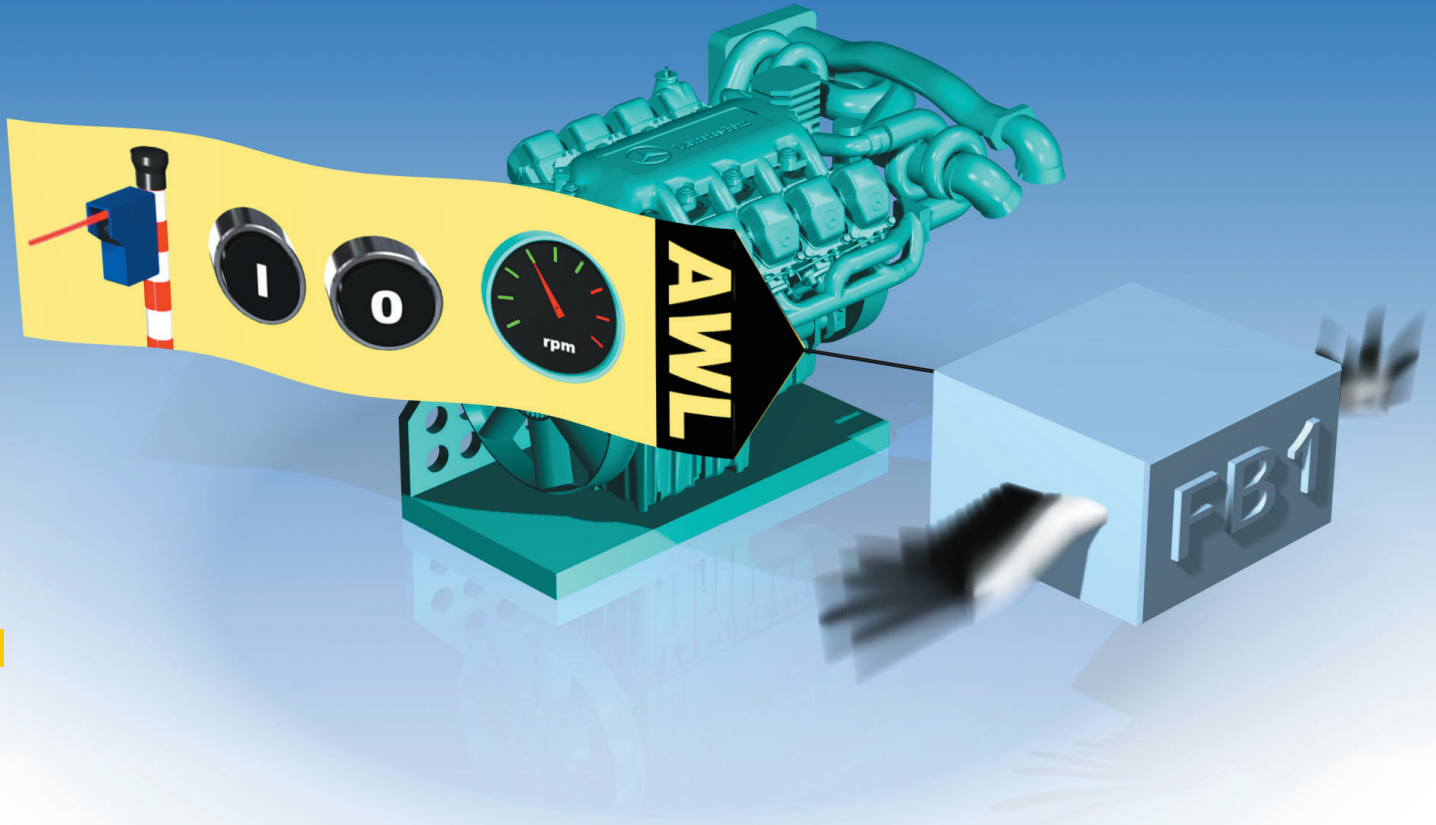
Guarde las modificaciones.



Las variables locales del bloque llevan el signo # y sólo se utilizan en el bloque en el que han sido declaradas. Las variables globales se indican entre comillas. Se definen en la tabla de símbolos y se utilizan en todo el programa. El estado de señal “Modo automático” se define en el OB1 (segmento 3, consulte las páginas 6-10) mediante otro elemento SR y se consulta ahora en el FB1.

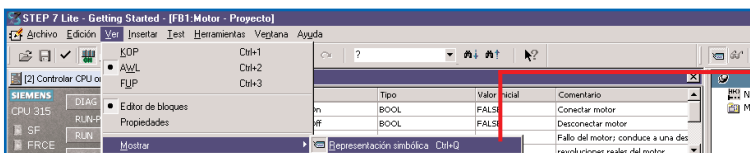
Pulse **F1 > Contenido > Programar bloques > Crear bloques lógicos y Editar instrucciones KOP** para obtener más información.

Programar FB en AWL



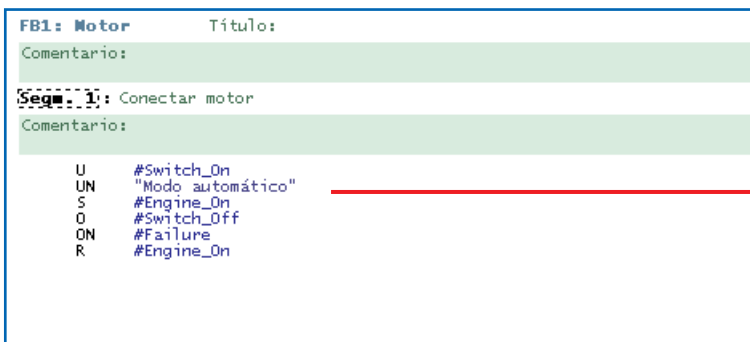
7.8

Programar el proceso de conexión y desconexión del motor



1

Compruebe si está activada la representación simbólica.



2

Introduzca estas instrucciones AWL en el segmento 1.

Programar la vigilancia de la velocidad de giro

```
FB1: Motor      Título:
Comentario:

Segm. 1: Conectar motor
Comentario:

U      #Switch_On
UN     "Modo automático"
S      #Engine_On
O      #Switch_Off
ON     #Failure
R      #Engine_On

Segm. 2: Vigilar el número de revoluciones
Comentario:

L      #Actual_Speed
L      #Preset_Speed
>=I
=      #Preset_Speed_Reached
```

3

Inserte un segmento nuevo e introduzca estas instrucciones AWL.

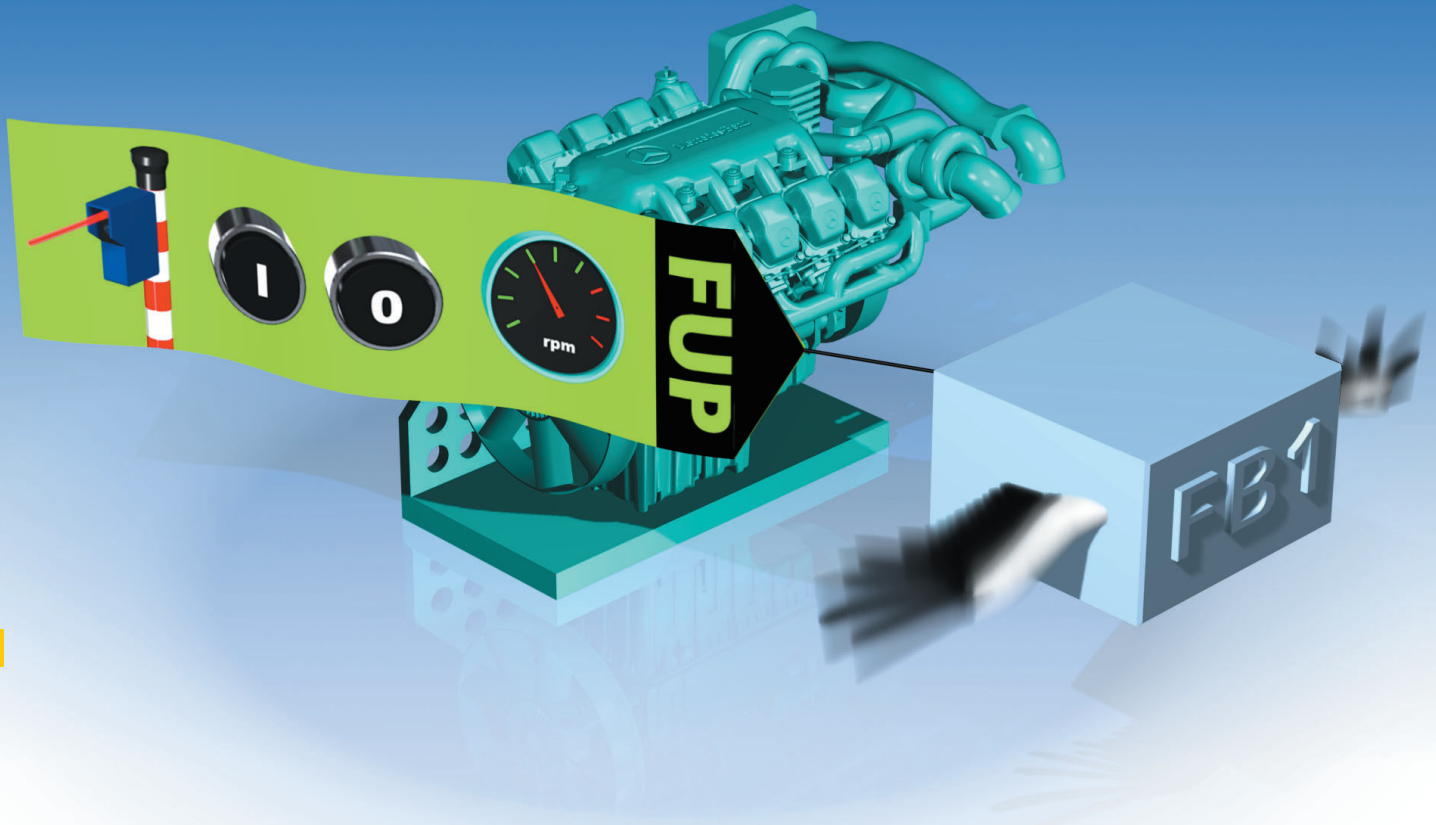
Guarde las modificaciones.



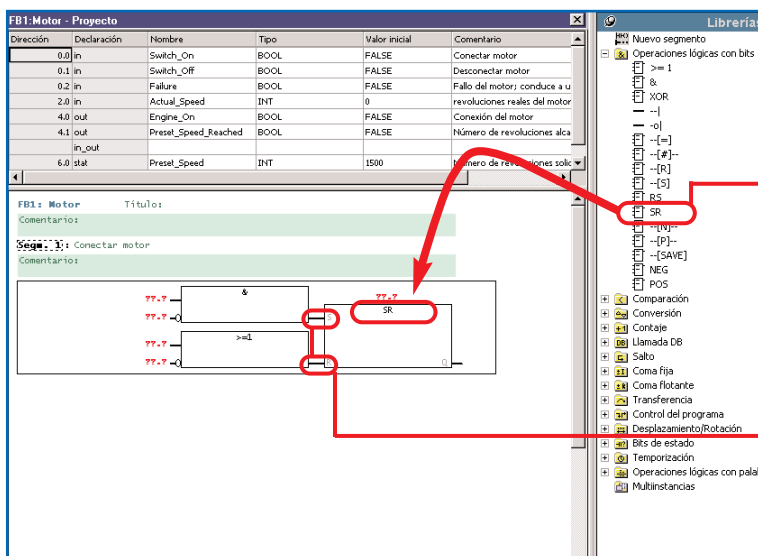
Las variables locales del bloque llevan el signo # y sólo se utilizan en el bloque. Las variables globales se indican entre comillas. Se definen en la tabla de símbolos y se utilizan en todo el programa. El estado de señal “Modo automático” se define en el OB1 (segmento 3, consulte las páginas 6-16) mediante otro elemento SR y se consulta ahora en el FB1.

Pulse **F1 > Contenido > Programar bloques > Crear bloques lógicos y Editar instrucciones AWL** para obtener más información.

Programar FB en FUP

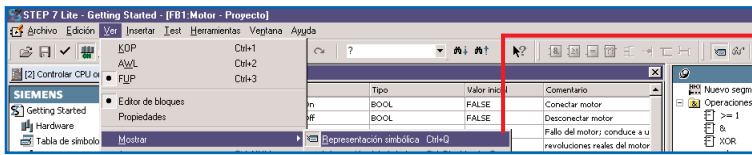


7.10



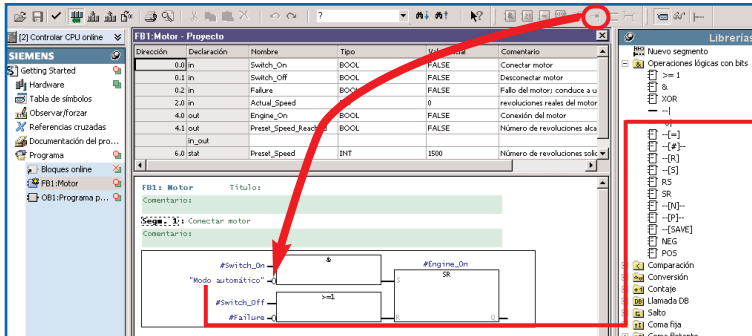
Programar el proceso de conexión y desconexión del motor

- 1 Inserte una función SR en el segmento 1 desde la vista de instrucciones.
- 2 Coloque en la entrada S (Set) un cuadro Y y en la entrada R (Reset) un cuadro O.



4 Compruebe si está activada la representación simbólica.

5 Marque todos los ??? e introduzca los nombres correspondientes de la tabla de declaración de variables (# se asigna automáticamente).



6 Dirige una entrada de la función Y con el nombre simbólico "Modo automático".
Niegue las entradas "Modo automático" y #Failure con el símbolo correspondiente de la barra de herramientas.

Programar la vigilancia de velocidad de giro

Inserte un segmento nuevo y marque el circuito.

Navegue por la vista general de instrucciones hasta el comparador e inserte **CMP>=I**. Dirige las entradas con los nombres de la tabla de declaración de variables.

8 Inserte una asignación de salida en el comparador y dirija la asignación con los nombres de la tabla de declaración de variables.

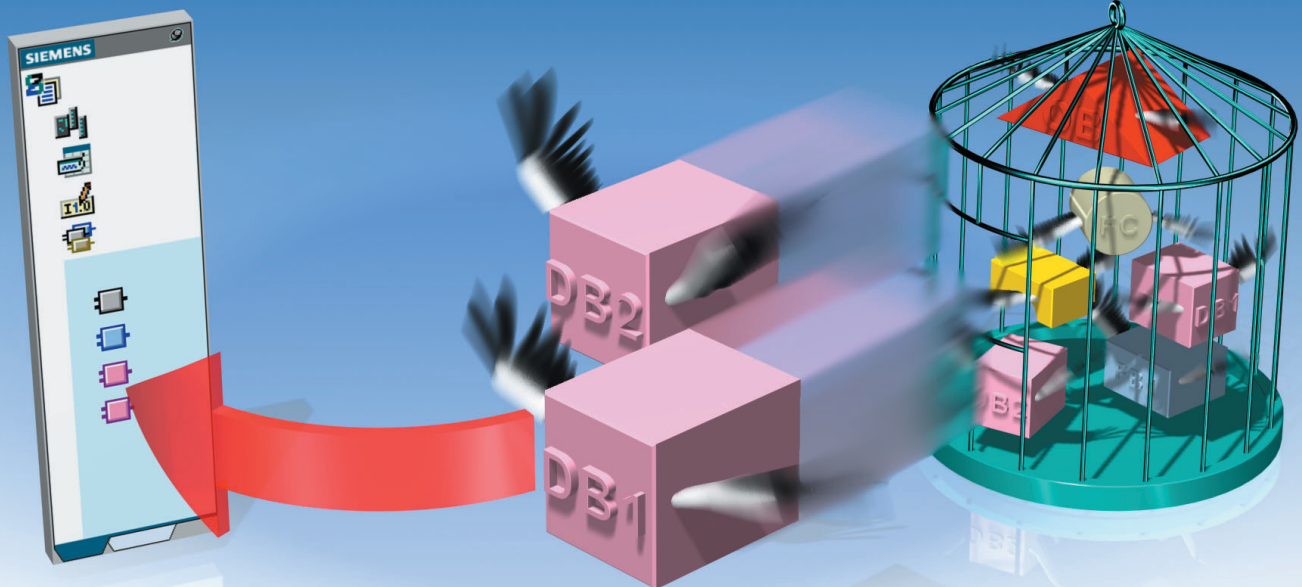
Guarde las modificaciones.



Las variables locales del bloque se identifican mediante # y sólo son válidas en el bloque. Las variables globales aparecen entre comillas. Se definen en la tabla de símbolos y son válidas en todo el programa. El estado de señal "Modo automático" se determina en el OB1 (segmento 3, consulte las páginas 6-22) mediante otro elemento SR y se consulta ahora en el FB1.

Pulse **F1 > Contenido > Programar bloques > Crear bloques lógicos y Editar instrucciones FUP** para obtener más información.

Crear bloques de datos de instancia y modificar valores actuales



7.12

Insertar un bloque de datos

Para poder programar a continuación la llamada (CALL) del FB1 en el OB1, deberá crear el bloque de datos correspondiente.

El FB debe controlar y supervisar un motor de gasolina o diesel. Las distintas velocidades teóricas de los motores se guardan en dos DB separados, donde se modifica el valor actual (#Preset_Speed).

Programando el bloque de función una sola vez de forma centralizada, se reducen los costes de programación.

Nuevo bloque

Número: OB1: CYCL_EXC

• Bloque de organización: FB 2

• Bloque de función: FC 1

• Función: DB 1 → FB1: Motor

• Bloque de datos: UDT 1

Tipo de datos: **Bloque de datos**

Símbolo: Gasolina

Comentario de símbolo: Datos del motor de gasolina

Lenguaje de creación: DB ☒ Apto para multistanancias

Propiedades de cabecera

Nombre: Versión: 1.0

Familia: Autor:

Aceptar Cancelar

1 Haga clic con el botón derecho del ratón en la ventana del proyecto y seleccione la opción **Nuevo > Bloque** del menú emergente. Aparecerá el cuadro de diálogo **Nuevo bloque**.

2 Active la opción **Bloque de datos** e introduzca el FB 1 como FB asignado (igual que en la figura).

3 Confirme los datos con **Aceptar**. El DB1 se insertará en el proyecto "Getting Started" y se abrirá inmediatamente.

Se inserta el bloque

STEP 7 Lite - Getting Started - [DB1:Gasolina - Proyecto]

Vista de datos

Tipo	Valor inicial	Valor actual	Comentario
BOOL	FALSE	FALSE	Conectar motor
BOOL	FALSE	FALSE	Desconectar motor
BOOL	FALSE	FALSE	Fallo del motor: conduce a una desconexión
INT	0	0	Revoluciones reales del motor
INT	FALSE	FALSE	Conexión del motor
INT	1500	1500	Número de revoluciones solicitado

DB1:Gasolina

4 Active la **vista Datos**. El DB sólo se puede modificar en esta vista.

5 Asegúrese de que en la columna **Valor actual** figura "1500" para el motor de gasolina.

Se inserta el bloque.

STEP 7 Lite - Getting Started - [DB2:Diesel - Proyecto]

Vista de datos

Dirección	Declaración	Nombre	Tipo	Valor inicial	Valor actual	Comentario
0.0	statin	Switch_On	BOOL	FALSE	FALSE	Conectar motor
0.1	statin	Switch_Off	BOOL	FALSE	FALSE	Desconectar motor
0.2	statin	Failure	BOOL	FALSE	FALSE	Fallo del motor: conduce a una desconexión
2.0	statin	Actual_Speed	INT	0	0	Revoluciones reales del motor
4.0	statout	Engine_On	BOOL	FALSE	FALSE	Conexión del motor
4.1	statout	Preset_Speed_Reached	BOOL	FALSE	FALSE	Número de revoluciones alcanzado
6.0	stat	Preset_Speed	INT	1500	1200	Número de revoluciones solicitado

DB2:Diesel

6 Inserte de la misma forma un DB 2 "Diesel".

7 Indique "1200" para el motor diesel en la columna **Valor actual**. Guarde las modificaciones.

Para programar la llamada del FB en el OB1, consulte el capítulo sobre KOP, FUP o AWL dependiendo del lenguaje de programación que haya utilizado.



Una vez ajustados los valores actuales concluyen los preparativos necesarios para controlar dos motores con un solo bloque de función. Para controlar más motores, basta con crear más bloques de datos.

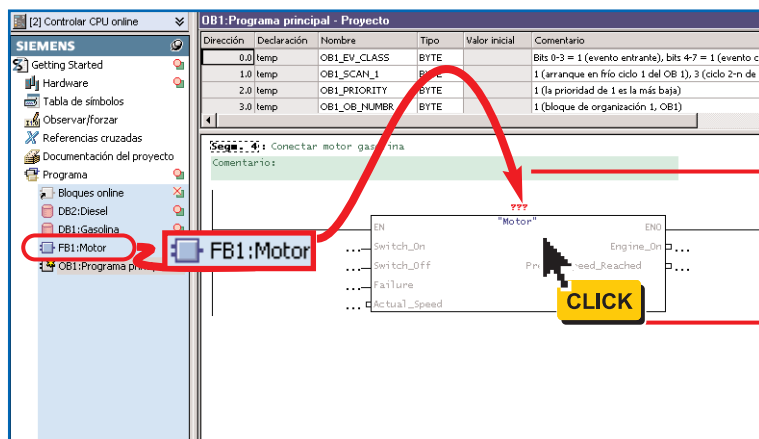
Pulse **F1 > Contenido > Programar bloques > Crear bloques de datos** para obtener más información.

Programar llamadas de bloque en KOP



7.14

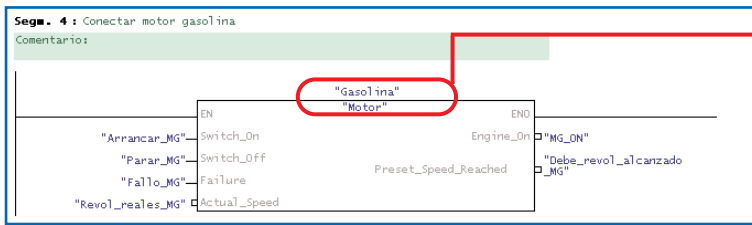
Toda la programación de un FB no tendrá efecto si no es llamado desde el OB1. Por cada llamada del FB se utiliza un bloque de datos diferente (uno por cada motor), y de este modo se controlan ambos motores.



1 Abra el OB1 e inserte el segmento 4.

2 Arrastre el **FB 1** desde la ventana de proyectos hasta el segmento 4 mediante la función Arrastrar y soltar. Se muestran todas las variables específicas del motor.

3 Haga clic en ???. Se abrirá la lista de selección de símbolos.

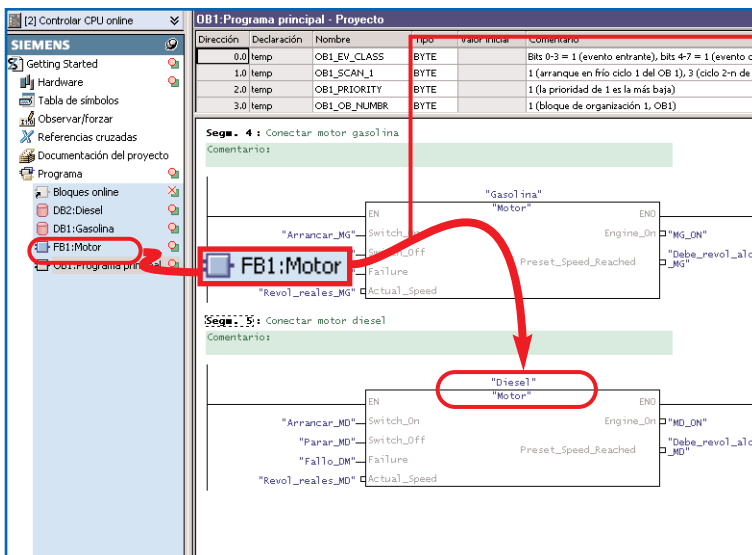


4 Seleccione el bloque de datos "Gasolina".

5 Direccione de forma análoga los demás parámetros del bloque de función con los nombres simbólicos correspondientes.

Las variables de entrada y salida específicas del motor (declaración "in" y "out") se muestran en el FB "Motor". Se asigna una señal "MG_XXX" para el motor de gasolina a las variables.

Todavía falta la llamada del motor diesel.



6 Inserte el segmento 5, arrastre de nuevo el FB1 desde la ventana de proyectos al segmento y programe de forma análoga la llamada del bloque de función "Motor" (FB1) con el bloque de datos "Diesel" (DB 2).

Se asigna una señal "MD_XXX" para el motor diesel a las variables.

Guarde las modificaciones y cierre el bloque.

7.15



Si crea programas estructurados con OBs, FBs y DBs deberá programar la llamada de un bloque subordinado (por ejemplo, FB1) en el bloque de un nivel superior (por ejemplo, OB1). El procedimiento es siempre el mismo. También puede asignar nombres simbólicos a los diversos bloques en la tabla de símbolos (por ejemplo, el FB1 tiene el nombre "Motor" y el DB1, el nombre "Gasolina").

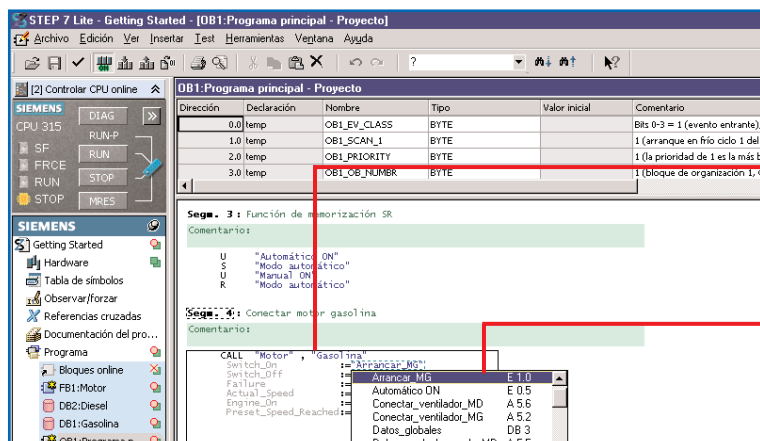
Los bloques programados se pueden imprimir en cualquier momento con el comando **Archivo > Imprimir**. Pulse **F1 > Contenido > Imprimir de la documentación del proyecto** para obtener más información al respecto.

Programar llamadas de bloque en AWL

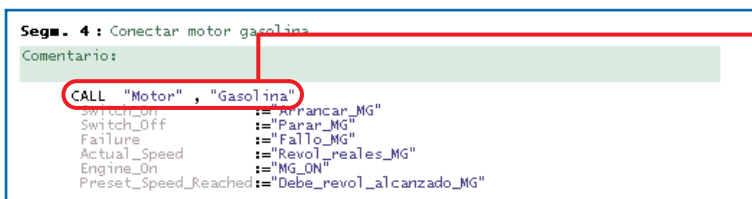


7.16

La programación de un FB no tendrá efecto si no se llama desde el OB1. En cada llamada del FB se utiliza un bloque de datos distinto (uno por cada motor), y de este modo se controlan ambos motores.



- 1 Abra el OB1 e inserte el segmento 4.
- 2 Escriba en el área de instrucciones **CALL "Motor", "Gasolina"** y pulse Entrar. Se muestran todas las variables específicas del motor.
- 3 Haga clic con el botón derecho del ratón detrás de := y seleccione el comando del menú emergente Insertar símbolo. Se abrirá la lista de selección de símbolos.

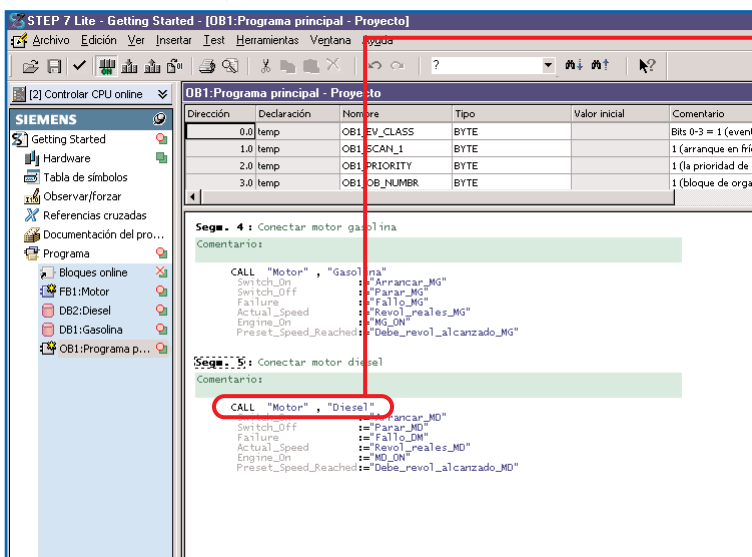


4

Direccione del mismo modo los demás parámetros del bloque de función con los nombres simbólicos correspondientes.

Las variables de entrada y salida específicas del motor (declaración "in" y "out") se muestran en el FB "Motor". Se asigna una señal "MG_xxx" para el motor de gasolina a las variables.

Todavía falta la llamada para el motor diesel.



6

Inserte el segmento 5 y programe de forma análoga la llamada del bloque de función "Motor" (FB1) con el bloque de datos "Diesel" (DB 2).

Se asigna una señal "MD_xxx" para el motor diesel a las variables.

Guarde las modificaciones y cierre el bloque.



Si crea programas estructurados con OBs, FBs y DBs, deberá programar la llamada de un bloque subordinado (por ejemplo, FB1) en el bloque de un nivel superior (por ejemplo, OB1). El procedimiento es siempre el mismo. También puede asignar nombres simbólicos a los diversos bloques en la tabla de símbolos (por ejemplo, el FB1 tiene el nombre "Motor" y el DB1, el nombre "Gasolina").

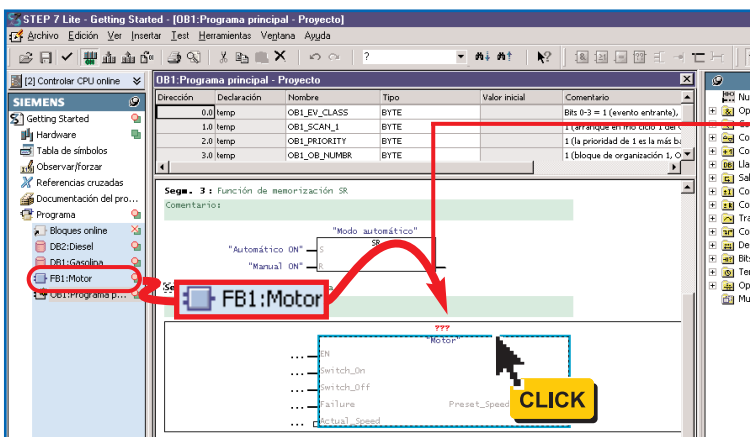
Los bloques programados en cualquier momento con el comando **Archivo > Imprimir**. Pulse **F1 > Contenido > Imprimir la documentación del proyecto** para obtener más información al respecto.

Programar llamadas de bloque en FUP

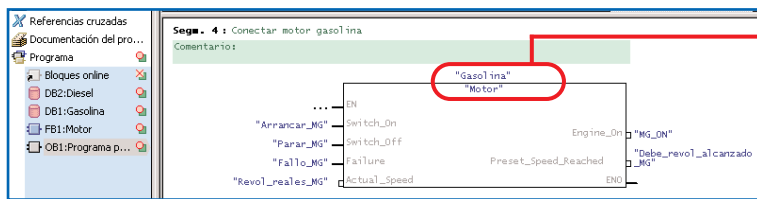


7.18

La programación de un FB no tendrá efecto si no se llama desde el OB1. En cada llamada del FB se utiliza un bloque de datos distinto (uno por cada motor), y de este modo se controlan ambos motores.



- 1 Abra el OB1 e inserte el segmento 4.
- 2 Arrastre el **FB 1** desde la ventana del proyecto hasta el segmento 4 mediante la función Arrastrar y soltar. Se mostrarán todas las variables específicas del motor.
- 3 Haga clic en ???. Se abrirá la lista de selección de símbolos.

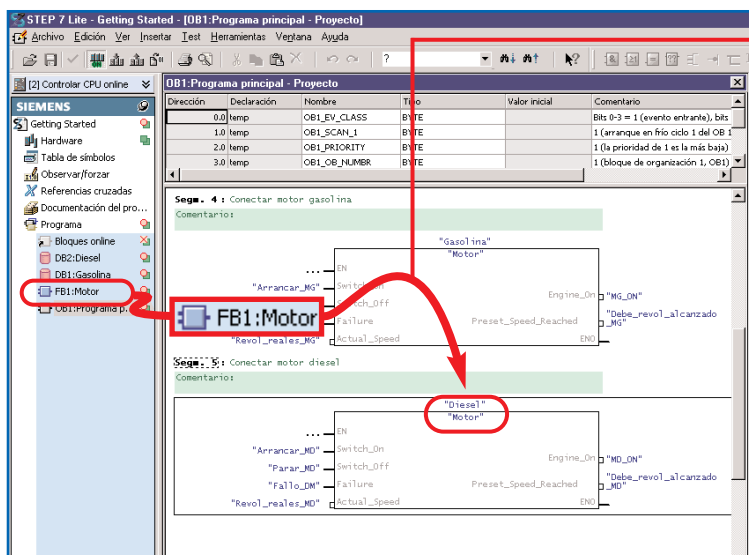


4 Seleccione el bloque de datos "Gasolina".

5 Direccione del mismo modo los demás parámetros del bloque de función con los nombres simbólicos correspondientes.

Las variables de entrada y salida específicas del motor (declaración "in" y "out") se muestran en el FB "Motor". Se asigna una señal "MG_XXX" para el motor de gasolina a las variables.

Todavía falta la llamada para el motor diesel.



6 Inserte el segmento 5, arrastre de nuevo el FB1 desde la ventana de proyectos al segmento y programe de forma análoga la llamada del bloque de función "Motor" (FB1) con el bloque de datos "Diesel" (DB 2).

Se asigna una señal "MD_XXX" para el motor diesel a las variables.

Guarde las modificaciones y cierre el bloque.



Si crea programas estructurados con OBs, FBs y DBs deberá programar la llamada de un bloque subordinado (por ejemplo, FB1) en el bloque de un nivel superior (por ejemplo, OB1). El procedimiento es siempre el mismo. También puede asignar nombres simbólicos a los diversos bloques en la tabla de símbolos (por ejemplo, el FB1 tiene el nombre "Motor" y el DB1, el nombre "Gasolina").

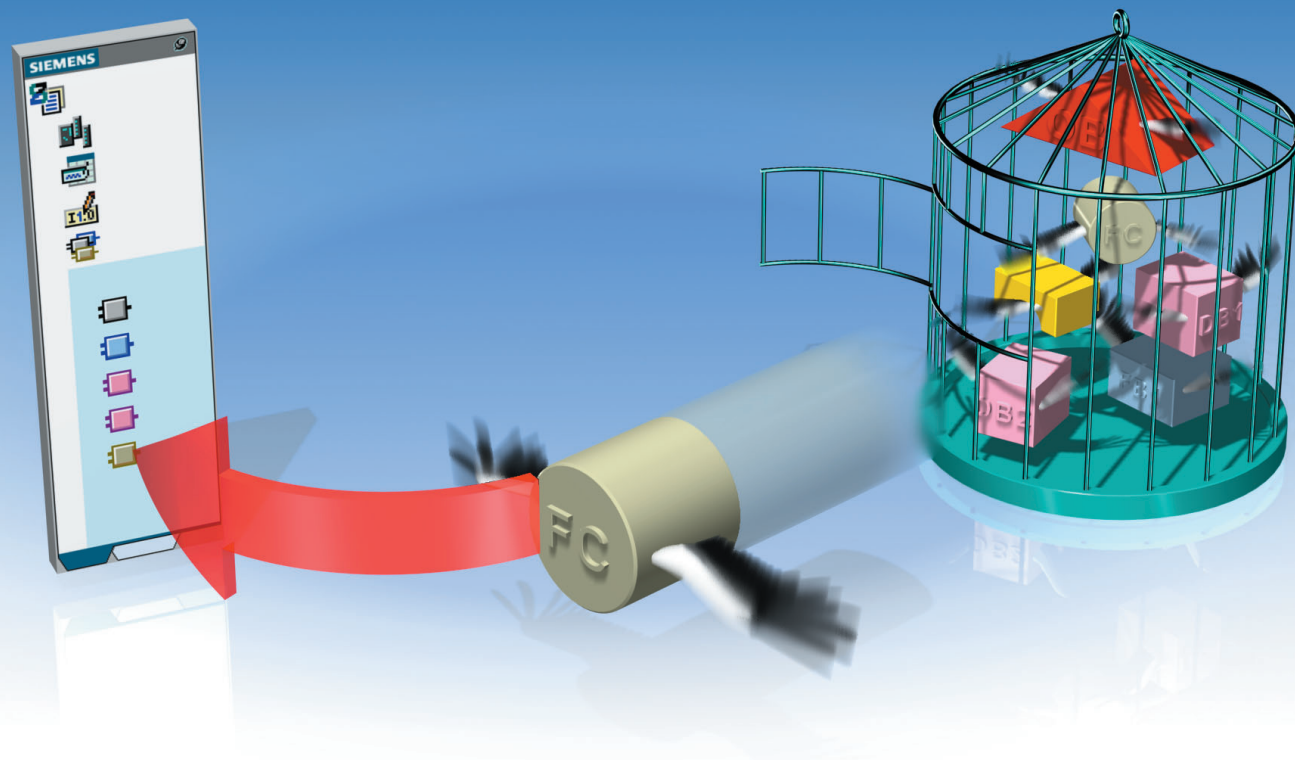
Los bloques programados se pueden imprimir en cualquier momento con el comando **Archivo > Imprimir**. Pulse **F1 > Contenido > Imprimir la documentación del proyecto** para obtener más información al respecto.

8

Uso de funciones



Crear y abrir funciones (FC)



8.2

Las funciones se utilizan para programar una función cuando no es necesario guardar resultados intermedios, ajustes o estados operativos hasta la siguiente llamada. Por este motivo, se denominan también “bloques sin memoria”.

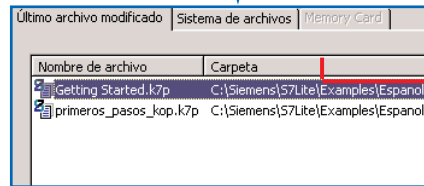
Si quiere continuar este capítulo, deberá copiar primero la tabla de símbolos en el proyecto “Getting Started” (consulte la página 5.5).



Se abre el cuadro de diálogo de selección de proyectos

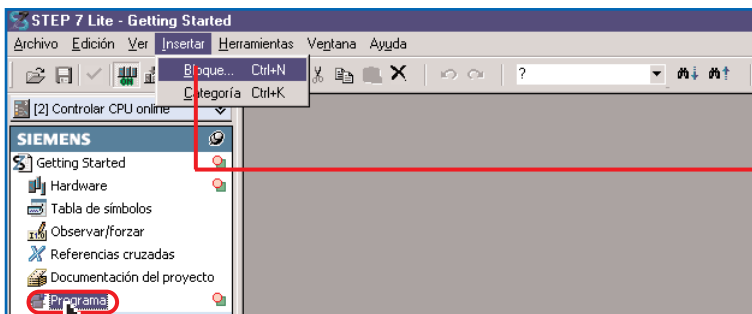
1

Si es necesario, abra STEP 7 Lite.



2

En el cuadro de diálogo **Abrir proyecto**, haga clic en el proyecto "Getting Started".



3

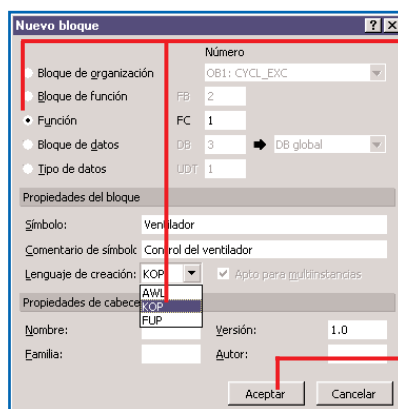
En la ventana de proyectos, haga clic en Programa.

4

Seleccione el comando de menú **Insertar > Bloque** o abra el menú contextual mediante el botón derecho del ratón y seleccione el comando **Nuevo > Bloque**.

CLICK

Se abre el cuadro de diálogo para crear bloques.



5

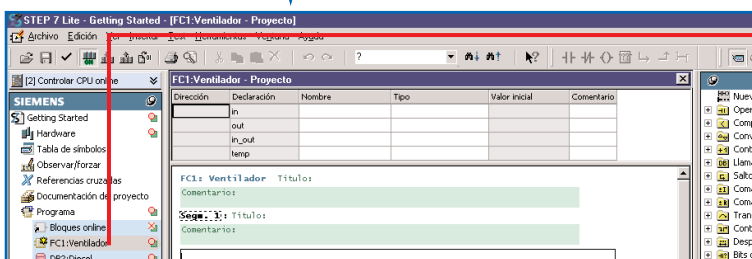
Seleccione la opción **Función**.

En el campo **Lenguaje de creación**, seleccione el lenguaje en el que haya creado el proyecto "Getting Started".

6

Confirme los datos con **Aceptar**.

Se inserta el bloque y se abre inmediatamente.



7

El bloque nuevo se inserta en la ventana del proyecto y se abre inmediatamente.



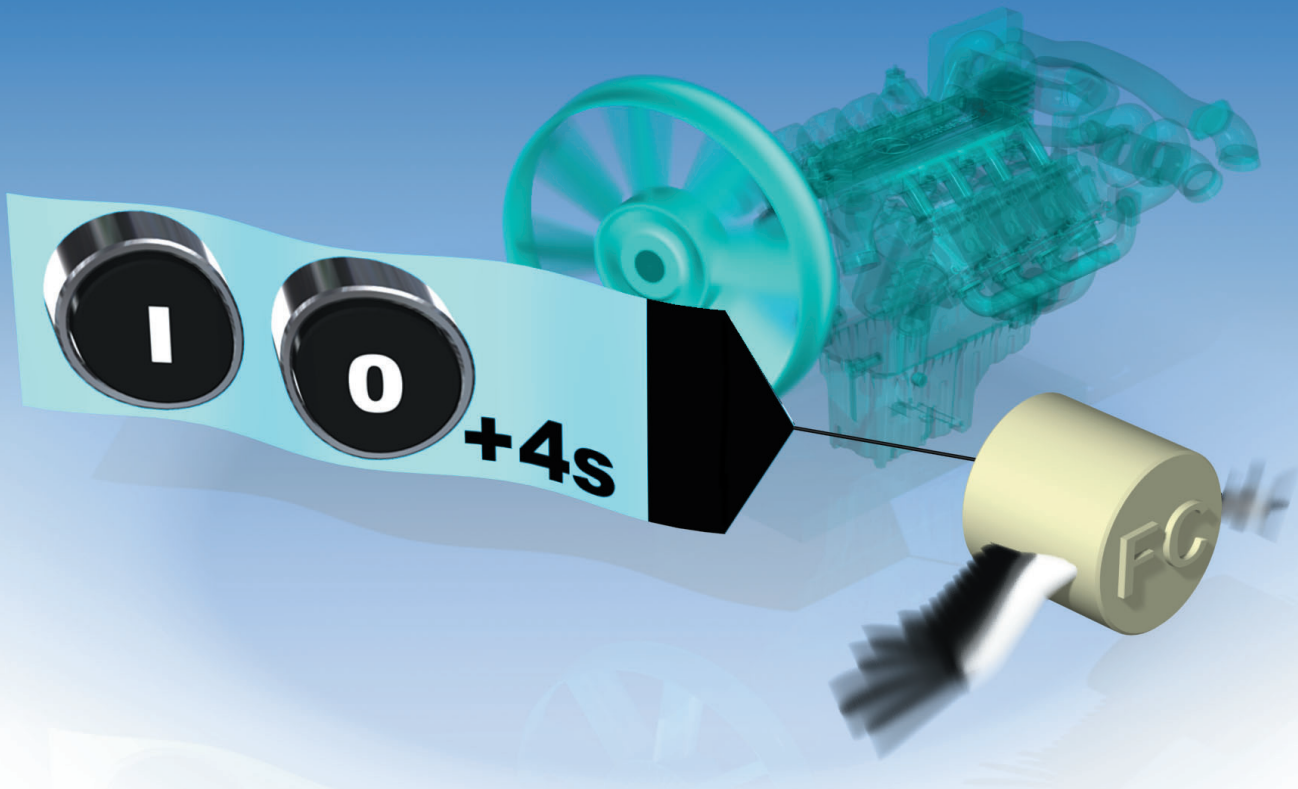
Al contrario de lo que ocurre con los bloques de función, en una función no se pueden definir datos estáticos en la tabla de declaración de variables.

Para programar la función se puede recurrir, como es habitual, a los nombres simbólicos de la tabla de símbolos.



Para más información, pulse **F1 > Contenido > Nociones básicas para diseñar programas > Bloques en el programa de usuario.**

Programar una función



8.6

En el ejemplo siguiente programaremos una función de temporización. La función de temporización hace que se active un ventilador al mismo tiempo que se enciende un motor. El ventilador seguirá en marcha hasta cuatro segundos después de apagarse el motor (retardo a la desconexión).

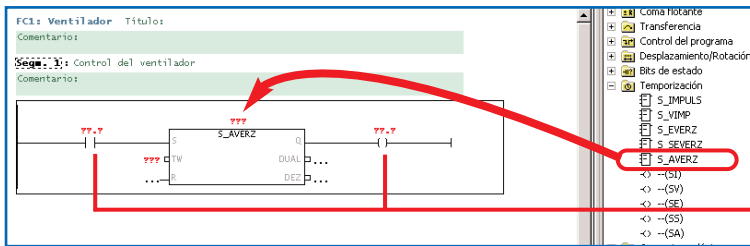
Tabla de declaración de variables

Dirección	Declaración	Nombre	Tipo	Valor inicial	Comentario
0.0	in	Engine_On	BOOL		Señal para la conexión del motor
2.0	in	Timer_Function	TIMER		Función de temporización utilizada para el retardo a la desconexión
4.0	out	Fan_On	BOOL		Señal para la conexión del ventilador
	in_out				
	temp				

1

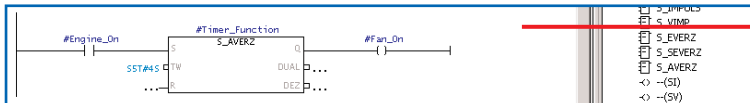
De mismo modo que con el bloque de función, introduzca primero los parámetros de entrada y salida de la función (declaración “in” y “out”) en la tabla de declaración de variables. Seleccione el tipo **TIMER** (temporizador) del menú emergente **Parámetros** (botón derecho del ratón).

Programar la función de temporización en KOP

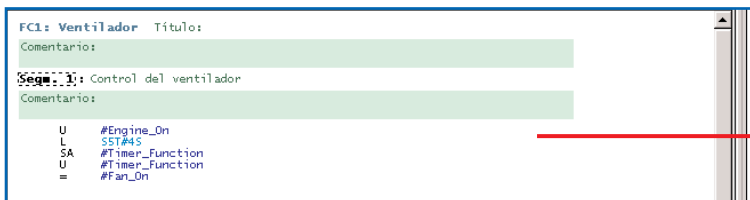


2 Inserte en el segmento 1 el elemento **S_AVERZ** que encontrará en **Librerías > Instrucciones > Temporizadores**.

3 Inserte además un contacto normalmente abierto y una bobina.

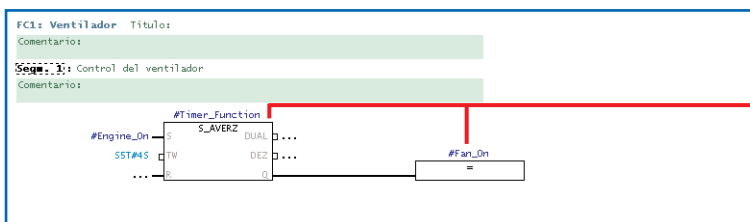


4 Introduzca en los ??? los nombres de la tabla de declaración de variables. Se indicarán automáticamente con el signo #. Introduzca además TW S5T#4s para el valor de temporización. Guarde las modificaciones y cierre el bloque.



Programar la función de temporización en AWL

2 Introduzca las instrucciones indicadas a la izquierda en la figura. Guarde las modificaciones y cierre el bloque.



Programar la función de temporización en FUP

2 Al igual que en KOP: inserte las instrucciones de la librería indicadas a la izquierda en el segmento, complete todos los ??? e indique la constante para el valor de tiempo. Guarde las modificaciones y cierre el bloque.



Con el parámetro de entrada “#Motor_on” se inicia la “#Funcion_de_tiempo”. Más tarde, durante la llamada en el OB1, se le asignarán una vez los parámetros del motor de gasolina y otra vez, los parámetros del motor diesel (por ejemplo T1 para “Marcha_MG”).

Llamar una función en el OB 1

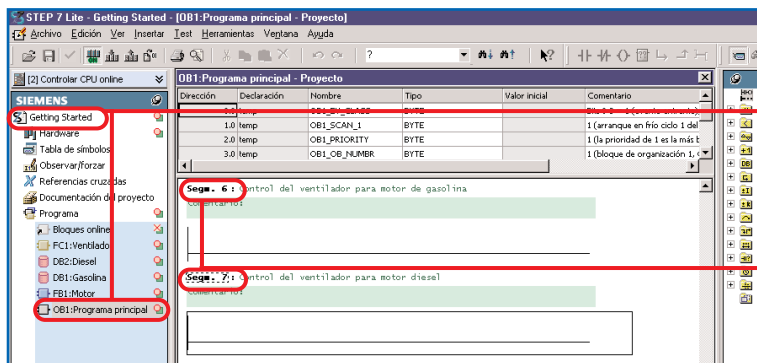


8.8

La llamada de la función FC1 se realiza del mismo modo que la llamada del bloque de función en el OB 1. A todos los parámetros de la función se les asignan en el OB 1 los operandos correspondientes del motor de gasolina o diesel.



En una instrucción de STEP 7 Lite, un operando es la parte que ordena a la CPU lo que debe hacer. Se puede direccionar de forma absoluta o simbólica.

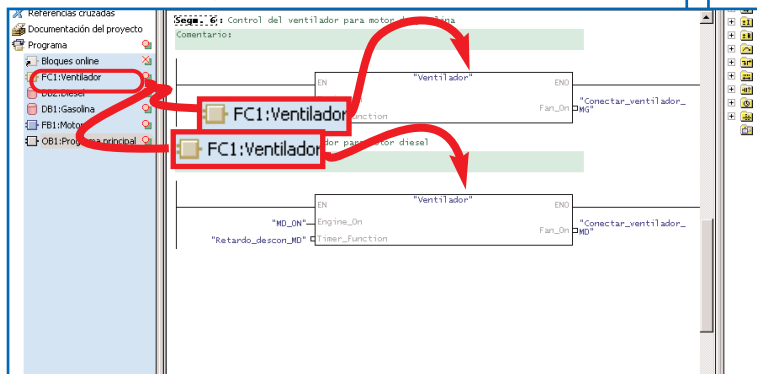


Abrir el OB 1

1 Abra el proyecto “Getting Started” creado en KOP, FUP o AWL y abra el OB 1.

2 Inserte el **segmento 6** para el motor de gasolina y el **segmento 7** para el motor diesel.

Una vez terminada la programación, se verá así en KOP, ...

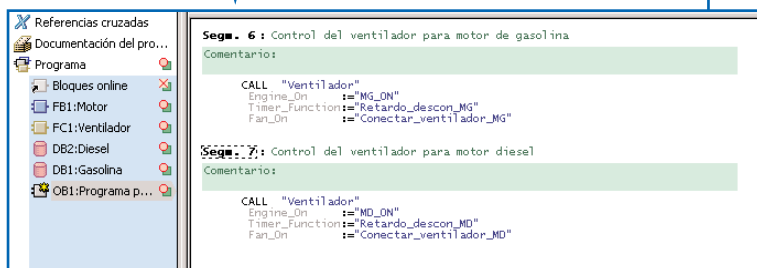


Programar llamadas de bloque en KOP

3 Arrastre la FC1 a los segmentos 6 y 7 mediante la función Arrastrar y soltar.

4 Complete todos los ??? como en la figura de la izquierda. Guarde el bloque y ciérrelo.

... en AWL,

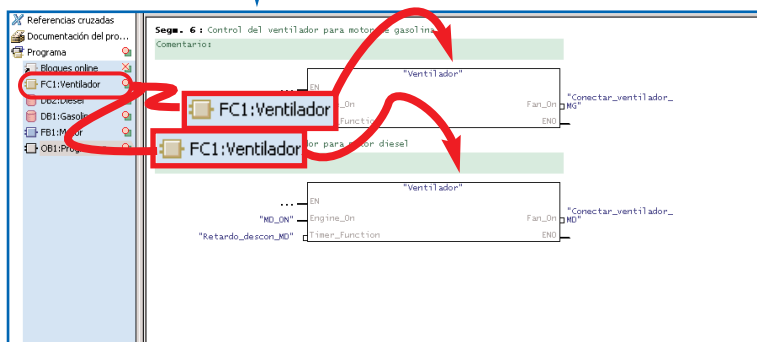


Programar llamadas de bloque en AWL

3 Introduzca las instrucciones AWL que aparecen en la figura de la izquierda.

4 Guarde y cierre el bloque.

... y en FUP.



Programar llamadas de bloque en FUP

3 Arrastre la FC1 al segmento 6 Motor de gasolina mediante la función Arrastrar y soltar. Arrastre la FC1 al segmento 7 Motor diesel mediante la función Arrastrar y soltar.

4 Complete todos los ??? como en la figura de la izquierda. Guarde el bloque y ciérrelo.



1. ¿La representación en pantalla es distinta a la de las figuras?

Seleccione la programación simbólica con el comando **Ver > Mostrar > Representación simbólica**.

2. ¿Desea ver más información en pantalla?

Active **Ver > Mostrar > Información del símbolo** para obtener información sobre las direcciones utilizadas en cada segmento.

Para representar varios segmentos en la pantalla, desactive **Ver > Mostrar > Comentario** y, si es necesario, **Ver > Mostrar > Información del símbolo**. Con **Ver > Factor de zoom** se puede modificar el tamaño de representación del segmento.

3. ¿Necesita información sobre los lenguajes de programación KOP, AWL o FUP?

Encontrará más información pulsando **F1 > Contenido > Llamar ayudas de referencia > Descripción de lenguajes y ayudas de bloques**.

4. ¿No desea llamar siempre la función?

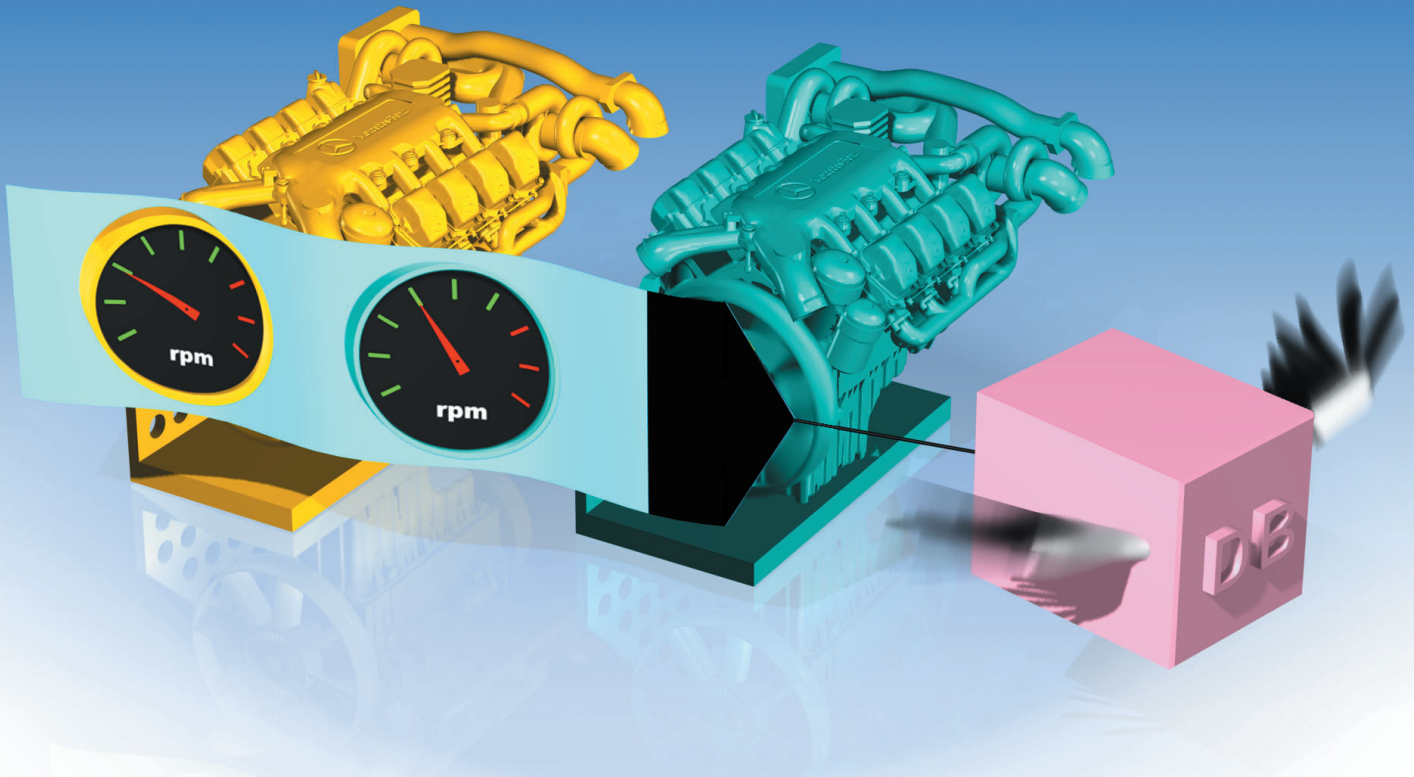
La llamada de las funciones está programada en este ejemplo como llamada incondicional, es decir, la función se procesa siempre. Dependiendo de los requisitos de la tarea de automatización en cuestión, la llamada de FCs o FBs también se puede vincular a ciertas condiciones: por ejemplo a una entrada o a una combinación lógica previa. Para programar condiciones, tiene a su disposición la entrada EN y la salida ENO.

9

Uso de bloques de datos globales



Crear y abrir un bloque de datos (DB) globales



9.2

Si la CPU no dispone del número de marcas internas (células de memoria) necesario para gestionar el volumen de datos, los datos seleccionados se pueden depositar en un bloque de datos global.

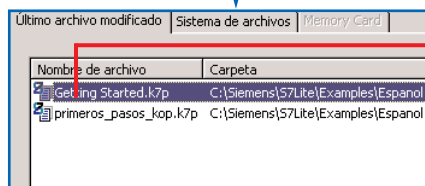
Todos los bloques pueden acceder a los datos del bloque de datos globales. En cambio, un bloque de datos de instancia está asignado a un bloque de función determinado y sus datos sólo están disponibles de forma local en dicho bloque de función (consulte el capítulo 7, “Crear bloques de datos de instancia y modificar valores actuales”). Si quiere continuar este capítulo, deberá copiar primero la tabla de símbolos en el proyecto “Getting Started” (consulte la página 5.5).



Se abre el cuadro de diálogo de selección de proyectos.

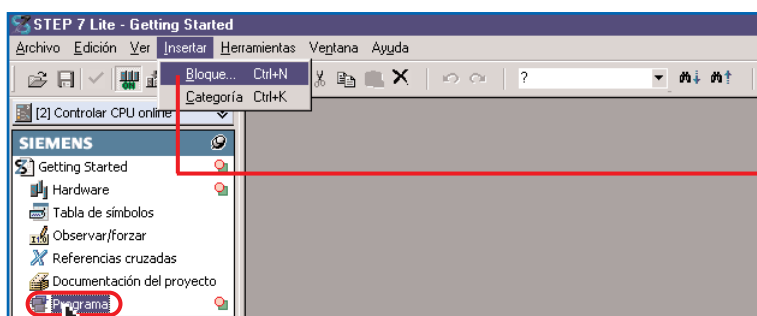
1

Si es necesario, abra STEP 7 Lite.



2

En el cuadro de diálogo **Abrir proyecto**, haga doble clic en el proyecto "Getting Started".



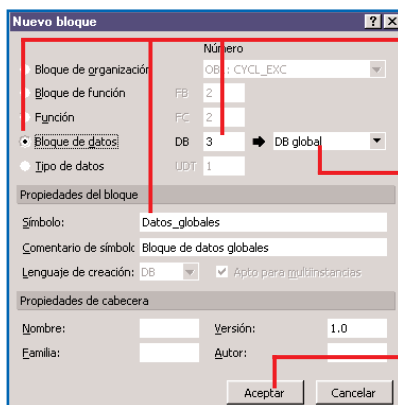
3

En la ventana de proyectos, haga clic en Programa.

4

Seleccione el comando de menú **Insertar > Bloque** o abra el menú emergente con el botón derecho del ratón y seleccione el comando **Nuevo > Bloque**.

Se abre el cuadro de diálogo para crear bloques.



5

Seleccione la opción **Bloque de datos**. En el campo **DB** se introduce un "3" automáticamente y en el campo **Símbolo**, "Datos_G".

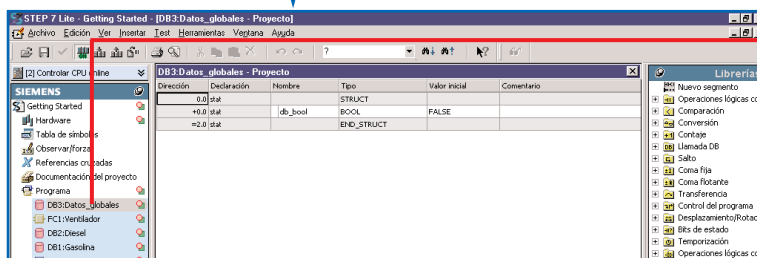
6

Seleccione "DB global".

7

Confirme los datos con **Aceptar**.

Se inserta el bloque y se abre inmediatamente.



8

El bloque nuevo se inserta en la ventana del proyecto y se abre inmediatamente.

Programar variables en el DB

DB3:Datos_globales - Proyecto					
Dirección	Declaración	Nombre	Tipo	Valor inicial	Comentario
0.0	stat		STRUCT		
+0.0	stat	PE_Actual_Speed	INT	0	Revoluciones reales del motor de gasolina
+2.0	stat	DE_Actual_Speed	INT	0	Revoluciones reales del motor diesel
+4.0	stat	Preset_Speed_Reached	BOOL	FALSE	Ambos motores han alcanzado el número teórico de revoluciones
=6.0	stat		END_STRUCT		

9.4

1

Introduzca "Revol_reales_MG" en la columna **Nombre**.

2

Abra el menú emergente con el botón derecho del ratón y seleccione la opción **Datos simples > INT** para **Tipo**.

Complete la lista tal como se muestra en la figura.

Guarde las modificaciones y cierre el bloque.



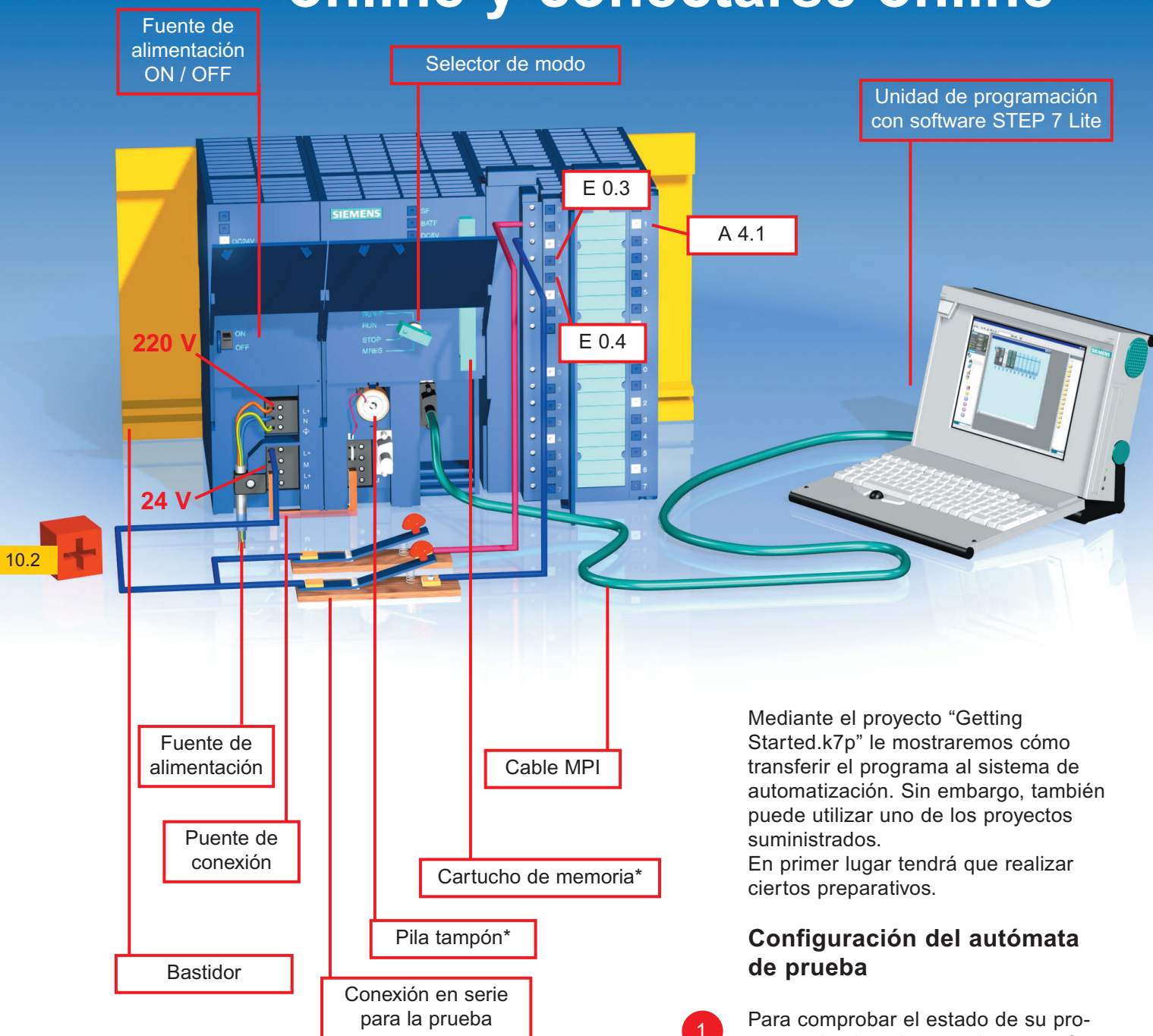
Pulse **F1 > Contenido > Programar bloques > Crear bloques de datos** para obtener más información.

10

Transferir
el programa
a la CPU



Establecer una conexión online y conectarse online



Mediante el proyecto "Getting Started.k7p" le mostraremos cómo transferir el programa al sistema de automatización. Sin embargo, también puede utilizar uno de los proyectos suministrados. En primer lugar tendrá que realizar ciertos preparativos.

Configuración del autómata de prueba

- 1 Para comprobar el estado de su programa de ejemplo necesita un autómata de prueba que le permita activar y desactivar las entradas (p. ej. en forma de pulsadores).

*= Accesorios opcionales

2

Comprobar el programa

Utilice el proyecto creado “Getting Started”. Además, debe haber configurado como mínimo el hardware (capítulo 4) y programado la conexión en paralelo (capítulo 6).

3

Comprobar el hardware

Configure el hardware y compruebe una vez más lo siguiente:

- ¿Está insertado el conector de bus en los módulos?
- ¿Se han colgado los módulos en los perfiles soporte, se han abatido después hacia abajo y se han atornillado?
- ¿Se ha conectado una fuente de alimentación de 220 V?
- ¿Ha colocado el peine de conexión?
- Siempre que estén disponibles, ¿ha colocado la pila de respaldo y el cartucho de memoria?

10.3

4

Conectarse online

Conectarse online significa establecer una conexión entre la CPU y la unidad de programación.

- Inserte el cable MPI en la unidad de programación y en la CPU.

En la CPU:

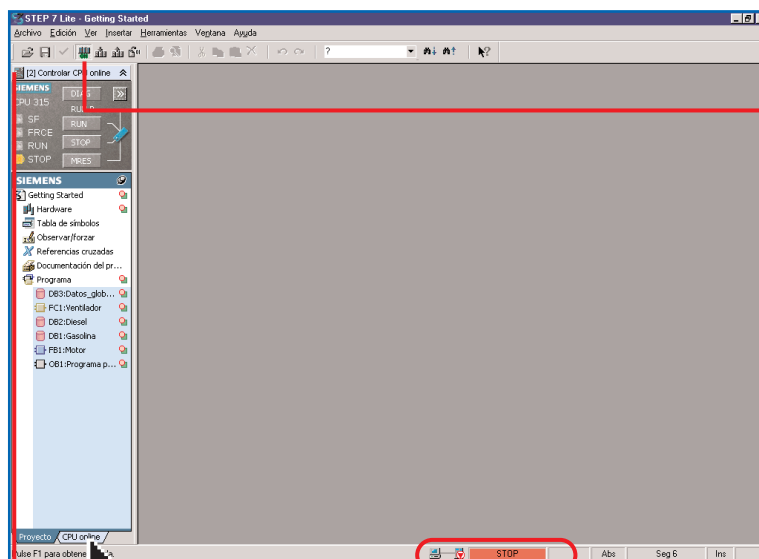
- **Conecte** la fuente de alimentación.
- Gire el selector al modo **STOP**.



En la unidad de programación:

- **Presione** el interruptor principal.
- Inicie STEP 7 Lite.
- Abra “Getting Started.k7p” o uno de los proyectos de ejemplo.

Transferir el programa a la CPU



Establecer la conexión online

5

Al arrancar, STEP 7 Lite intenta establecer una conexión online inmediatamente.

El botón verde **Conectar online** deberá permanecer iluminado y en la barra de estado se indicará en primer lugar el estado de conexión **Online** y, a continuación, el estado operativo de la CPU **STOP**.

Si no puede acceder a ninguna CPU, STEP7 Lite permanecerá offline y en la barra de estado aparecerá el estado de conexión **Desconectado**.

En tal caso, resuelva la causa que ha provocado el fallo al establecer la conexión online (p. ej. el cable no se ha enchufado correctamente o la CPU está apagada) y haga clic en el botón **Conectar online**.

Con este botón podrá establecer y deshacer la conexión online tantas veces como lo desee.

6

Compruebe si aparece **Manejar CPU online**. Los botones del panel de mandos de la CPU se activarán.

7

Compruebe: En la barra de estado aparecerá ahora el estado operativo actual de la CPU.

8

Ahora puede pasar de la vista offline del **proyecto** a la vista online de la **CPU online**, pruebe a hacerlo.

En la ficha **CPU online** aparecerán todos los módulos que se encuentren en la CPU.

Si todavía no ha transferido su proyecto a la CPU, los símbolos de la ventana del proyecto indicarán que los datos de la unidad de programación todavía no coinciden con los de la CPU.

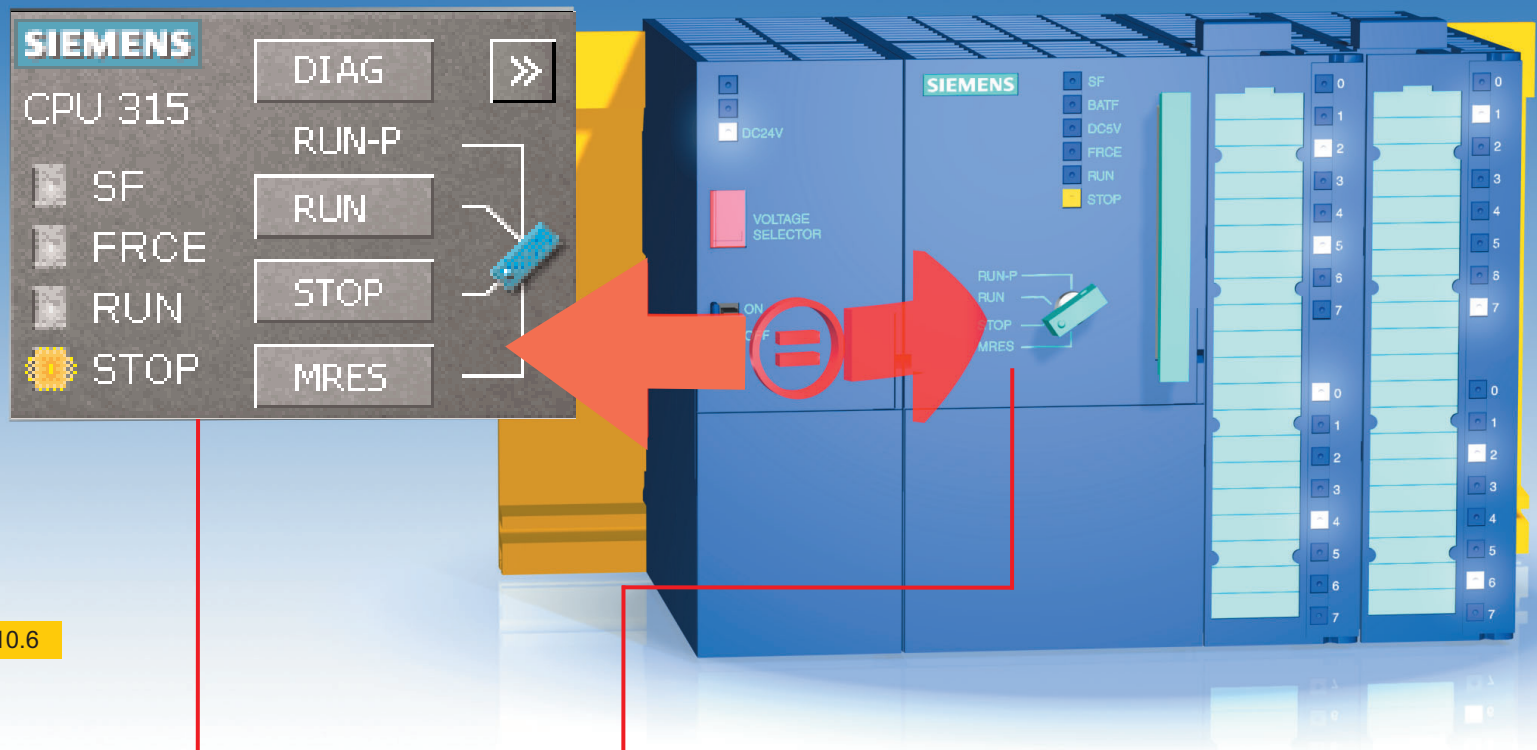


También es posible conectar online y comprobar el programa (capítulo 11) utilizando otro hardware, tal y como se ilustra en la página 10.2. Simplemente observe el direccionamiento de las entradas y salidas.

Encontrará más información sobre cómo configurar los módulos centrales en los manuales “Sistema de automatización S7-300, Configuración e instalación”.

Si desea obtener más información sobre cómo conectar online, consulte la Ayuda en pantalla con el comando de menú **F1 > Contenido > Establecer un enlace online**.

Borrado total de la CPU y transferencia del programa



Panel de mandos de la CPU en STEP 7 Lite

Panel de mandos de la CPU real

El panel de mandos de la CPU

Antes de transferir su programa es necesario realizar un borrado total de la CPU. Puede hacerlo con el panel de mandos directamente en la CPU o con el panel de mandos simulado en STEP 7 Lite.

El panel de mandos también permite cambiar los estados operativos de la CPU.

Por razones de seguridad, en el panel de STEP 7 Lite sólo se podrán activar aquellos botones que puedan ser seleccionados para el estado operativo de la CPU real.

Ejemplo:

- En la CPU: Selector en **RUN** -
En el panel de mandos de STEP 7 Lite se puede seleccionar **STOP**.
- En la CPU: Selector en **STOP** -
En el panel de mandos de STEP 7 Lite no se puede seleccionar **RUN**.

Si se produce una situación de peligro también podrá conmutar la CPU a STOP desde STEP 7 Lite.

Borrado total directamente en la CPU

Antes de transferir el programa a la CPU, borre todos los datos y programas antiguos de la CPU con el comando Borrado total. Para ello:

-  1 Conecte la CPU.
-  2 Gire el selector a **STOP** (si todavía no está en STOP). El LED se iluminará en rojo.
-  3 Gire el selector a **MRES** y manténgalo en esta posición durante al menos 3 segundos, hasta que el LED STOP comience a parpadear en rojo lentamente.

3 sec
↓
3 sec
↓
-  4 Suelte el selector y, después de un máximo de 3 segundos, vuelva a seleccionar la posición MRES hasta que el LED STOP comience a parpadear rápidamente.

El borrado total de la CPU ha concluido.

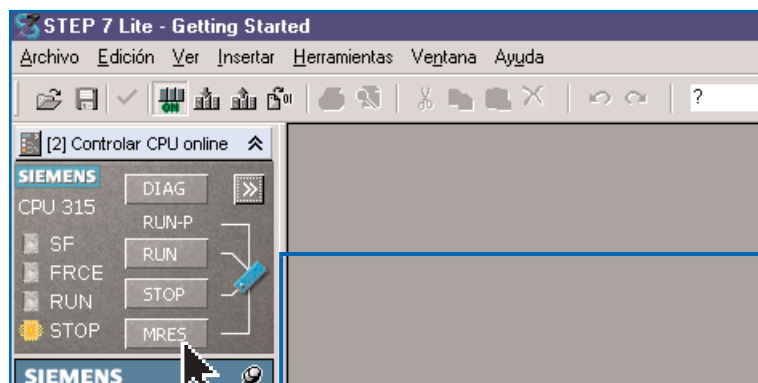
Borrado total de la CPU con STEP7 Lite

Además del proceso explicado anteriormente, también puede realizar un borrado total de la CPU con el software STEP 7 Lite.

En la CPU: Sitúe el selector en **STOP**.



1

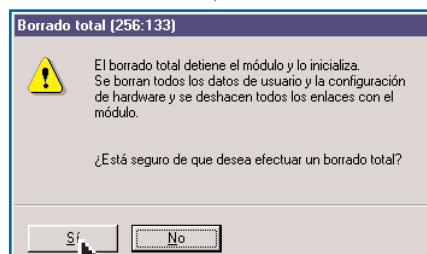


2

En STEP7 Lite: Haga clic en el botón **MRES**.

CLICK

Aparece una consulta de seguridad.



CLICK

3

Confirme la consulta de seguridad con **Sí**. El borrado total de la CPU habrá concluido.

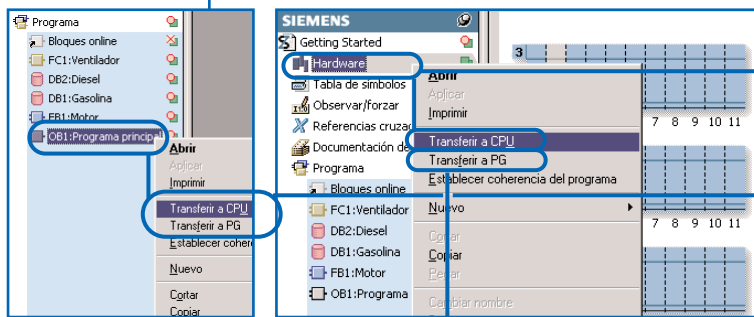
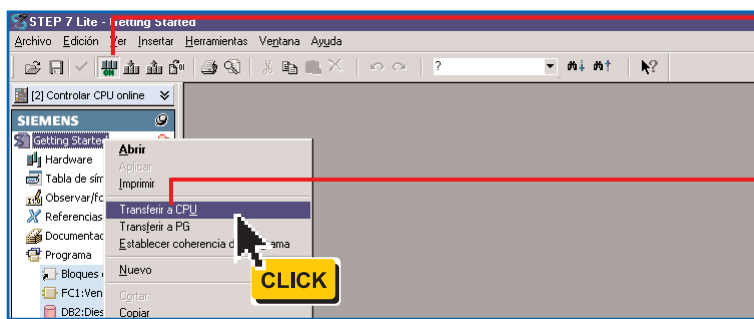
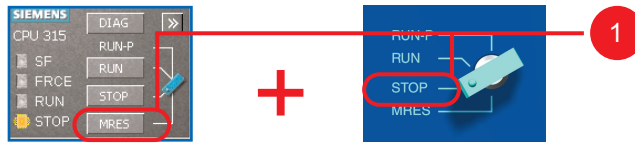


Encontrará notas adicionales sobre los estados operativos en la ayuda de STEP 7 Lite, a la que se puede acceder con **F1 > Contenido > Anexo > Estados operativos**.

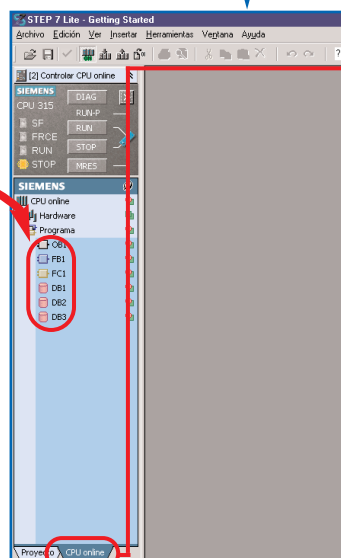
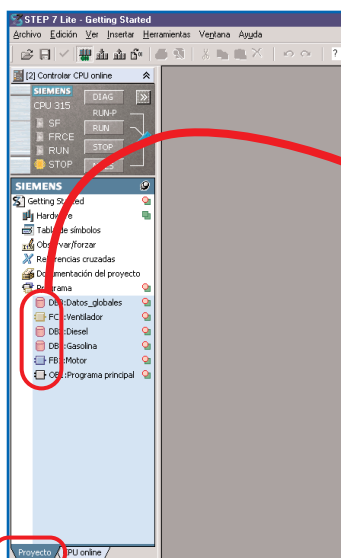
Atención:
El conocimiento de los estados operativos de las CPU resulta útil para programar el arranque, comprobar el programa de control y para el diagnóstico de errores.

Transferir el programa a la CPU

Compruebe que, al transferir el programa, se encuentren en **STOP** el selector de la CPU y el de STEP 7 Lite.



Ha cambiado la vista "CPU online".



- 2 Transferir el proyecto completo a la CPU:
Selecione el proyecto "Getting Started" en la ventana del proyecto.
- 3 Haga clic con el botón derecho del ratón en el proyecto y elija **Transferir a CPU** en el menú emergente.
Todo el proyecto incluido en la configuración de hardware se transferirá a la CPU.

También puede transferir bloques individuales o la configuración de hardware en la CPU.

- 4 En este ejemplo, la configuración de hardware está marcada para cargarla en la CPU.
- 5 En este ejemplo, el bloque está marcado para cargarlo en la CPU.
- 5 Dependiendo de qué elementos estén marcados, STEP 7 Lite ofrecerá la opción **Transferir a PG** (unidad de programación).

- 6 Haga clic en **CPU online**.
Aquí aparecerán todos los datos que se encuentran en la CPU.

- **Proyecto** (vista offline)
= Datos en la unidad de programación
- **CPU online** (vista online)
= Datos en la CPU

Transferir el programa a la CPU

Encender la CPU y comprobar el estado operativo



1

Gire el selector a la posición **RUN-P**. El LED **RUN** se encenderá en verde y el LED **STOP** rojo se apagará. La CPU está lista para el funcionamiento.

Se indica la ejecución cíclica del programa.

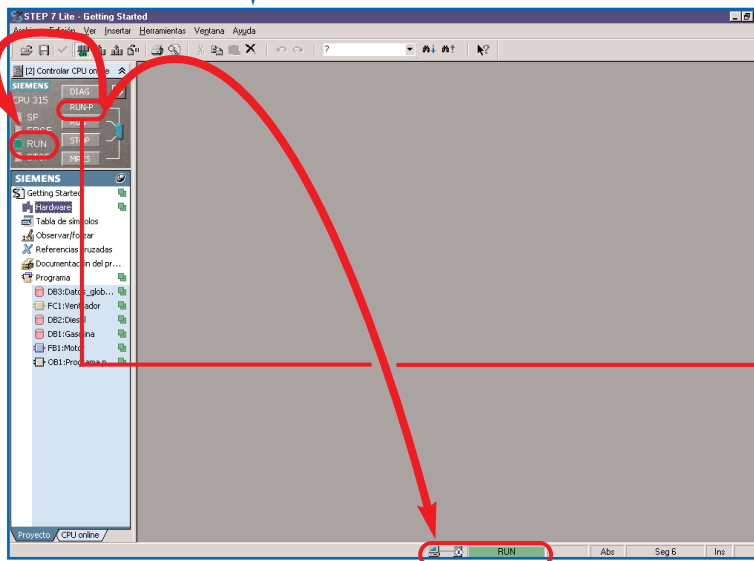
2

Compruebe lo siguiente en la CPU: Si se enciende el LED verde, puede comenzar a comprobar el programa.

Si el LED rojo sigue encendido, hay algún error. Para realizar el diagnóstico de errores, puede hacer clic en el botón "DIAG" para evaluar el búfer de diagnóstico (consulte también el apartado "Información del módulo e historial de errores", pág. 12.5).

3

Compruebe lo siguiente en STEP 7 Lite:
Al conmutar directamente en la CPU, se conmutó también aquí a **RUN-P**, indicándose la ejecución cíclica del programa sobre fondo verde.



10.10



Borrado total:

Aunque se efectúe un borrado total, la CPU seguirá conteniendo los bloques de función de sistema (SFBs) y funciones de sistema (SFCs). Estas funciones del sistema operativo siempre están disponibles en la CPU. No necesitan ser transferidos, pero no se pueden borrar.

Transferir bloques individuales:

Para que durante el funcionamiento real pueda reaccionar rápidamente a fallos, puede transferir bloques a la CPU de forma individual. Al transferirlos, hay que vigilar que el selector de la CPU esté en la posición "RUN-P" o en "STOP". En "RUN-P", los bloques transferidos se activan inmediatamente.



Atención:

Si sobrescribe bloques con otros que contengan errores, la instalación no funcionará correctamente.



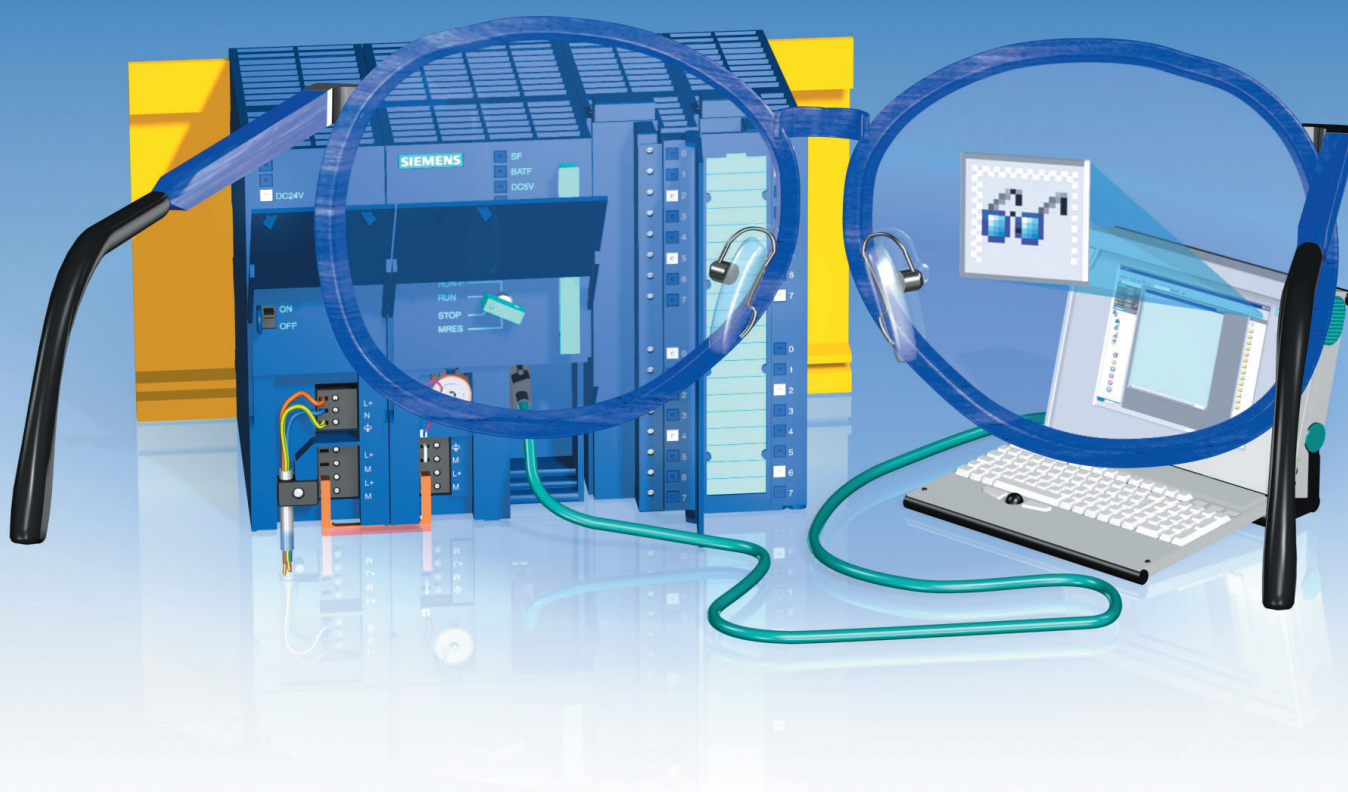
Si al transferir los bloques no se tiene en cuenta el orden (p.ej. se llama un bloque en el OB 1, pero todavía no está disponible en la CPU), la CPU pasará al estado operativo "STOP".

CPU 31xC:

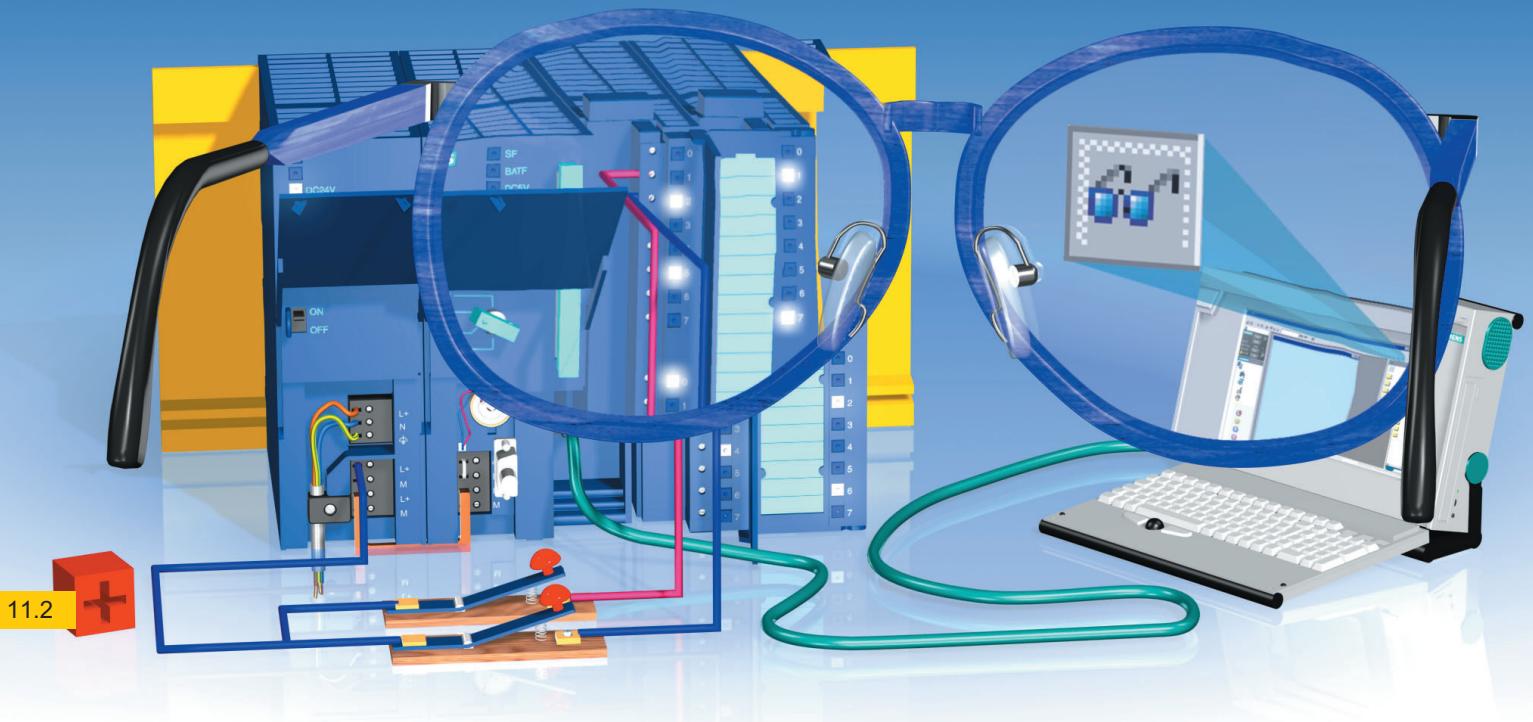
El selector de modo de la CPU 31xC no es un interruptor giratorio sino de balancín y no dispone de la posición RUN-P. Sin embargo, el borrado total se efectúa del mismo modo. Encontrará más información sobre Micro Memory Cards pulsando **F1 > Index Micro Memory Card**.

11

Comprobar el programa



Comprobar el estado del programa (Observar)



STEP 7 Lite ofrece la posibilidad de comprobar el funcionamiento del programa directamente en el sistema de automatización. Dispone de las siguientes posibilidades de comprobación:

1. Test en modo Observar – permite seguir de cerca la ejecución del programa (consulte pág. 11.3 a 11.5).
2. Test con tablas de variables – para observar y modificar los operandos, p. ej. entradas, salidas o marcas (consulte pág. 11.6 a 11.10).

Para comprobar el estado del programa es necesario que el proyecto completo esté cargado en la CPU.

Preparativos

1

Establezca la conexión online.



2

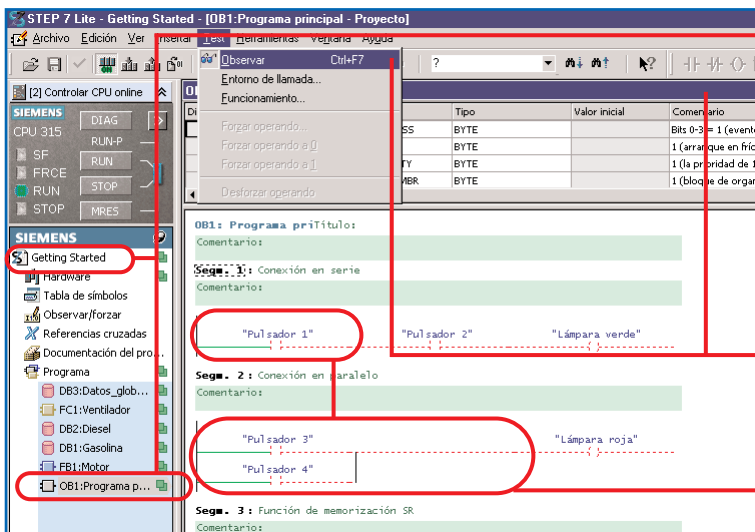
Gire el selector de la CPU a RUN o RUN-P.

3

Para el segmento 1: Configure con los cables una conexión en serie. Para el segmento 2: Configure con los cables una conexión en paralelo (véase el gráfico)

4

Abra el proyecto "Getting Started" o uno de los proyectos de ejemplo cargados en la CPU. Abra el OB 1.



5

Ejecute la función Observar con el comando **Test > Observar**. La función sólo se podrá activar si antes se ha ejecutado "Conectar online".

6

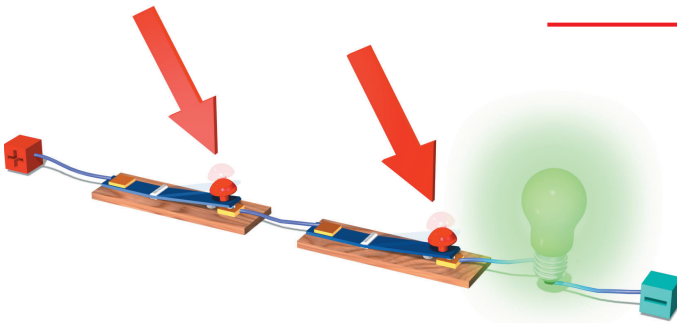
Los circuitos negros de los segmentos se representarán ahora en color.

Circuito **VERDE**: La corriente fluye.
Circuito **ROJO**: La corriente no fluye.

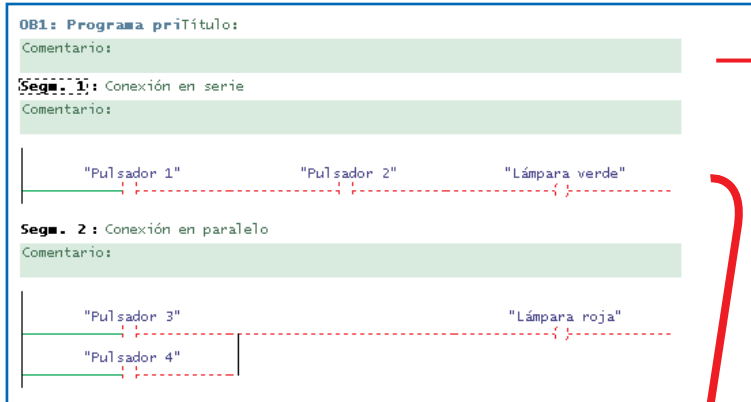
7

En su configuración de prueba, accione consecutivamente los pulsadores y observe:

- En STEP 7 Lite: cómo cambian los colores de los circuitos.
- En los módulos: cómo se encienden y se apagan los LED en los módulos de entrada y salida.



Comprobar el programa en KOP

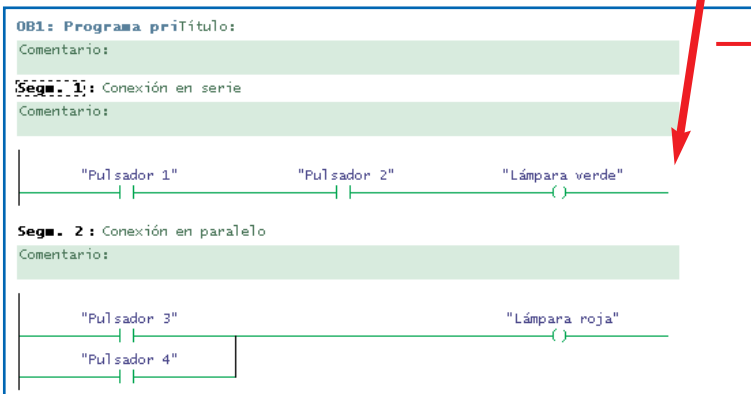


1

Deje abiertos todos los pulsadores.

En los segmentos 1 y 2 habrá tensión hasta los pulsadores 1, 3 y 4. Estos circuitos aparecerán en verde. A partir de los pulsadores 1, 3 y 4 no fluir la corriente, por lo que los circuitos se representarán en rojo.

El cambio de colores simboliza que el resultado lógico se cumple hasta ese punto.

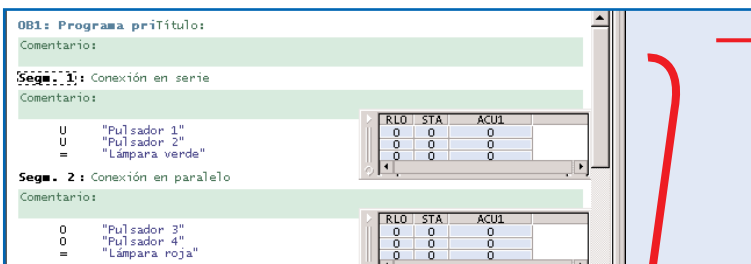


2

Ahora, pulse los pulsadores 1, 2, 3 y 4. Fluirá corriente por todos los circuitos.

Si ha abierto uno de nuestros proyectos de ejemplo, puede consultar en los comentarios qué diodos deben iluminarse en los módulos de entrada y salida.

Comprobar el programa en AWL



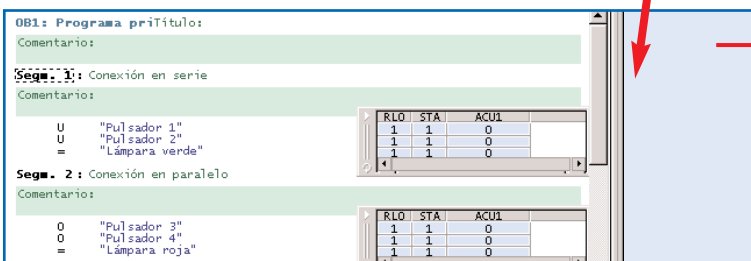
1

Deje abiertos todos los pulsadores.

En AWL, aparecerán

- Resultado lógico (RLO)
- Bit de estado (STA)
- Acumulador (ACU1)

en forma de tabla.

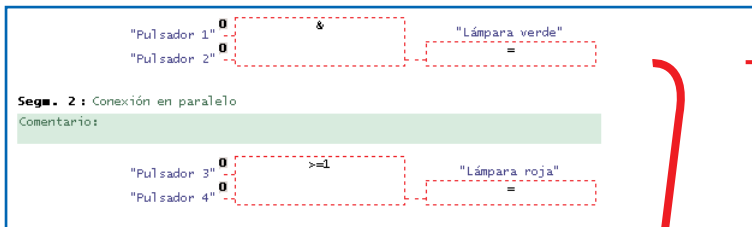


2

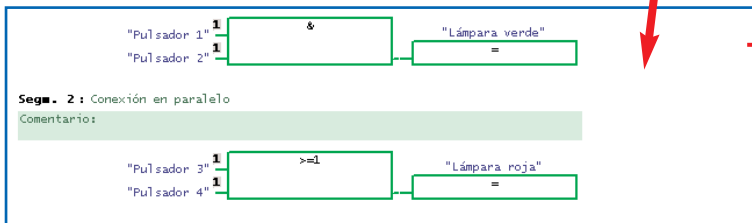
Ahora, accione los pulsadores 1, 2, 3 y 4.

Se cumplirá el resultado lógico en todos los puntos.

Comprobar el programa en FUP



1 Deje abiertos todos los pulsadores.



2 Ahora, accione los pulsadores 1, 2, 3 y 4.
El cambio de color señala que el resultado lógico se cumple en todos los puntos.

3 Desactive el comando de menú **Test > Observar** y cierre la ventana.

11.5



Bajo **Herramientas > Preferencias > Editor de bloques** se puede modificar la representación de los resultados de la comprobación.

Encontrará más información sobre el tema en **F1 > Contenido > Test > Test con el estado de programa**.

Observar y forzar variables

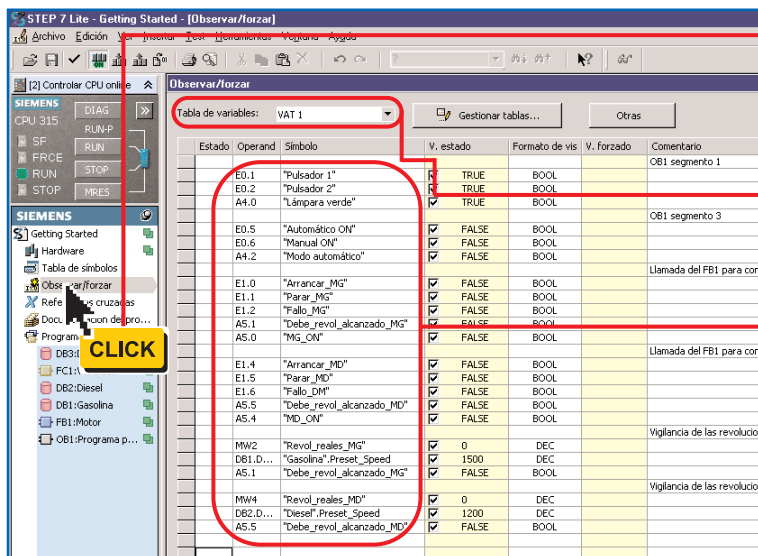
Diagram illustrating the STEP 7 Lite - Getting Started - [Observar/forzar] window interface and its components:

- Iniciar observación**: Button to start observation.
- Iniciar forzado**: Button to start forcing.
- Observar operandos**: Column for observing operands.
- Forzar operandos**: Column for forcing operands.
- Ampliar o reducir la pantalla Observar/Forzar**: Window title bar.
- Iniciar función Observar/Forzar**: Button to start the function.
- Campos de entrada de la tabla de variables**: Table headers.
- Campos para mostrar el valor de estado, p. ej., "true" o "false"**: Column for showing the state value.
- Campo de entrada del valor de forzado**: Column for entering the forced value.

Estado	Operand	Símbolo	V. estado	Formato de vis	V. forzado	Comentario
	E0.1	"Pulsador 1"	<input checked="" type="checkbox"/>	TRUE	BOOL	OB1 segmento 1
	E0.2	"Pulsador 2"	<input checked="" type="checkbox"/>	TRUE	BOOL	
	A4.0	"Lámpara verde"	<input checked="" type="checkbox"/>	TRUE	BOOL	
	E0.5	"Automático ON"	<input checked="" type="checkbox"/>	FALSE	BOOL	OB1 segmento 3
	E0.6	"Manual ON"	<input checked="" type="checkbox"/>	FALSE	BOOL	
	A4.2	"Modo automático"	<input checked="" type="checkbox"/>	FALSE	BOOL	
	E1.0	"Arrancar_MG"	<input checked="" type="checkbox"/>	FALSE	BOOL	Llamada del FB1 para conectar el motor de gasolina
	E1.1	"Parar_MG"	<input checked="" type="checkbox"/>	FALSE	BOOL	
	E1.2	"Fallo_MG"	<input checked="" type="checkbox"/>	FALSE	BOOL	
	A5.1	"Debe_revol_alcanzado_MG"	<input checked="" type="checkbox"/>	FALSE	BOOL	
	A5.0	"MG_ON"	<input checked="" type="checkbox"/>	FALSE	BOOL	
	E1.4	"Arrancar_MD"	<input checked="" type="checkbox"/>	FALSE	BOOL	Llamada del FB1 para conectar el motor diesel
	E1.5	"Parar_MD"	<input checked="" type="checkbox"/>	FALSE	BOOL	
	E1.6	"Fallo_DM"	<input checked="" type="checkbox"/>	FALSE	BOOL	
	A5.5	"Debe_revol_alcanzado_MD"	<input checked="" type="checkbox"/>	FALSE	BOOL	
	A5.4	"MD_ON"	<input checked="" type="checkbox"/>	FALSE	BOOL	
	MW2	"Revol_reales_MG"	<input checked="" type="checkbox"/>	0	DEC	Vigilancia de las revoluciones del motor de gasolina
	DB1.D...	"Gasolina".Preset_Speed	<input checked="" type="checkbox"/>	1900	DEC	
	A5.1	"Debe_revol_alcanzado_MG"	<input checked="" type="checkbox"/>	FALSE	BOOL	
	MW4	"Revol_reales_MD"	<input checked="" type="checkbox"/>	0	DEC	Vigilancia de las revoluciones del motor diesel
	DB2.D...	"Diesel".Preset_Speed	<input checked="" type="checkbox"/>	1200	DEC	
	A5.5	"Debe_revol_alcanzado_MD"	<input checked="" type="checkbox"/>	FALSE	BOOL	

Para observar operandos introdúzcalos primero en la tabla de variables. Para ello es necesario que exista una conexión online con la CPU.

Para forzar operandos deberá indicar un valor de forzado y activar la casilla correspondiente. Además de la conexión online, también es necesario que la CPU se encuentre en el estado operativo RUN-P.



Crear la tabla de variables

- 1 Abra el proyecto "Getting Started". Haga doble clic en **Observar/Forzar**.
- 2 En el campo **Tabla de variables**, cree una tabla con el nombre "VAT 1".
- 3 En el ejemplo "Getting Started", indique todas las variables o solamente las que desee forzar.

Para ello:

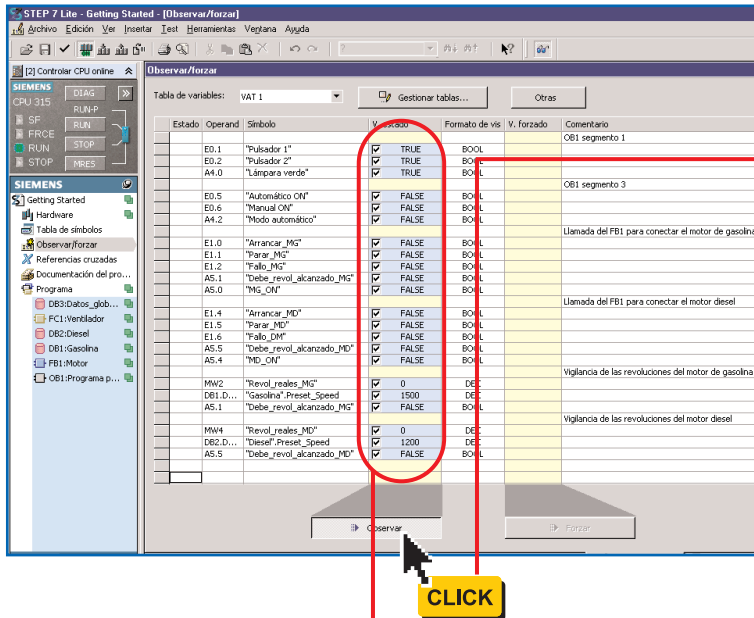
En la columna **Operando**, escriba "E0.1". En cuanto pulse la tecla Entrar, el nombre simbólico "Pulsador 1" aparecerá automáticamente tal y como figura en la tabla de símbolos. También puede situar el puntero del ratón en la columna Operando y seleccionar operandos en la lista con las teclas **Ctrl + j**.

También puede copiar la tabla de variables de uno de los proyectos de ejemplo.

- 4 Abra uno de los proyectos de ejemplo en una segunda instancia de STEP 7 Lite. Seguidamente, haga clic en **Observar/Forzar** y abra **VAT 1**. Seleccione con **Ctrl + a** toda la tabla y cópiela con **Ctrl + c** al portapapeles.
- 5 Vaya ahora al proyecto "Getting Started" e inserte los datos del portapapeles con **Ctrl + v**.

Observar variables

Al observar las variables no sólo se comprueba el funcionamiento del programa, sino también el del hardware.



1 En su configuración de prueba, accione el "Pulsador 1" y el "Pulsador 2".

2 Haga clic en **Observar**. La columna Valor de estado aparecerá ahora sobre fondo azul y se observarán las variables.

3 Aquí podrá ver:

- cómo cambian los indicadores de la columna **Valor de estado** de "FALSE" a "TRUE"
- cómo se encienden o apagan simultáneamente los LED de los módulos de entrada y de salida al accionar los pulsadores en la configuración de prueba.

4 Para continuar el test del programa y del hardware, compruebe ahora si la combinación de los siguientes estados es correcta

- Pulsador abierto/cerrado
- LED encendido/apagado
- Variable true/false

Forzar variables

Al forzar se asignan valores a las variables para simular así determinadas situaciones del funcionamiento del programa.

Ejemplo:

1 Para poder llevar a cabo el forzado, la CPU deberá estar de nuevo en RUN-P y la función **Observar** activa.

2 Indique en la columna **Valor de forzado**, por ejemplo el valor "TRUE". Ahora, el valor de forzado dejará de estar activo.

3 Active el valor de forzado marcando la casilla que aparece en cuanto se introduce el valor de forzado.

4 Haga clic en **Forzar**.

5 Observe el efecto de las variables forzadas en la columna **Valor de estado**.

Estado	Operand	Símbolo	V. estado	Formato de vis.	V. forzado	Comentario
E0.1	"Pulsador 1"		TRUE	BOOL	TRUE	OB1 segmento 1
E0.2	"Pulsador 2"		TRUE	BOOL	TRUE	
A4.0	"Lámpara verde"		TRUE	BOOL	TRUE	
E0.5	"Automático ON"		FALSE	BOOL		OB1 segmento 3
E0.6	"Manual ON"		FALSE	BOOL		
A4.2	"Modo automático"		FALSE	BOOL		Llamada del FB1 para conectar el motor de gasolina
E1.0	"Arrancar_MG"		FALSE	BOOL		
E1.1	"Parar_MG"		FALSE	BOOL		
E1.2	"Fallo_MG"		FALSE	BOOL		
A5.1	"Debe_revol_ alcanzado_MG"		FALSE	BOOL		Llamada del FB1 para conectar el motor diesel
A5.0	"MG_ON"		FALSE	BOOL		
E1.4	"Arrancar_MD"		FALSE	BOOL		
E1.5	"Parar_MD"		FALSE	BOOL		
E1.6	"Fallo_MD"		FALSE	BOOL		
A5.5	"Debe_revol_ alcanzado_MD"		FALSE	BOOL		Vigilancia de las revoluciones del motor de gasolina
A5.4	"MD_ON"		FALSE	BOOL		
MW2	"Revol_reales_MG"		0	DEC		Vigilancia de las revoluciones del motor diesel
D61.D...	"Gasolina_Preset_Speed"		1000	DEC		
A5.1	"Debe_revol_ alcanzado_MG"		FALSE	BOOL		
MW4	"Revol_reales_MD"		0	DEC		
D62.D...	"Diesel_Preset_Speed"		1000	DEC		
A5.5	"Debe_revol_ alcanzado_MD"		FALSE	BOOL		

Comprobar el programa

SIEMENS CPU 315

DIAG RUN-P SF RUN FRCE STOP MRES

SIEMENS

Getting Started Hardware Tabla de símbolos Observar/forzar Referencias cruzadas Documentación del pro... Programa DB3:Datos_glob... FC1:Ventilador DB2:Diesel DB1:Gasolina FB1:Motor OB1:Programa p...

Tabla de variables: VAT 1

Estado	Operand	Símbolo	V. estado	Formato de vis	V. forzado	Comentario
	E0.1	"Pulsador 1"	TRUE	BOOL	TRUE	OB1 segmento 1
	E0.2	"Pulsador 2"	TRUE	BOOL	TRUE	
	A4.0	"Lámpara verde"	TRUE	BOOL		OB1 segmento 3
	E0.5	"Automático ON"	FALSE	BOOL		
	E0.6	"Manual ON"	FALSE	BOOL		
	A4.2	"Modo automático"	FALSE	BOOL		
	E1.0	"Arrancar_MG"	FALSE	BOOL		Llamada del FB1 para cone
	E1.1	"Parar_MG"	FALSE	BOOL		
	E1.2	"Fallo_MG"	FALSE	BOOL		
	A5.1	"Debe_revol_alcanzado_MG"	TRUE	BOOL		
	A5.0	"MG_ON"	FALSE	BOOL		Llamada del FB1 para cone
	E1.4	"Arrancar_MD"	FALSE	BOOL		
	E1.5	"Parar_MD"	FALSE	BOOL		
	E1.6	"Fallo_MD"	FALSE	BOOL		
	A5.5	"Debe_revol_alcanzado_MD"	TRUE	BOOL		
	A5.4	"MD_ON"	FALSE	BOOL		
	MW2	"Revol_reales_MG"	1500	DEC	1500	Vigilancia de las revolucione
	DB1.D...	"Gasolina".Preset_Speed	1500	DEC		
	A5.1	"Debe_revol_alcanzado_MG"	TRUE	BOOL		
	MW4	"Revol_reales_MD"	1200	DEC	1200	Vigilancia de las revolucione
	DB2.D...	"Diesel".Preset_Speed	1200	DEC		
	A5.5	"Debe_revol_alcanzado_MD"	TRUE	BOOL		

Observar Forzar

Además de operandos binarios se pueden forzar operandos no binarios. Para ello seleccione primero el formato de visualización y, a continuación, indique el valor de forzado correspondiente.

Si introduce DEC en la columna **Formato de visualización** para forzar la velocidad teórica,

puede indicar como valores de forzado las revoluciones "1200" y "1500" en decimal.

Si indica un valor que no admita el formato de visualización, el fondo de la celda se representará en rojo y la columna **Valor de forzado** no mostrará más casillas.

El formato de visualización de una variable se puede cambiar haciendo clic sobre el tipo de formato en la columna **Formato de visualización**.

Vista ampliada de la pantalla Observar/forzar

Observar/forzar

Tabla de variables: VAT 1

Gestione tablas... Otras Desbloquear salidas

Estado	Operand	Símbolo	V. estado	Formato de vis	V. forzado	Comentario
	E0.1	"Pulsador 1"	TRUE	BOOL	TRUE	OB1 segmento 1
	E0.2	"Pulsador 2"	TRUE	BOOL	TRUE	
	A4.0	"Lámpara verde"	TRUE	BOOL		OB1 segmento 2
	E0.5	"Automático ON"	FALSE	BOOL		
	E0.6	"Manual ON"	FALSE	BOOL		
	A4.2	"Modo automático"	FALSE	BOOL		
	E1.0	"Arrancar_MG"	FALSE	BOOL		Llamada del FB1 para conectar el motor de gasolina
	E1.1	"Parar_MG"	FALSE	BOOL		
	E1.2	"Fallo_MG"	FALSE	BOOL		
	A5.1	"Debe_revol_alcanzado_MG"	TRUE	BOOL		
	A5.0	"MG_ON"	FALSE	BOOL		Llamada del FB1 para conectar el motor diesel
	E1.4	"Arrancar_MD"	FALSE	BOOL		
	E1.5	"Parar_MD"	FALSE	BOOL		
	E1.6	"Fallo_MD"	FALSE	BOOL		
	A5.5	"Debe_revol_alcanzado_MD"	TRUE	BOOL		
	A5.4	"MD_ON"	FALSE	BOOL		
	MW2	"Revol_reales_MG"	1500	DEC	1500	Vigilancia de las revoluciones del motor de gasolina
	DB1.D...	"Gasolina".Preset_Speed	1500	DEC		
	A5.1	"Debe_revol_alcanzado_MG"	TRUE	BOOL		
	MW4	"Revol_reales_MD"	1200	DEC	1200	Vigilancia de las revoluciones del motor diesel
	DB2.D...	"Diesel".Preset_Speed	1200	DEC		
	A5.5	"Debe_revol_alcanzado_MD"	TRUE	BOOL		

Modo de observación Fin de ciclo permanente

Observar inmediatamente Observar

Inicio de ciclo permanente Modo de forzado

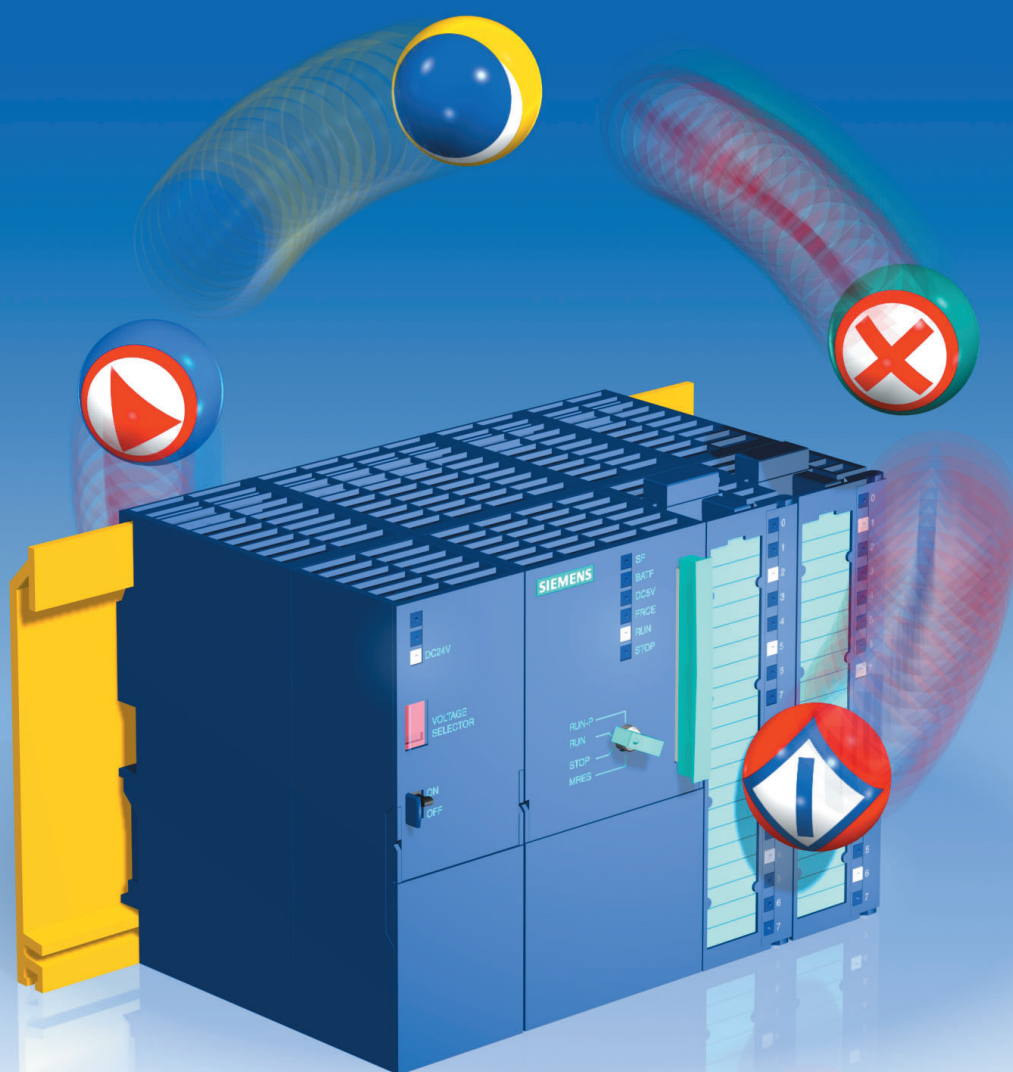
Forzar Forzar inmediatamente

En la vista ampliada de la pantalla Observar/forzar puede elegir distintos modos de observación y forzado y, p. ej., determinar si el valor de forzado

- deberá ser activado solamente en determinados estados del ciclo del OB1, p. ej., al comienzo, o bien
- inmediatamente, es decir, durante la ejecución del programa.

12

Diagnóstico de errores



Diagnóstico HW a simple vista

The screenshot shows the 'Hardware - Proyecto' window in STEP 7 Lite. It displays a rack diagram on the left and a detailed module data table on the right. The table lists modules in slots 1 through 11, including a CPU 315 and two SM 32 modules. A status window at the bottom right provides detailed information for the selected CPU module, showing it is 'Módulo O.K.'.

1 Abrir el proyecto bajo **Archivo**

2 Conexión online

3 Observar el estado operativo de la CPU

4 Indicador colectivo de fallos
Comprobar el hardware

5 Vista: Abrir diagnóstico HW

6 Indicación de los módulos diagnosticables

7 Datos globales de los módulos – con indicación de su estado

8 Datos detallados – se especifica el error

9 Mostrar búfer de diagnóstico

10 Datos completos para pedido de repuestos

Cuando se produce un fallo de hardware en el equipo PLC, STEP 7 Lite le ofrece un diagnóstico completo en una sola pantalla. Los números indican qué orden seguir para realizar el diagnóstico. Esta vista sólo está disponible si existe una conexión online.

¿Se ha producido algún error?



En una instalación, la CPU ha cambiado a STOP. El selector de modo de la CPU se encuentra en la posición **RUN**.

Significa que hay un fallo de hardware.

Solución

Gire el selector de modo a la posición **STOP**.



1

Abra el proyecto que corresponde a esta CPU (equipo) y en cuya configuración de hardware se ha producido el error.

2

Establezca un enlace online entre la unidad de programación y el equipo (consulte el capítulo 10).

12.3

3

Ya sabe que la CPU ha pasado a STOP. Estos indicadores resultan muy útiles, cuando, p. ej., el técnico no puede ver la CPU al realizar trabajos de mantenimiento.

4

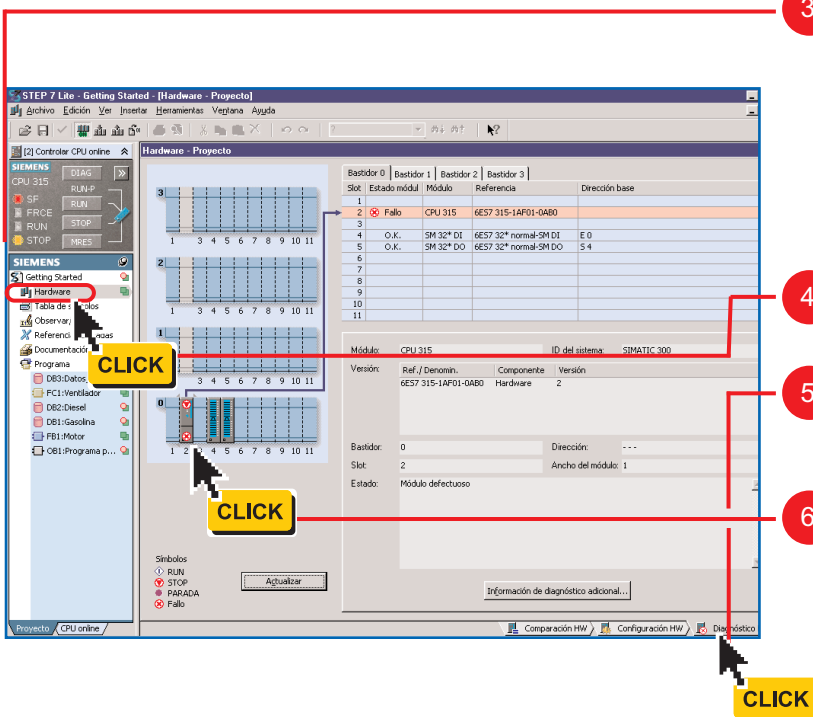
Haga doble clic sobre **Hardware** en la parte izquierda de la ventana del proyecto.

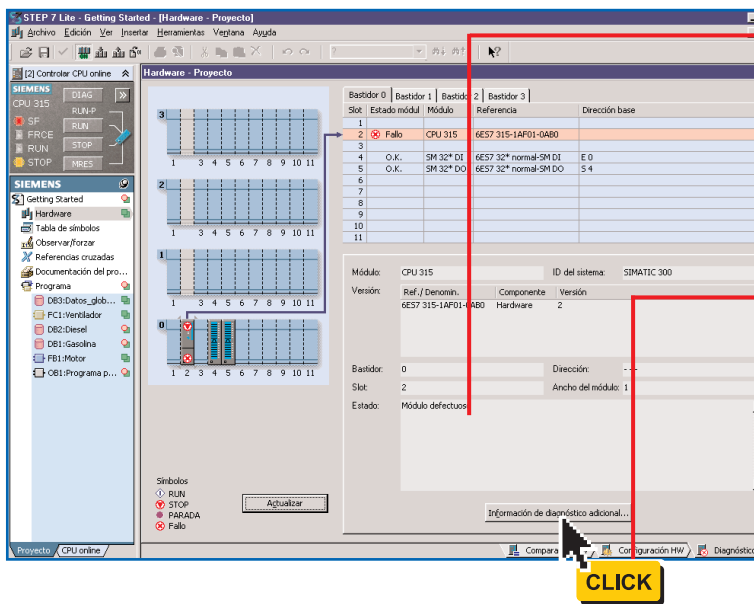
5

Abra la vista **Diagnóstico HW**.

6

El módulo defectuoso está marcado con un pictograma en el bastidor. Haga clic en este módulo.





8

Aparecerá un mensaje:

- Módulo en orden (OK)
- Módulo defectuoso.

9

Para más información, haga clic en **Información de diagnóstico adicional**.

Estado de los módulos e historial de errores

STOP, RUN, PARADA, ...

OK, defectuoso, ...

Consultar otros datos de la CPU

Información del módulo - CPU 315 ONLINE

Estado operativo de la CPU: STOP Información del módulo: Error

Sistema de reloj | Datos característicos | Comunicaciones | General

Búfer de diagnóstico | Pilas | Memoria | Tiempo de ciclo

Eventos: ☐ Activar filtros ajustados ☐ Mostrar eventos en hexadecimal

N°	Hora	Fecha	Evento
1	08:23:57.490	01.03.02	BAF: fallo de la tensión de respaldo en el aparato central
2	08:20:31.342	01.03.02	CONEXIÓN respaldada
3	20:08:10.877	28.02.02	Corte de alimentación
4	20:06:55.187	28.02.02	STOP mediante el selector de modos de operación
5	19:09:37.780	28.02.02	Cambio de ARRANQUE a RUN
6	19:09:37.779	28.02.02	Petición manual de arranque completo (en caliente)
7	19:09:37.760	28.02.02	Cambio de STOP a ARRANQUE
8	19:07:21.815	28.02.02	Borrado total efectuado

Detalles: 1 de 100 ID del evento: 16# 3922

BAF: fallo de la tensión de respaldo en el aparato central
N° de DB: 81
DB no existente o bloqueado o no ejecutable en el estado operativo actual
Error externo, Evento entrante

Guardar como... Ajustes... Abrir bloque Ayuda del evento

1 Búfer de diagnóstico

2 Detalles sobre el evento marcado en el búfer de diagnóstico

3 Ayuda sobre el evento indicado

Filtrar eventos en el búfer de diagnóstico

Guardar búfer de diagnóstico en formato TXT

12.5

Antes de cambiar un módulo supuestamente defectuoso, utilice la función **Información del módulo** (para acceder a esta función, consulte la página 12.4).

El búfer de diagnóstico guarda todos los eventos de la CPU, no sólo los fallos. Necesitará estos datos especialmente para distinguir el verdadero error de los errores aparecidos en consecuencia.



Para más información pulse **F1 > Contenido > Llamar la información del módulo.**

**En caso de rotura de hilo:**

Compruebe el cableado o asegúrese de que la posición de los adaptadores del margen de medida sea correcta.

Si la CPU cambia a STOP:

Evalúe los mensajes del búfer de diagnóstico. Haciendo clic en el botón "DIAG" del panel de mandos de la CPU accederá más rápidamente al búfer de diagnóstico.

Si el módulo está defectuoso:

Desconecte la tensión de carga antes de extraer el módulo.

13

Índice
alfabético

A

Aplicar 4.15, 6.8, 6.14, 6.20
Automation License Manager 1.10
AWL 6.2
Ayuda 2.8
Ayuda básica 2.9
Ayuda breve 2.9
Ayuda de referencia 2.10
Ayuda directa 2.9

B

Barra de alimentación 6.9
Bastidor 4.2, 10.2
Bloque de datos global 9.2
Bloque de función 7.2
Bloques de datos de instancia 7.12
Borrado total de la CPU 10.6, 10.7, 10.8
Búfer de diagnóstico 12.2

C

Cable MPI 10.2
Cambiar el lenguaje de programación 6.4
Campo de aplicación de STEP 7 Lite 0.3
Cartucho de memoria 10.2
Catálogo de hardware 4.6
Chincheta 4.5
Clave de licencia 1.8
Comparación HW 4.18
Comparación online/offline 4.17
Completar la tabla de declaración de variables 7.4
Componentes de la lista de verificación 1.5
Comprobar el programa 11.2
Conectar online 10.3
Conexión en paralelo 1.3
Conexión en serie 1.3
Conexión y desconexión del motor 7.6
Configuración HW 4.6
Configuración del autómatas de prueba 10.2
Configurar el interface de programación 6.11
Configurar los módulos 4.2
Copiar la tabla de símbolos 5.5
Crear tabla de variables 11.7

D

DB 6.5
Definición de los requisitos de seguridad 3.5
Diagnóstico de HW 12.2
Diagnósticos de error 12.2
Diagrama de funciones 6.2
Dirección 0.2
Dirección absoluta 4.8, 5.2
Documentación 1.5

E

Editor de bloques 6.4
Ejemplos de programación 2.3
Elemento de memoria 1.3
Entrada 5.4
Error de configuración 4.10
Esquema de contactos 6.2
Establecer conexión online 10.4
Estado del programa 11.2

F

FB 6.5
FC 6.5, 8.2
Flipflop 1.3
Forzar 11.9
Forzar inmediatamente 11.10
Forzar variables 11.9
Fuente de alimentación 10.2
Función 8.2
Función de temporización 8.6
FUP 6.2

G

Guardar 4.15, 6.8, 6.14, 6.20
Guardar datos de configuración 4.14

H

Hardware real 4.19
Historial de errores 12.5

I

Información del módulo 12.5
Insertar módulos 4.9
Insertar segmento nuevo 6.4
Instalación 1.8

K

KOP 6.2

L

Lenguaje de programación 6.2
Librerías 2.5
Lista de instrucciones 6.2
Llamada de bloque 7.14

M

Manejo de archivos 2.7
Mensajes de error 2.10
Motor de gasolina 3.3
Motor diesel 3.3

N

Nuevo bloque 7.13
Nuevo proyecto 4.4

O

Offline 4.19
Online 4.19
OB 6.5
Observar 11.3, 11.8
Observar variables 11.6, 11.8
Operando 8.8
Ordenador 1.5

P

Panel de mandos de la CPU 2.5
Parametrización 4.13
Parametrizar módulos 4.12
Parametrizar módulos 4.6

Peine de conexión 10.2

Pictograma 4.6

Pila de respaldo 10.2

Planta de inspección de motores 3.2

Programación absoluta 5.2

Programación simbólica 5.4

Programación simbólica y absoluta, cambiar 6.4

Programar funciones 8.6

Programar la función de temporización 8.7

Programar la vigilancia de velocidad de giro 7.7

Programar llamadas de bloque 8.9

R

Referencia 1.5

Referencias cruzadas 6.24

Representación simbólica 6.7

Retardo a la desconexión 8.6

Rotura de hilo 12.5

S

Símbolos 4.20

Software de simulación 1.5

STOP 12.5

T

Tabla de declaración de variables 8.6

Tabla de símbolos 5.5

Test 11.4

Tipos de datos 5.6

Transferir bloques individuales 10.11

Transferir el programa 10.6

Transferir el programa a la CPU 10.9

Transferir la configuración de hardware 4.16

V

Ventana de proyectos 2.4

Vista general 1.6



6ES7810-3CC07-0YA05

Order No. 6ES7810-3CC07-0YA05

Siemens Aktiengesellschaft

Bereich Automation and Drives
Geschäftsgebiet Industrial Automation Systems
Postfach 4848, D-90327 Nuernberg

www.siemens.com/automation