

SIMATIC

Sistema de automatización S7-200

Manual de sistema

Referencia del manual:
6ES7298-8FA21-8DH0

03/2000
Edición 02

Prólogo, Índice

Introducción a los Micro-PLCs S7-200	1
Instalar un Micro-PLC S7-200	2
Instalación y configuración del sistema de automatización S7-200	3
Nociones básicas para programar una CPU S7-200	4
Memoria de la CPU: tipos de datos y direccionamiento	5
Configurar las entradas y salidas	6
Configurar el hardware para la comunicación en redes	7
Convenciones para las operaciones S7-200	8
Operaciones SIMATIC	9
Operaciones IEC 1131-3	10
Utilizar operaciones del protocolo USS para la comunicación con accionamientos	11
Datos técnicos S7-200	A
Códigos de error	B
Marcas especiales (SM)	C
Eliminar errores	D
Números de referencia	E
Tiempos de ejecución de las operaciones AWL	F
Breviario del S7-200	G
Índice alfabético	

Consignas de seguridad para el usuario

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 las medidas preventivas adecuadas, se producirá la muerte, o bien lesiones corporales graves o daños materiales considerables.



Precaución

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



Cuidado

Junto con el símbolo de aviso, significa que, si no se adoptan las medidas preventivas adecuadas, pueden producirse lesiones corporales leves o moderadas.

Cuidado

Sin el símbolo de aviso, significa que, si no se adoptan las medidas preventivas adecuadas, pueden producirse daños materiales.

Nota

Sin el símbolo de aviso, significa que, si no se adoptan las medidas preventivas adecuadas, pueden producirse resultados o estados impredecibles.

Personal cualificado

La puesta en funcionamiento y el servicio del equipo sólo deben ser llevados a cabo conforme a lo descrito en este manual. Sólo está autorizado a intervenir en este equipo el **personal cualificado**. En el sentido del manual se trata de personas que disponen de los conocimientos técnicos necesarios para poner en funcionamiento, conectar a tierra y marcar los aparatos, sistemas y circuitos de acuerdo con las normas estándar de seguridad.

Uso conforme

Considere lo siguiente:



Precaución

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, un almacenamiento, una instalación y un montaje conforme a las prácticas de la buena ingeniería, así como un manejo y un mantenimiento rigurosos.

Marcas registradas

SIMATIC ®, SIMATIC NET ® y SIMATIC HMI ® son marcas registradas por SIEMENS AG.

Los restantes nombres y designaciones contenidos en el presente impreso pueden ser marcas registradas cuya utilización por terceros para sus fines pueden violar los derechos de los propietarios.

Copyright © Siemens AG 2000 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 para ello. Los infractores quedan obligados a la indemnización de los daños. Se reservan todos los derechos, en particular para el caso de concesión de patentes o de modelos de utilidad.

Siemens AG
Bereich Automatisierungs- und Antriebstechnik
Geschäftsgebiet Industrie-Automatisierungssysteme
Postfach 4848, D-90327 Nuremberg

Siemens Aktiengesellschaft

Exención de responsabilidad

Hemos probado el contenido de esta publicación con la concordancia descrita para el hardware y el software. Sin embargo, es posible que se den algunas desviaciones que nos impiden tomar garantía completa de esta concordancia. El contenido de esta publicación está sometido a revisiones regularmente y en caso necesario se incluyen las correcciones en la siguiente edición. Agradecemos sugerencias.

© Siemens AG 2000
Sujeto a cambios sin previo aviso.

6ES7298-8FA21-8DH0



Prólogo

Finalidad del manual

La gama S7-200 comprende diversos sistemas de automatización pequeños (Micro-PLCs) que se pueden utilizar para numerosas tareas. Gracias a su diseño compacto, su bajo costo y su amplio juego de operaciones, los sistemas de automatización S7-200 son idóneos para controlar tareas sencillas. La gran variedad de modelos de CPUs y el software de programación basado en Windows ofrecen la flexibilidad necesaria para solucionar las tareas de automatización.

La gama S7-200 se ha rediseñado, ofreciendo ahora productos más pequeños, más rápidos y con más funciones. Los nuevos productos S7-200 han de sustituir a los anteriores.

El *Manual del sistema de automatización S7-200* contiene informaciones acerca de cómo instalar y programar los nuevos Micro-PLCs S7-200, incluyendo los siguientes temas:

- Montaje y cableado.
- Descripción de las funciones de la CPU, tipos de datos y modos de direccionamiento, ciclo de la CPU, protección con contraseña y comunicación en redes.
- Datos técnicos.
- Descripciones y ejemplos de las operaciones de programación SIMATIC e IEC 1131–3.
- Utilización de las operaciones del protocolo USS para la comunicación con accionamientos.
- Tiempos de ejecución típicos de las operaciones AWL (SIMATIC).

Destinatarios

Este manual va dirigido a ingenieros, programadores, técnicos de instalación y electricistas que dispongan de conocimientos básicos sobre los sistemas de automatización.

Ámbito de validez

Las informaciones contenidas en este manual son aplicables en particular a los siguientes productos:

- CPUs S7-200: CPU 221, CPU 222 y CPU 224 (firmware 1.1) y CPU 226 (firmware 1.0).
- STEP 7-Micro/WIN 32 (versión 3.1), un paquete de software de programación (de 32 bits) para los entornos Windows 95, Windows 98 y Windows NT.
- STEP 7-Micro/WIN 32 Toolbox, un paquete de software de programación (de 32 bits) para los entornos Windows 95, Windows 98 y Windows NT. STEP 7-Micro/WIN 32 Toolbox se ha diseñado para clientes que utilicen la CPU S7-200 junto con otros componentes de la gama baja de autómatas programables (p.ej. un panel táctil TP070 o un accionamiento MicroMaster).

Homologaciones

La gama S7-200 SIMATIC cumple las siguientes normas:

- Comunidad Europea (CE), Directiva de Baja Tensión 73/23/EEC
- Directiva EMC de la Comunidad Europea (CE) 89/336/CEE
- Underwriters Laboratories, Inc.: UL 508 registrado (Industrial Control Equipment)
- Canadian Standards Association: CSA C22.2 número 142 certificado (Process Control Equipment)
- Factory Mutual Research: FM clase I, categoría 2, grupos A, B, C y D Hazardous Locations y clase I, sección 2, IIC, T4.

Consulte el Anexo A para obtener más información sobre el cumplimiento de normas.

Informaciones adicionales

Los siguientes productos y manuales proporcionan informaciones más detalladas acerca de temas especiales:

- STEP 7-Micro/WIN 32 en CD o disquetes: incorpora la Ayuda en pantalla, la *Guía de iniciación a STEP 7-Micro/WIN* (un manual electrónico imprimible), así como proyectos con ejemplos y consejos.
- STEP 7-Micro/WIN 32 Toolbox en CD: incorpora el software de configuración del panel táctil TP070, las operaciones del protocolo USS, la Ayuda en pantalla, la *Guía de iniciación a STEP 7-Micro/WIN* (un manual electrónico imprimible), así como proyectos con ejemplos y consejos.
- Estándar PROFIBUS (Process Field Bus) (EN 50170): describe el protocolo estándar para la comunicación DP de las CPUs S7-200.
- *Manual del usuario del visualizador de textos TD 200*: describe cómo instalar y utilizar el TD 200 con un sistema de automatización S7-200.

Cómo utilizar el manual

Si es la primera vez que trabaja con los Micro-PLCs S7-200, es recomendable que lea todo el *Manual del sistema de automatización S7-200*. Si ya dispone de los conocimientos necesarios, consulte el índice para encontrar la información que precise.

El *Manual del sistema de automatización S7-200* se divide en los siguientes capítulos:

- El capítulo 1 ("Introducción a los Micro-PLCs S7-200") ofrece una panorámica de algunas de las funciones de los sistemas de automatización.
- El capítulo 2 ("Instalar un Micro-PLC S7-200") informa acerca de los procedimientos, dimensiones y reglas básicas para instalar las CPUs y los módulos de ampliación.
- El capítulo 3 ("Instalación y configuración del sistema de automatización S7-200") describe cómo instalar y configurar un sistema de programación S7-200.
- El capítulo 4 ("Nociones básicas para programar una CPU S7-200"), el capítulo 5 ("Memoria de la CPU: tipos de datos y direccionamiento") y el capítulo 6 ("Configurar las entradas y salidas") proporcionan información acerca de cómo la CPU S7-200 procesa los datos y ejecuta el programa de usuario.
- El capítulo 7 ("Configurar el hardware para la comunicación en redes") informa acerca de cómo instalar y desinstalar equipos de hardware de comunicación y sobre cómo conectar la CPU S7-200 a diferentes tipos de redes.
- El capítulo 8 ("Convenciones para las operaciones S7-200") ofrece una panorámica de los diferentes lenguajes de programación y de la correspondiente terminología.
- El capítulo 9 ("Operaciones SIMATIC") contiene descripciones y ejemplos de las operaciones KOP, FUP y AWL SIMATIC.
- El capítulo 10 ("Operaciones IEC 1131-3") contiene descripciones y ejemplos de las operaciones LD y FBD IEC 1131-3.
- El capítulo 11 ("Utilizar operaciones del protocolo USS para la comunicación con accionamientos") contiene descripciones y ejemplos de las operaciones del protocolo USS, así como informaciones acerca de cómo utilizar estas operaciones para la comunicación con accionamientos.

En los anexos se proporcionan informaciones adicionales (datos técnicos de los equipos, descripciones de los códigos de error, eliminación de errores y tiempos de ejecución de las operaciones AWL).

Asistencia complementaria

Si tiene preguntas técnicas, si necesita información sobre los cursos de entrenamiento o para efectuar pedidos, diríjase por favor a su representante de Siemens más próximo.

En las siguientes direcciones de Internet puede obtener informaciones acerca de los productos y servicios de Siemens, asistencia técnica y respuestas a las preguntas más frecuentes:

http://www.ad.siemens.de	(informaciones generales sobre Siemens)
http://www.siemens.com/S7-200	(informaciones sobre los productos S7-200)

Índice

1	Introducción a los Micro-PLCs S7-200	1-1
1.1	Comparativa de los diversos Micro-PLCs S7-200	1-2
1.2	Principales componentes de un Micro-PLC S7-200	1-5
1.3	Configuraciones máximas de E/S	1-7
2	Instalar un Micro-PLC S7-200	2-1
2.1	Preparar el montaje	2-2
2.2	Montar y desmontar un Micro-PLC S7-200 o un módulo de ampliación	2-6
2.3	Instalar el cableado de campo	2-9
2.4	Utilizar circuitos de supresión	2-16
2.5	Alimentación de corriente	2-18
3	Instalación y configuración del sistema de automatización S7-200	3-1
3.1	Resumen breve	3-2
3.2	Instalar STEP 7-Micro/WIN 32	3-3
3.3	Configurar la comunicación utilizando el cable PC/PPI	3-5
3.4	Establecer la comunicación con la CPU	3-9
3.5	Cambiar los parámetros de comunicación de la CPU	3-10
4	Nociones básicas para programar una CPU S7-200	4-1
4.1	Crear una solución de automatización con un Micro-PLC	4-2
4.2	Programas S7-200	4-5
4.3	Lenguajes y editores de programación S7-200	4-6
4.4	Diferencias entre las operaciones SIMATIC e IEC 1131-3	4-10
4.5	Elementos básicos para estructurar un programa	4-18
4.6	El ciclo de la CPU	4-22
4.7	Ajustar el modo de operación de la CPU	4-25
4.8	Definir una contraseña para la CPU	4-27
4.9	Comprobar y observar el programa	4-30
4.10	Editar programas en modo RUN	4-39
4.11	Seleccionar el tiempo en segundo plano	4-42
4.12	Eliminar errores de las CPUs S7-200	4-43

5	Memoria de la CPU: tipos de datos y direccionamiento	5-1
5.1	Direccionamiento directo de las áreas de memoria de la CPU	5-2
5.2	Direccionamiento indirecto (SIMATIC) de las áreas de memoria de la CPU	5-13
5.3	Respaldar datos en la CPU S7-200	5-15
5.4	Utilizar el programa para guardar datos en la memoria no volátil	5-20
5.5	Guardar el programa en un cartucho de memoria	5-22
6	Configurar las entradas y salidas	6-1
6.1	Entradas y salidas integradas y adicionales	6-2
6.2	Utilizar filtros de entrada para suprimir interferencias	6-4
6.3	Capturar impulsos	6-5
6.4	Configurar los estados de señal de las salidas	6-8
6.5	Filtrar entradas analógicas	6-9
6.6	Entradas y salidas rápidas	6-10
6.7	Potenciómetros analógicos	6-13
7	Configurar el hardware para la comunicación en redes	7-1
7.1	Opciones de comunicación	7-2
7.2	Instalar y desinstalar interfaces de comunicación	7-7
7.3	Seleccionar y cambiar parámetros	7-9
7.4	Comunicación con módems	7-16
7.5	Redes y protocolos	7-27
7.6	Componentes de redes	7-32
7.7	Utilizar el cable PC/PPI con otros dispositivos y en modo Freeport	7-36
7.8	Rendimiento de la red	7-42
8	Convenciones para las operaciones S7-200	8-1
8.1	Conceptos y convenciones para programar con STEP 7-Micro/WIN 32	8-2
8.2	Márgenes válidos para las CPUs S7-200	8-7

9	Operaciones SIMATIC	9-1
9.1	Operaciones lógicas con bits (SIMATIC)	9-2
9.2	Operaciones de comparación (SIMATIC)	9-10
9.3	Operaciones de temporización (SIMATIC)	9-15
9.4	Operaciones con contadores (SIMATIC)	9-23
9.5	Operaciones de reloj (SIMATIC)	9-71
9.6	Operaciones aritméticas con enteros (SIMATIC)	9-73
9.7	Operaciones aritméticas con números reales (SIMATIC)	9-82
9.8	Operaciones con funciones numéricas (SIMATIC)	9-85
9.9	Operaciones de transferencia (SIMATIC)	9-102
9.10	Operaciones de tabla (SIMATIC)	9-107
9.11	Operaciones lógicas (SIMATIC)	9-114
9.12	Operaciones de desplazamiento y rotación (SIMATIC)	9-120
9.13	Operaciones de conversión (SIMATIC)	9-130
9.14	Operaciones de control del programa (SIMATIC)	9-145
9.15	Operaciones de interrupción y comunicación (SIMATIC)	9-169
9.16	Operaciones lógicas de pilas (SIMATIC)	9-197
10	Operaciones IEC 1131-3	10-1
10.1	Operaciones lógicas con bits (IEC)	10-2
10.2	Operaciones de comparación (IEC)	10-8
10.3	Operaciones de temporización (IEC)	10-11
10.4	Operaciones con contadores (IEC)	10-15
10.5	Operaciones aritméticas (IEC)	10-18
10.6	Operaciones con funciones numéricas (IEC)	10-21
10.7	Operaciones de transferencia (IEC)	10-23
10.8	Operaciones lógicas (IEC)	10-25
10.9	Operaciones de desplazamiento y rotación (IEC)	10-27
10.10	Operaciones de conversión (IEC)	10-30
11	Utilizar operaciones del protocolo USS para la comunicación con accionamientos	11-1
11.1	Requisitos de las operaciones del protocolo USS	11-2
11.2	Secuencia de programación	11-4
11.3	Operaciones del protocolo USS	11-5
11.4	Conectar los accionamientos	11-17
11.5	Configurar los accionamientos	11-18

A	Datos técnicos S7-200	A-1
A.1	Datos técnicos generales	A-2
A.2	Datos técnicos de la CPU 221 (firmware 1.1)	A-6
A.3	Datos técnicos de la CPU 222 (firmware 1.1)	A-11
A.4	Datos técnicos de la CPU 224 (firmware 1.1)	A-16
A.5	Datos técnicos de la CPU 226	A-21
A.6	Datos técnicos del módulo de ampliación EM 221 de entradas digitales	A-26
A.7	Datos técnicos de los módulos de ampliación EM 222 de salidas digitales	A-28
A.8	Datos técnicos de los módulos de ampliación EM 223, 4 entradas digitales/4 salidas digitales	A-30
A.9	Datos técnicos de los módulos de ampliación EM 223, 8 entradas digitales/8 salidas digitales	A-33
A.10	Datos técnicos de los módulos de ampliación EM 223, 16 entradas digitales/16 salidas digitales	A-36
A.11	Datos técnicos de los módulos de ampliación EM 231, EM 232 y EM 235 de entradas y/o salidas analógicas	A-39
A.12	Datos técnicos del módulo EM 277 PROFIBUS-DP	A-50
A.13	Datos técnicos de los módulos de ampliación EM 231 Termopar y RTD	A-67
A.14	Procesador de comunicaciones CP 243-2	A-85
A.15	Cartuchos opcionales	A-88
A.16	Cable de módulo de ampliación	A-89
A.17	Cable PC/PPI	A-90
A.18	Simuladores de entradas	A-93
B	Códigos de error	B-1
B.1	Códigos de errores fatales y mensajes	B-2
B.2	Errores de programación del tiempo de ejecución	B-3
B.3	Violación de reglas de compilación	B-4
C	Marcas especiales (SM)	C-1
D	Eliminar errores	D-1
E	Números de referencia	E-1
F	Tiempos de ejecución de las operaciones AWL	F-1
G	Breviario del S7-200	G-1
	Índice alfabético	Índice alfabético-1

Introducción a los Micro-PLCs S7-200

La gama S7-200 comprende diversos sistemas de automatización pequeños (Micro-PLCs) que se pueden utilizar para numerosas tareas. La figura 1-1 muestra un Micro-PLC S7-200. Gracias a su diseño compacto, su capacidad de ampliación, su bajo costo y su amplio juego de operaciones, los Micro-PLCs S7-200 son especialmente apropiados para solucionar tareas de automatización sencillas. Además, los diversos tamaños y fuentes de alimentación de las CPUs ofrecen la flexibilidad necesaria para solucionar las tareas de automatización.

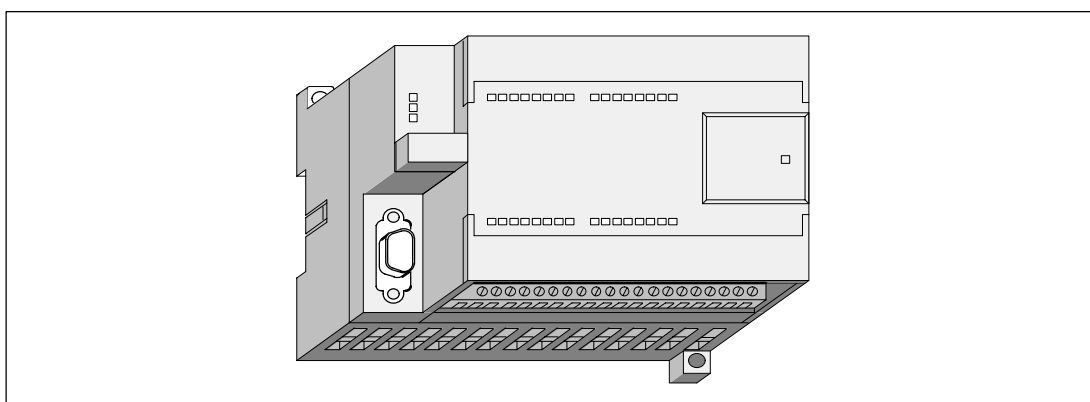


Figura 1-1 Micro-PLC S7-200

Índice del capítulo

Apartado	Descripción	Página
1.1	Comparativa de los diversos Micro-PLCs S7-200	1-2
1.2	Principales componentes de un Micro-PLC S7-200	1-5
1.3	Configuraciones máximas de E/S	1-7

1.1 Comparativa de los diversos Micro-PLCs S7-200

Equipos necesarios

La figura 1-2 muestra los componentes básicos de un sistema Micro-PLC S7-200, incluyendo una CPU S7-200, un PC, el software de programación STEP 7-Micro/WIN 32 (versión 3.1) y un cable de comunicación.

Si desea utilizar un PC, debe disponer de uno de los siguientes equipos adicionales:

- Un cable PC/PPI.
- Un procesador de comunicaciones (CP) y un cable de interface multipunto (MPI)
- Una tarjeta de interface multipunto (MPI). El cable de comunicación se suministra junto con la tarjeta MPI.

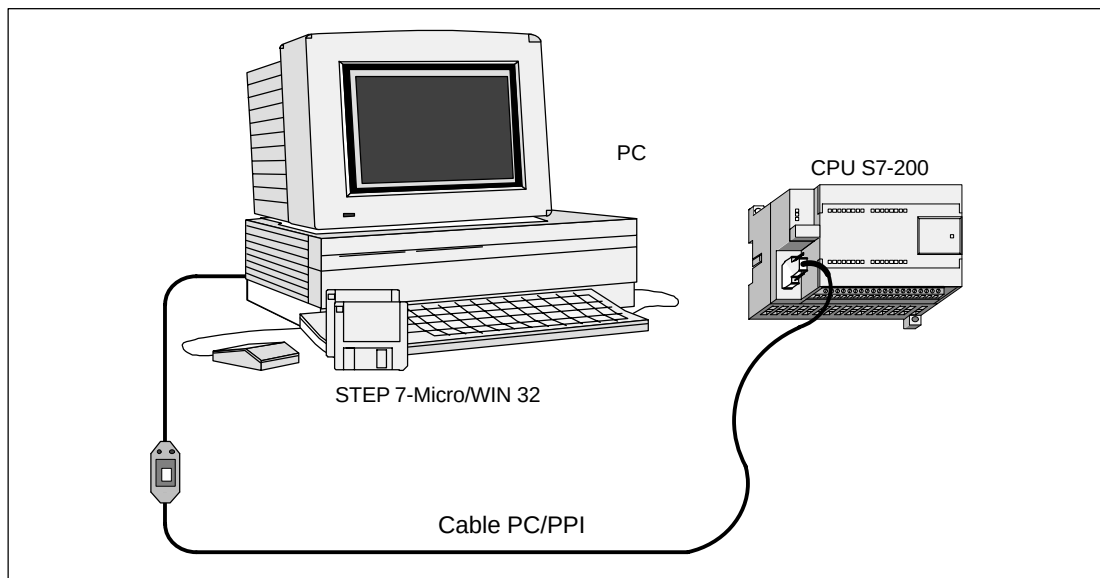


Figura 1-2 Componentes de un Micro-PLC S7-200

Capacidad de las CPUs S7-200

La gama S7-200 comprende diversas CPUs. Por lo tanto, se dispone de una gran variedad de funciones que permiten diseñar soluciones de automatización a un precio razonable. La tabla 1-1 resume las principales funciones de las CPUs S7-200 (firmware 1.1).

Tabla 1-1 Resumen de las CPUs S7-200

Función	CPU 221	CPU 222	CPU 224	CPU 226
Tamaño físico	90 mm x 80 mm x 62 mm	90 mm x 80 mm x 62 mm	120,5 mm x 80 mm x 62 mm	190 mm x 80 mm x 62 mm
Memoria				
Programa	2048 palabras	2048 palabras	4096 palabras	4096 palabras
Datos de usuario	1024 palabras	1024 palabras	2560 palabras	2560 palabras
Memoria para el programa de usuario	EEPROM	EEPROM	EEPROM	EEPROM
Respaldo (condensador de alto rendimiento)	50 horas (típ.)	50 horas (típ.)	190 horas (típ.)	190 horas (típ.)
E/S físicas				
E/S físicas	6 E / 4 S	8 E / 6 S	14 E / 10 S	24 E / 16 S
Número de módulos de ampliación	Ninguno	2 módulos	7 módulos	7 módulos
E/S (total)				
Tamaño de la imagen de E/S digitales	256 (128 E / 128 S)	256 (128 E / 128 S)	256 (128 E / 128 S)	256 (128 E / 128 S)
Tamaño de la imagen de E/S analógicas	Ninguno	16 E / 16 S	32 E / 32 S	32 E / 32 S
La cantidad real de E/S que se puede contar con las CPUs se puede ver limitada por el tamaño de la imagen del proceso, la cantidad de módulos de ampliación, la corriente de 5 V y la cantidad de E/S físicas de cada componente.(v. apt. 1.3).				
Operaciones				
Velocidad de ejecución booleana a 33 MHz	0,37 µs/operación	0,37 µs/operación	0,37 µs/operación	0,37 µs/operación
Imagen del proceso de las E/S	128 E / 128 S	128 E / 128 S	128 E / 128 S	128 E / 128 S
Relés internos	256	256	256	256
Contadores/temporizadores	256/256	256/256	256/256	256/256
Palabra IN / palabra OUT	Ninguno	16/16	32/32	32/32
Relés de control secuencial	256	256	256	256
Bucles FOR/NEXT	Sí	Sí	Sí	Sí
Aritmética en coma fija (+ – * /)	Sí	Sí	Sí	Sí
Aritmética en coma flotante (+ – * /)	Sí	Sí	Sí	Sí
Funciones adicionales				
Contadores rápidos	4 H/W (20 KHz)	4 H/W (20 KHz)	6 H/W (20 KHz)	6 H/W (20 KHz)
Potenciómetros analógicos	1	1	2	2
Salidas de impulsos	2 (20 KHz, sólo DC)	2 (20 KHz, sólo DC)	2 (20 KHz, sólo DC)	2 (20 KHz, sólo DC)
Interrupciones de comunicación	1 transmisión/ 2 recepción	1 transmisión/ 2 recepción	1 transmisión/ 2 recepción	2 transmisión/ 4 recepción
Interrupciones temporizadas	2 (1 ms a 255 ms)	2 (1 ms a 255 ms)	2 (1 ms a 255 ms)	2 (1 ms a 255 ms)
Entradas de interrupción de hardware	4 filtros de entrada	4 filtros de entrada	4 filtros de entrada	4 filtros de entrada
Reloj de tiempo real	Sí (cartucho)	Sí (cartucho)	Sí (incorporado)	Sí (incorporado)
Protección con contraseña	Sí	Sí	Sí	Sí

Tabla 1-1 Resumen de las CPUs S7-200, continuación

Comunicación				
Número de puertos de comunicación:	1 (RS-485)	1 (RS-485)	1 (RS-485)	2 (RS-485)
Protocolos soportados				
Puerto 0:	PPI, DP/T, Freeport	PPI, DP/T, Freeport	PPI, DP/T, Freeport	PPI, DP/T, Freeport
Puerto 1:	No aplicable	No aplicable	No aplicable	PPI, DP/T, Freeport
PROFIBUS punto a punto	(NETR/NETW)	(NETR/NETW)	(NETR/NETW)	(NETR/NETW)

1.2 Principales componentes de un Micro-PLC S7-200

Un Micro-PLC S7-200 puede comprender una CPU S7-200 sola o conectada a diversos módulos de ampliación opcionales.

CPU S7-200

La CPU S7-200 es un equipo autónomo compacto que incorpora una unidad central de procesamiento (CPU), una fuente de alimentación, así como entradas y salidas digitales.

- La CPU ejecuta el programa y almacena los datos para la tarea de automatización o el proceso.
- El sistema se controla mediante entradas y salidas digitales (E/S). Las entradas vigilan las señales de los dispositivos de campo (p.ej. sensores e interruptores), mientras que las salidas supervisan las bombas, motores u otros aparatos del proceso.
- La fuente de alimentación suministra corriente a la CPU y a los módulos de ampliación conectados.
- El (los) puerto(s) de comunicación permite(n) conectar la CPU a una unidad de programación o a otros dispositivos que intervengan en el proceso.
- Los diodos luminosos indican el modo de operación de la CPU (RUN o STOP), el estado de las entradas y salidas físicas, así como los posibles fallos del sistema que se hayan detectado.
- Utilizando módulos de ampliación se pueden agregar entradas y salidas (E/S) adicionales a la CPU. (La CPU 221 no se puede ampliar.)
- El rendimiento de la comunicación se puede incrementar utilizando módulos de ampliación.
- Algunas CPUs tienen un reloj de tiempo real incorporado, en tanto que otras pueden disponer de un cartucho (opcional) de reloj de tiempo real.
- Un cartucho enchufable EEPROM en serie (opcional) sirve para almacenar programas de la CPU y transferir programas de una CPU a otra.
- Un cartucho enchufable de pila (opcional) permite prolongar el respaldo de los datos en la RAM.

La figura 1-3 muestra una CPU S7-200.

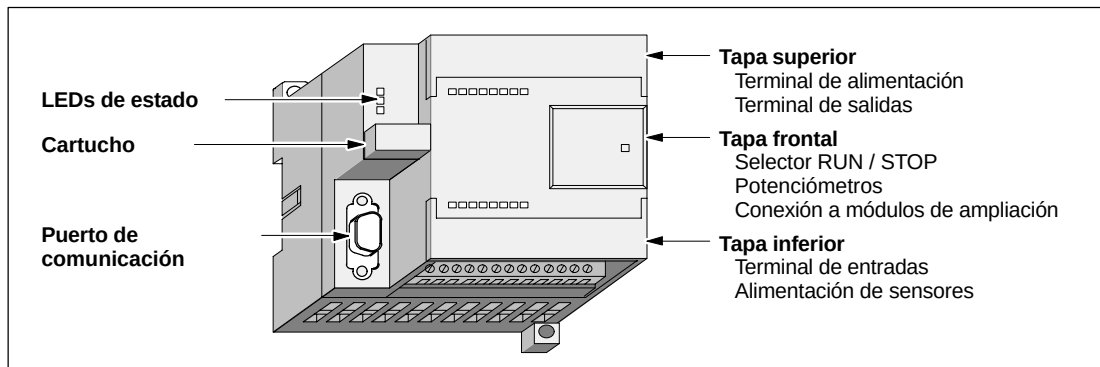


Figura 1-3 CPU S7-200

Módulos de ampliación

La CPU S7-200 dispone de un número determinado de entradas y salidas físicas. Conectando un módulo de ampliación se dispondrá de más entradas y salidas (v. fig. 1-4).

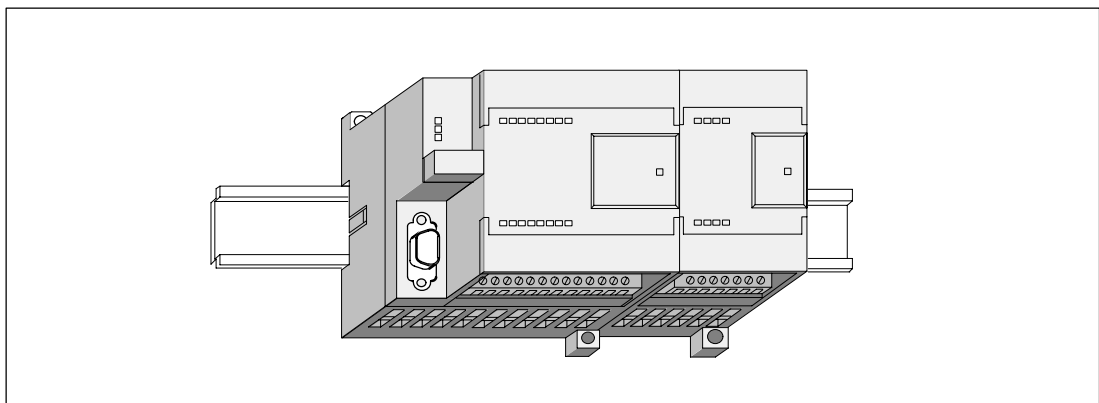


Figura 1-4 CPU con un módulo de ampliación

1.3 Configuraciones máximas de E/S

La configuración máxima de E/S para cada CPU está sujeta a los siguientes límites:

- Cantidad de módulos de ampliación:
CPU 221: no se pueden conectar módulos de ampliación.
CPU 222: 2 módulos de ampliación como máximo.
CPU 224 y CPU 226: 7 módulos de ampliación como máximo.
De estos 7 módulos, sólo 2 pueden ser módulos de ampliación inteligentes (EM 277 PROFIBUS-DP).
- Tamaño de la imagen de las E/S digitales: La memoria imagen de todas las CPUs S7-200 permite manejar 128 entradas digitales y 128 salidas digitales. Algunas E/S físicas no se pueden gestionar en la memoria imagen, puesto que el espacio está dispuesto en bloques de 8 E/S. Puede suceder que un módulo determinado no pueda gestionar por completo un bloque de 8 E/S. Por ejemplo, la CPU 224 tiene 10 salidas físicas, pero necesita 16 salidas de la memoria imagen. Un módulo de 4 entradas y 4 salidas físicas requiere 8 entradas y 8 salidas de la memoria imagen. En el momento de imprimir el presente manual no existía ninguna combinación de un modelo de CPU actual y módulos de ampliación que pudiera representar un límite a este respecto. Pueden existir límites en CPUs anteriores a la gama 22x o en futuras versiones.
- Tamaño de la imagen de las E/S analógicas: A continuación se indica la memoria imagen prevista para las E/S analógicas:
CPU 222: 16 entradas y 16 salidas
CPU 224 y CPU 226: 32 entradas y 32 salidas
- Corriente de 5 V: La tabla 1-2 muestra la corriente de 5 V máxima que puede suministrar cada una de las CPUs. La corriente total de todos los módulos de ampliación integrados en el sistema no puede exceder el límite mencionado. Para obtener más información a este respecto, consulte el apartado 2.5 ("Alimentación de corriente").

La tabla 1-3 muestra la configuración máxima de E/S permitida para cada CPU S7-200.

Tabla 1-2 Corriente suministrada por la CPU S7-200

CPU 22x Corriente DC 5 V para los módulos de ampliación (en miliamperios)		Módulo de ampliación Consumo de corriente DC 5 V (en miliamperios)	
CPU 222	340	EM 221, 8 entradas digitales x DC 24 V	30
CPU 224	660	EM 222, 8 salidas digitales x DC 24 V	50
CPU 226	1000	EM 222, 8 salidas digitales x relé	40
		EM 223, 4 entradas digitales/4 salidas digitales x DC 24 V	40
		EM 223, 4 entradas digitales/4 salidas de relé x DC 24 V	40
		EM 223, 8 entradas digitales/8 salidas digitales x DC 24 V	80
		EM 223, 8 entradas digitales/8 salidas de relé x DC 24 V	80
		EM 223, 16 entradas digitales/16 salidas digitales x DC 24 V	160
		EM 223, 16 entradas digitales/16 salidas de relé x DC 24 V	150
		EM 231, 4 entradas analógicas x 12 bits	20
		EM 231, 4 entradas analógicas x termopar	60
		EM 231, 4 entradas analógicas x RTD	60
		EM 232, 2 salidas analógicas x 12 bits	20
		EM 235, 4 entradas analógicas/4 salidas analógicas x 12 bits	30
		EM 277 PROFIBUS-DP	150

Tabla 1-3 Configuraciones máximas de E/S para las CPUs S7-200

Módulo	5 V (mA)	Entradas digitales	Salidas digitales	Entradas analógicas	Salidas analógicas
CPU 221 No se puede ampliar					
CPU 222					
E/S digitales máx. CPU 2 x EM 223 DI16/DO16 x DC24V ó 2 x EM 223 DI16/DO16 x DC24V/Relé Total =	+340 -320 ó -300 0	8 32 40	6 32 38		
Entradas analógicas máx. CPU 2 x EM 235 AI4/AQ1 Total =	+340 -60 0	8 8	6 6	8 8	2 2
Salidas analógicas máx. CPU 2 x EM 232 AQ2 Total =	+340 -40 0	8 8	6 6	0 0	4 4
CPU 224					
Entradas digitales/salidas de relé máx. CPU 4 x EM 223 DI16/DO16 x DC24V/Relé 2 x EM 221 DI8 x DC24V Total =	+660 -600 -60 0	14 64 16 94	10 64 74		
Entradas digitales/salidas DC máx. CPU 4 x EM 223 DI16/DO16 x DC24V Total =	+660 -640 0	14 64 78	10 64 74		
Entradas digitales/salidas de relé máx. CPU 4 x EM 223 DI16/DO16 x DC24V/Relé 1 x EM 222 DO8 x Relé Total =	+660 -600 -40 0	14 64 78	10 64 8 82		
CPU 226					
Entradas digitales/salidas de relé máx. CPU 6 x EM 223 DI16/DO16 x DC24V/Relé 1 x EM 223 DI8/DO8 x DC24V/Relé Total =	+1000 -900 -80 0	24 96 8 128	16 96 8 120		
Entradas digitales/salidas DC máx. CPU 6 x EM 223 DI16/DO16 x DC24V 1 x EM 221 DI8 x DC24V Total =	+1000 -960 -30 0	24 96 8 128	16 96 112		
CPU 224 o CPU 226					
Entradas analógicas máx. CPU 7 x EM 235 AI4/AQ1 Total =	>660 -210 0	14 (24) 14 (24)	10 (16) 10 (16)	28 28	7 7
Salidas analógicas máx. CPU 7 x EM 232 AQ2 Total =	>660 -140 0	14 (24) 14 (24)	10 (16) 10 (16)	0 0	14 14

Instalar un Micro-PLC S7-200

Los sistemas de automatización S7-200 (Micro-PLCs) son fáciles de instalar. Se pueden montar bien sea en un armario eléctrico, utilizando los orificios de sujeción previstos para ello, o bien en un raíl normalizado (DIN), usando ganchos de retención. Sus pequeñas dimensiones permiten ahorrar espacio.

Este capítulo explica cómo instalar y cablear un sistema de automatización S7-200.

Índice del capítulo

Apartado	Descripción	Página
2.1	Preparar el montaje	2-2
2.2	Montar y desmontar un Micro-PLC S7-200 o un módulo de ampliación	2-6
2.3	Instalar el cableado de campo	2-9
2.4	Utilizar circuitos de supresión	2-16
2.5	Alimentación de corriente	2-18

2.1 Preparar el montaje

Disposición

Los sistemas de automatización S7-200 se pueden disponer en un armario eléctrico o en un raíl DIN. Es posible montarlos de forma horizontal o vertical. Un sistema de automatización S7-200 se puede conectar a un módulo de ampliación utilizando uno de los métodos siguientes:

- Un cable plano con el correspondiente conector está incorporado en el módulo de ampliación para poder conectarlo fácilmente a la CPU o a otro módulo de ampliación.
- Con objeto de flexibilizar aún más el montaje, se ofrecen también cables de conexión para los módulos de ampliación.

La figura 2-1 muestra dos ejemplos típicos de disposición.

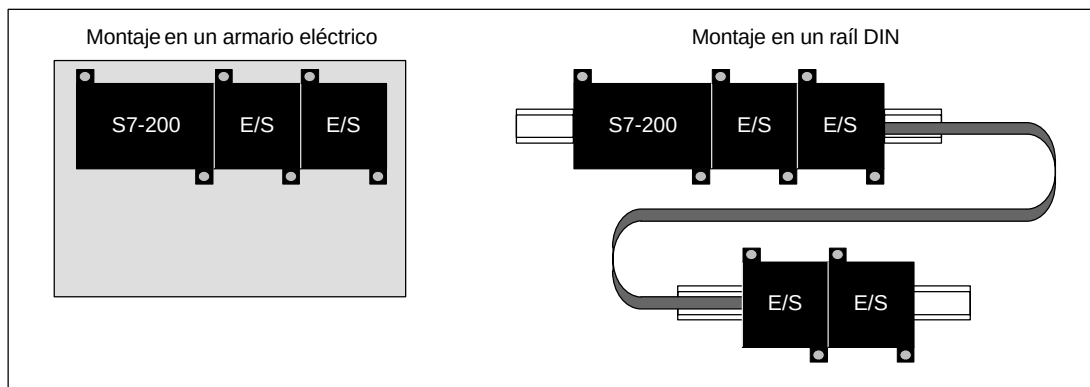


Figura 2-1 Disposición para el montaje

Espacio necesario para montar una CPU S7-200

Al planificar la disposición de los módulos en el bastidor se deberán respetar las siguientes reglas:

- Para las CPUs S7-200 y los módulos de ampliación se ha previsto la ventilación por convección natural. Por lo tanto, se deberá dejar un margen mínimo de 25 mm por encima y por debajo de las unidades para garantizar su ventilación (v. fig. 2-2). El funcionamiento continuo a una temperatura ambiente máxima y con una carga muy elevada reduce la vida útil de cualquier dispositivo electrónico.
- Para el montaje vertical, la temperatura ambiente máxima se reduce en 10° C. La CPU se debe montar debajo de los módulos de ampliación. Si el montaje se efectúa en un raíl DIN vertical es preciso utilizar un tope.
- Para la profundidad de montaje se deben prever 75 mm (v. fig. 2-2).
- Al planificar la disposición de los módulos, prevea suficiente espacio para el cableado de las entradas y salidas, así como para las conexiones de los cables de comunicación.

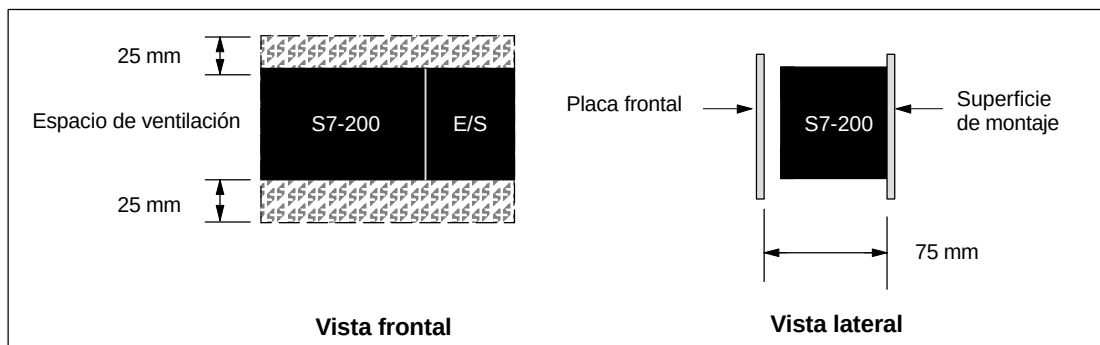


Figura 2-2 Espacio necesario para montar una CPU S7-200 de forma horizontal o vertical

Montaje en un raíl DIN

Las CPUs S7-200 y los módulos de ampliación se pueden montar en un raíl DIN estándar (DIN EN 50 022). La figura 2-3 muestra las dimensiones de dicho raíl.

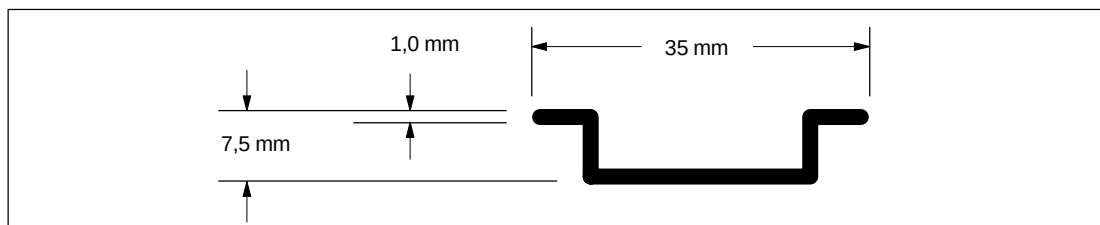


Figura 2-3 Dimensiones del raíl DIN

Dimensiones del armario eléctrico

Las CPUs S7-200 y los módulos de ampliación disponen de orificios para facilitar su montaje en armarios eléctricos. En las figuras 2-4 a 2-7 se indican las dimensiones necesarias para montar las diversas CPUs S7-200 y los módulos de ampliación.

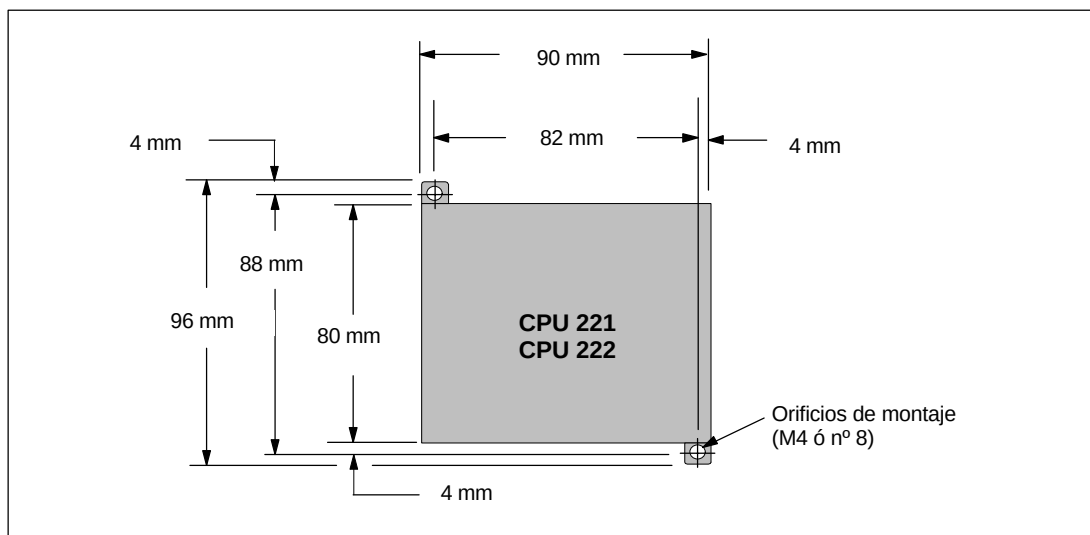


Figura 2-4 Dimensiones de montaje para una CPU 221 y una CPU 222

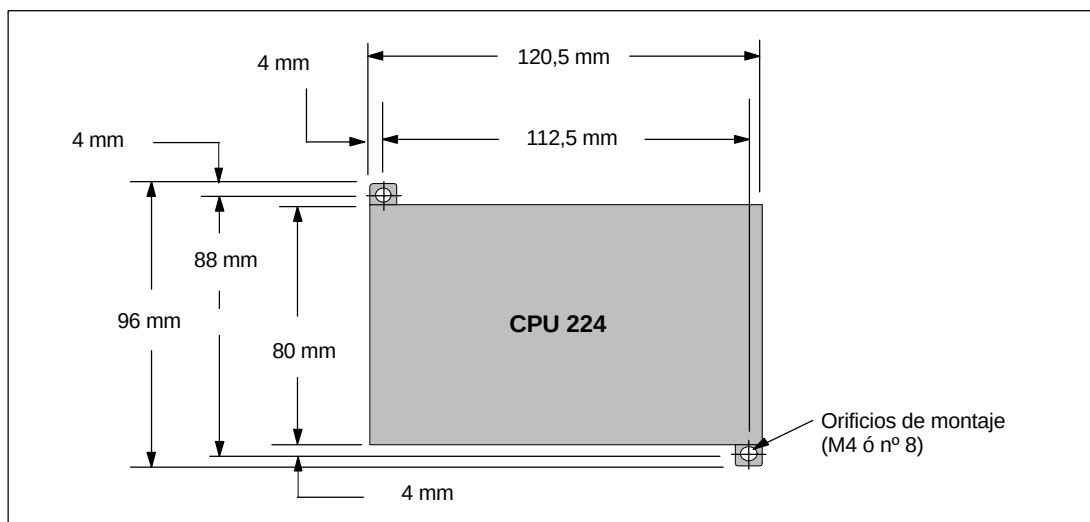


Figura 2-5 Dimensiones de montaje para una CPU 224

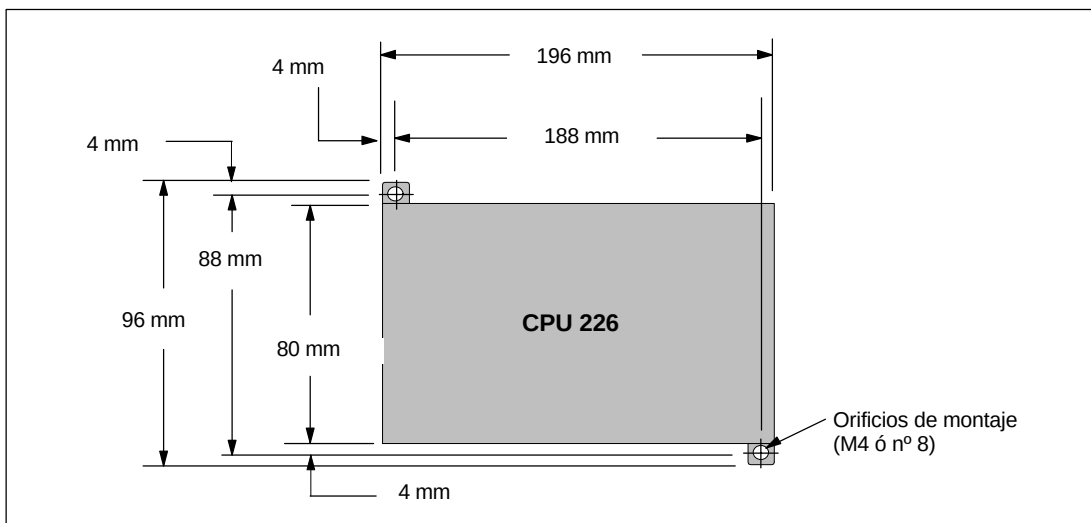


Figura 2-6 Dimensiones de montaje para una CPU 226

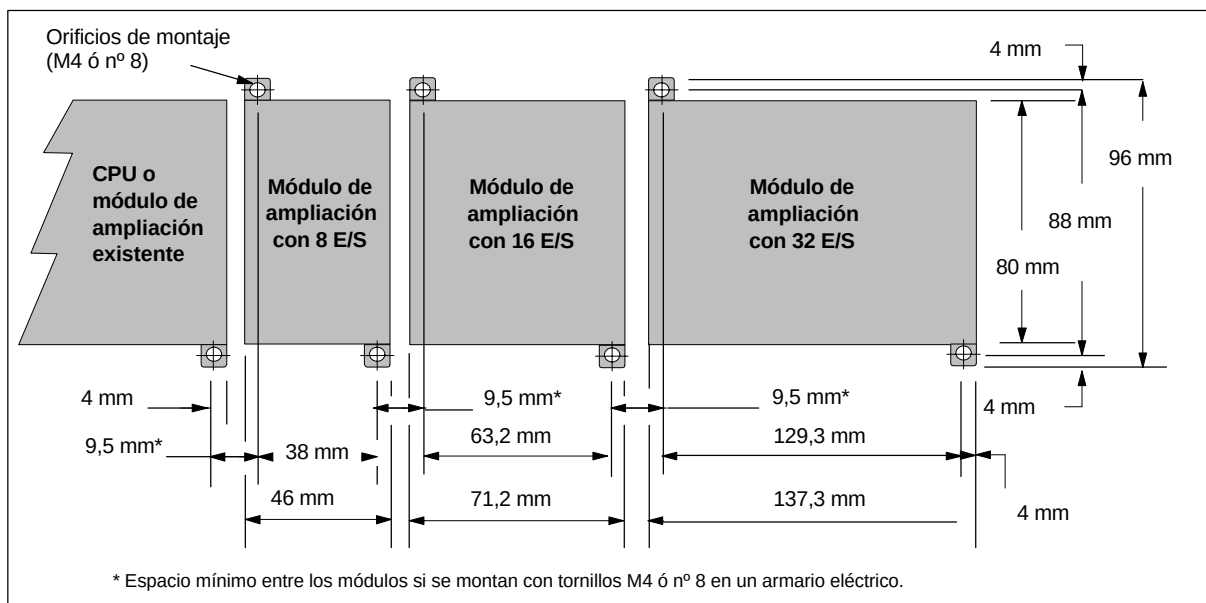


Figura 2-7 Dimensiones de montaje para módulos de ampliación

2.2 Montar y desmontar un Micro-PLC S7-200 o un módulo de ampliación

Montar un Micro-PLC S7-200 o un módulo de ampliación en un armario eléctrico



Precaución

Si se intenta montar o desmontar los sistemas de automatización y/o los equipos conectados a los mismos durante el encendido, puede producirse un choque eléctrico o fallos en los equipos.

Si antes del montaje o desmontaje no se ha desconectado por completo la alimentación eléctrica de los módulos S7-200 y de los equipos conectados, ello podría causar la muerte o heridas graves al personal, y/o daños materiales.

Respete siempre las medidas de seguridad necesarias y verifique que la alimentación eléctrica de los módulos S7-200 se haya desconectado antes del montaje.

Para montar una CPU S7-200 en un armario eléctrico, siga los siguientes pasos:

1. Posicione y taladre los orificios de sujeción para los tornillos de tamaño DIN M4 (estándar americano nº 8). En el apartado 2.1 se indican las dimensiones de montaje y otros datos importantes al respecto.
2. Atornille la CPU S7-200 al armario eléctrico, utilizando tornillos de tamaño DIN M4 (estándar americano nº 8).

Para montar un módulo de ampliación en un armario eléctrico, siga los siguientes pasos:

1. Posicione y taladre los orificios de sujeción para los tornillos de tamaño DIN M4 (estándar americano nº 8). En el apartado 2.1 se indican las dimensiones de montaje y otros datos importantes al respecto.
2. Coloque el módulo de ampliación cerca de la CPU o del módulo de ampliación y fíjelo correctamente.
3. Enchufe el cable plano del módulo de ampliación en el conector de la CPU ubicado debajo de la tapa frontal. El cable muestra la orientación correcta.

Así se finaliza el montaje.

Montar un Micro-PLC S7-200 o un módulo de ampliación en un raíl DIN



Precaución

Si se intenta montar o desmontar los sistemas de automatización y/o los equipos conectados a los mismos durante el encendido, puede producirse un choque eléctrico o fallos en los equipos.

Si antes del montaje o desmontaje no se ha desconectado por completo la alimentación eléctrica de las CPUs S7-200 y de los equipos conectados a las mismas, ello podría causar la muerte o heridas graves al personal, y/o daños materiales.

Respete siempre las medidas de seguridad necesarias y verifique que la alimentación eléctrica de los módulos S7-200 se haya desconectado antes del montaje.

Para montar una CPU S7-200 en un raíl DIN, siga los siguientes pasos:

1. Atornille el raíl DIN al armario eléctrico dejando un espacio de 75 mm entre tornillo y tornillo.
2. Abra el gancho de retención (ubicado en el lado inferior de la CPU S7-200) y enganche la parte posterior de la CPU al raíl DIN.
3. Cierre el gancho de retención y verifique que la CPU S7-200 haya enganchado correctamente en el raíl.

Para montar un módulo de ampliación en un raíl DIN, siga los siguientes pasos:

1. Abra el gancho de retención y enganche la parte posterior del módulo en el raíl próximo a la CPU o al módulo de ampliación.
2. Cierre el gancho de retención para fijar el módulo de ampliación al raíl. Asegúrese de que el módulo se haya enganchado correctamente en el raíl.
3. Enchufe el cable plano del módulo de ampliación en el conector de la CPU ubicado debajo de la tapa frontal. El cable muestra la orientación correcta.
4. Así se finaliza el montaje.

Nota

Si los módulos se montan en entornos donde se presenten vibraciones fuertes, o bien en posición vertical, puede ser necesario asegurarlos con topes.

Desmontar un Micro-PLC o un módulo de ampliación S7-200



Precaución

Si se intenta montar o desmontar los sistemas de automatización y/o los equipos conectados a los mismos durante el encendido, puede producirse un choque eléctrico o fallos en los equipos.

Si antes del montaje o desmontaje no se ha desconectado por completo la alimentación eléctrica de las CPUs S7-200 y de los equipos conectados a las mismas, ello podría causar la muerte o heridas graves al personal, y/o daños materiales.

Respete siempre las medidas de seguridad necesarias y verifique que la alimentación eléctrica de los módulos S7-200 se haya desconectado antes de proceder a su montaje.

Para desmontar una CPU o un módulo de ampliación S7-200, siga los siguientes pasos:

1. Desconecte todos los cables enchufados al módulo que desee desmontar (v. figura 2-8). Algunas CPUs y algunos módulos de ampliación disponen de conectores extraíbles.
2. Abra la tapa de acceso frontal y desconecte el cable plano de los módulos adyacentes.
3. Desatornille los tornillos de montaje o abra el gancho de retención y desmonte el módulo.



Precaución

Si instala un módulo incorrecto, es posible que el programa instalado en la CPU funcione de forma impredecible.

Si un módulo y un cable de ampliación se sustituyen con otro modelo o si no se instalan con la orientación correcta, ello podría causar la muerte o heridas graves al personal, y/o daños materiales.

Sustituya un módulo de ampliación con el mismo modelo y oriéntelo correctamente.

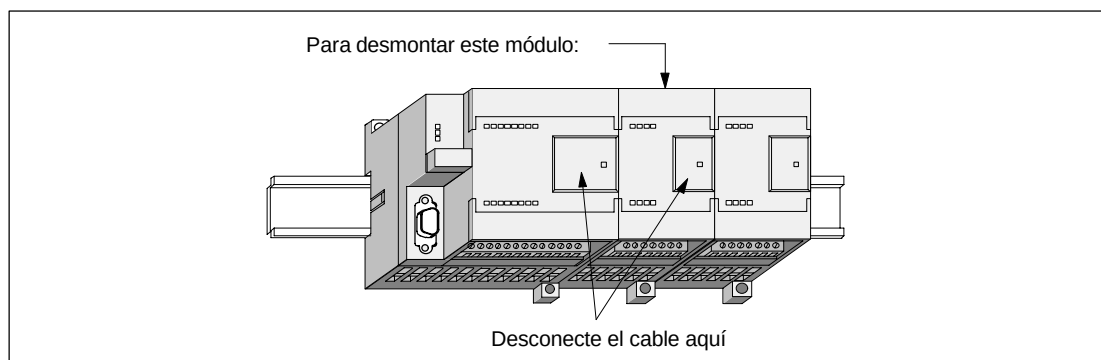


Figura 2-8 Desmontar un módulo de ampliación

2.3 Instalar el cableado de campo



Precaución

Si se intenta montar o desmontar los sistemas de automatización y/o los equipos conectados a los mismos durante el encendido, puede producirse un choque eléctrico o fallos en los equipos.

Si antes del montaje o desmontaje no se ha desconectado por completo la alimentación eléctrica de las CPUs S7-200 y de los equipos conectados a las mismas, ello podría causar la muerte o heridas graves al personal, y/o daños materiales.

Antes de instalar el cableado de campo es necesario tomar las precauciones de seguridad adecuadas y verificar que estén desconectadas las fuentes de alimentación de los sistemas de automatización S7-200.

Reglas de carácter general

Los puntos siguientes constituyen reglas de carácter general para la instalación y el cableado de los Micro-PLCs S7-200:

- Al cablear un Micro-PLC S7-200 es necesario respetar todos los reglamentos, códigos y normas eléctricas vinculantes. Instale y utilice el equipo conforme a todas las normas nacionales y locales vigentes. Diríjase a las autoridades locales para informarse acerca de qué reglamentos, códigos o normas rigen en el lugar de instalación.
- Utilice siempre cables con una sección adecuada para la intensidad. Los S7-200s aceptan cables con sección de 1,50 mm² a 0,50 mm² (14 AWG a 22 AWG).
- No apriete excesivamente los bornes de tornillo, pues podrían pasarse de rosca. El par máximo de apriete es de 0.56 N-m.
- Utilice siempre un cable lo más corto posible (apantallado o blindado, como máximo 500 metros, sin pantalla o blindaje, 300 metros). El cableado deberá efectuarse por pares; con el cable de neutro o común combinado con un cable de fase o uno de señal.
- Separe el cableado de corriente alterna y el cableado de corriente continua de alta tensión y rápida conmutación de los cables de señal de baja tensión.
- Identifique y disponga adecuadamente el cableado hacia los S7-200s. De ser necesario, prevea un alivio de tracción. Para obtener más información sobre la identificación de terminales o bornes, consulte las hojas de datos técnicos en el Anexo A.
- Instale dispositivos de supresión de sobretensiones apropiados en el cableado susceptible de recibir sobretensiones causadas por rayos.
- Ninguna alimentación externa deberá aplicarse a una carga de salida en paralelo con una salida de corriente continua (DC). En caso contrario puede circular corriente inversa a través de la salida a menos que se instale un diodo u otra barrera.



Precaución

Un funcionamiento anormal de los equipos de control puede causar un funcionamiento inesperado del equipo controlado.

Dicho funcionamiento inesperado puede causar la muerte o lesiones personales graves y/o daños al equipo.

Prevea dispositivos de parada de emergencia, dispositivos electromecánicos de mayor jerarquía y otras medidas redundantes de seguridad que sean independientes del sistema de automatización.

Reglas de puesta a tierra de referencia de potencial para circuitos aislados

Seguidamente se indican las reglas de puesta a tierra para circuitos aislados:

- Deberá identificarse el punto de referencia (referencia de tensión 0) para cada circuito de la instalación así como los puntos donde puedan interconectarse circuitos con referencias de potencial diferentes. Tal tipo de conexiones puede causar circulaciones parásitas de corriente con consecuencias indeseadas, tales como errores lógicos o circuitos deteriorados. Una causa muy común de diferentes potenciales de referencia son tomas de tierra que están separadas físicamente por una gran distancia. Cuando se interconectan dispositivos con tierras muy separadas a través de un cable de comunicación o de sensor, por el circuito creado por el cable y tierra pueden circular corrientes inesperadas. Las corrientes de carga de maquinaria pesada pueden causar, incluso con distancias reducidas, diferencias de potencial de tierra o generar corrientes indeseadas por fenómenos de inducción electromagnética. Las fuentes de alimentación que no tengan coordinada su referencia de potencial 0 pueden causar corrientes dañinas al circular entre sus circuitos asociados.
- Si una CPU con potencial de tierra diferente se conecta a una misma red PPI, es preciso utilizar un repetidor RS 485 aislado.
- Los productos S7-200 incluyen aislamientos en ciertos puntos para prevenir la circulación de corrientes indeseadas en la instalación. Al planear la instalación, se deberá considerar dónde existen tales elementos de aislamiento y dónde no. También se deberán considerar los puntos de aislamiento en fuentes de alimentación asociadas y otros equipos, así como los puntos que utilizan como referencia las fuentes de alimentación asociadas.
- Los puntos de referencia de tierra y los aislamientos que ofrece el equipo deberán elegirse de forma que se interrumpan bucles de circuito innecesarios que pueden causar la circulación de corrientes indeseadas. No olvide considerar aquellas conexiones temporales que pueden introducir cambios en el potencial de referencia de los circuitos, p.ej. la conexión de una unidad de programación a la CPU.
- Al definir físicamente las tierras es necesario considerar los requisitos de puesta a tierra de protección y el correcto funcionamiento de los aparatos de protección por corte.
- Para obtener el mayor nivel posible de inmunidad a interferencias en la mayoría de las instalaciones, la conexión M de la fuente de alimentación de sensores se deberá conectar a tierra.

Las descripciones siguientes constituyen una introducción a las características de aislamiento generales de la gama S7-200. Sin embargo, algunas prestaciones pueden diferir en determinados productos. Las hojas de datos técnicos en el Anexo A incluyen información sobre los puntos de aislamiento y los valores nominales de los mismos para determinados productos. Los aislamientos con valores nominales inferiores a AC 1.500 V han sido diseñados únicamente como aislamiento funcional y no deberán tomarse para definir barreras de seguridad.

- El potencial de referencia de la lógica de circuito es el mismo que el de la conexión M de la fuente de alimentación DC de sensores.
- El potencial de referencia de la lógica de circuito es el mismo que el punto M de la alimentación de entrada en el caso de una CPU con alimentación en corriente continua.
- Los puertos de comunicación de la CPU tienen el mismo potencial de referencia que la lógica de circuito.
- Las entradas y salidas (E/S) analógicas no están aisladas con respecto a la lógica de circuito. Las entradas analógicas son de tipo diferencial, es decir tienen una baja razón de rechazo en modo común.
- La lógica de circuito está aislada de la tierra hasta AC 500 V.
- Las E/S digitales en DC están aisladas de la lógica de la CPU hasta AC 500 V.
- Los grupos de E/S digitales en DC están aislados unos de otros hasta AC 500 V.
- Las salidas de relé están aisladas de la lógica de circuito hasta AC 1.500 V.
- Los grupos de salida están aislados unos de otros hasta AC 1.500 V.
- La fase y el neutro de la alimentación en alterna están aislados de tierra, la lógica de circuito y todas las E/S hasta AC 1.500 V.

Utilizar el bloque de bornes opcional en CPUs sin conector extraíble

El bloque de bornes opcional para el cableado de campo (v. fig. 2-9) tiene la ventaja de que las conexiones permanecen fijas aun al desmontar o montar la CPU S7-200. En el Anexo E se indica el número de referencia de dicho bloque.

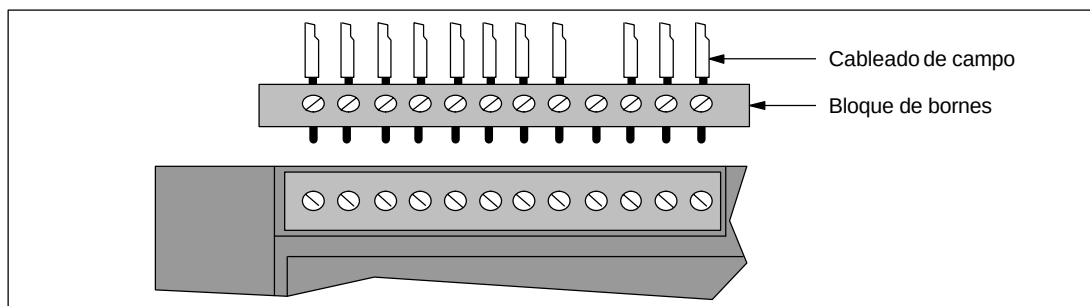


Figura 2-9 Bloque de bornes opcional

Bloque de terminales extraíble

El bloque de terminales extraíble para el cableado de campo (v. fig. 2-10) tiene la ventaja de que las conexiones permanecen fijas aun al desmontar o montar la CPU S7-200 y los módulos de ampliación.

Para soltar el bloque de terminales de la CPU o del módulo de ampliación, siga los siguientes pasos:

1. Levante la tapa de acceso superior de la CPU o del módulo de ampliación.
2. En la mitad del bloque de terminales, inserte un destornillador en la ranura como muestra la figura 2-10.
3. Empuje hacia abajo y haga palanca para soltar el bloque de terminales como se muestra abajo.

Para enganchar nuevamente un bloque de terminales en una CPU o en un módulo de ampliación, siga los siguientes pasos:

1. Levante la tapa de acceso superior de la CPU o del módulo de ampliación.
2. Verifique que el nuevo bloque de terminales esté alineado correctamente con los pines de la CPU o del módulo de ampliación.
3. Empuje el bloque de terminales hacia abajo hasta que enganche correctamente en la CPU o en el módulo de ampliación.

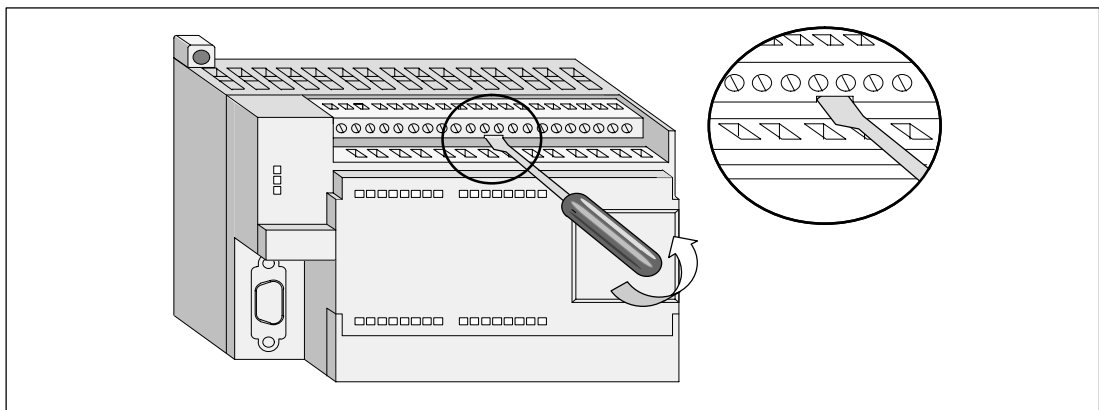


Figura 2-10 Bloque de terminales extraíble para la CPU 224 y los módulos de ampliación.

Reglas para la instalación con corriente alterna

Seguidamente se indican reglas de carácter general para instalaciones con corriente alterna (v. fig. 2-11).

[a] Instale un interruptor unipolar para cortar la alimentación de la CPU, todos los circuitos de entrada y todos los circuitos de salida (de carga).

[b] Prevea dispositivos de sobrecorriente para proteger la alimentación de la CPU, las salidas y las entradas. Para mayor protección es posible instalar un fusible en cada salida.

[c] No se precisa protección de sobrecorriente externa para las entradas si se utiliza la fuente de alimentación de DC 24 V para sensores integrada en el Micro-PLC. Esta fuente para sensores está protegida contra cortocircuitos.

[d] Conecte todos los terminales de tierra del S7-200 por el camino más corto a tierra para obtener el mayor nivel posible de inmunidad a interferencias. Es recomendable conectar todos los terminales de masa a un solo punto eléctrico. Para establecer esta conexión, utilice un cable con una sección de 14 AWG ó 1.5 mm².

[e] La fuente de alimentación DC para sensores integrada en la CPU puede usarse también para alimentar las entradas de dicho componente

[f] las entradas DC de ampliación y [g] las bobinas de los relés del módulo de ampliación. Esta fuente para sensores está protegida contra cortocircuitos.

[h] Para obtener el mayor nivel posible de inmunidad a interferencias en la mayoría de las instalaciones, la conexión M de la fuente de alimentación de sensores se deberá conectar a tierra.

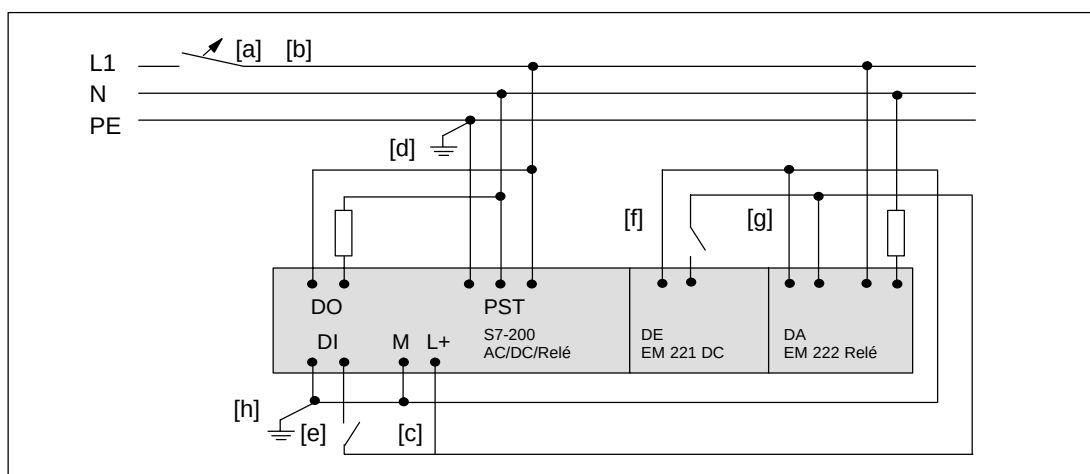


Figura 2-11 AC 120/230 V Uso de un interruptor de sobrecorriente único para proteger la CPU y la carga

Reglas para la instalación con corriente continua

Seguidamente se indican reglas de carácter general para instalaciones con corriente continua (v. fig. 2-12).

[a] Instale un interruptor unipolar para cortar la alimentación de la CPU, todos los circuitos de entrada y todos los circuitos de salida (de carga).

[b] Prevea dispositivos de sobrecorriente para proteger la alimentación de la CPU, [c] las salidas y [d] las entradas. Para mayor protección es posible instalar un fusible en cada salida. No se precisa protección de sobrecorriente externa para las entradas si se utiliza la fuente de alimentación de DC 24 V para sensores integrada en el Micro-PLC. Esta última fuente de alimentación dispone de una función de limitación interna de corriente.

[e] Verifique que la fuente de alimentación DC tenga suficiente capacidad para mantener la tensión en caso de que se produzcan cambios bruscos de carga. De no ser así, prevea condensadores externos adecuados.

[f] Para obtener el mayor nivel posible de inmunidad a interferencias en la mayoría de las instalaciones, es recomendable conectar a tierra todas las fuentes de alimentación DC. Equipe las fuentes de alimentación DC no puestas a tierra con una resistencia y un condensador en paralelo [g] conectado entre el común de la alimentación y el conductor de protección. Dicha resistencia ofrece una vía de fuga para prevenir acumulaciones de carga estática; el condensador permite derivar las interferencias de alta frecuencia. Los valores típicos son 1 MΩ y 4.700 pF.

[h] Conecte todos los terminales de tierra del S7-200 por el camino más corto a tierra para obtener el mayor nivel posible de inmunidad a interferencias. Es recomendable conectar todos los terminales de masa a un solo punto eléctrico. Para establecer esta conexión, utilice un conductor con un sección de 14 AWG ó 1,5 mm².

Para alimentar circuitos de DC 24 V, utilice siempre una fuente que ofrezca separación eléctrica segura de la red de AC 120/230 V y fuentes de peligro similares.

Los documentos siguientes incluyen definiciones de separación segura de circuitos:

- Protected extra low voltage: conforme a EN60204-1
- Clase 2 ó Limited Voltage/Current Circuit conforme a UL 508

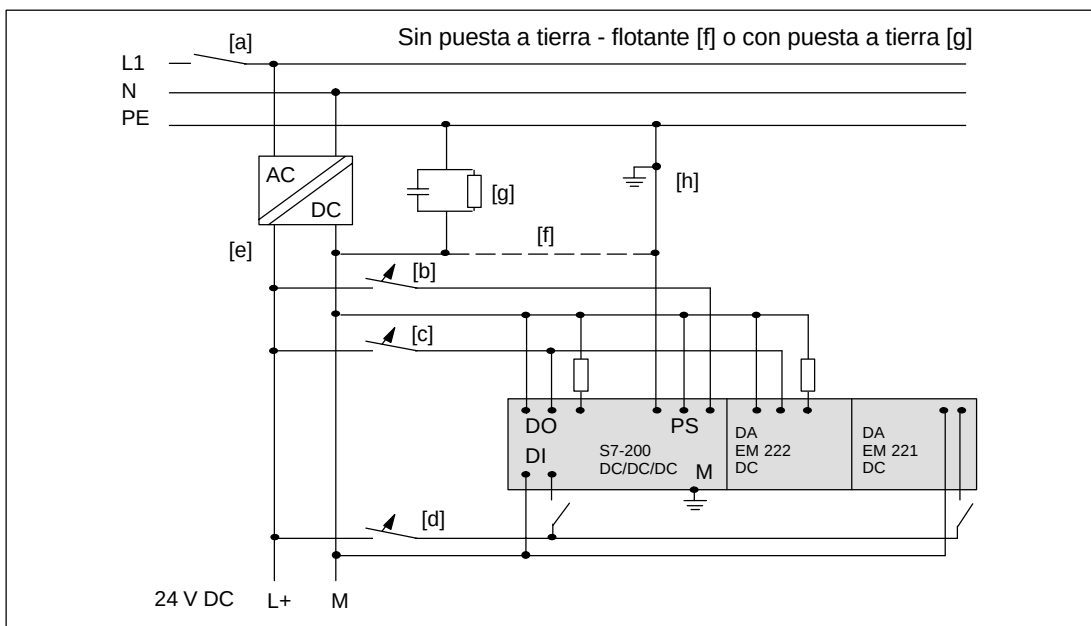


Figura 2-12 Instalación en una red de corriente continua

2.4 Utilizar circuitos de supresión

Reglas de carácter general

Las cargas inductivas deberán equiparse con circuitos supresores destinados a limitar el incremento de tensión producido al cortarse la alimentación. Tenga en cuenta las reglas siguientes al diseñar la supresión adecuada. La eficacia de un determinado diseño depende de la aplicación. Por tanto, deberá verificarse para cada caso particular. Verifique que las características nominales de todos los componentes sean adecuadas para la aplicación en cuestión.

Proteger transistores en DC

Las salidas en DC a transistores del S7-200 incorporan fijadores controlados por diodos Zener, que se adecúan para múltiples aplicaciones. Para prevenir la sobrecarga de los circuitos internos, utilice diodos supresores externos en caso de cargas inductivas elevadas o que se conmuten con frecuencia. Las figuras 2-13 y 2-14 muestran aplicaciones típicas para salidas DC a transistor.

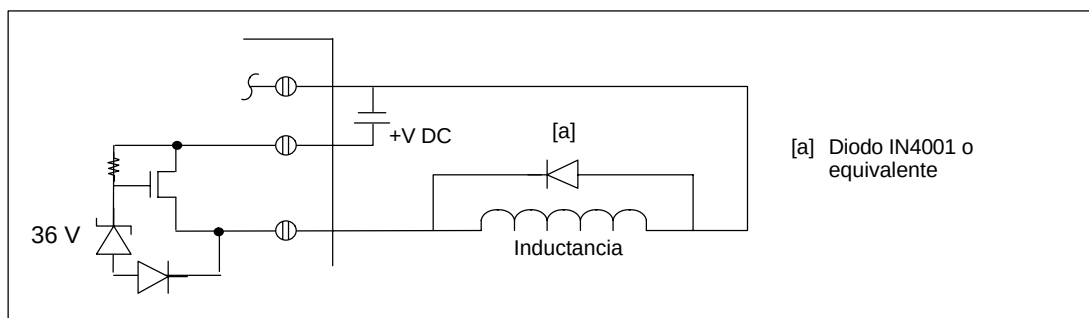


Figura 2-13 Protección por diodo para salidas en DC a transistores

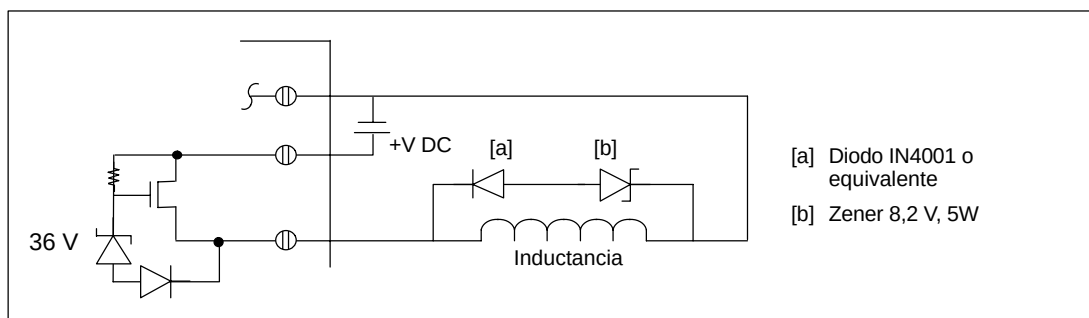


Figura 2-14 Protección por diodo Zener para salidas en DC a transistores

Proteger relés que controlan corriente continua

La figura 2-15 muestra las redes de resistencia/condensador que se pueden utilizar para aplicaciones de relé en baja tensión (30 V) DC. Conecte la red en los terminales de la carga.

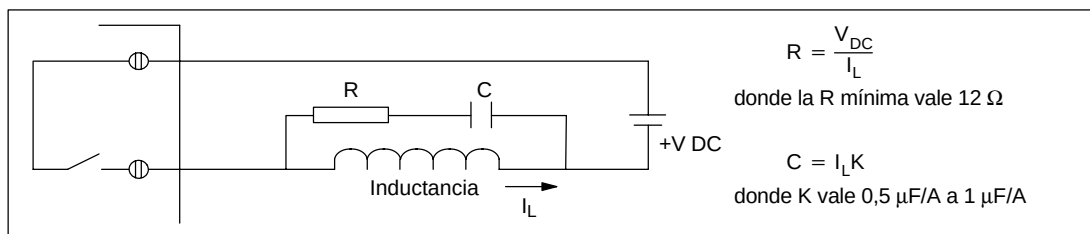


Figura 2-15 Red resistencia/condensador conectada a una carga DC controlada por relés

Para las aplicaciones con relé en corriente continua se puede utilizar también la protección por diodo supresor como muestran las figuras 2-13 y 2-14. Se permite una tensión de umbral de hasta 36 V si se utiliza un diodo Zener conectado de forma inversa.

Proteger relés que controlan corriente alterna

Si se utiliza un relé para conmutar cargas inductivas con AC 115 V/230 V, se deben conectar redes de resistencia/condensador entre los contactos del relé como muestra la figura 2-16. También pueden utilizarse varistores de óxido metálico (MOV) para limitar la tensión de pico. Verifique que la tensión de trabajo del varistor MOV sea como mínimo un 20% superior a la tensión nominal de fase.

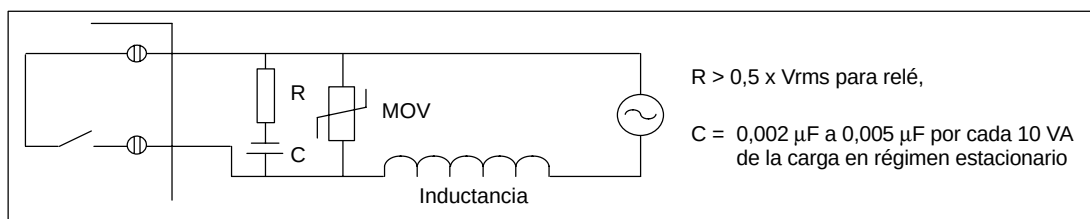


Figura 2-16 Carga AC con red conectada entre contactos del relé

El condensador permite la circulación de la corriente de fugas aunque esté abierto el contacto. Verifique que la corriente de fugas, $I \text{ (fuga)} = 2 \times 3,14 \times f \times C \times V_{ef}$, sea aceptable para la aplicación.

Por ejemplo: Las especificaciones de un contactor NEMA de tamaño 2 muestran un consumo transitorio de la bobina de 183 VA y un consumo de 17 VA en régimen estacionario. Con AC 115 V, la corriente transitoria vale $183 \text{ VA}/115 \text{ V} = 1,59 \text{ A}$, es decir, es inferior a la capacidad de cierre, que vale 2A, de los contactos del relé.

Dimensionamiento de la resistencia = $0,5 \times 115 = 57,5 \Omega$; elegir 68 Ω como valor estándar. Dimensionamiento del condensador = $(17 \text{ VA}/10) \times 0,005 = 0,0085 \mu F$; elegir 0,01 μF como valor estándar.

La corriente de fugas vale = $2 \times 3,14 \times 60 \times 0,01 \times 10^{-6} \times 115 = 0,43 \text{ mA}$ eficaces (rms).

2.5 Alimentación de corriente

Las CPU del S7-200 tienen integrada una fuente de alimentación capaz de abastecer la CPU, los módulos de ampliación y otras cargas que precisen DC 24 V. Utilice la información siguiente como guía para determinar cuánta energía (o corriente) puede suministrar la CPU a la configuración deseada.

Requisitos de alimentación

Cada CPU S7-200 ofrece alimentación tanto en DC 5 V como DC 24 V:

- Cada CPU dispone de una fuente de alimentación para sensores de DC 24 V que puede suministrar esta tensión para las entradas locales o para las bobinas de relés en los módulos de ampliación. Si el consumo de DC 24 V supera la corriente que es capaz de aportar el módulo CPU, entonces puede añadirse una fuente de alimentación externa de DC 24 V para abastecer con DC 24 V los módulos de ampliación. La alimentación de DC 24 V se debe conectar manualmente a dichas entradas o bobinas de relé.
- La CPU alimenta también con DC 5 V los módulos de ampliación cuando se conectan al módulo base. Si el consumo de DC 5 V de los módulos de ampliación supera la corriente aportable por la CPU, entonces es necesario desconectar tantos módulos de ampliación como sean necesarios para no superar la corriente aportable por la CPU.

Las hojas de datos técnicos que se incluyen en el Anexo A informan sobre las corrientes suministrables por las CPUs y sobre el consumo de los módulos de ampliación.



Precaución

Si se conecta una fuente de alimentación externa de DC 24 V en paralelo con la fuente de alimentación para sensores DC del S7-200, esto puede causar un conflicto entre ambas fuentes ya que cada una intenta establecer su propio nivel de tensión de salida.

Este conflicto puede tener como consecuencia una reducción de la vida útil o la avería inmediata de una o ambas fuentes de alimentación y, en consecuencia, el funcionamiento imprevisible del sistema de automatización. Un funcionamiento imprevisible puede ocasionar la muerte o lesiones graves al personal, y/o daños al equipo o a bienes materiales.

La fuente de alimentación para sensores del S7-200 y la fuente de alimentación externa deben alimentar diferentes puntos. Se permite una conexión común de los cables neutros.

Ejemplo de cálculo de los requisitos de alimentación

La tabla 2-1 muestra un ejemplo de cálculo de los requisitos de alimentación de un Micro-PLC S7-200 compuesto de los módulos siguientes:

- CPU 224 AC/DC/Relé
- 3 módulos de ampliación EM 223, 8 entradas DC / 8 salidas de relé
- 1 módulo de ampliación EM 221, 8 entradas DC

Esta instalación tiene un total de 46 entradas y 34 salidas.

La CPU de este ejemplo suministra suficiente corriente (DC 5 V) para los módulos de ampliación, pero la alimentación de sensores no suministra suficiente corriente DC 24 V para todas las entradas y salidas de relé. Las E/S requieren 400 mA, pero la CPU sólo puede suministrar 280 mA. Para esta instalación se necesita una fuente adicional de alimentación de 120 mA (como mínimo) y con corriente DC 24 V para que todas las entradas y salidas puedan funcionar correctamente.

Tabla 2-1 Cálculo de requisitos de alimentación en una configuración de ejemplo

Corriente máx. CPU	DC 5 V	DC 24 V
CPU 224 AC/DC/Relé	660 mA	280 mA
menos		
Consumo del sistema	DC 5 V	DC 24 V
CPU 224, 14 entradas		14 * 4 mA = 56 mA
3 EM 223, alimentación necesaria de 5 V	3 * 80 mA = 240 mA	
1 EM 221, alimentación necesaria de 5V	1 * 30 mA = 30 mA	
3 EM 223, 8 entradas c/u		3 * 8 * 4 mA = 96 mA
3 EM 223, 8 salidas de relé c/u		3 * 8 * 9 mA = 216 mA
1 EM 221, 8 entradas		8 * 4 mA = 32 mA
Consumo total	270 mA	400 mA
igual a		
Balance de corriente	DC 5 V	DC 24 V
Balance total de corriente	390 mA	[120 mA]

Calcular la corriente necesaria

Utilice la tabla siguiente para determinar cuánta corriente (o energía) puede suministrar la CPU a la configuración en cuestión. Consulte el anexo A para obtener información sobre las corrientes suministrables por la CPU y el consumo de los módulos de ampliación.

Alimentación	DC 5 V	DC 24 V

menos

Consumo del sistema	DC 5 V	DC 24 V
Consumo total		

igual a

Balance de corriente	DC 5 V	DC 24 V
Balance total de corriente		

Instalación y configuración del sistema de automatización S7-200

3

El presente capítulo describe cómo instalar y configurar un sistema de automatización S7-200. El sistema de automatización S7-200 descrito aquí comprende:

- Una CPU S7-200.
- Un PC o una unidad de programación con STEP 7-Micro/WIN 32 instalado.
- Un cable de interconexión.

Índice del capítulo

Apartado	Descripción	Página
3.1	Resumen breve	3-2
3.2	Instalar STEP 7-Micro/WIN 32	3-3
3.3	Configurar la comunicación utilizando el cable PC/PPI	3-5
3.4	Establecer la comunicación con la CPU	3-9
3.5	Cambiar los parámetros de comunicación de la CPU	3-10

3.1 Resumen breve

Información general

Para la instalación se deben tener en cuenta los siguientes criterios:

- El sistema operativo utilizado (Windows 95, Windows 98, o Windows NT 4.0).
- El tipo de hardware utilizado, p.ej.:
 - PC con cable PC/PPI
 - PC o unidad de programación SIMATIC con procesador de comunicaciones (CP)
 - CPU 221, CPU 222, CPU 224, CPU 226
 - Módem
- La velocidad de transferencia utilizada.

Hardware y software recomendados

STEP 7-Micro/WIN 32, versión 3.1 y STEP 7-Micro/WIN 32 Toolbox son aplicaciones de software que soportan los entornos Windows 95 (de 32 bits), Windows 98 y Windows NT. Para poder utilizar el software se recomiendan los siguientes componentes:

- Un panel táctil TP070 para su utilización con STEP 7-Micro/WIN 32 Toolbox.
- Un ordenador personal (PC) con un procesador 80586 o superior y 16 MB de RAM, o bien una unidad de programación Siemens con STEP 7-Micro/WIN 32 instalado (p.ej. una PG 740). Como mínimo se necesita un procesador 80486 con 8 MB de RAM.
- Uno de los componentes siguientes:
 - Un cable PC/PPI conectado al puerto de comunicación.
 - Una tarjeta de procesador de comunicaciones (CP).
- Una pantalla VGA o cualquier otra pantalla soportada por Microsoft Windows que tenga una resolución de 1024 X 768.
- 50 MB libres en el disco duro (como mínimo).
- Windows 95, Windows 98 o Windows NT 4.0.
- Opcional pero recomendable: un ratón asistido por Microsoft Windows.

STEP 7-Micro/WIN 32 incorpora una extensa Ayuda en pantalla y una *Guía de iniciación*, también en pantalla. Elija el comando de menú **Ayuda** o pulse la tecla F1 para obtener las informaciones más recientes.

STEP 7-Micro/WIN 32 Toolbox incluye además el software de configuración del panel táctil TP070, así como las operaciones del protocolo USS para los accionamientos MicroMaster.

3.2 Instalar STEP 7-Micro/WIN 32

Instrucciones previas a la instalación

Siga los siguientes pasos antes de instalar el programa:

- Si ya está instalada una versión anterior de STEP 7-Micro/WIN 32, cree una copia de seguridad de todos los proyectos de STEP 7-Micro/WIN en un disquete.
- Verifique que todas las demás aplicaciones estén cerradas, incluyendo la barra de herramientas de Microsoft Office.
- Verifique que esté conectado el cable entre el PC y la CPU. Consulte el apartado 3.3 para obtener más información al respecto.

Instalación de STEP 7-Micro/WIN 32

Para instalar STEP 7-Micro/WIN 32, siga los siguientes pasos:

1. Inserte el CD o el disquete en la correspondiente unidad del PC.
2. Haga clic en el botón "Inicio" para abrir el menú de Windows.
3. Haga clic en **Ejecutar....**
4. Si la instalación se efectúa desde un disquete: En el cuadro de diálogo "Ejecutar", teclee **a:\setup** y haga clic en el botón "Aceptar" o pulse la tecla INTRO. Así se inicia la instalación.

Si la instalación se efectúa desde un CD: En el cuadro de diálogo "Ejecutar", teclee **e:\setup** (donde "e" es la letra correspondiente a la unidad de CD-ROM) y haga clic en el botón "Aceptar" o pulse la tecla INTRO. Así se inicia la instalación.

5. Siga las instrucciones que van apareciendo en pantalla hasta finalizar la instalación.
6. Al final de la instalación aparecerá automáticamente el cuadro de diálogo "Ajustar interface PG/PC". El ajuste de los parámetros del interface PG/PC se explica más adelante en el presente capítulo. Haga clic en el botón "Cancelar" para continuar.
7. Aparecerá el cuadro de diálogo "Fin de la instalación", con una de las siguientes opciones:
 - 1ª opción: Sí, deseo reiniciar mi ordenador en este momento. (Ajuste estándar)
No, reiniciaré mi ordenador más adelante.

Si se visualiza la 1ª opción, es recomendable que acepte los ajustes estándar y haga clic en el botón "Finalizar" para concluir la instalación y leer el archivo README (LÉAME) que contiene las informaciones más recientes acerca de STEP 7-Micro/WIN 32.

- 2ª opción: Sí, deseo leer el archivo LÉAME ahora mismo. (Ajuste estándar)
Sí, deseo arrancar STEP 7-Micro/WIN 32 ahora.

Si se visualiza la 2ª opción, es recomendable que acepte los ajustes estándar y haga clic en el botón "Finalizar" para concluir la instalación y leer el archivo README (LÉAME) que contiene las informaciones más recientes acerca de STEP 7-Micro/WIN 32.

En la 2ª opción se pueden seleccionar las dos casillas de verificación. Haciendo clic en el botón "Finalizar" al estar seleccionadas las dos casillas, se concluirá la instalación, se visualizará el archivo LÉAME y se arrancará STEP 7-Micro/WIN 32 (v. fig. 3-1).

Lea el archivo READMEX.TXT incluido en el CD o en los disquetes de instalación para obtener las informaciones más recientes acerca de STEP 7-Micro/WIN 32. (En lugar de x, aparecerá la letra A = alemán, B = inglés, C = francés, D = español, E = italiano).

Nota

Si STEP 7-Micro/WIN se instala en el mismo directorio en el que ya esté instalado el software, se desinstalará la versión existente.

Fallos de comunicación

Las siguientes situaciones pueden causar fallos de comunicación:

- Velocidad de transferencia incorrecta: corrija la velocidad de transferencia.
- Dirección de estación incorrecta: corrija la dirección de estación.
- Cable PC/PPI ajustado incorrectamente: verifique los ajustes de los interruptores DIP del cable PC/PPI.
- Puerto de comunicaciones incorrecto en el PC: verifique el puerto COM.
- CPU en modo Freeport (puerto de comunicaciones bajo control del programa de usuario): cambie la CPU a modo STOP.
- Conflicto con otros maestros: desconecte la CPU de la red.

3.3 Configurar la comunicación utilizando el cable PC/PPI

Aquí se explica cómo configurar la comunicación entre la CPU S7-200 y el PC utilizando el cable PC/PPI. Esta es una configuración con un solo maestro y sin ningún otro equipo de hardware instalado (como p.ej. un módem o una unidad de programación).

Conectar el PC a la CPU

La figura 3-1 muestra una configuración típica para conectar el PC a la CPU utilizando el cable PC/PPI. Para establecer un enlace correcto entre los componentes:

1. Ajuste los interruptores DIP del cable PC/PPI a la velocidad de transferencia asistida por su PC. Seleccione también las opciones "11 bits" y "DCE" si su cable PC/PPI las asiste.
2. Conecte el extremo RS-232 ("PC") del cable PC/PPI al puerto de comunicaciones de su PC (COM1 ó COM2) y apriete los tornillos.
3. Conecte el extremo RS-485 ("PPI") del cable PC/PPI al puerto de comunicaciones de la CPU y apriete los tornillos.

Los datos técnicos del cable PC/PPI se indican en el Anexo A. Su número de referencia figura en el Anexo E. Para obtener informaciones más detalladas acerca de redes con varios enlaces, consulte el capítulo 7.

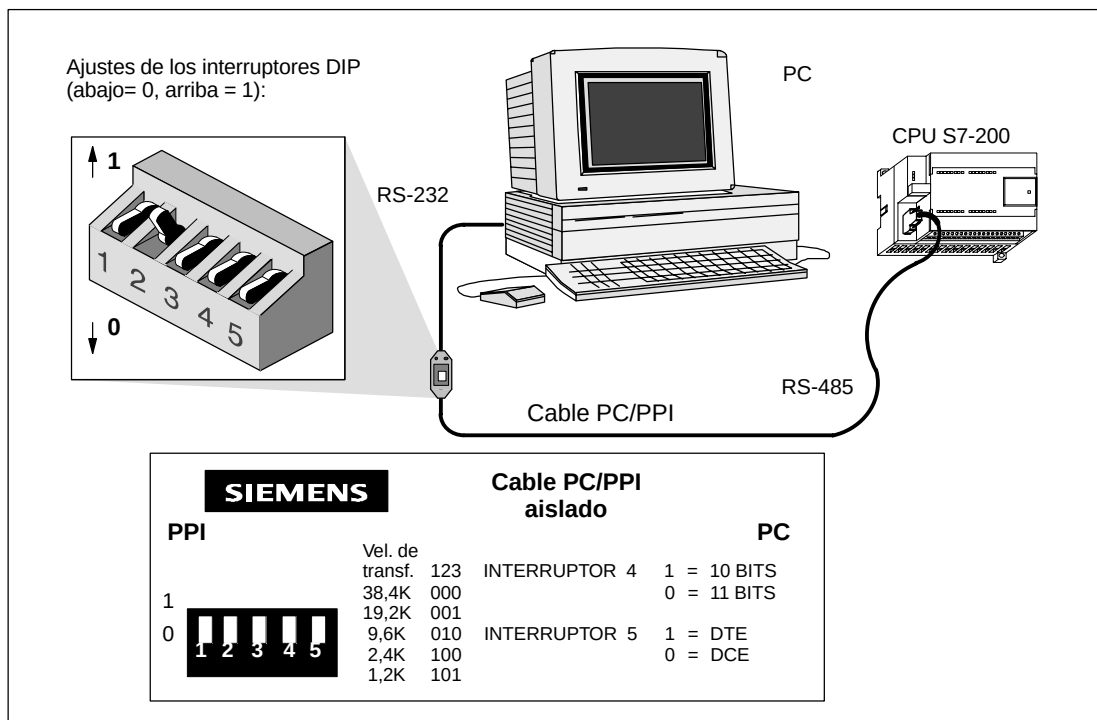


Figura 3-1 Comunicación con una CPU en modo PPI

Verificar los parámetros estándar del interface de comunicación

Para verificar los parámetros estándar de su interface, siga los siguientes pasos:

1. En la ventana de STEP 7-Micro/WIN 32, haga clic en el icono "Comunicación" o elija el comando de menú **Ver > Comunicación**. Aparecerá el cuadro de diálogo "Enlaces de comunicación".
2. En el cuadro de diálogo "Enlaces de comunicación", haga doble clic en el icono del cable PC/PPI. Aparecerá el cuadro de diálogo "Ajustar interface PG/PC" (v. fig. 3-2).
3. Haga clic en el botón "Propiedades" para acceder al cuadro de diálogo donde se visualizan las propiedades del interface (v. fig. 3-3). Verifique las propiedades. La velocidad de transferencia estándar es de 9.600 kbit/s.

Para obtener información sobre cómo cambiar los parámetros estándar, consulte el apartado 7.3 del capítulo 7.

Nota

Si el hardware que está utilizando no aparece en la lista visualizada en el cuadro de diálogo "Ajustar interface PG/PC", deberá instalar el hardware correcto (v. apt. 7.2).



Figura 3-2 Ajustes en el cuadro de diálogo "Interface PG/PC"

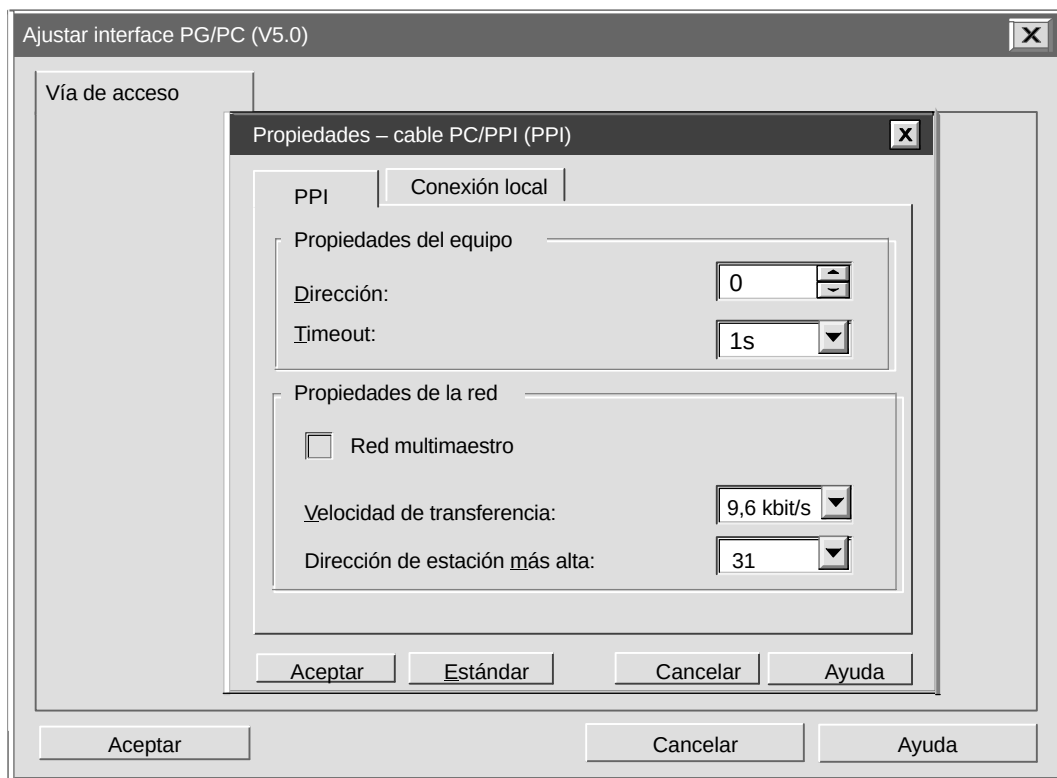


Figura 3-3 Cuadro de diálogo "Propiedades – Interface PG/PC"

3.4 Establecer la comunicación con la CPU

Tras haber instalado el software STEP 7-Micro/WIN 32 en el PC y determinado los ajustes de comunicación del cable PC/PPI, podrá establecer un enlace con la CPU S7-200. (Si está utilizando una unidad de programación, STEP 7-Micro/WIN 32 ya estará instalado).

Para establecer la comunicación con la CPU S7-200, siga los siguientes pasos:

1. En la pantalla de STEP 7-Micro/WIN 32, haga clic en el icono "Comunicación" o elija el comando de menú **Ver > Comunicación**. Aparecerá el cuadro de diálogo "Enlaces de comunicación" donde se indica que no hay ninguna CPU conectada.
2. En el cuadro de diálogo "Enlaces de comunicación", haga doble clic en el icono "Actualizar". STEP 7-Micro/WIN 32 verifica si hay CPUs S7-200 (estaciones conectadas), hasta la dirección de estación más alta indicada en la configuración de los parámetros de comunicación. Por cada estación conectada aparecerá un icono de CPU en el cuadro de diálogo "Enlaces de comunicación" (v. fig. 3-4).
3. Haga doble clic en el icono de la CPU con la que desea establecer la comunicación. Como podrá apreciar, los parámetros de comunicación visualizados en el cuadro de diálogo corresponden a la estación seleccionada.
4. Así queda establecido el enlace con la CPU S7-200.

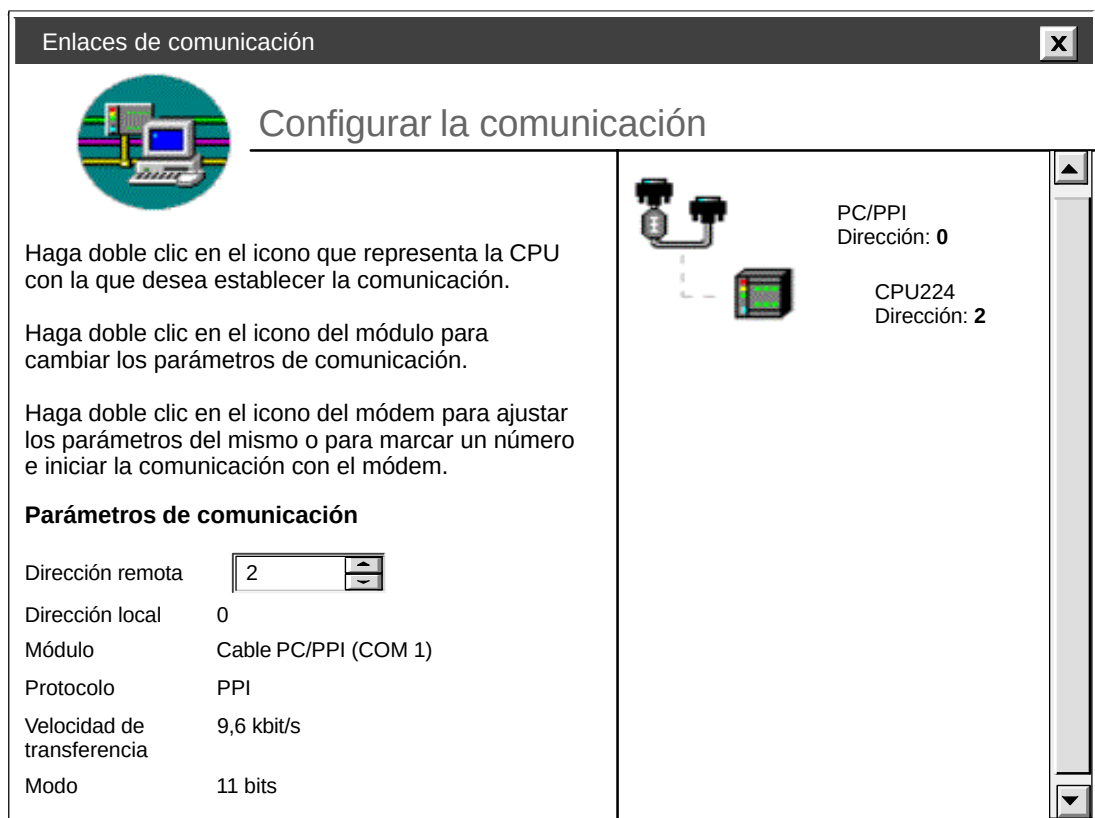


Figura 3-4 Cuadro de diálogo "Enlaces de comunicación"

3.5 Cambiar los parámetros de comunicación de la CPU

Tras haber establecido un enlace con la CPU S7-200, puede verificar o cambiar los parámetros de comunicación de la CPU.

Para cambiar los parámetros de comunicación, siga los siguientes pasos:

1. En la barra de navegación, haga clic en el icono "Bloque de sistema" o elija el comando de menú **Ver > Bloque de sistema**.
2. Aparecerá el cuadro de diálogo "Bloque de sistema". Haga clic en la ficha "Puerto(s)" (v. fig. 3-5). El ajuste estándar de la dirección de estación es 2 y el de la velocidad de transferencia es de 9,6 kbit/s.
3. Haga clic en "Aceptar" para conservar esos parámetros. Si desea modificar la parametrización, efectúe los cambios deseados y haga clic en el botón "Aceptar".
4. En la barra de herramientas, haga clic en el botón "Cargar en CPU" para cargar los cambios en la CPU.

Así se adopta la parametrización deseada para la comunicación.

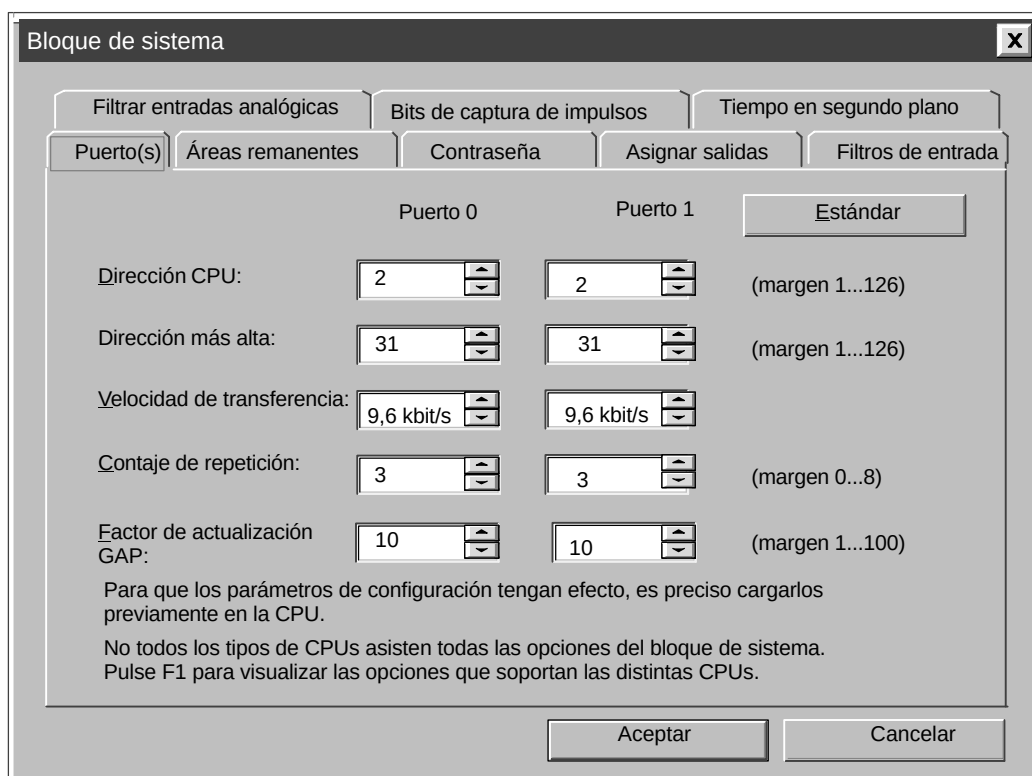


Figura 3-5 Cambiar los parámetros de comunicación

Nociones básicas para programar una CPU S7-200

4

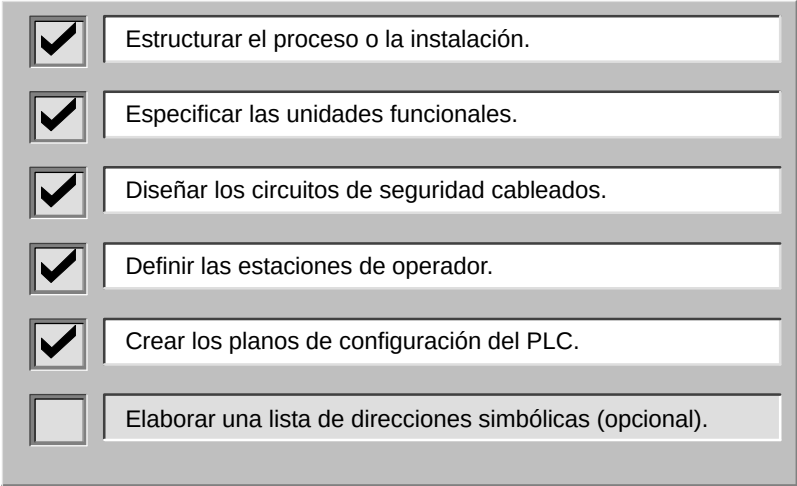
Antes de comenzar a programar aplicaciones para la CPU S7-200, es recomendable que se familiarice con algunas funciones básicas de la misma.

Índice del capítulo

Apartado	Descripción	Página
4.1	Crear una solución de automatización con un Micro-PLC	4-2
4.2	Programas S7-200	4-5
4.3	Lenguajes y editores de programación S7-200	4-6
4.4	Diferencias entre las operaciones SIMATIC e IEC 1131-3	4-10
4.5	Elementos básicos para estructurar un programa	4-18
4.6	El ciclo de la CPU	4-22
4.7	Ajustar el modo de operación de la CPU	4-25
4.8	Definir una contraseña	4-27
4.9	Comprobar y observar el programa	4-30
4.10	Editar programas en modo RUN	4-39
4.11	Seleccionar el tiempo en segundo plano	4-42
4.12	Eliminar errores de las CPUs S7-200	4-43

4.1 Crear una solución de automatización con un Micro-PLC

Existen diversos métodos para crear una solución de automatización con un Micro-PLC. En el presente apartado se indican algunas reglas generales aplicables a numerosos proyectos. No obstante, también deberá tener en cuenta las reglas de su empresa y su propia experiencia. La figura 4-1 muestra los pasos básicos al respecto.



☒	Estructurar el proceso o la instalación.
☒	Especificar las unidades funcionales.
☒	Diseñar los circuitos de seguridad cableados.
☒	Definir las estaciones de operador.
☒	Crear los planos de configuración del PLC.
☐	Elaborar una lista de direcciones simbólicas (opcional).

Figura 4-1 Pasos básicos para crear una solución de automatización

Estructurar el proceso o la instalación

Divida el proceso o la instalación en secciones independientes. Dichas secciones determinan los límites entre los diversos sistemas de automatización e influyen en las descripciones de las áreas de funciones y en la asignación de recursos.

Especificar las unidades funcionales

Describa las funciones de cada sección del proceso o de la instalación. Incorpore los siguientes aspectos:

- Entradas y salidas (E/S)
- Descripción del funcionamiento de la operación
- Condiciones de habilitación (es decir, los estados que se deben alcanzar antes de ejecutar una función) de cada actuador (electroválvulas, motores, accionamientos, etc).
- Descripción del interface de operador
- Interfaces con otras secciones del proceso o de la instalación

Diseñar los circuitos de seguridad cableados

Determine qué aparatos requieren un cableado permanente por motivos de seguridad. Si fallan los sistemas de automatización, puede ocurrir un arranque inesperado o un cambio en el funcionamiento de las máquinas. En tal caso, pueden producirse heridas graves o daños materiales. Por lo tanto, es preciso utilizar dispositivos de protección contra sobrecargas electromecánicas que funcionen independientemente de la CPU, evitando así las condiciones inseguras.

Para diseñar los circuitos de seguridad cableados:

- Defina el funcionamiento erróneo o inesperado de los actuadores que pudieran causar peligros.
- Defina las condiciones que garantizan un funcionamiento seguro y determine cómo detectar dichas condiciones, independientemente de la CPU.
- Defina cómo la CPU y los módulos de ampliación deberán influir en el proceso cuando se conecte y desconecte la alimentación, así como al detectarse errores. Estas informaciones se deberán utilizar únicamente para diseñar el funcionamiento normal y el funcionamiento anormal esperado, sin poderse aplicar para fines de seguridad.
- Prevea dispositivos de parada de emergencia manual o de protección contra sobrecargas electromagnéticas que impidan un funcionamiento peligroso, independientemente de la CPU.
- Desde los circuitos independientes, transmita informaciones de estado apropiadas a la CPU para que el programa y los interfaces de operador dispongan de los datos necesarios.
- Defina otros requisitos adicionales de seguridad para que el proceso se lleve a cabo de forma segura y fiable.

Definir las estaciones de operador

Conforme a las funciones exigidas, cree planos de las estaciones de operador incorporando los siguientes puntos:

- Panorámica de la ubicación de todas las estaciones de operador con respecto al proceso o a la instalación.
- Disposición mecánica de los aparatos (pantalla, interruptores, lámparas, etc) de la estación de operador.
- Esquemas eléctricos con las correspondientes entradas y salidas de la CPU o de los módulos de ampliación.

Crear los planos de configuración

Conforme a las funciones exigidas, cree planos de configuración del sistema de automatización incorporando los siguientes puntos:

- Ubicación de todas las CPUs y de todos los módulos de ampliación con respecto al proceso o a la instalación.
- Disposición mecánica de las CPUs y de los módulos de ampliación (incluyendo armarios, etc.)
- Esquemas eléctricos de todas las CPUs y de los módulos de ampliación (incluyendo los números de referencia, las direcciones de comunicación y las direcciones de las entradas y salidas).

Elaborar una lista de nombres simbólicos

Si desea utilizar nombres simbólicos para el direccionamiento, elabore una lista de nombres simbólicos para las direcciones absolutas. Incluya no sólo las entradas y salidas físicas, sino también todos los demás elementos que utilizará en su programa.

4.2 Programas S7-200

Referencias a las entradas y salidas en el programa

El funcionamiento básico de la CPU S7-200 es muy sencillo:

- La CPU lee el estado de las entradas.
- El programa almacenado en la CPU utiliza dichas entradas para evaluar la lógica. Durante la ejecución del programa, la CPU actualiza los datos.
- La CPU escribe los datos en las salidas.

La figura 4-2 muestra cómo se procesa un esquema de circuitos simple en una CPU S7-200. En este ejemplo, el estado del interruptor de la estación de operador para abrir la electroválvula de vaciado se suma a los estados de otras entradas. El resultado obtenido establece entonces el estado de la salida que corresponde a dicha electroválvula.

La CPU procesa el programa cíclicamente, leyendo y escribiendo los datos.

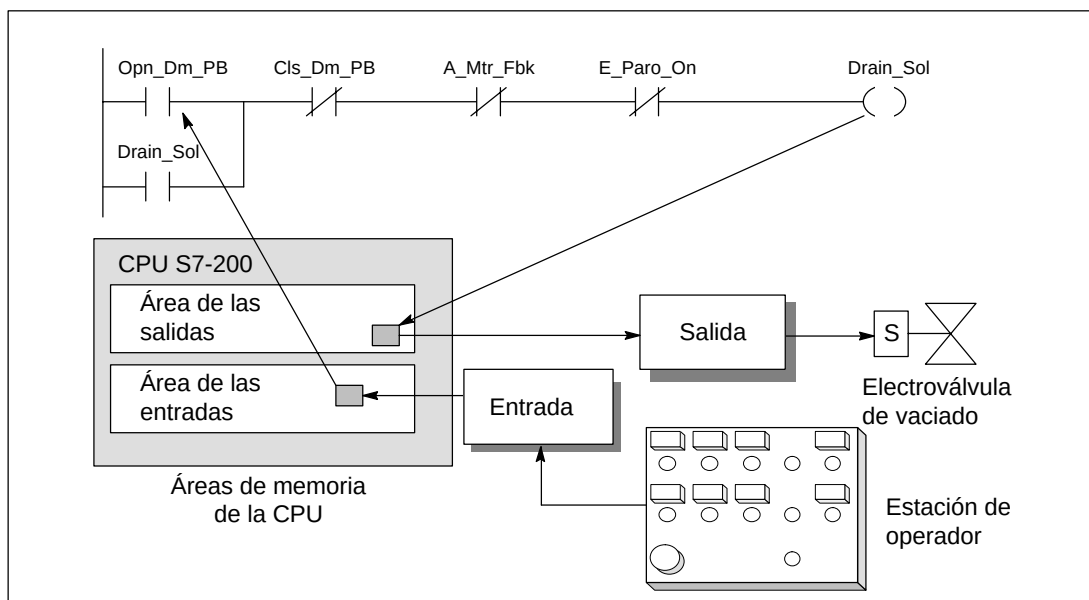


Figura 4-2 Referencias a las entradas y salidas en el programa

4.3 Lenguajes y editores de programación S7-200

Las CPUs S7-200 ofrecen numerosos tipos de operaciones que permiten solucionar una gran variedad de tareas de automatización. Disponen de dos juegos básicos de operaciones, a saber: SIMATIC e IEC 1131-3. El software de programación STEP 7-Micro/WIN 32 permite elegir entre diferentes editores para crear programas de control utilizando dichas operaciones. Por ejemplo, puede ser que Ud. prefiera crear programas en un entorno de programación gráfico, en tanto que otra persona que trabaje en su misma empresa opta por utilizar un editor textual, similar al lenguaje ensamblador.

Para crear sus programas debe hacer dos selecciones básicas:

- El tipo de juego de operaciones a utilizar (SIMATIC o IEC 1131-3).
- El tipo de editor a utilizar (Lista de instrucciones, Esquema de contactos o Diagrama de funciones)

La figura 4-1 muestra las combinaciones posibles de juegos de operaciones y de editores S7-200.

Tabla 4-1 Juegos de operaciones y editores SIMATIC e IEC 1131-3

Juego de operaciones SIMATIC	Juegos de operaciones IEC 1131-3
Editor AWL (Lista de instrucciones)	No disponible
Editor KOP (Esquema de contactos)	Editor LD (Diagrama de escalera)
Editor FUP (Diagrama de funciones)	Editor FBD (Diagrama de bloques funcionales)

Editor AWL (Lista de instrucciones)

El editor AWL (Lista de instrucciones) de STEP 7-Micro/WIN 32 permite crear programas de control introduciendo la nemotécnica de las operaciones. Por lo general, el editor AWL se adecúa especialmente para los programadores expertos ya familiarizados con los sistemas de automatización (PLCs) y con la programación lógica. El editor AWL también sirve para crear ciertos programas que, de otra forma, no se podrían programar con los editores KOP ni FUP. Ello se debe a que AWL es el lenguaje nativo de la CPU, a diferencia de los editores gráficos en los que son aplicables ciertas restricciones para poder dibujar los diagramas correctamente. La figura 4-3 muestra un ejemplo de un programa AWL.

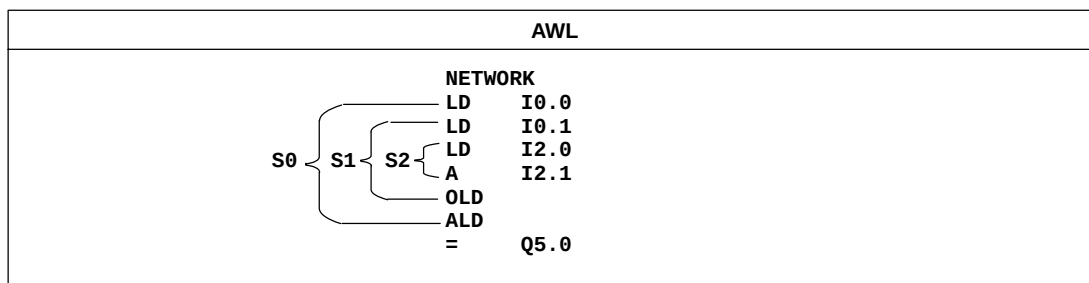


Figura 4-3 Ejemplo de un programa AWL

Como muestra la figura 4-3, esta forma textual es muy similar a la programación en lenguaje ensamblador. La CPU ejecuta cada operación en el orden determinado por el programa, de arriba a abajo, reiniciando luego arriba nuevamente. AWL y el lenguaje ensamblador también son similares en otro sentido. Las CPUs S7-200 utilizan una pila lógica para resolver la lógica de control (v. fig. 4-4). Los editores KOP y FUP insertan automáticamente las operaciones necesarias para procesar la pila. En AWL, es el usuario quien debe insertar dichas operaciones.

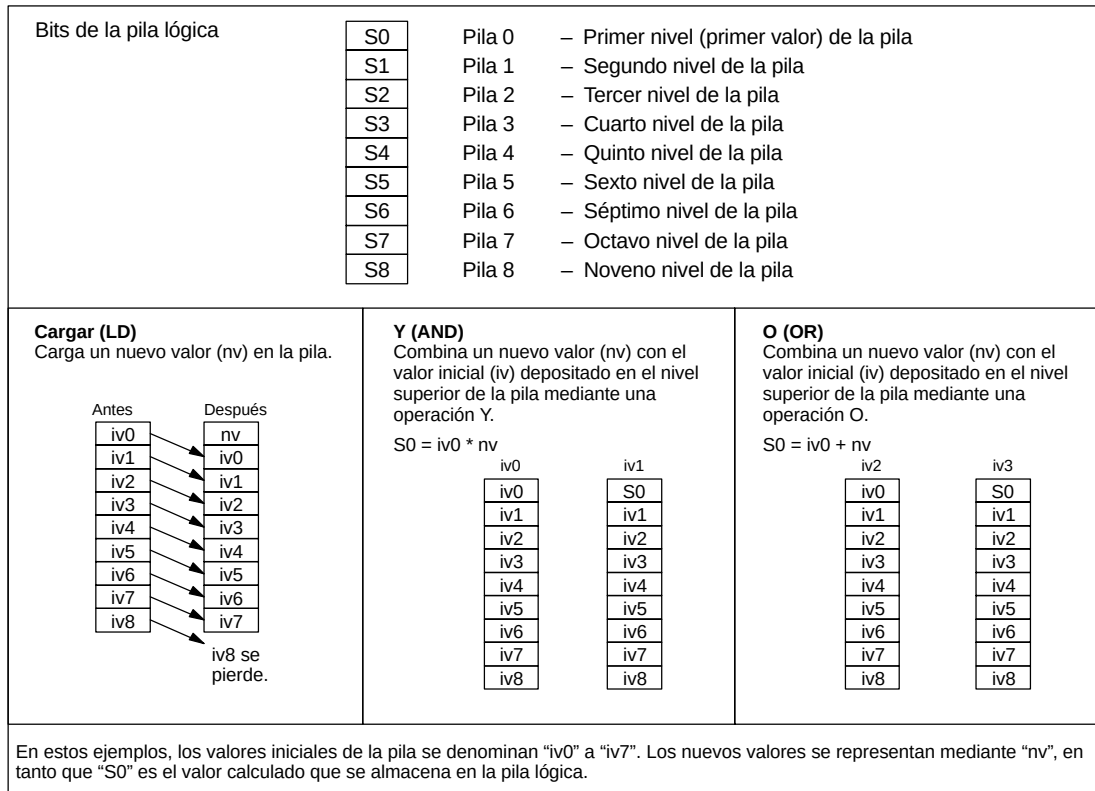


Figura 4-4 Pila lógica de la CPU S7-200

A continuación se indican los aspectos principales a considerar cuando se desee utilizar el editor AWL:

- El lenguaje AWL es más apropiado para los programadores expertos.
- En algunos casos, AWL permite solucionar problemas que no se podrían resolver muy fácilmente con los editores KOP o FUP.
- El editor AWL sólo se puede utilizar con el juego de operaciones SIMATIC.
- En tanto que el editor AWL se puede utilizar siempre para ver o editar un programa creado con los editores KOP o FUP SIMATIC, lo contrario no es posible en todos los casos. Los editores KOP o FUP SIMATIC no siempre se pueden utilizar para visualizar un programa que se haya creado en AWL.

Editor KOP (Esquema de contactos)

El editor KOP (Esquema de contactos) de STEP 7-Micro/WIN 32 permite crear programas con componentes similares a los elementos de un esquema de circuitos. KOP es probablemente el lenguaje predilecto de numerosos programadores y encargados del mantenimiento de sistemas de automatización. Básicamente, los programas KOP hacen que la CPU emule la circulación de corriente eléctrica desde una fuente de alimentación, a través de una serie de condiciones lógicas de entrada que, a su vez, habilitan condiciones lógicas de salida. Por lo general, la lógica se divide en unidades pequeñas y de fácil comprensión llamadas “segmentos” o “networks”. El programa se ejecuta segmento por segmento, de izquierda a derecha y luego de arriba a abajo. Tras alcanzar la CPU el final del programa, comienza nuevamente en la primera operación del mismo.

La figura 4-5 muestra un ejemplo de un programa KOP.

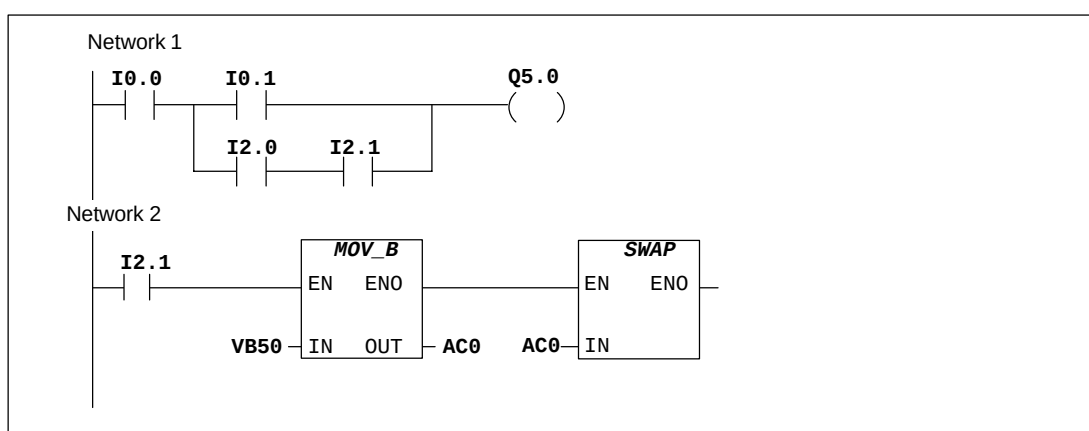


Figura 4-5 Ejemplo de un programa KOP

Las operaciones se representan mediante símbolos gráficos que incluyen tres formas básicas. Como muestra la figura 4-5, se pueden conectar en serie incluso varias operaciones de cuadros.

- Contactos - representan condiciones lógicas de “entrada” tales como interruptores, botones, condiciones internas, etc.
- Bobinas - representan condiciones lógicas de “salida” tales como lámparas, arrancadores de motor, relés interpuestos, condiciones internas de salida, etc.
- Cuadros - representan operaciones adicionales tales como temporizadores, contadores u operaciones aritméticas.

A continuación se indican los aspectos principales a considerar cuando se desee utilizar el editor KOP:

- El lenguaje KOP les facilita el trabajo a los programadores principiantes.
- La representación gráfica es a menudo fácil de comprender, siendo popular en el mundo entero.
- El editor KOP se puede utilizar con los juegos de operaciones SIMATIC e IEC 1131–3.
- El editor AWL siempre se puede utilizar para visualizar un programa creado en SIMATIC KOP.

Editor FUP (Diagrama de funciones)

El editor FUP (Diagrama de funciones) de STEP 7-Micro/WIN 32 permite visualizar las operaciones en forma de cuadros lógicos similares a los circuitos de puertas lógicas. En FUP no existen contactos ni bobinas como en el editor KOP, pero sí hay operaciones equivalentes que se representan en forma de cuadros. La lógica del programa se deriva de las conexiones entre las operaciones de cuadro. Ello significa que la salida de una operación (p.ej. un cuadro AND) se puede utilizar para habilitar otra operación (p.ej. un temporizador), con objeto de crear la lógica de control necesaria. Dichas conexiones permiten solucionar numerosos problemas lógicos.

La figura 4-6 muestra un ejemplo de un programa creado con el editor FUP.

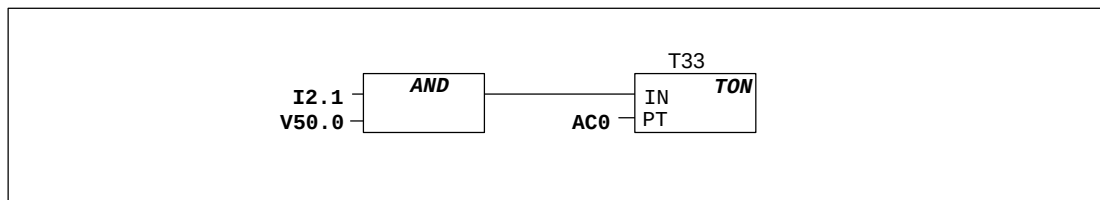


Figura 4-6 Ejemplo de un programa FUP

A continuación se indican los aspectos principales a considerar cuando se desee utilizar el editor FUP:

- El estilo de representación en forma de puertas gráficas se adecúa especialmente para observar el flujo del programa.
- El editor FUP se puede utilizar con los juegos de operaciones SIMATIC e IEC 1131-3.
- El editor AWL siempre se puede utilizar para visualizar un programa creado en SIMATIC FUP.

4.4 Diferencias entre las operaciones SIMATIC e IEC 1131-3

Juego de operaciones SIMATIC

La mayoría de los sistemas de automatización ofrecen los mismos tipos básicos de operaciones, pero por lo general existen pequeñas diferencias en cuanto al aspecto, al funcionamiento, etc. de los productos de los distintos fabricantes. El juego de operaciones SIMATIC se ha diseñado para los sistemas de automatización S7-200. Un gran número de estas operaciones tienen un aspecto y un funcionamiento diferentes si se comparan con otras marcas de sistemas de automatización (autómatas programables). Tenga en cuenta los siguientes aspectos al seleccionar el juego de operaciones SIMATIC:

- Por lo general, el tiempo de ejecución de las operaciones SIMATIC es más breve.
- El juego de operaciones SIMATIC se puede utilizar con los tres editores (KOP, AWL y FUP).

Juego de operaciones IEC 1131-3

La Comisión Electrotécnica Internacional (CEI) o International Electrotechnical Commission (IEC) es un organismo mundial que desarrolla normas globales para todos los campos de la electrotécnica. (Nota: En el presente manual se utilizan las siglas inglesas de dicho organismo). Durante los últimos años, dicha comisión ha desarrollado una norma que se dedica a numerosos aspectos de la programación de autómatas programables (denominados "sistemas de automatización" en la terminología SIMATIC). El objetivo de dicha norma es que los diferentes fabricantes de autómatas programables ofrezcan operaciones similares tanto en su aspecto como en su funcionamiento. Existen algunas diferencias básicas entre los juegos de operaciones SIMATIC e IEC 1131-3.

- El juego de operaciones IEC 1131-3 se limita a las operaciones estándar comunes entre los fabricantes de autómatas programables. Algunas operaciones incluidas en el juego SIMATIC no están normalizadas en la norma IEC 1131-3. (Éstas se pueden utilizar en calidad de operaciones no normalizadas. No obstante, entonces el programa ya no será absolutamente compatible con la norma IEC 1131-3).
- Algunos cuadros aceptan varios formatos de datos. A menudo, esto se denomina sobrecarga. Por ejemplo, en lugar de tener cuadros aritméticos por separado, tales como ADD_I (Sumar enteros), ADD_R (Sumar reales) etc., la operación ADD definida en la norma IEC 1131-3 examina el formato de los datos a sumar y selecciona automáticamente la operación correcta en la CPU. Así se puede ahorrar tiempo al diseñar los programas.
- Si se utilizan las operaciones IEC-1131, se comprueba automáticamente si los parámetros de la operación corresponden al formato de datos correcto. Dicha comprobación no es obvia para el usuario. Por ejemplo, si se ha intentado introducir un valor de entero en una operación para la que se deba utilizar un valor binario (on/off), se indica un error. Esta función permite reducir los errores de sintaxis de programación.

A continuación se indican los aspectos principales a considerar cuando se desee utilizar las operaciones IEC 1131–3:

- Por lo general, es más fácil aprender a crear programas para autómatas programables de diferentes fabricantes.
- Aunque se dispone de menos operaciones (conforme a lo especificado en la norma), también se pueden utilizar numerosas operaciones SIMATIC.
- El funcionamiento de algunas operaciones IEC–1131 es diferente al de sus equivalentes en SIMATIC (temporizadores, contadores, multiplicación, división, etc.)
- Es posible que el tiempo de ejecución de las operaciones IEC–1131 sea más largo.
- Las operaciones sólo se pueden utilizar en los editores KOP y FUP.
- La norma IEC 1131–3 especifica que las variables se deben declarar tipificadas, soportando que el sistema verifique el tipo de datos.

Tipificación de variables en SIMATIC e IEC 1131–3

Cada operación o subrutina parametrizada se identifica en SIMATIC e IEC 1131–3 mediante una definición precisa denominada *signatura*. En todas las operaciones estándar, los tipos de datos permitidos para cada operando de la operación se indican en la *signatura*. En el caso de las subrutinas parametrizadas, el usuario crea la *signatura* de la subrutina en la tabla de variables locales.

STEP 7-Micro/WIN 32 implementa una verificación simple de los tipos de datos en el modo SIMATIC y una verificación precisa en el modo IEC 1131–3. Cuando un tipo de datos se especifica para una variable (bien sea local o global), STEP 7-Micro/WIN 32 verifica que el tipo de datos del operando concuerde con la *signatura* de la operación en el nivel indicado. En la tabla 4-2 se definen los tipos de datos elementales y la tabla 4-3 muestra los tipos de datos complejos disponibles en STEP 7-Micro/WIN 32.

Tabla 4-2 Tipos de datos elementales IEC 1131–3

Tipos de datos elementales	Tamaño	Descripción	Margen de datos
BOOL (1 bit)	1 bit	Valor booleano	0 a 1
BYTE (8 bits)	8 bits	Byte sin signo	0 a 255
WORD (16 bits)	16 bits	Entero sin signo	0 a 65.535
INT (16 bits)	16 bits	Entero con signo	–32768 a +32767
DWORD (32 bits)	32 bits	Entero doble sin signo	0 a $2^{32} - 1$
DINT (32 bits)	32 bits	Entero doble con signo	-2^{31} a $+2^{31} - 1$
REAL (32 bits)	32 bits	Valor de 32 bits en coma flotante (IEEE)	-10^{38} a $+10^{38}$

Tabla 4-3 Tipos de datos complejos IEC 1131-3

Tipos de datos complejos	Descripción	Margen de direcciones
TON	Temporizador con retardo al conectar	1 ms T32, T96 10 ms T33 a T36, T97 a T100 100 ms T37 a T63, T101 a T255
TOF	Temporizador con retardo al desconectar	1 ms T32, T96 10 ms T33 a T36, T97 a T100 100 ms T37 a T63, T101 a T255
TP	Temporizador por impulsos	1 ms T32, T96 10 ms T33 a T36, T97 a T100 100 ms T37 a T63, T101 a T255
CTU	Contador ascendente	0 a 255
CTD	Contador descendente	0 a 255
CTUD	Contador ascendente–descendente	0 a 255
SR	Bloque funcional biestable (posicionar dominante)	—
RS	Bloque funcional biestable (rearmar dominante)	—

Verificación de los tipos de datos Hay tres niveles de verificación de los tipos de datos: verificación precisa, verificación simple y sin verificación.

Verificación precisa de los tipos de datos En este modo, el tipo de datos del parámetro debe concordar con el tipo de datos del símbolo o de la variable. Cada parámetro formal tiene sólo un tipo de datos (con excepción de las operaciones sobrecargadas). Por ejemplo, el parámetro IN de una operación SRW (Desplazar palabra a la derecha) tiene el tipo de datos WORD. Sólo las variables asignadas al tipo de datos WORD se compilarán correctamente. Las variables tipificadas como INT no serán válidas para los parámetros de operaciones de tipo WORD si se exige la verificación precisa de los tipos de datos.

La verificación precisa de los tipos de datos se efectúa sólo en el modo IEC 1131-3 (v. tabla 4-4).

Tabla 4-4 Verificación precisa de los tipos de datos seleccionados por el usuario y sus equivalentes

Tipo de datos seleccionado por el usuario	Tipo de datos equivalente
BOOL	BOOL
BYTE	BYTE
WORD	WORD
INT	INT
DWORD	DWORD
DINT	DINT
REAL	REAL

Verificación simple de los tipos de datos En este modo, cuando se le asigna un tipo de datos a un símbolo o a una variable, se asocia automáticamente también a todos los tipos cuyo tamaño binario concuerde con el tipo de datos seleccionado. Por ejemplo, si se selecciona el tipo de datos DINT, la variable local asignará automáticamente el tipo de datos DWORD, puesto que ambos son tipos de datos de 32 bits. El tipo de datos REAL no se asigna automáticamente, aunque se trata también de un tipo de datos de 32 bits. En el caso del tipo de datos REAL, se considera que no tiene ningún otro tipo de datos equivalente, siendo siempre único. La verificación simple de los tipos de datos se efectúa sólo en el modo SIMATIC cuando se utilizan variables locales (v. tabla 4-5).

Tabla 4-5 Verificación simple de los tipos de datos seleccionados por el usuario y sus equivalentes

Tipo de datos seleccionado por el usuario	Tipo de datos equivalente
BOOL	BOOL
BYTE	BYTE
WORD	WORD, INT
INT	WORD, INT
DWORD	DWORD, DINT
DINT	DWORD, DINT
REAL	REAL

Sin verificación de los tipos de datos Este modo se encuentra disponible únicamente para las variables globales SIMATIC que no permitan seleccionar los tipos de datos. Todos los tipos de datos de tamaño equivalente se asignan automáticamente al símbolo. Por ejemplo, a un símbolo que se le haya asignado la dirección VD100, STEP 7-Micro/WIN 32 le asignará automáticamente los tipos de datos que muestra la tabla 4-6.

Tabla 4-6 Tipo de datos de tamaño determinado para los símbolos globales SIMATIC

Dirección seleccionada por el usuario	Tipo de datos equivalente
V0.0	BOOL
VB0	BYTE
VW0	WORD, INT
VD0	DWORD, DINT, REAL

Ventajas de la verificación de los tipos de datos

La verificación de los tipos de datos contribuye a evitar errores de programación. Si una operación soporta números con signo y se utiliza un número sin signo para el operando de esa operación, STEP 7-Micro/WIN 32 marcará el número sin signo. Por ejemplo, la comparación $< I$ es una operación con signo. -1 es menor que 0 en el caso de los operandos con signo. No obstante, si la operación $< I$ puede soportar tipos de datos sin signo, el programador debe tener en cuenta que durante la ejecución del programa, un valor sin signo de 40.000 es menor que 0 en la operación $< I$.



Precaución

Verifique que la utilización de números sin signo en las operaciones con signo no cruce el límite entre los números positivos y negativos.

En caso contrario se pueden producir resultados impredecibles en el programa y en el funcionamiento del sistema de automatización. Un funcionamiento imprevisible de los equipos controlados puede ocasionar la muerte o lesiones graves al personal, y/o considerables daños materiales.

Tenga siempre en cuenta que el número sin signo de una operación con signo no cruce el límite entre los números positivos y negativos.

En resumen, la verificación precisa de los tipos de datos en el modo de edición IEC 1131-3 contribuye a identificar dichos errores durante la compilación, indicando si se utilizan tipos de datos no válidos para la operación. Esta función no se encuentra disponible en los editores SIMATIC.

Seleccionar entre los modos de programación SIMATIC e IEC 1131-3

Puesto que, en contraposición a SIMATIC, el modo IEC 1131-3 exige una tipificación precisa de los datos, STEP 7-Micro/WIN 32 no permite transferir programas entre los dos modos de edición. Por tanto, el usuario debe elegir uno de dichos modos.

Operaciones sobrecargadas

Las operaciones sobrecargadas soportan diversos tipos de datos. No obstante, se efectúa aún una verificación precisa de los tipos de datos, puesto que todos los tipos de datos de los operandos deben concordar para que la operación se pueda compilar correctamente. La tabla 4-7 muestra un ejemplo de la operación ADD sobrecargada (IEC).

Tabla 4-7 Ejemplo de la operación ADD sobrecargada (IEC)

Operación	Tipos de datos admisibles (verificación precisa)	Tipos de datos admisibles (verificación simple)	Operación compilada
ADD	INT	WORD, INT	ADD_I (Sumar enteros)
ADD	DINT	DWORD, DINT	ADD_D (Sumar enteros dobles)
ADD	REAL	REAL	ADD_R (Sumar reales)

Si todos los operandos tienen el tipo de datos DINT, el compilador generará una operación Sumar enteros dobles. Si se mezclan los tipos de datos de la operación sobrecargada, ocurrirá un error de compilación. El nivel de verificación de los tipos de datos determina lo que es válido o no. El ejemplo siguiente generará un error de compilación sólo si se efectúa una verificación precisa de los tipos de datos, mas no si se realiza una verificación simple.

ADD IN1 = INT, IN2 = WORD, IN3 = INT

Verificación precisa: error de compilación

Verificación simple: se compila a ADD_I (Sumar enteros)

La verificación simple de los tipos de datos no evita que ocurran errores de programación en el tiempo de ejecución. Si se efectúa una verificación simple de los tipos de datos, el compilador no detectará el siguiente error de programación: ADD 40000, 1 será un número negativo, mas no un valor 40.001 sin signo.

Utilizar el direccionamiento directo en IEC para operaciones sobrecargadas

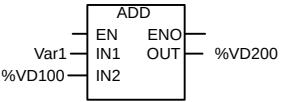
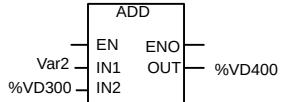
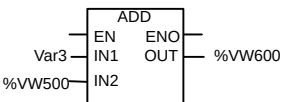
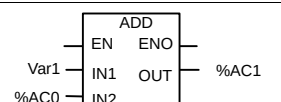
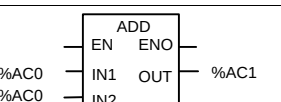
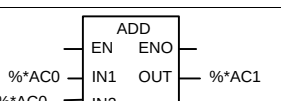
El modo de programación IEC 1131-3 permite utilizar direcciones directas como parte de la configuración de los parámetros de la operación. Tanto las variables con las direcciones se pueden utilizar dentro de los parámetros. Recuerde que las direcciones directas no contienen información explícita sobre el tipo de datos. Además, la información de tipo no se puede deducir de ninguna de las operaciones IEC sobrecargadas, toda vez que éstas aceptan diversos tipos de datos.

Los tipos de datos de los parámetros direccionados directamente se determinan examinando otros parámetros tipificados incluidos en la operación. Si se configura que un parámetro utilice una variable de un tipo específico, se supone que todos los parámetros representados directamente sean de ese mismo tipo. Las tablas 4-8 y 4-9 muestran ejemplos de tipos de datos de parámetros representados directamente. El signo de porcentaje (%) se utiliza para indicar una dirección directa.

Tabla 4-8 Ejemplo de tipos de datos para el direccionamiento directo

Nombre	Dirección	Tipo de datos	Comentario
Var1		REAL	Esta es una variable en coma flotante.
Var2		DINT	Esta es una variable de entero doble.
Var3		INT	Esta es una variable de entero.

Tabla 4-9 Ejemplos de direccionamiento directo en operaciones sobrecargadas

Ejemplo	Descripción
	Se supone que VD100 y VD200 son de tipo REAL, puesto que el tipo de Var1 es REAL.
	Se supone que VD300 y VD400 son de tipo DINT, puesto que el tipo de Var2 es DINT.
	Se supone que VW500 y VW600 son de tipo INT, puesto que el tipo de Var3 es INT.
	Se supone que AC0 y AC1 son de tipo REAL, puesto que el tipo de Var1 es REAL.
	Esta configuración no es válida, puesto que el tipo no se puede determinar. El tipo de datos de los acumuladores podría ser uno cualquiera.
	Esta configuración no es válida, puesto que el tipo no se puede determinar. El tipo de datos de los punteros de los acumuladores podría ser uno cualquiera.
% indica una dirección directa.	

Utilizar operaciones de conversión

Las operaciones de conversión permiten transferir valores de un tipo de datos a otro. STEP 7-Micro/WIN 32 soporta las operaciones de conversión que figuran en la tabla 4-10 para transferir valores entre los tipos de datos elementales.

Tabla 4-10 Operaciones de conversión

Operación de conversión	Operandos admisibles en la verificación precisa de los tipos de datos	Operandos admisibles en la verificación simple de los tipos de datos
BYTE a INT	IN: BYTE OUT: INT	IN: BYTE OUT: WORD, INT
INT a BYTE	IN: INT OUT: BYTE	IN: WORD, INT OUT: BYTE
INT a DINT	IN: DINT OUT: DINT	IN: WORD, INT OUT: DWORD, DINT
DINT a INT	IN: DINT OUT: INT	IN: DWORD, DINT OUT: WORD, INT
DINT a REAL	IN: DINT OUT: REAL	IN: DWORD, DINT OUT: REAL
REAL a DINT (ROUND)	IN: REAL OUT: DINT	IN: REAL OUT: DWORD, DINT

En el modo de edición IEC 1131-3, la operación MOVE sobrecargada se puede utilizar para convertir entre INT y WORD, DINT y DWORD. La operación MOVE permite transferir tipos de datos de un mismo tamaño, sin que el compilador genere errores (v. tabla 4-11).

Tabla 4-11 Utilizar la operación sobrecargada MOVE

MOVE sobrecargada (IEC 1131-3)	IN	OUT
MOVE (INT a WORD)	INT	WORD
MOVE (WORD a INT)	WORD	INT
MOVE (DINT a DWORD)	DINT	DWORD
MOVE (DWORD a DINT)	DWORD	DINT

4.5 Elementos básicos para estructurar un programa

La CPU S7-200 ejecuta continuamente el programa para controlar una tarea o un proceso. El programa se crea con STEP 7-Micro/WIN 32 y se carga en la CPU. Desde el programa principal se pueden invocar diversas subrutinas o rutinas de interrupción.

Estructurar el programa

Los programas para la CPU S7-200 comprenden tres partes básicas: el programa principal, las subrutinas (opcional) y las rutinas de interrupción (opcional). Un programa S7-200 se divide en los siguientes elementos:

- Programa principal: En esta parte del programa se disponen las operaciones que controlan la aplicación. Las operaciones del programa principal se ejecutan de forma secuencial en cada ciclo de la CPU.
- Rutinas de interrupción: Estos elementos opcionales del programa se ejecutan cada vez que ocurra el correspondiente evento de interrupción.
- Subrutinas: Estos elementos opcionales del programa se ejecutan sólo cuando se llaman desde el programa principal o desde una rutina de interrupción.

Programa de ejemplo con subrutinas y rutinas de interrupción

A continuación se muestran programas de ejemplo para una interrupción temporizada que se puede utilizar en aplicaciones tales como leer el valor de una entrada analógica. En este ejemplo, el intervalo de muestreo de la entrada analógica es de 100 ms.

Las figuras 4-7 a 4-11 muestran programas que utilizan una subrutina y una rutina de interrupción en los diversos lenguajes de programación S7-200.

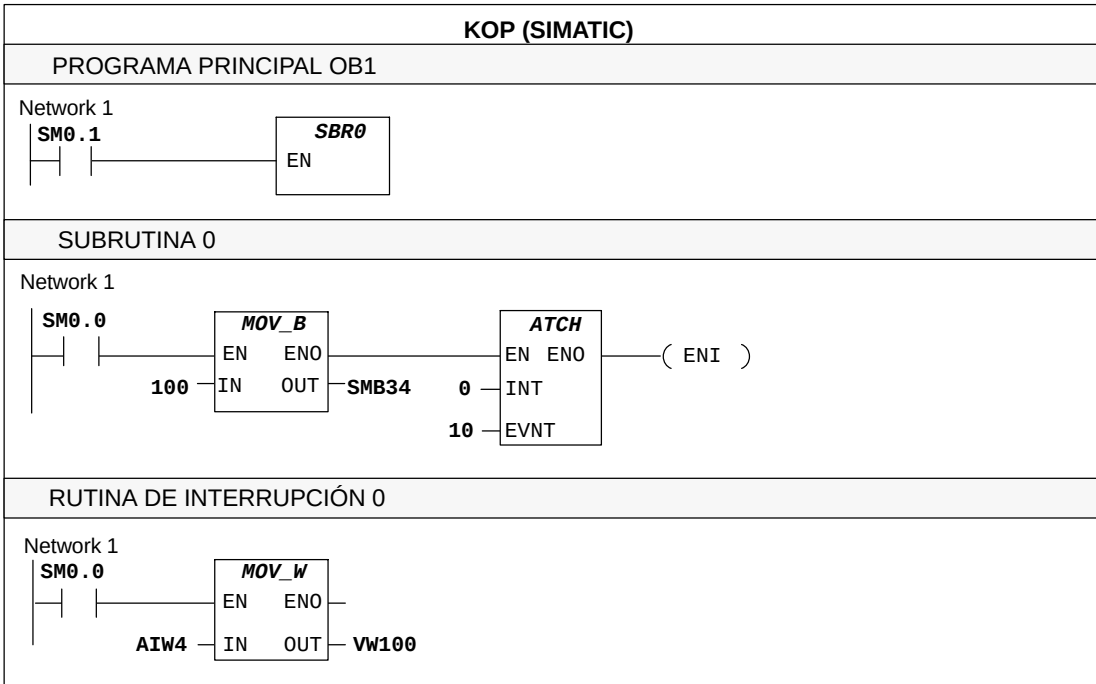


Figura 4-7 Programa KOP (SIMATIC) con una subrutina y una rutina de interrupción

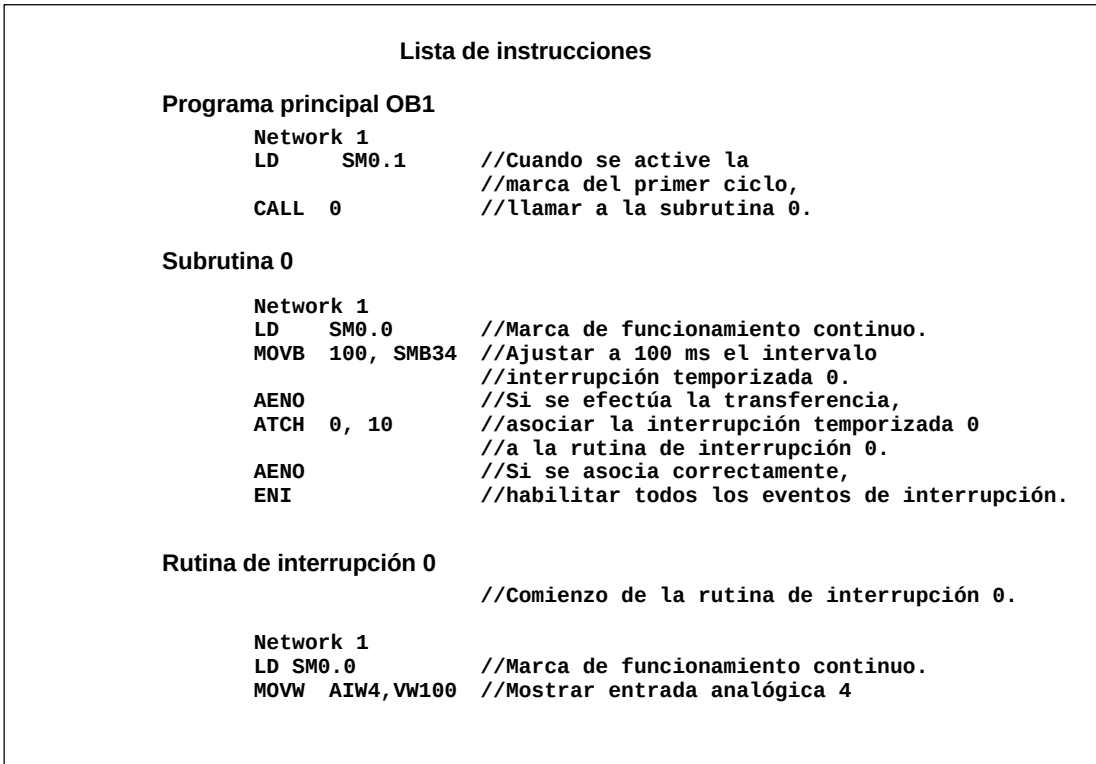


Figura 4-8 Programa AWL (SIMATIC) con una subrutina y una rutina de interrupción

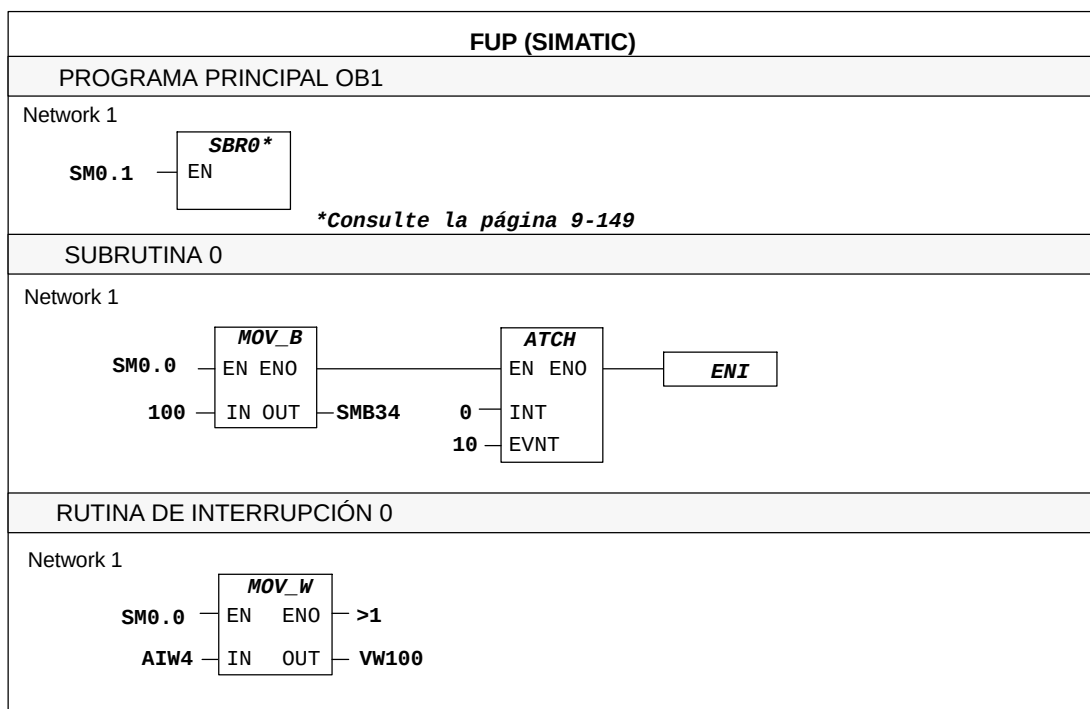


Figura 4-9 Programa FUP (SIMATIC) con una subrutina y una rutina de interrupción

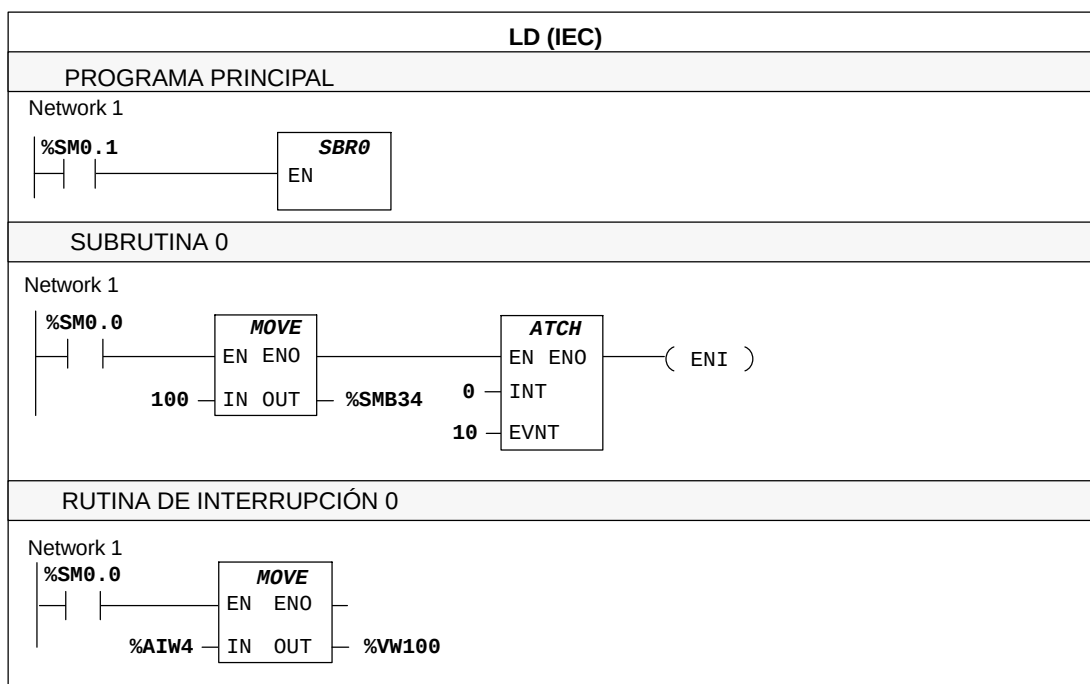


Figura 4-10 Programa LD (IEC) con una subrutina y una rutina de interrupción

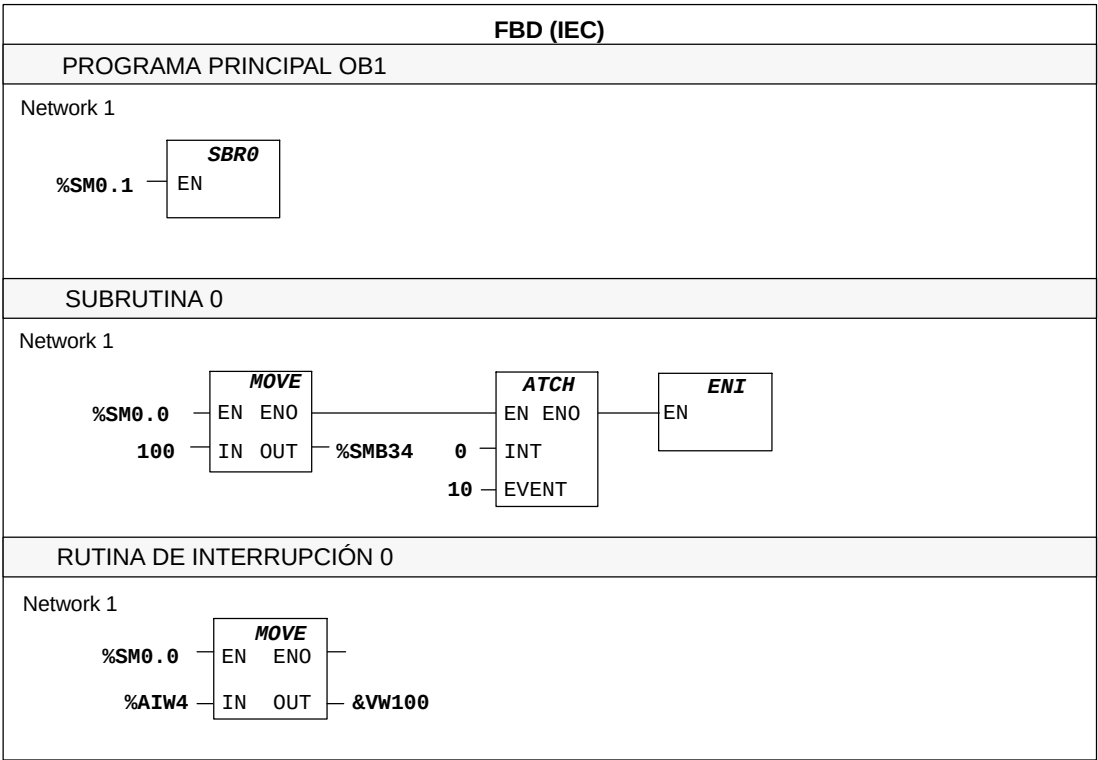


Figura 4-11 Programa FBD (IEC) con una subrutina y una rutina de interrupción

4.6 El ciclo de la CPU

La CPU S7-200 se ha previsto para que ejecute cíclicamente una serie de tareas, incluyendo el programa de usuario. Dicha ejecución se denomina ciclo. Durante el ciclo que muestra la figura 4-12, la CPU ejecuta la mayoría de las tareas siguientes (o todas ellas):

- Lee las entradas.
- Ejecuta el programa.
- Procesa las peticiones de comunicación.
- Efectúa un autodiagnóstico.
- Escribe en las salidas.

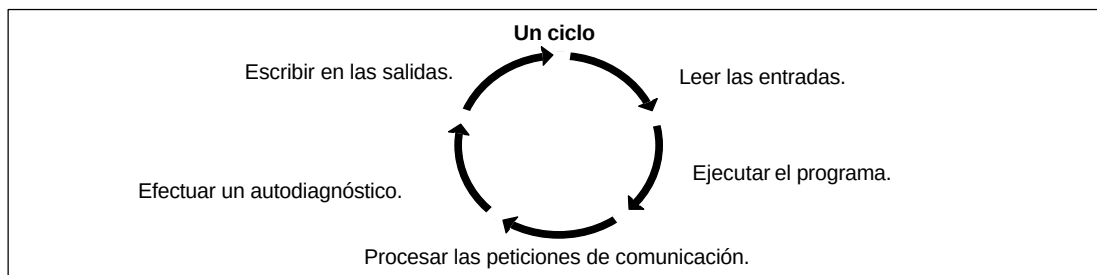


Figura 4-12 Ciclo de la CPU S7-200

La serie de tareas que se ejecutan durante el ciclo depende del modo de operación de la CPU. La CPU S7-200 tiene dos modos de operación: STOP y RUN. Con respecto al ciclo, la principal diferencia entre STOP y RUN es que el programa se ejecuta al estar la CPU en modo RUN, mas no en STOP.

Leer las entradas digitales

Al principio de cada ciclo se leen los valores actuales de las entradas digitales y se escriben luego en la imagen del proceso de las entradas.

La CPU reserva un espacio de la imagen del proceso de las entradas en incrementos de ocho bits (un byte). Si la CPU o el módulo de ampliación no proveen una entrada física para cada bit del byte reservado, no será posible asignar dichos bits a los módulos siguientes en la cadena de E/S o utilizarlos en el programa de usuario. Al comienzo de cada ciclo, la CPU pone a 0 estos bits no utilizados en la imagen del proceso. No obstante, si la CPU soporta varios módulos de ampliación y no se está utilizando su capacidad de E/S (porque no se han instalado los módulos de ampliación), los bits de entradas de ampliación no utilizados se pueden usar como marcas internas adicionales.

La CPU no actualiza las entradas analógicas como parte del ciclo normal, a menos que se haya habilitado la filtración digital de las mismas. Dicha filtración es una opción seleccionable por el usuario, pudiéndose habilitar individualmente para cada una de las entradas analógicas.

La filtración digital se ha previsto para su utilización en módulos analógicos de bajo costo que no disponen de una filtración interna al módulo. Es recomendable utilizar la filtración digital en aplicaciones donde la señal de entrada cambia lentamente. Si la señal es rápida, no es recomendable habilitar la filtración digital.

Si se habilita la filtración de una entrada analógica, la CPU actualiza dicha entrada una vez por ciclo, efectúa la filtración y almacena internamente el valor filtrado. El valor filtrado se suministra entonces cada vez que el programa accede a la entrada analógica.

Si no se habilita la filtración de una entrada analógica, la CPU lee su valor del módulo físico cada vez que el programa de usuario accede a la entrada analógica.

Ejecutar el programa

Durante esta fase del ciclo, la CPU ejecuta el programa desde la primera operación hasta la última (= Finalizar programa). El control directo de las entradas y salidas permite acceder directamente a las mismas mientras se ejecuta el programa o una rutina de interrupción.

Si se utilizan interrupciones, las rutinas asociadas a los eventos de interrupción se almacenan como parte del programa (v. apt. 4.5). Las rutinas de interrupción no se ejecutan como parte del ciclo, sino sólo cuando ocurre el evento (en cualquier punto del ciclo).

Procesar las peticiones de comunicación

Durante esta fase del ciclo, la CPU procesa los mensajes que haya recibido por el puerto de comunicación.

Efectuar el autodiagnóstico de la CPU

Durante el autodiagnóstico se comprueba el firmware de la CPU y la memoria del programa (sólo en modo RUN), así como el estado de los módulos de ampliación.

Escribir en las salidas digitales

Al final de cada ciclo, la CPU escribe los valores de la imagen del proceso de las salidas en las salidas digitales.

La CPU reserva un espacio de la imagen del proceso de las salidas en incrementos de ocho bits (un byte). Si la CPU o el módulo de ampliación no proveen una salida física para cada bit del byte reservado, no será posible asignar dichos bits a los módulos siguientes en la cadena de E/S.

Cuando el modo de operación de la CPU se cambia de RUN a STOP, las salidas digitales adoptan los valores definidos en la tabla de salidas o conservan su estado actual (v. apt. 6.4). El ajuste estándar es que las salidas digitales se pongan a 0. Las salidas analógicas conservan su último valor.

Interrumpir el ciclo

Si se utilizan interrupciones, las rutinas asociadas a los eventos de interrupción se almacenan como parte del programa. Las rutinas de interrupción no se ejecutan como parte del ciclo, sino sólo cuando ocurre el evento (en cualquier punto del ciclo). La CPU procesa las interrupciones según su prioridad y después en el orden que aparecen.

Imagen del proceso de las entradas y salidas

Por lo general, es recomendable utilizar la imagen del proceso, en vez de acceder directamente a las entradas o salidas durante la ejecución del programa. Las imágenes del proceso existen por tres razones:

- El sistema verifica todas las entradas al comenzar el ciclo. De este modo se sincronizan y "congelan" los valores de estas entradas durante la ejecución del programa. La imagen del proceso actualiza las salidas cuando termina de ejecutarse el programa. Ello tiene un efecto estabilizador en el sistema.
- El programa de usuario puede acceder a la imagen del proceso mucho más rápido de lo que podría acceder directamente a las entradas y salidas físicas, con lo cual se acelera su tiempo de ejecución.
- Las entradas y salidas son unidades de bit a las que se debe acceder en formato de bit. No obstante, la imagen del proceso permite acceder a ellas en formato de bits, bytes, palabras y palabras dobles, lo que ofrece flexibilidad adicional.

Control directo de las entradas y salidas

Las operaciones de control directo de las entradas y salidas (E/S) permiten acceder a la entrada o salida física, aunque el acceso a las E/S se efectúa por lo general a través de las imágenes del proceso. El acceso directo a una entrada no modifica la dirección correspondiente en la imagen del proceso de las entradas. En cambio, el acceso directo a una salida actualiza simultáneamente la dirección correspondiente en la imagen del proceso de las salidas.

La CPU gestiona las E/S analógicas como datos directos, a menos que se haya habilitado la filtración digital de las entradas analógicas (v. apt. 6.5). Cuando se escribe un valor en una salida analógica, la salida se actualiza inmediatamente.

4.7 Ajustar el modo de operación de la CPU

La CPU S7-200 tiene dos modos de operación:

- STOP: La CPU no ejecuta el programa. Cuando está en modo STOP, es posible cargar programas o configurar la CPU.
- RUN: La CPU ejecuta el programa.

El diodo luminoso (LED) en la parte frontal de la CPU indica el modo de operación actual.

El modo de operación se puede cambiar como se indica a continuación:

- Accionando manualmente el selector de modos de operación de la CPU.
- Utilizando el software de programación STEP 7-Micro/WIN 32 y colocando el selector de la CPU en posición TERM o RUN.
- Insertando una operación STOP en el programa.

Cambiar el modo de operación con el selector

El modo de operación de la CPU se puede cambiar manualmente accionando el selector (ubicado debajo de la tapa de acceso frontal de la CPU):

- Si el selector se pone en STOP, se detendrá la ejecución del programa.
- Si el selector se pone en RUN, se iniciará la ejecución del programa.
- Si el selector se pone en TERM, no cambiará el modo de operación de la CPU.

Si se interrumpe la alimentación estando el selector en posición STOP o TERM, la CPU pasará a modo STOP cuando se le aplique tensión. Si se interrumpe la alimentación estando el selector en posición RUN, la CPU pasará a modo RUN cuando se le aplique tensión.

Cambiar el modo de operación con STEP 7-Micro/WIN 32

Como muestra la figura 4-13, el modo de operación de la CPU se puede cambiar también con STEP 7-Micro/WIN 32. Para que ello sea posible, el selector de la CPU deberá estar en posición TERM o RUN.

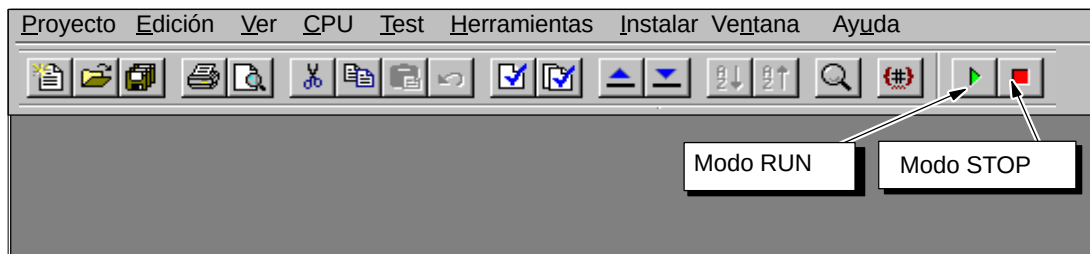


Figura 4-13 Utilizar STEP 7-Micro/WIN 32 para cambiar el modo de operación de la CPU

Cambiar el modo de operación desde el programa

Para cambiar la CPU a modo STOP es posible introducir la correspondiente operación (STOP) en el programa. Ello permite detener la ejecución del programa en función de la lógica. Para obtener más información acerca de la operación STOP, consulte el capítulo 9 (operaciones SIMATIC) o el capítulo 10 (operaciones IEC 1131-3).

4.8 Definir una contraseña para la CPU

Todas las CPUs S7-200 ofrecen una protección con contraseña para restringir el acceso a determinadas funciones. Con una contraseña se puede acceder a las funciones y a la memoria de la CPU. Si no se utiliza la opción de contraseña, la CPU permite un acceso ilimitado. Si está protegida con una contraseña, la CPU prohíbe todas las operaciones restringidas conforme a la configuración definida al definir la contraseña.

Restringir el acceso a la CPU

Como muestra la tabla 4-12, las CPUs S7-200 ofrecen tres niveles de protección para acceder a sus funciones. Cada uno de dichos niveles permite ejecutar determinadas funciones sin la contraseña. Si se introduce la contraseña correcta, es posible acceder a todas las funciones de la CPU. El ajuste estándar para la CPU S7-200 es el nivel 1 (privilegios totales).

Si se introduce la contraseña a través de una red, no se afecta la protección con contraseña de la CPU. Si un usuario tiene acceso a las funciones restringidas de la CPU, ello no autoriza a los demás usuarios a acceder a dichas funciones. El acceso ilimitado a las funciones de la CPU sólo se permite a un usuario a la vez.

Nota

Una vez introducida la contraseña, el nivel de protección se conservará aproximadamente durante un minuto después de haber desconectado la unidad de programación de la CPU. Si otro usuario se conecta inmediatamente a la CPU durante ese tiempo es posible que pueda acceder a la unidad de programación.

Tabla 4-12 Restringir el acceso a la CPU S7-200

Tarea	Nivel 1	Nivel 2	Nivel 3
Leer y escribir datos de usuario	No restringido	No restringido	No restringido
Arrancar, detener y rearrancar la CPU		No restringido	No restringido
Leer y escribir el reloj de tiempo real			
Cargar en la PG el programa de usuario, los datos y la configuración		Restringido	Restringido
Cargar en la CPU			
Consultar el estado AWL			
Borrar el programa de usuario, los datos y la configuración			
Forzar datos o ejecutar uno/varios ciclo(s)			
Copiar en el cartucho de memoria			
Escribir en las salidas en modo STOP			

Configurar la contraseña para la CPU

STEP 7-Micro/WIN 32 permite definir una contraseña para acceder a las funciones de la CPU. Elija el comando de menú **Ver > Bloque de sistema** y haga clic en la ficha "Contraseña" (v. fig. 4-14). Indique el nivel de protección deseado. Introduzca y verifique luego la contraseña.

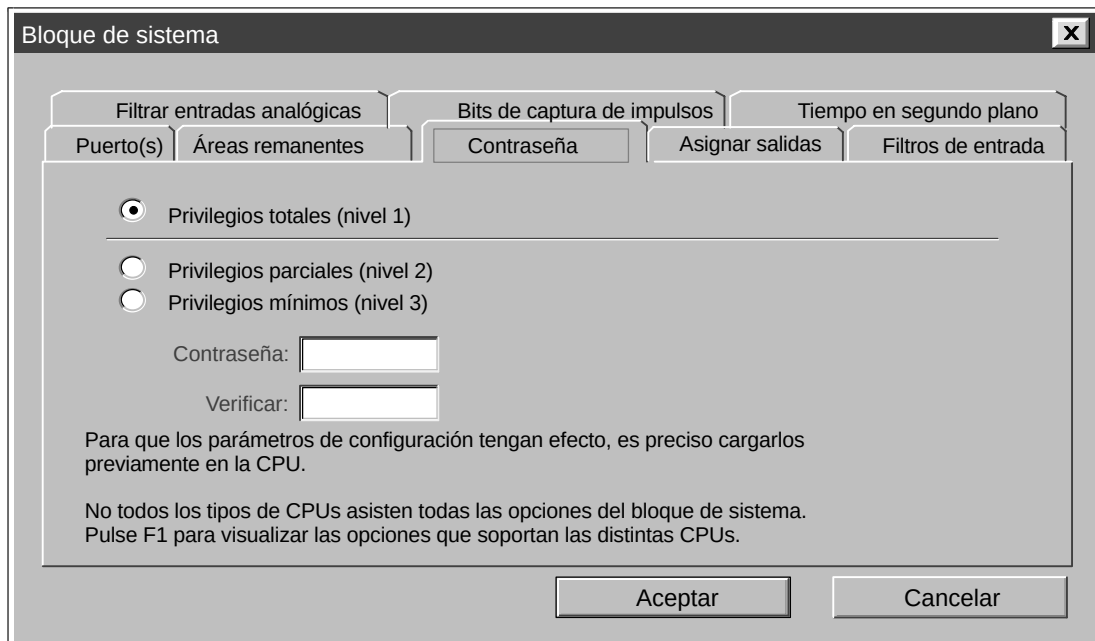


Figura 4-14 Configurar una contraseña para la CPU

Remedio si se olvida la contraseña de la CPU

Si se olvida la contraseña, es preciso efectuar un borrado total de la memoria de la CPU y volver a cargar el programa. Al borrar la memoria de la CPU, ésta pasa a modo STOP y recupera los ajustes estándar, con excepción de la dirección de estación, la velocidad de transferencia y el reloj de tiempo real.

Si desea borrar el programa de la CPU, elija el comando de menú **CPU > Borrar...** con objeto de visualizar el cuadro de diálogo "Borrar CPU". Seleccione los tres bloques y confirme haciendo clic en el botón "Aceptar". Si ha configurado una contraseña, aparecerá el cuadro de diálogo "Contraseña". Introduciendo la contraseña "clearplc" podrá iniciar el borrado total.

La función de borrado total no borra el programa contenido en el cartucho de memoria. Puesto que en éste último se encuentra almacenado no sólo el programa, sino también la contraseña, es preciso volver a programar también dicho cartucho para borrar la contraseña olvidada.



Precaución

Al efectuarse un borrado total de la CPU, se desactivan las salidas (si son salidas analógicas, éstas se congelan en un valor determinado).

Si la CPU S7-200 está conectada a otros equipos durante el borrado total, es posible que los cambios de las salidas se transfieran también a dichos equipos. Si ha determinado que el "estado seguro" de las salidas sea diferente al ajustado de fábrica, es posible que los cambios de las salidas provoquen reacciones inesperadas en los equipos, lo que podría causar la muerte o heridas graves personales y/o daños materiales.

Adopte siempre las medidas de seguridad apropiadas y asegúrese de que su proceso se encuentra en un estado seguro antes de efectuar un borrado total de la CPU.

4.9 Comprobar y observar el programa

STEP 7-Micro/WIN 32 ofrece diversas posibilidades para comprobar y observar el programa de usuario.

Ejecutar el primer ciclo o varios ciclos para observar el programa

Es posible indicar que la CPU ejecute el programa durante un número limitado de ciclos (entre 1 y 65.535 ciclos). Seleccionando el número de ciclos que la CPU debe ejecutar, se puede observar el programa a medida que van cambiando las variables del proceso. Elija el comando de menú **Test > Varios ciclos** para especificar el número de ciclos a ejecutar. La figura 4-15 muestra el cuadro de diálogo para introducir el número de ciclos a ejecutar por la CPU.

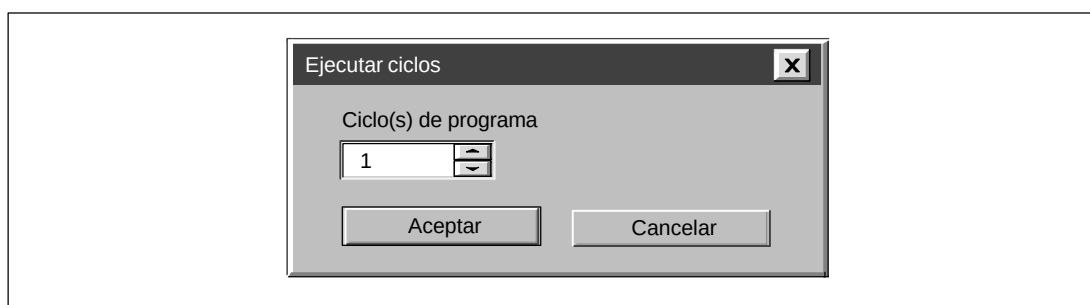


Figura 4-15 Ejecutar el programa un número determinado de ciclos

Utilizar una tabla de estado para observar y modificar el programa

Como muestra la figura 4-16, la tabla de estado se puede utilizar para leer, escribir, forzar y observar las variables mientras se ejecuta el programa. Elija el comando de menú **Ver > Tabla de estado**.

- Los botones de la barra de herramientas de la tabla de estado se visualizan en el área de la barra de herramientas de STEP 7-Micro/WIN 32. Dichos botones (Orden ascendente, Orden descendente, Lectura sencilla, Escribir todo, Forzar, Desforzar, Desforzar todo y Leer todo) se pueden activar al seleccionar la tabla de estado.
- Es posible crear varias tablas de estado.
- Para seleccionar el formato de una celda, seleccione la celda y pulse el botón derecho del ratón a fin de abrir la lista desplegable (v. fig. 4-16).



Figura 4-16 Observar y modificar variables con una tabla de estado

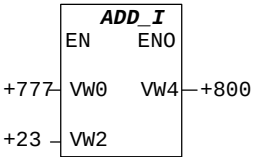
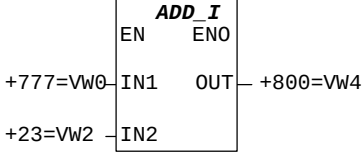
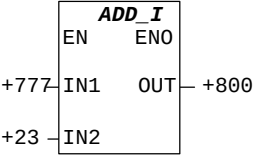
Visualizar el estado del programa en KOP

El estado del programa KOP se puede ver en STEP 7-Micro/WIN 32. STEP 7-Micro/WIN 32 debe estar visualizando el programa KOP. El estado KOP muestra el estado de todos los valores de los operandos de las operaciones (v. fig. 4-17). Todas las informaciones de estado se basan en los valores leídos al final de un ciclo de la CPU. STEP 7-Micro/WIN 32 adquiere los valores para visualizar el estado durante varios ciclos de la CPU, actualizando luego la ventana de estado KOP. Por consiguiente, el estado KOP visualizado no refleja el estado real de ejecución de cada elemento KOP.

Utilice el cuadro de diálogo "Opciones" para configurar la pantalla del estado KOP. Elija el comando de menú **Herramientas > Opciones** y seleccione la ficha "Estado KOP". La tabla 4-13 muestra las opciones de visualización del estado KOP.

Para abrir la ventana del estado KOP, haga clic en el botón "Estado del programa" de la barra de herramientas (v. fig. 4-17).

Tabla 4-13 Seleccionar las opciones de visualización para el estado KOP

Opción de visualización	Visualización del estado KOP
Mostrar la dirección dentro de la operación y el valor fuera de la operación.	
Mostrar la dirección y el valor fuera de la operación.	
Mostrar sólo el valor de estado.	

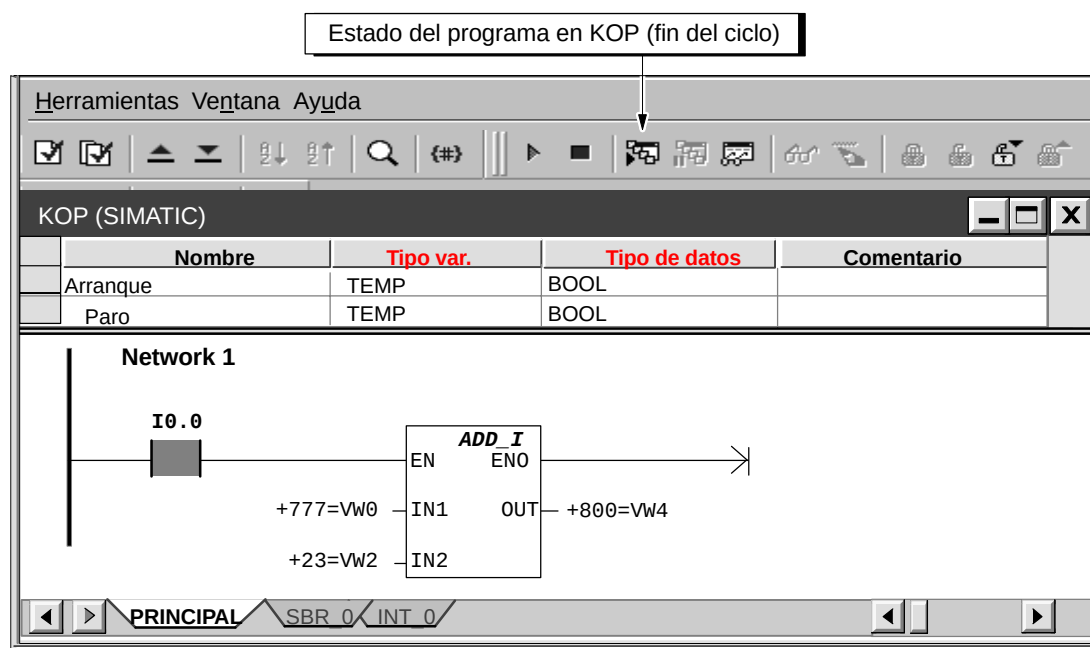


Figura 4-17 Visualizar el estado de un programa en KOP

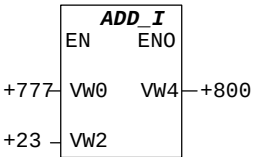
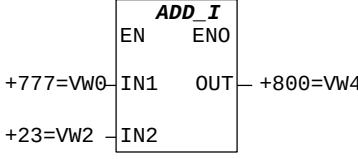
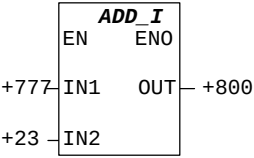
Visualizar el estado del programa en FUP

El estado del programa FUP se puede ver en STEP 7-Micro/WIN 32. STEP 7-Micro/WIN 32 debe estar visualizando el programa FUP. El estado FUP muestra el estado de todos los valores de los operandos de las operaciones. Todas las informaciones de estado se basan en los valores leídos al final de un ciclo de la CPU. STEP 7-Micro/WIN 32 adquiere los valores para visualizar el estado durante varios ciclos de la CPU, actualizando luego la ventana de estado FUP. Por consiguiente, el estado FUP visualizado no refleja el estado real de ejecución de cada elemento FUP.

Utilice el cuadro de diálogo "Opciones" para configurar la pantalla del estado FUP. Elija el comando de menú **Herramientas > Opciones** y seleccione la ficha "Estado FUP". La tabla 4-14 muestra las opciones de visualización del estado FUP.

Para abrir la ventana de estado FUP, haga clic en el botón "Estado del programa" de la barra de herramientas (v. fig. 4-18).

Tabla 4-14 Seleccionar las opciones de visualización para el estado FUP

Opción de visualización	Visualización del estado FUP
Mostrar la dirección dentro de la operación y el valor fuera de la operación.	
Mostrar la dirección y el valor fuera de la operación.	
Mostrar sólo el valor de estado.	

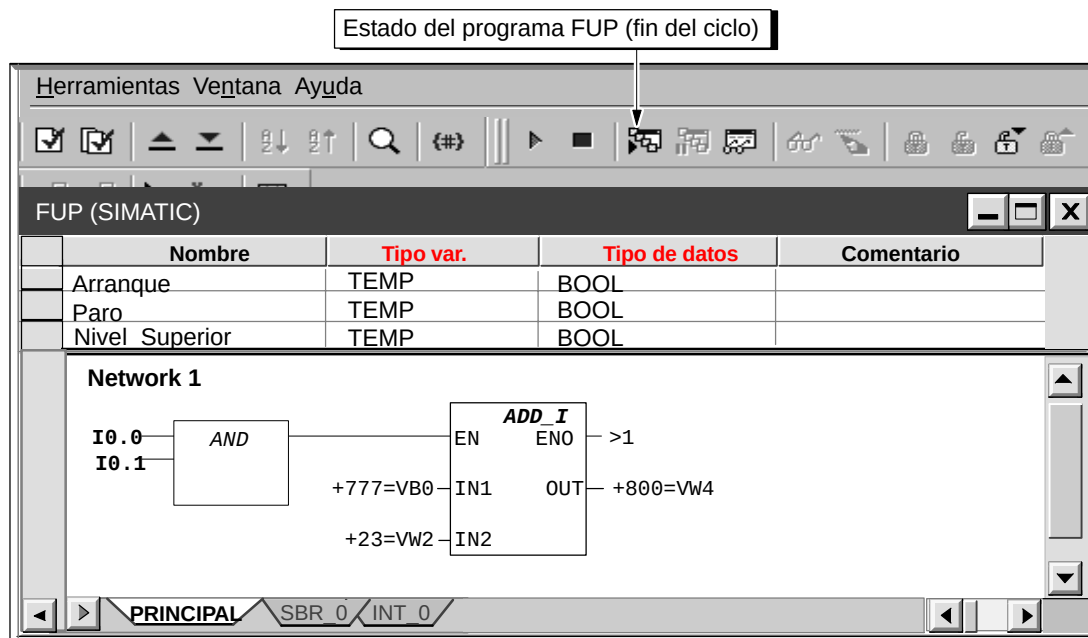


Figura 4-18 Visualizar el estado de un programa en FUP

Visualizar el estado del programa en AWL

Si el programa se visualiza utilizando el editor AWL, STEP 7-Micro/WIN 32 ofrece un método para observar el estado de ejecución del programa tras evaluarse cada una de las operaciones. Este método de observación del estado se denomina "estado AWL". La sección del programa para la que se ha habilitado el estado AWL se denomina "ventana de estado AWL". El tamaño de esta ventana es aproximadamente igual al de la pantalla de STEP 7-Micro/WIN 32. Las informaciones recogidas de la CPU se limitan a 200 bytes ó 25 líneas. Las líneas del programa AWL que excedan dichos límites, aparecerán marcadas con "-" en la pantalla de estado. Las informaciones de estado se recogen comenzando en la primera operación AWL, en el lado superior de la ventana del editor. A medida que se navega hacia abajo por la ventana del editor, se obtienen nuevas informaciones de la CPU. La figura 4-19 muestra una ventana de estado AWL típica.

Para abrir la ventana de estado AWL, haga clic en el botón "Estado del programa" de la barra de herramientas (v. fig. 4-19). Arrastre la línea divisoria derecha de la ventana del programa para visualizar la ventana del estado AWL.

Cuando se activa el estado del programa en AWL, el editor de programas se divide en una sección de código (a la izquierda) y una sección de estado (a la derecha) que contiene los valores de los operandos. Los operandos de direcciones indirectas muestran tanto la dirección del puntero como el valor de ésta. La dirección del puntero aparece entre paréntesis.

Activando el botón "Estado del programa" en AWL, los valores se actualizan continuamente en la pantalla. Si desea detener la actualización en la pantalla hasta la siguiente actualización de los datos, haga clic en el botón "Detener" (v. fig. 4-19). Los datos actuales permanecerán en la pantalla hasta que el botón se desactive nuevamente.

Los valores de los operandos se muestran en columnas en el orden en el que aparecen en cada operación. Estos valores se recogen cada vez que se ejecuta la operación, por lo que reflejan su estado real de ejecución.

El color de los valores indica el estado de ejecución de la operación:

- El color negro indica que la operación se ha ejecutado correctamente. Las operaciones incondicionales no incluidas en un bloque SCR se ejecutarán siempre, sin depender del estado de la pila lógica. La ejecución de las operaciones condicionales depende de la pila lógica.
- El color rojo indica que se ha detectado un error en la ejecución. Consulte la descripción de la operación en cuestión en el capítulo 9 para determinar cuáles condiciones de error ponen ENO a 0.
- El color gris indica que la operación no se ha ejecutado, debido a que el valor superior de la pila (s0) es igual a 0, o bien porque la operación se encuentra en un bloque SCR que no se ha habilitado.
- Un espacio en blanco indica que la operación no se ha ejecutado.

A continuación se indican condiciones que impiden la ejecución de las operaciones:

- El valor superior de la pila lógica es igual a 0.
- La operación se ha saltado a causa de operaciones de salto u otras instrucciones del programa.
- La CPU no está en modo RUN.

Para seleccionar las categorías de valores que desea visualizar en la pantalla de estado AWL, elija el comando de menú **Herramientas > Opciones** y haga clic en la ficha "Estado AWL". En la ventana de estado AWL se pueden observar tres categorías de valores, a saber:

- Operandos (máx. tres por operación)
- Pila lógica (los cuatro valores más recientes de la pila lógica, como máximo)
- Bits de estado de las operaciones (máx. 12)

Para consultar la información del primer ciclo, cambie la CPU a modo STOP, active el estado AWL y elija luego el comando de menú **Test > Primer ciclo**.

Estado del programa AWL (en tiempo de ejecución) Detener

Nombre	Tipo var.	Tipo de datos	Comentario
	TEMP		
	TEMP		

	Op 1	Op 2	Op 3	0
OD<= *VD0 *VD4				
// V SCRR S2 S1 S0 / Neg				
OD<= *VD0 *VD4				
// V SCRR S2 S1 S0 / Neg				
OD<= *VD0 *VD4				
// V SCRR S2 S1 S0 / Neg				
OD<= *VD0 *VD4				
SCRE				1 1
NETWORK.7				
// *****				
// Comprobar el esquema de colores				
// Función de comunicación				
//				
LD SM0.0	+1	+1		1 1
MOVW +1, VW0	-1	-1		1 1
*I -1, VW0				1 1
LD SM0.0				1 1
MOVB &VB4, VD0	16#8000004	+13217732		1 1
MOVB 1, VB4	1	1		1 1
MOVB 0, VB1500	0	0		1 1
MOVB 3, VB1501	3	3		1 1
MOVB &VB0, VD1502	16#8000000	+134217728		1 1
MOVB 0, VB1506	0	0		1 1
MOVB &VB1500, VD1508	16#0005DC	+134219228		1 1
MOVB 16#80, SMB87	16#80	128		1 1
MOVB &VB1520, VD1524	16#80005F0	+13421948		1 1
// Verificar el esquema de color rojo				
// V S0 / Neg / 1 (VB4)				
XMT *VD0, 0	1 (VB4)			1 0
// V S0 / Neg / 1 (VB4)				
RCV *VD0, 0	1 (VB4)			1 0
// V / S0 / Neg / 165 (VB1500)				
NETR *VD1508, 0	165			1 0
// V / S0 / Neg / 165 (VB1500)				
NETW *VD1508, 0				- -

PRINCIPAL SBR_0

Figura 4-19 Visualizar el estado de un programa en AWL

Forzar valores específicos

La CPU S7-200 permite forzar algunas o todas las entradas y salidas (I y Q). Además es posible forzar hasta 16 marcas internas (V o M) o bien los valores de las entradas y salidas analógicas (AI o AQ). Los valores de la memoria V o de las marcas se pueden forzar en formato de bytes, palabras o palabras dobles. Los valores analógicos se fuerzan únicamente como palabras, en bytes de número par (p.ej. AIW6 ó AQW14). Todos los valores forzados se almacenan en la memoria EEPROM no volátil de la CPU.

Puesto que los valores forzados se pueden modificar durante el ciclo (por el programa, al actualizarse las entradas y salidas o al procesarse las comunicaciones), la CPU los vuelve a forzar en diversos puntos del ciclo. La figura 4-20 muestra el ciclo, indicando dónde la CPU actualiza las variables forzadas.

La función Forzar se impone a las operaciones de lectura y de escritura directas. Asimismo, tiene prioridad sobre las salidas que deban adoptar un valor específico cuando la CPU pase a modo STOP. En este último caso, la salida conservará el valor forzado y no el valor configurado.

Como muestra la figura 4-21, la tabla de estado se puede utilizar para forzar valores. Para forzar un nuevo valor, introduzca el valor en la columna "Nuevo valor" de la tabla de estado y haga clic en el botón "Forzar" en la barra de herramientas. Para forzar un valor existente, destaque el valor en la columna "Valor actual" y pulse luego el botón "Forzar".

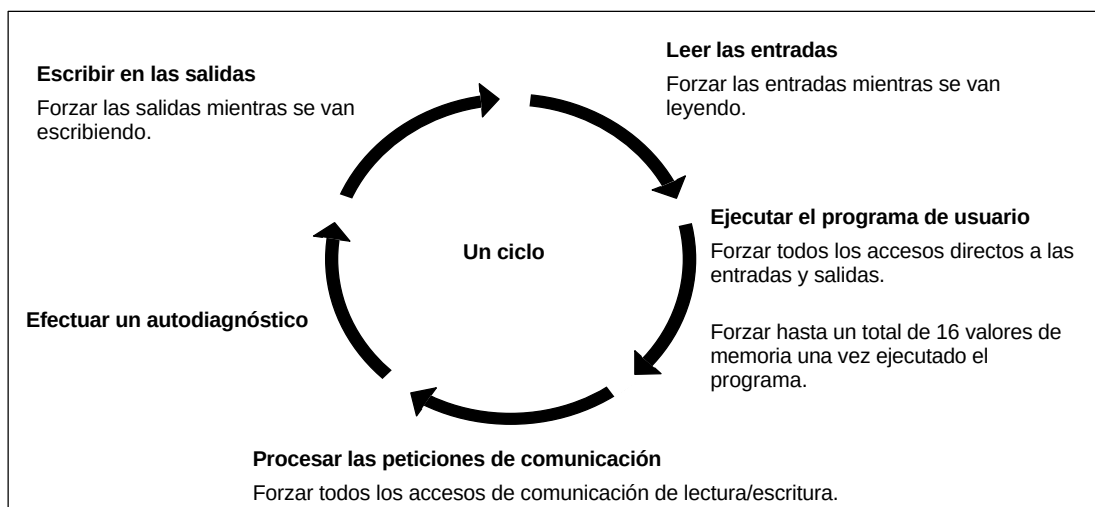


Figura 4-20 Ciclo de la CPU S7-200

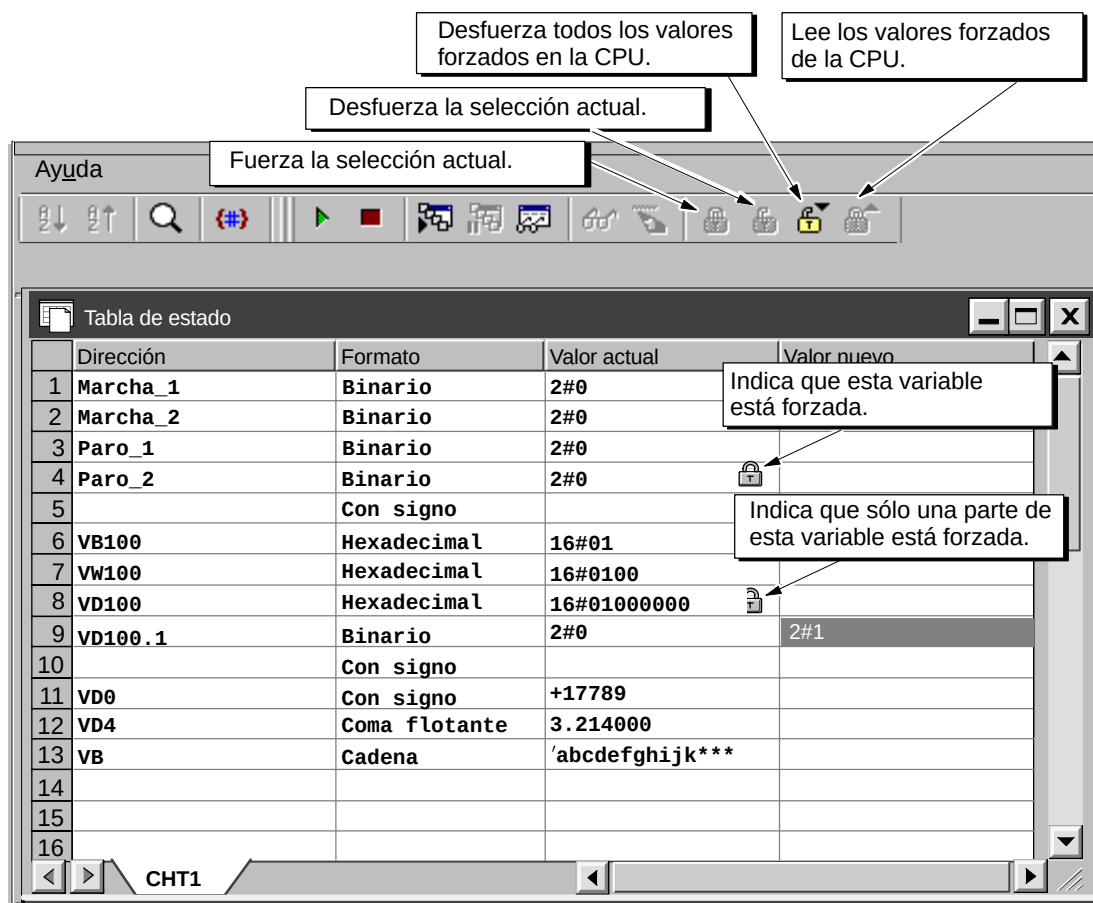


Figura 4-21 Forzar variables mediante la tabla de estado

4.10 Editar programas en modo RUN

La CPU 224 (a partir de la versión 1.10) y la CPU 226 (a partir de la versión 1.00) soportan la edición de programas en modo RUN. Esta función permite efectuar cambios pequeños en el programa de usuario sin afectar demasiado a los equipos controlados. No obstante, también es posible realizar modificaciones considerables que podrían ser perjudiciales o incluso peligrosas.



Precaución

Si los cambios se cargan en una CPU que se encuentre en modo RUN, afectarán directamente al proceso. Si el programa se modifica estando la CPU en modo RUN, es posible que se produzcan reacciones inesperadas en los equipos, lo que podría causar la muerte o lesiones graves personales y/o daños materiales.

Sólo el personal cualificado que tenga pleno conocimiento de los efectos de esta función debería editar programas en modo RUN.

Para poder editar un programa en modo RUN, es preciso cumplir los siguientes requisitos previos:

- La CPU conectada (online) debe soportar la edición de programas en modo RUN.
- La CPU conectada (online) debe estar en modo RUN.

Para editar en modo RUN, siga los pasos siguientes:

1. Elija el comando de menú **Test > Editar programa en RUN**. (La figura 4-22 muestra el menú Test en el lado derecho de la pantalla.)
2. Si el proyecto es diferente al programa contenido en la CPU, se le pregunta si desea guardarlo. En modo RUN sólo se puede editar el programa de la CPU.

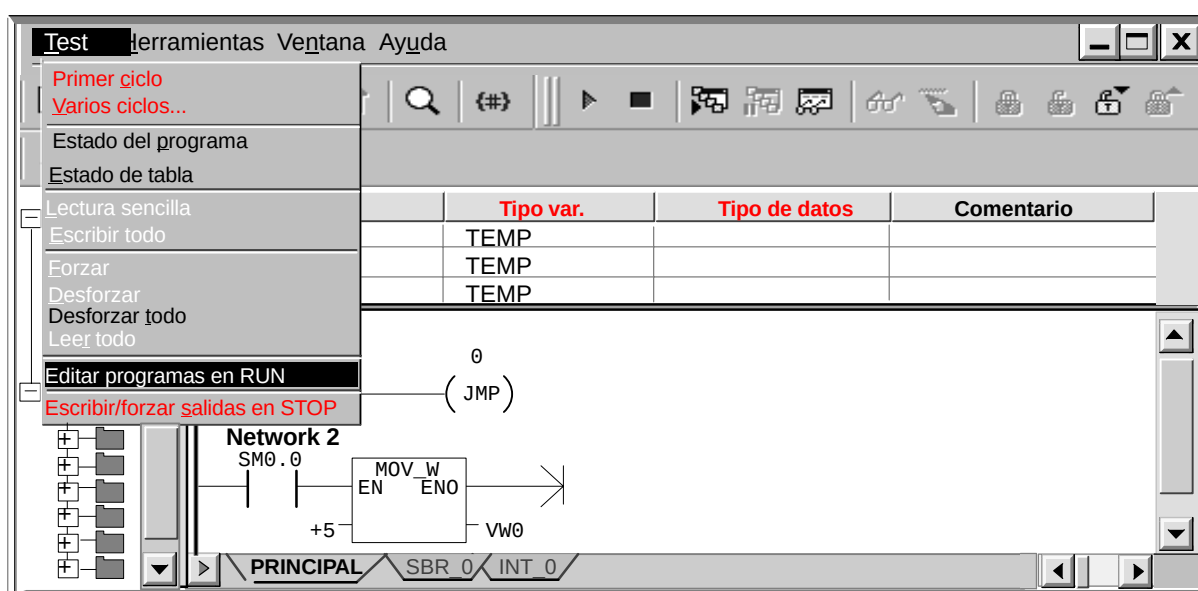


Figura 4-22 Visualizar el estado de un programa en FUP

3. Aparecerá la advertencia que muestra la figura 4-23. Tras hacer clic en el botón "Continuar", el programa contenido en la CPU se cargará en la PG/el PC y se activará la función de edición en modo RUN. El programa se podrá editar sin limitación alguna.

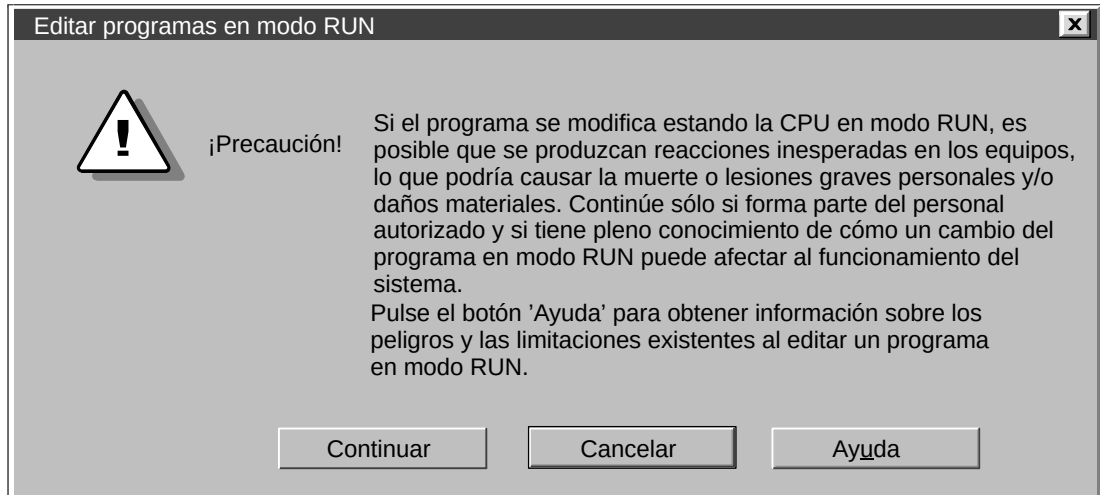


Figura 4-23 Cuadro de diálogo "Editar programas en modo RUN"

Nota

Las operaciones Detectar flanco positivo (EU) y Detectar flanco negativo (ED) se muestran con un operando. Para visualizar informaciones sobre el estado anterior de las operaciones de flanco, haga clic en el icono "Referencias cruzadas" de la sección "Ver" de la pantalla. En la ficha "Flancos usados" se indican los números de los flancos utilizados en el programa. Tenga cuidado de no asignar números de flancos repetidos cuando edite el programa.

Consideraciones antes de cargar el programa en la CPU en modo RUN

La función de edición en modo RUN permite cargar el programa en la CPU mientras ésta se encuentra en modo RUN. Para decidir si los cambios del programa se deben cargar en la CPU en modo RUN o STOP, tenga en cuenta los siguientes aspectos:

- Si, editando en modo RUN, se borra la lógica de control de una salida, ésta conservará su último estado hasta la próxima vez que la CPU se desconecte y se conecte de nuevo, o bien, hasta que cambie a modo STOP.
- Si, editando en modo RUN, se están ejecutando las operaciones HSC o PTO/PWM, éstas se seguirán ejecutando hasta la próxima vez que la CPU se desconecte y se conecte de nuevo, o bien, hasta que cambie a modo STOP.
- Si, editando en modo RUN, se borra una operación ATCH, sin borrarse también la rutina de interrupción asociada, ésta se seguirá ejecutando hasta la próxima vez que la CPU se desconecte y se conecte de nuevo, o bien, hasta que cambie a modo STOP. De forma similar, si se borra una operación DTCH, la correspondiente rutina de interrupción no se culminará hasta que la CPU se desconecte y se conecte de nuevo, o bien, hasta que cambie a modo STOP.
- Si, editando en modo RUN, se añade una operación ATCH, deseando que se ejecute en el primer ciclo, la operación no se activará hasta que la CPU se desconecte y se conecte de nuevo, o bien, hasta el próximo cambio de modo STOP a RUN.
- Si, editando en modo RUN, se borra una operación ENI, las correspondientes rutinas de interrupción se continuarán ejecutando hasta que la CPU se desconecte y se conecte de nuevo, o bien, hasta que cambie a modo STOP.
- Si, editando en modo RUN, se modifica la dirección de tabla de un cuadro RCV y éste se encuentra activo en el momento de cambiar del programa antiguo al programa nuevo, los datos se escribirán en la antigua dirección de tabla. Ello mismo es aplicable a las operaciones NETR y NETW.
- La lógica determinada por la marca del primer ciclo no se ejecutará hasta que no se cambie de nuevo a modo RUN, bien sea tras desconectar y conectar la CPU, o bien después de un cambio de STOP a RUN. Ello se debe a que la marca del primer ciclo no se ve afectada por la edición en modo RUN.

Cargar un programa en la CPU en modo RUN

Para cargar un programa en la CPU en modo RUN, haga clic en el botón "Cargar en CPU" de la barra de herramientas o elija el comando de menú **Archivo > Cargar en CPU**. El programa se cargará en la CPU si se compila correctamente. El proceso de carga se indica en una barra de progreso. Tenga en cuenta que sólo es posible cargar el bloque de programa en la CPU. Para poder activar la operación de carga se deberán cumplir los siguientes requisitos previos:

- La CPU deberá soportar la edición de programas en modo RUN.
- El programa se deberá haber compilado sin errores.
- No puede haber fallos en la comunicación entre el ordenador donde se está ejecutando STEP 7-Micro/WIN 32 y la CPU.

Salir de la edición en modo RUN

Para salir de la edición en modo RUN, elija el comando de menú **Test > Editar programa en RUN** y haga clic en la marca de verificación para desactivarla. Si ha efectuado cambios que no se hayan guardado todavía, puede decidir si desea continuar editando, cargar los cambios en la CPU y salir de la edición en modo RUN, o bien salir sin cargar en la CPU.

4.11 Seleccionar el tiempo en segundo plano

Es posible configurar el porcentaje del tiempo de ciclo que se dedicará a procesar las peticiones de comunicación correspondientes a la compilación del programa tras editar en modo RUN, o bien en relación con la consulta del estado AWL. Si se incrementa el porcentaje de tiempo dedicado a procesar las peticiones de comunicación, aumentará también el tiempo de ciclo, por lo que el proceso de control se ejecutará más lentamente.

Haga clic en la ficha "Tiempo en segundo plano" para ajustar el porcentaje del tiempo de ciclo permitido para la comunicación en segundo plano (v. fig. 4-24). Edite las propiedades del tiempo de comunicación en segundo plano y cargue luego los cambios en la CPU.

El ajuste estándar de dicho porcentaje es 10%. Este ajuste se ha elegido como compromiso apropiado para procesar la compilación y/o las operaciones de estado, reduciendo al mismo tiempo el impacto en el proceso de control. El valor se puede ajustar en incrementos de 5% hasta un máximo de 50%.

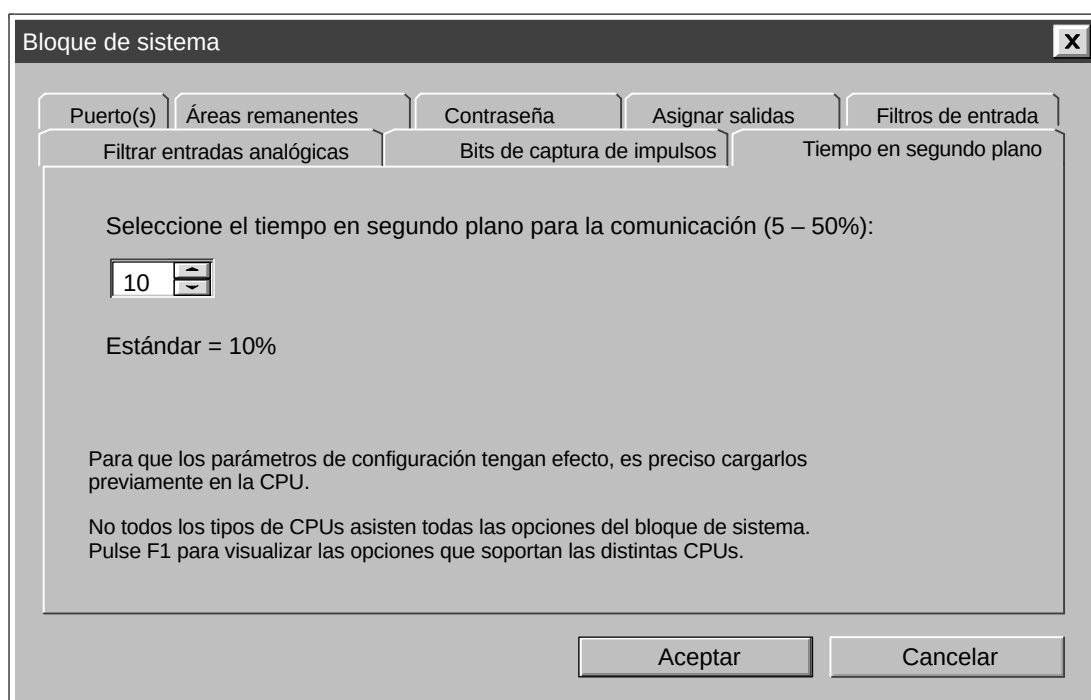


Figura 4-24 Ventana de configuración del tiempo en segundo plano

4.12 Eliminar errores de las CPUs S7-200

Las CPUs S7-200 clasifican los errores en errores fatales y no fatales. STEP 7-Micro/WIN 32 permite visualizar los códigos generados por los errores. Elija el comando de menú **CPU > Información** para visualizar los errores. La figura 4-25 muestra un cuadro de diálogo donde se visualizan el código y la descripción del error. El Anexo B incluye una lista completa de los códigos de error.

En la figura 4-25, el campo "Último fatal" muestra el último código de error fatal generado por la CPU. Al desconectarse la alimentación, este valor se conserva si se respalda la RAM. El valor se pone a 0 si se efectúa un borrado total de la CPU o si la RAM no se respalda tras un corte prolongado de la alimentación.

El campo "Total fatales" muestra el contaje de los errores fatales generados por la CPU desde la última vez que se efectuó un borrado total de la misma. Al desconectarse la alimentación, este valor se conserva si se respalda la RAM. El valor se pone a 0 si se efectúa un borrado total de la CPU o si la RAM no se respalda tras un corte prolongado de la alimentación.

Información CPU

Modo de operación:

Versiones

CPU

Firmware

ASIC

Tiempos de ciclo (ms)

Último

Errores

Fatales No se han presentado errores fatales.

No fatales No se han presentado errores no fatales.

Último fatal No se han presentado errores no fatales.

Total fatales

Módulo	Tipo	E/S	Dirección	Estado
CPU	E/S digitales	16 E/16 S	10.0 / Q0.0	Sin error
	E/S digitales	0 E/8 S	10.0 / Q2.0	Sin error
2	Nombre del módulo			No existente
3				No existente
4				No existente
5				No existente

Información del módulo inteligente Reseteear tiempos ciclo Cerrar

Utilice la descripción y el código para eliminar la posible causa del error.

Figura 4-25 Cuadro de diálogo "Información CPU": ficha "Estado de error"

Eliminar errores fatales

Cuando ocurre un error fatal, la CPU detiene la ejecución del programa. Según la gravedad del error, es posible que la CPU no pueda ejecutar todas las funciones, o incluso ninguna de ellas. El objetivo del tratamiento de errores fatales es conducir a la CPU a un estado seguro, en el que se puedan analizar y eliminar las condiciones que hayan causado el error. Cuando la CPU detecta un error fatal, cambia a modo STOP, enciende los indicadores "SF" y "STOP" y desactiva las salidas. La CPU permanece en dicho estado hasta que haya eliminado la causa del error fatal.

Una vez efectuados los cambios para eliminar el error fatal, es preciso rearmar la CPU. La CPU se puede rearmar utilizando uno de los métodos siguientes:

- Desconectando la alimentación y conectándola luego nuevamente.
- Cambiando el selector de modos de RUN o TERM a STOP.
- Utilizando STEP 7-Micro/WIN. STEP 7-Micro/WIN 32 incorpora el comando de menú **CPU > Reset arranque** que obliga a la CPU a efectuar un rearmado y a borrar todos los errores fatales.

Al rearmar la CPU se borra la condición de error fatal y se ejecuta un diagnóstico de arranque para verificar si se ha corregido el error. En caso de detectarse otro error fatal, se encenderá de nuevo el indicador "SF". De lo contrario, la CPU comenzará a funcionar con normalidad.

Existen diversas condiciones posibles de error que incapacitan a la CPU para la comunicación. En esos casos no es posible visualizar el código de error de la CPU. Dichos errores indican fallos de hardware, por lo que es necesario reparar la CPU. No se pueden solucionar modificando el programa ni borrando la memoria de la CPU.

Eliminar errores no fatales

Los errores no fatales pueden mermar parcialmente el funcionamiento de la CPU, pero no le impiden ejecutar el programa o actualizar las entradas y salidas. Como muestra la figura 4-25, STEP 7-Micro/WIN 32 permite visualizar los códigos generados por los errores no fatales. Hay tres categorías básicas de errores no fatales:

- Errores durante el tiempo de ejecución. Todos los errores no fatales que se detectan en modo RUN se depositan en marcas especiales (SM). El programa puede observar y evaluar dichas marcas. Consulte el Anexo C para obtener más información acerca de las marcas especiales utilizadas para indicar los errores no fatales durante el tiempo de ejecución.

Cuando se enciende la CPU, ésta lee la configuración de las entradas y salidas, almacenando dicha información en la memoria de datos del sistema y en las marcas especiales. Durante el funcionamiento normal de la CPU, el estado de las entradas y salidas se actualiza periódicamente y se almacena en las marcas especiales. Si la CPU detecta una configuración de E/S diferente, activa el correspondiente bit del byte de error en el módulo. El módulo de ampliación no se actualizará hasta que dicho bit se desactive de nuevo. Para que la CPU pueda desactivar ese bit, las entradas y salidas del módulo deberán coincidir nuevamente con la configuración almacenada en la memoria de datos del sistema.

- Errores de compilación del programa. Al cargar un programa en la CPU, ésta lo compila. Si durante la compilación se detecta una violación de las reglas, el proceso de carga se suspenderá, generándose entonces un código de error. (Si ya se ha cargado un programa en la CPU, seguirá existiendo en la EEPROM, por lo que no se perderá). Una vez corregido el programa, se podrá cargar de nuevo.
- Errores de programación durante el tiempo de ejecución. El programa puede crear condiciones de error mientras se ejecuta el programa. Por ejemplo, un puntero de direccionamiento indirecto que era válido cuando se compiló el programa puede haber cambiado durante la ejecución del programa, señalando entonces a una dirección fuera de área. Esto se considera un error de programación durante el tiempo de ejecución. Utilice el cuadro de diálogo que muestra la figura 4-25 en la página 4-43 para determinar el tipo de error que ha ocurrido.

La CPU no cambia a modo STOP cuando detecta un error no fatal. Tan sólo deposita el evento en la marca especial en cuestión y continúa ejecutando el programa. No obstante, es posible programar que la CPU cambie a modo STOP cuando se detecte un error no fatal. La figura 4-26 muestra un segmento de un programa que controla una marca especial. La operación prevé que la CPU cambie a modo STOP si se detecta un error de E/S.

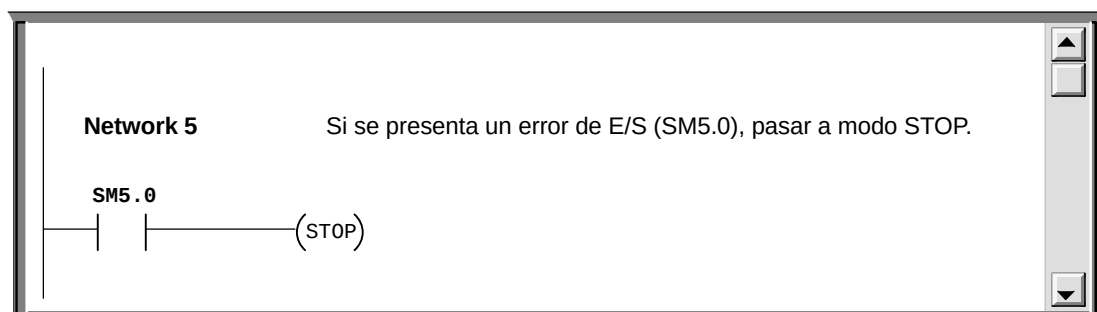


Figura 4-26 Detectar errores no fatales mediante el programa de usuario

Memoria de la CPU: tipos de datos y direccionamiento

5

La CPU S7-200 dispone de áreas de memoria especiales para que los datos se puedan procesar de forma más rápida y eficiente.

Índice del capítulo

Apartado	Descripción	Página
5.1	Direccionamiento directo de las áreas de memoria de la CPU	5-2
5.2	Direccionamiento indirecto (SIMATIC) de las áreas de memoria de la CPU	5-13
5.3	Respaldar datos en la CPU S7-200	5-15
5.4	Utilizar el programa para guardar datos en la memoria no volátil	5-20
5.5	Guardar el programa en el cartucho de memoria	5-22

5.1 Direccionamiento directo de las áreas de memoria de la CPU

La CPU S7-200 almacena información en diferentes áreas de la memoria que tienen direcciones unívocas. Es posible indicar explícitamente la dirección a la que se desea acceder. El programa puede acceder entonces directamente a la información.

Acceder a los datos a través de direcciones

Para acceder a un bit en un área de memoria es preciso indicar la dirección del mismo, la cual está formada por un identificador de área, la dirección del byte y el número del bit. La figura 5-1 muestra un ejemplo de direccionamiento de un bit (denominado también direccionamiento "byte.bit"). En el ejemplo, el área de memoria y la dirección del byte (I=entrada y 3=byte 3) van seguidas de un punto decimal (".") que separa la dirección del bit (bit 4).

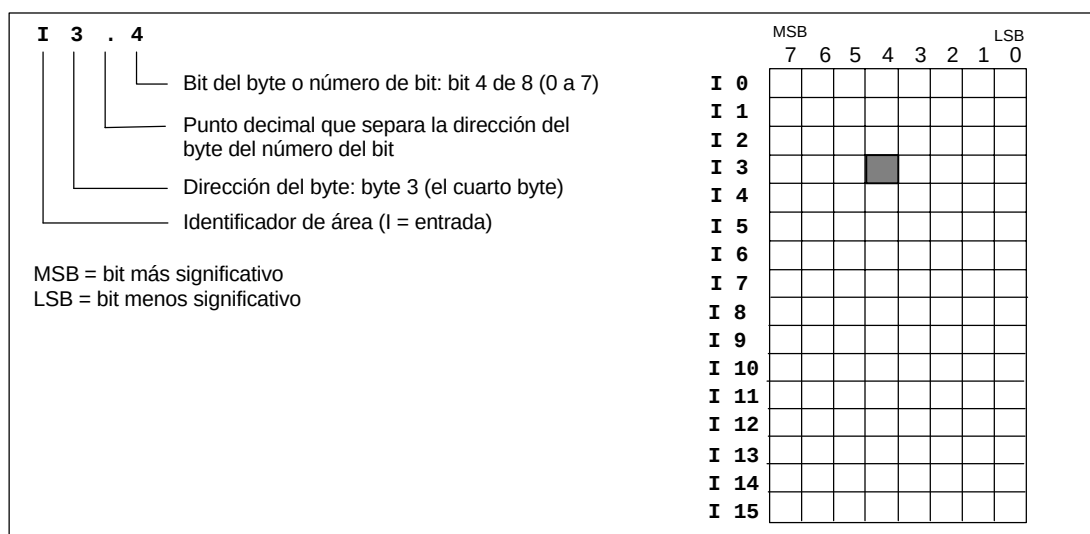


Figura 5-1 Acceder a un bit de datos en la memoria de la CPU (direccionamiento "byte.bit")

Utilizando el formato de dirección de byte se puede acceder a los datos de numerosas áreas de la memoria de la CPU (V, I, Q, M, S y SM) en formato de bytes, palabras o palabras dobles. La dirección de un byte, de una palabra o de una palabra doble de datos en la memoria de la CPU se indica de forma similar a la dirección de un bit. Esta última está formada por un identificador de área, el tamaño de los datos y la dirección inicial del valor del byte, de la palabra o de la palabra doble, como muestra la figura 5-2. Para acceder a los datos comprendidos en otras áreas de la memoria de la CPU (p.ej. T, C, HC y acumuladores) es preciso utilizar una dirección compuesta por un identificador de área y un número de elemento.

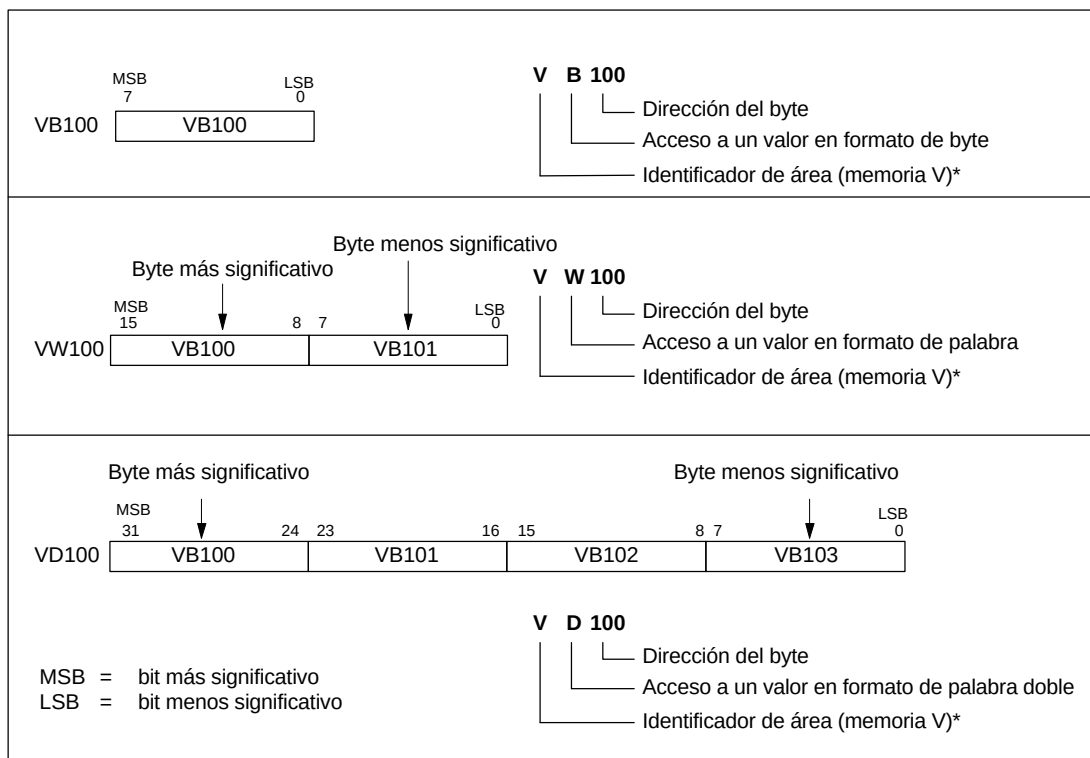


Figura 5-2 Acceso a la misma dirección en formato de byte, palabra y palabra doble

Representación numérica

La tabla 5-1 muestra el margen de números enteros representables en diversos tamaños de datos.

Los números reales (en coma flotante) se representan como números de precisión simple de 32 bits, siendo su formato: +1,175495E-38 a +3,402823E+38 (positivo), y -1,175495E-38 a -3,402823E+38 (negativo). A los valores de números reales se accede en formato de palabra doble. Para obtener más información sobre los números reales (o en coma flotante), consulte la norma ANSI/IEEE 754-1985.

Tabla 5-1 Indicadores de tamaño (y sus respectivos márgenes de números enteros)

Tamaño de los datos	Margen de enteros sin signo		Margen de enteros con signo	
	Decimal	Hexadecimal	Decimal	Hexadecimal
B (byte): valor de 8 bits	0 a 255	0 a FF	-128 a 127	80 a 7F
W (palabra): valor de 16 bits	0 a 65.535	0 a FFFF	-32.768 a 32.767	8000 a 7FFF
D (palabra doble): valor de 32 bits	0 a 4.294.967.295)	0 a FFFF FFFF	-2.147.483.648 a 2.147.483.647	8000 0000 a 7FFF FFFF

Direccionamiento de la imagen del proceso de las entradas (I)

Como se describe en el apartado 4.6, la CPU lee las entradas físicas al comienzo de cada ciclo y escribe los correspondientes valores en la imagen del proceso de las entradas. A ésta última se puede acceder en formato de bit, byte, palabra o palabra doble.

Formato:

Bit **I[direcc. del byte].[direcc. del bit] I0.1**
 Byte, palabra, palabra doble **I[tamaño][direcc. del byte inicial] IB4**

Direccionamiento de la imagen del proceso de las salidas (Q)

Al final de cada ciclo, la CPU copia en las salidas físicas el valor almacenado en la imagen del proceso de las salidas. A ésta última se puede acceder en formato de bit, byte, palabra o palabra doble.

Formato:

Bit **Q[direcc. del byte].[direcc. del bit] Q1.1**
 Byte, palabra, palabra doble **Q[tamaño][direcc. del byte inicial] QB5**

Direccionamiento de la memoria de variables (V)

La memoria de variables (memoria V) se puede utilizar para depositar los resultados intermedios calculados por las operaciones en el programa. La memoria V también permite almacenar otros datos que pertenezcan al proceso o a la tarea actuales. A la memoria de variables se puede acceder en formato de bit, byte, palabra o palabra doble.

Formato:

Bit	V[direcc. del byte].[direcc. del bit]	V10.2
Byte, palabra, palabra doble	V[tamaño][direcc. del byte inicial]	VW100

Direccionamiento del área de marcas (M)

El área de marcas (memoria M) se puede utilizar en calidad de relés de control para almacenar el estado inmediato de una operación u otra información de control. Al área de marcas se puede acceder en formato de bit, byte, palabra o palabra doble.

Formato:

Bit	M[direcc. del byte].[direcc. del bit]	M26.7
Byte, palabra, palabra doble	M[tamaño][direcc. del byte inicial]	MD20

Direccionamiento de los relés de control secuencial (S)

Los relés de control secuencial (S) permiten organizar los pasos del funcionamiento de una máquina en segmentos equivalentes en el programa. Los SCRs permiten segmentar lógicamente el programa de usuario. A los relés de control secuencial (SCR) se puede acceder en formato de bit, byte, palabra o palabra doble.

Formato:

Bit	S[direcc. del byte].[direcc. del bit]	S3.1
Byte, palabra, palabra doble	S[tamaño][direcc. del byte inicial]	SB4

Direccionamiento de las marcas especiales (SM)

Las marcas especiales permiten intercambiar datos entre la CPU y el programa. Dichas marcas se puede utilizar para seleccionar y controlar algunas funciones especiales de la CPU S7-200, tales como:

- Una marca que se activa sólo en el primer ciclo.
- Marcas que se activan y se desactivan en determinados intervalos.
- Marcas que muestran el estado de operaciones matemáticas y de otras operaciones.

Para obtener más información acerca de las marcas especiales, consulte el Anexo C. Aunque el área de las marcas especiales se basa en bits, es posible acceder a los datos en formato de bit, byte, palabra o palabra doble.

Formato:

Bit	SM[direcc. del byte].[direcc. del bit]	SM0.1
Byte, palabra, palabra doble	SM[tamaño][direcc. del byte inicial]	SMB86

Direccionamiento del área de memoria local (L)

Las CPUs S7-200 disponen de 64 bytes de memoria local (L), de los cuales 60 se pueden utilizar como memoria "borrador" para transferir parámetros formales a las subrutinas. Si se programa en KOP o FUP, STEP 7-Micro/WIN 32 reserva los últimos cuatro bytes de la memoria local para su propio uso. Si se programa en AWL, **se podrá acceder a todos los 64 bytes de la memoria L, pero se recomienda no utilizar los últimos cuatro bytes de la misma.**

La memoria local es similar a la memoria V (memoria de variables), con una excepción: la memoria V tiene un alcance global, en tanto que la memoria L tiene un alcance local. El término "alcance global" significa que a una misma dirección de la memoria se puede acceder desde cualquier parte del programa (programa principal, subrutinas o rutinas de interrupción). El término "alcance local" significa que la dirección de la memoria está asociada a una determinada parte del programa. Las CPUs S7-200 asignan 64 bytes de la memoria L al programa principal, 64 bytes a cada nivel de anidado de las subrutinas y 64 bytes a las rutinas de interrupción.

A los bytes de la memoria L asignados al programa principal no se puede acceder ni desde las subrutinas ni desde las rutinas de interrupción. Una subrutina no puede acceder a la asignación de la memoria L del programa principal, ni a la de una rutina de interrupción, ni tampoco a la de otra subrutina. De igual manera, una rutina de interrupción no puede acceder a la asignación de la memoria L del programa principal ni tampoco a la de una subrutina.

La CPU S7-200 asigna la memoria L según sea necesario en ese momento. Ello significa que mientras se está ejecutando la parte principal del programa, no existen las asignaciones de la memoria L para las subrutinas y las rutinas de interrupción. Cuando ocurre una interrupción o cuando se llama a una subrutina, la memoria local se asigna según sea necesario. La nueva asignación de la memoria L puede reutilizar las mismas direcciones de la memoria L de una subrutina o de una rutina de interrupción diferentes.

La CPU no inicializa la memoria L durante la asignación de direcciones, pudiendo contener cualquier valor. Al transferir parámetros formales a una llamada de subrutina, los valores de los parámetros transferidos se depositarán en las direcciones de la memoria L que se hayan asignado a dicha subrutina. Las direcciones de la memoria L que no reciban un valor como resultado de la transferencia de parámetros formales no se inicializarán, pudiendo contener cualquier valor en el momento de la asignación.

Se puede acceder a la memoria L en formato de bits, bytes, palabras o palabras dobles. La memoria L se puede utilizar como puntero de direccionamiento indirecto, pero a las direcciones de la memoria L no se puede acceder indirectamente.

Formato:

Bit	<i>L [direcc. del byte].[direcc. del bit]</i>	<i>L0.0</i>
Byte, palabra, palabra doble	<i>L [tamaño][direcc. del byte inicial]</i>	<i>LB33</i>

Direccionamiento del área de temporizadores (T)

En las CPUs S7-200, los temporizadores son elementos que cuentan intervalos de tiempo. Los temporizadores de las CPUs S7-200 tienen resoluciones (intervalos) de 1 ms, 10 ms y 100 ms. Hay dos variables asociadas a los temporizadores:

- Valor actual: en este número entero de 16 bits con signo se deposita el valor de tiempo contado por el temporizador.
- Bit del temporizador (bit T): este bit se activa o se desactiva como resultado de la comparación del valor actual con el valor de preselección. Este último se introduce como parte de la operación del temporizador.

A estas dos variables se accede utilizando la dirección del temporizador (T + número del temporizador). Dependiendo de la operación utilizada, se accede al bit del temporizador o al valor actual. Las operaciones con operandos en formato de bit acceden al bit del temporizador, en tanto que las operaciones con operandos en formato de palabra acceden al valor actual. Como muestra la figura 5-3, la operación Contacto normalmente abierto accede al bit del temporizador, en tanto que la operación Transferir palabra (MOV_W) accede al valor actual del temporizador. Para obtener más información acerca de las operaciones S7-200, consulte el capítulo 9 (operaciones SIMATIC) o el capítulo 10 (operaciones IEC 1131–3).

Formato: **T[número del temporizador]** **T24**

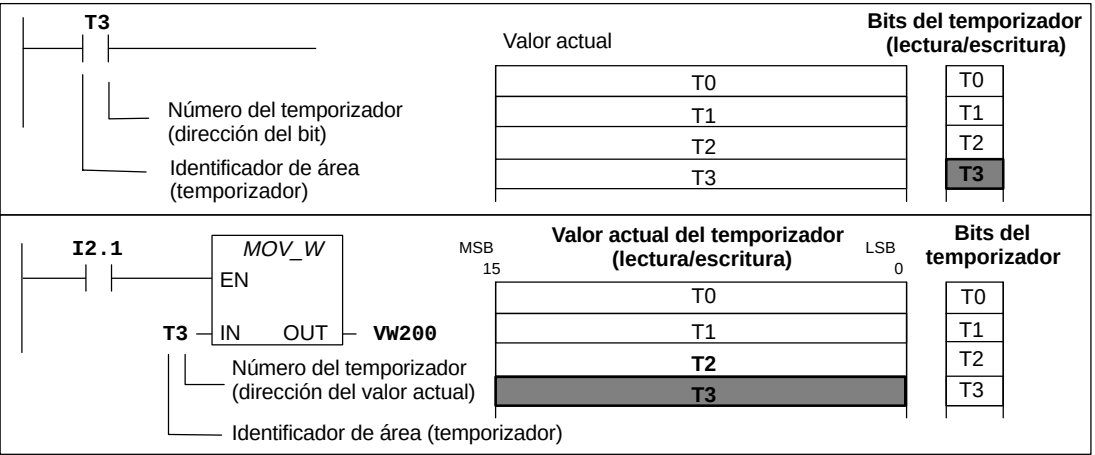


Figura 5-3 Acceso a los datos del temporizador SIMATIC

Direccionamiento de los contadores (C)

Los contadores de las CPUs S7-200 son elementos que cuentan los cambios de negativo a positivo en la(s) entrada(s) de conteo. Hay tres tipos de contadores: uno que cuenta sólo adelante, uno que cuenta atrás y uno que cuenta tanto adelante como atrás. Hay dos variables asociadas a los contadores:

- Valor actual: en este número entero de 16 bits con signo se deposita el valor de conteo acumulado.
- Bit del contador (bit C): este bit se activa o se desactiva como resultado de la comparación del valor actual con el valor de preselección. El valor de preselección se introduce como parte de la operación del contador.

A estas dos variables se accede utilizando la dirección del contador (C + número del contador). Dependiendo de la operación utilizada, se accede al bit del contador o al valor actual. Las operaciones con operandos en formato de bit acceden al bit del contador, en tanto que las operaciones con operandos en formato de palabra acceden al valor actual. Como muestra la figura 5-4, la operación Contacto normalmente abierto accede al bit del contador, en tanto que la operación Transferir palabra (MOV_W) accede al valor actual del contador. Para obtener más información acerca de las operaciones S7-200, consulte el capítulo 9 (operaciones SIMATIC) o el capítulo 10 (operaciones IEC 1131-3).

Formato:

C[número del contador]

C20

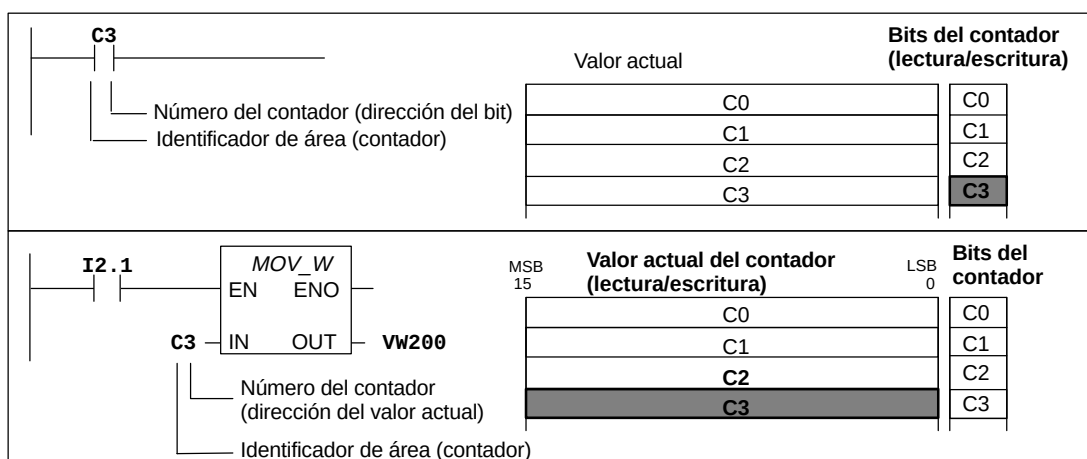


Figura 5-4 Acceso a los datos del contador SIMATIC

Direccionamiento de las entradas analógicas (AI)

La CPU S7-200 convierte valores reales analógicos (p.ej. temperatura, tensión, etc.) en valores digitales en formato de palabra (de 16 bits). A estos valores se accede con un identificador de área (AI), seguido del tamaño de los datos (W) y de la dirección del byte inicial. Puesto que las entradas analógicas son palabras que comienzan siempre en bytes pares (p.ej. 0, 2, 4, etc.), es preciso utilizar direcciones con bytes pares (p.ej. AIW0, AIW2, AIW4, etc.) para acceder a las mismas, como muestra la figura 5-5. Las entradas analógicas son valores de sólo lectura.

Formato: **AIW[dirección del byte inicial]** **AIW4**

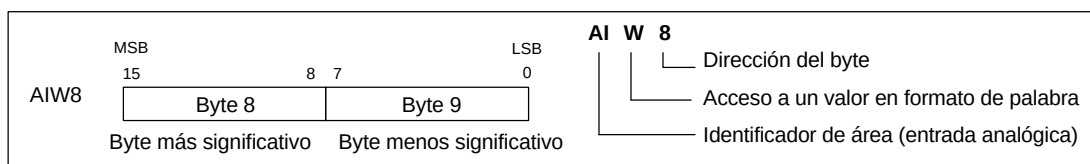


Figura 5-5 Acceso a una entrada analógica

Direccionamiento de las salidas analógicas (AQ)

La CPU S7-200 convierte valores digitales en formato de palabra (de 16 bits) en valores reales analógicos (p.ej. intensidad o tensión), proporcionales al valor digital. A estos valores se accede con un identificador de área (AQ), seguido del tamaño de los datos (W) y de la dirección del byte inicial. Puesto que las salidas analógicas son palabras que comienzan siempre en bytes pares (p.ej. 0, 2, 4, etc.), es preciso utilizar direcciones con bytes pares (p.ej. AQW0, AQW2, AQW4, etc.) para acceder a las mismas, como muestra la figura 5-6. Las salidas analógicas son valores de sólo escritura.

Formato: **AQW[dirección del byte inicial]** **AQW4**

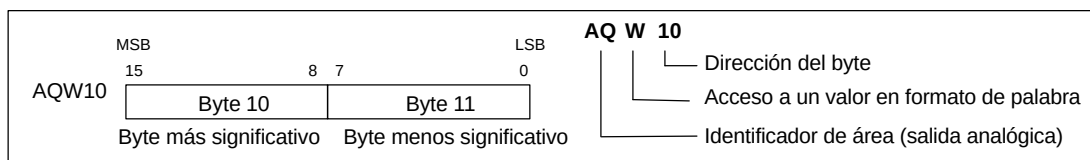


Figura 5-6 Acceso a una salida analógica

Direccionamiento de los acumuladores (AC)

Los acumuladores son elementos de lectura/escritura que se utilizan igual que una memoria. Los acumuladores se pueden usar p.ej. para transferir parámetros de y a subrutinas, así como para almacenar valores intermedios utilizados en cálculos. La CPU dispone de cuatro acumuladores de 32 bits (AC0, AC1, AC2 y AC3). A los acumuladores se puede acceder en formato de byte, palabra o palabra doble. Como muestra la figura 5-7, cuando se accede a un acumulador en formato de byte o de palabra se utilizan los 8 ó 16 bits menos significativos del valor almacenado en el acumulador. Cuando se accede a un acumulador en formato de palabra doble, se usan todos los 32 bits. La operación utilizada para el acceso al acumulador determina el tamaño de los datos a los que se accede.

Formato: **AC[número del acumulador]** **AC0**

Nota

Consulte el apartado 9.15 (Operaciones de comunicación SIMATIC) en el capítulo 9 para obtener más información acerca de cómo utilizar los acumuladores en las rutinas de interrupción.

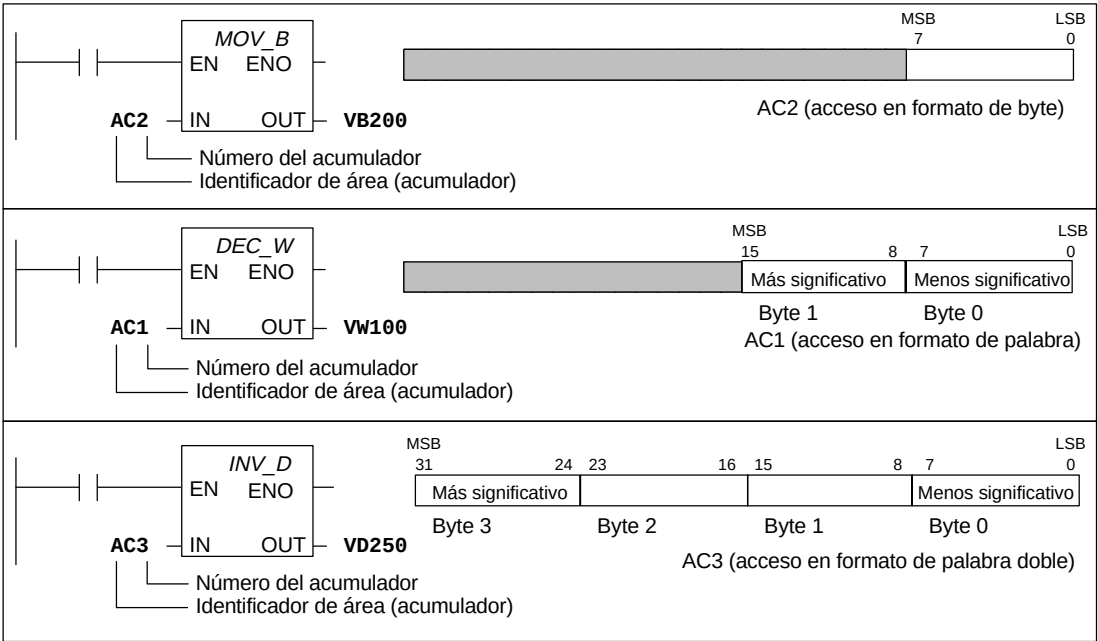


Figura 5-7 Acceso a los acumuladores

Direccionamiento de los contadores rápidos (HC)

Los contadores rápidos se han diseñado para contar eventos muy rápidos, independientemente del ciclo de la CPU. Tienen un valor de conteo de entero de 32 bits con signo (denominado también valor actual). Para acceder al valor de conteo del contador rápido, se indica la dirección del mismo (utilizando el identificador HC) y el número del contador (p.ej. HC0). El valor actual del contador rápido es de sólo lectura, pudiéndose acceder al mismo sólo en formato de palabra doble de 32 bits, como muestra la figura 5-8.

Formato: **HC[número del contador rápido]** **HC1**

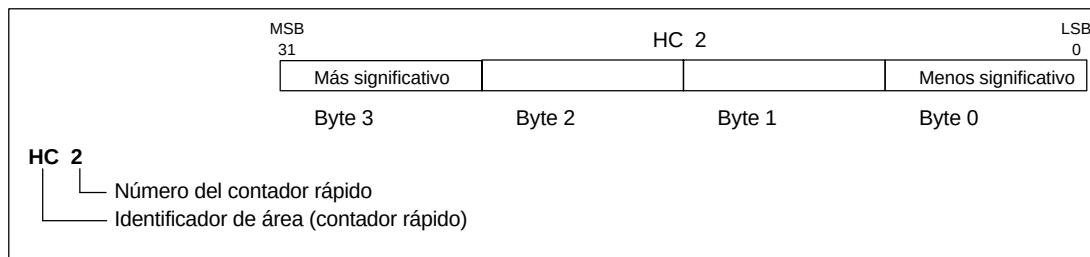


Figura 5-8 Acceso a los valores actuales del contador rápido

Utilizar valores de constantes

Las constantes se pueden utilizar en numerosas operaciones de la CPU S7-200. Pueden ser valores de bytes, palabras o palabras dobles. La CPU almacena todas las constantes como números binarios que se pueden representar en formato decimal, hexadecimal, ASCII o en coma flotante.

Formato decimal:	[valor decimal]
Formato hexadecimal:	16#[valor hexadecimal]
Formato ASCII:	'[texto ASCII]'
Formato real o en coma flotante:	ANSI/IEEE 754-1985
El formato binario es el siguiente:	2#1010_0101_1010_0101

La CPU S7-200 no permite indicar tipos de datos específicos ni comprobar datos (p.ej. indicar si la constante es un entero de 16 bits, un entero con signo o un entero de 32 bits). Por ejemplo, la operación Sumar puede utilizar el valor depositado en VW100 como entero con signo, en tanto que una operación de combinación con O-exclusiva puede emplear ese mismo valor de VW100 como valor binario sin signo.

A continuación se indican ejemplos de constantes en formato decimal, hexadecimal, ASCII y en coma flotante:

- Constante decimal: **20047**
- Constante hexadecimal: **16#4E4F**
- Constante ASCII: **'El texto aparece entre comillas sencillas.'**
- Formato real o en coma flotante: **+1.175495E-38 (positivo)**
-1.175495E-38 (negativo)
- Formato binario **2#1010_0101_1010_0101**

5.2 Direccionamiento indirecto (SIMATIC) de las áreas de memoria de la CPU

El direccionamiento indirecto utiliza un puntero para acceder a los datos de la memoria. La CPU S7-200 permite utilizar punteros para direccionar indirectamente las siguientes áreas de memoria: I, Q, V, M, S, T (sólo el valor actual) y C (sólo el valor actual). Los valores analógicos o de bits individuales no se pueden direccionar de forma indirecta.

Crear un puntero

Para acceder indirectamente a una dirección en la memoria es preciso crear primero un puntero que señale a esa dirección. Los punteros son valores de palabra doble que señalan a otra dirección en la memoria. Como punteros sólo se pueden utilizar direcciones de la memoria V y L, o bien los acumuladores (AC1, AC2 y AC3). Para crear un puntero se debe utilizar la operación Transferir palabra doble (MOVD) con objeto de transferir la dirección indirecta a la del puntero. El operando de entrada de la operación debe ir precedido de un carácter "&" para determinar que a la dirección indicada por el operando de salida (es decir, el puntero) se debe transferir la dirección y no su contenido.

Ejemplo:

```

MOVD    &VB100, VD204
MOVD    &MB4, AC2
MOVD    &C4, LD6

```

Utilizar un puntero para acceder a los datos

Introduciendo un asterisco (*) delante de un operando de una operación, se indica que el operando es un puntero. En el ejemplo que muestra la figura 5-9, *AC1 significa que AC1 es el puntero del valor de palabra indicado por la operación Transferir palabra (MOVW). En este ejemplo, los valores almacenados en V200 y V201 se transfieren al acumulador AC0.

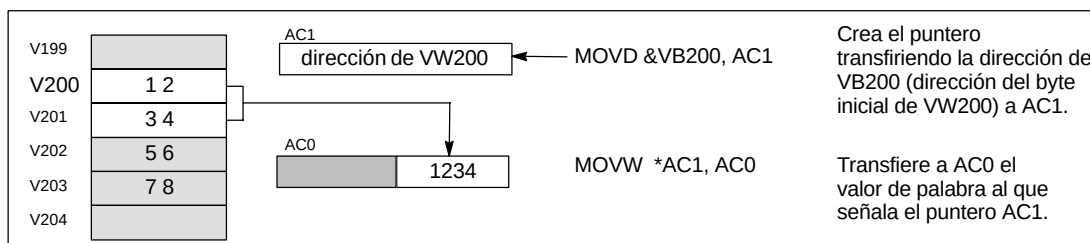


Figura 5-9 Utilizar un puntero para el direccionamiento indirecto

Modificar punteros

Los valores de los punteros se puede modificar. Puesto que los punteros son valores de 32 bits, para cambiarlos es preciso utilizar operaciones de palabra doble. Las operaciones aritméticas simples, tales como sumar o incrementar, se pueden utilizar para modificar los valores de los punteros. Recuerde que debe indicar el tamaño de los datos a los que desee acceder:

- Para acceder a un byte, sume o incremente el valor del puntero en 1.
- Para acceder a una palabra, o bien al valor actual de un temporizador o de un contador, sume o incremente el valor del puntero en 2.
- Para acceder a una palabra doble, sume o incremente el valor del puntero en 4.

La figura 5-10 muestra un ejemplo de cómo crear un puntero de direccionamiento indirecto y de cómo acceder indirectamente a los datos e incrementar el puntero.

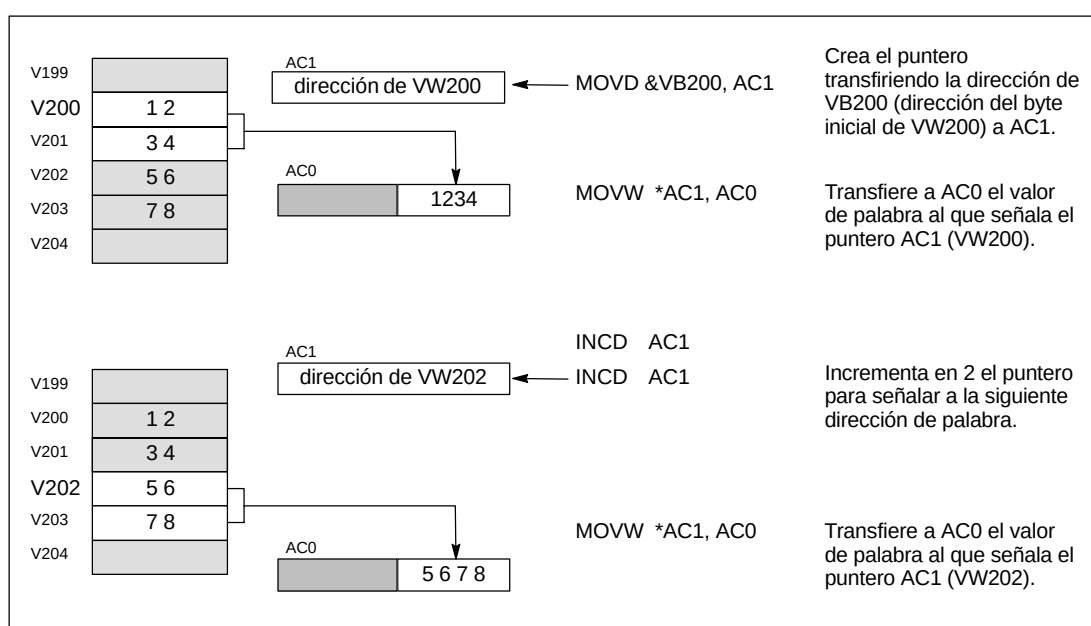


Figura 5-10 Modificar un puntero al acceder a un valor de palabra

5.3 Respaldo datos en la CPU S7-200

La CPU S7-200 ofrece diversos métodos para garantizar que el programa, los datos del mismo y los datos de configuración de la CPU se almacenen de forma segura (v. fig. 5-11).

- La CPU dispone de una EEPROM no volátil para almacenar todo el programa, así como las áreas de datos de usuario y la configuración de la CPU.
- La CPU dispone de un condensador de alto rendimiento que conserva todo el contenido de la memoria RAM después de un corte de alimentación. Según el tipo de CPU, el condensador puede respaldar la memoria durante varios días.
- La CPU soporta un cartucho de pila opcional que prolonga el tiempo durante el que se puede respaldar la memoria RAM después de un corte de alimentación. El cartucho de pila se activa sólo cuando se descarga el condensador de alto rendimiento.

En el presente apartado se describe el almacenamiento no volátil y el respaldo de los datos en la RAM en determinadas circunstancias.

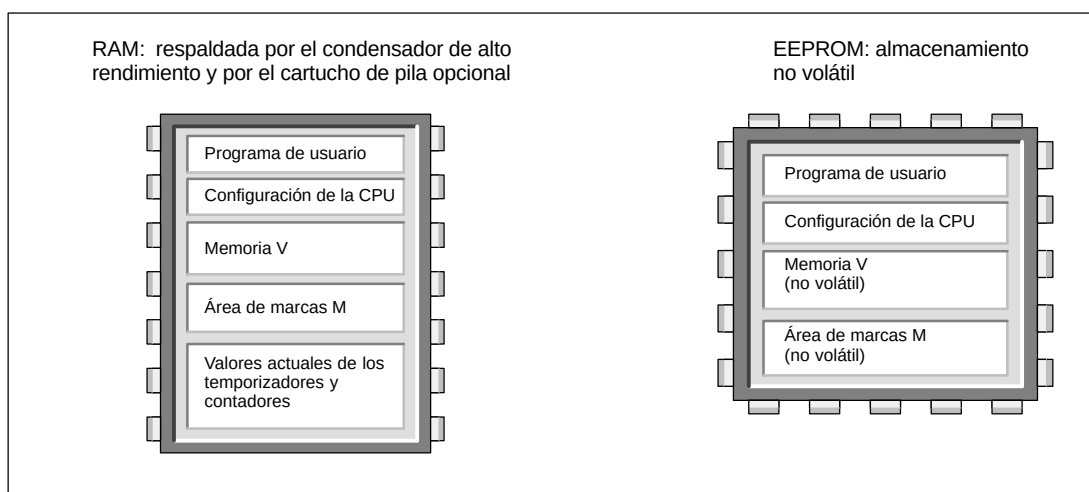


Figura 5-11 Áreas de memoria de la CPU S7-200

Cargar el proyecto en la CPU y en la PG

El proyecto comprende tres elementos: el programa de usuario, el bloque de datos (opcional) y la configuración de la CPU (opcional). Como muestra la figura 5-12, cargando el proyecto en la CPU se almacenan dichos elementos en la memoria RAM (de la CPU). La CPU también copia automáticamente el programa de usuario, el bloque de datos (DB1) y la configuración de la CPU en la EEPROM no volátil para que se almacenen allí.

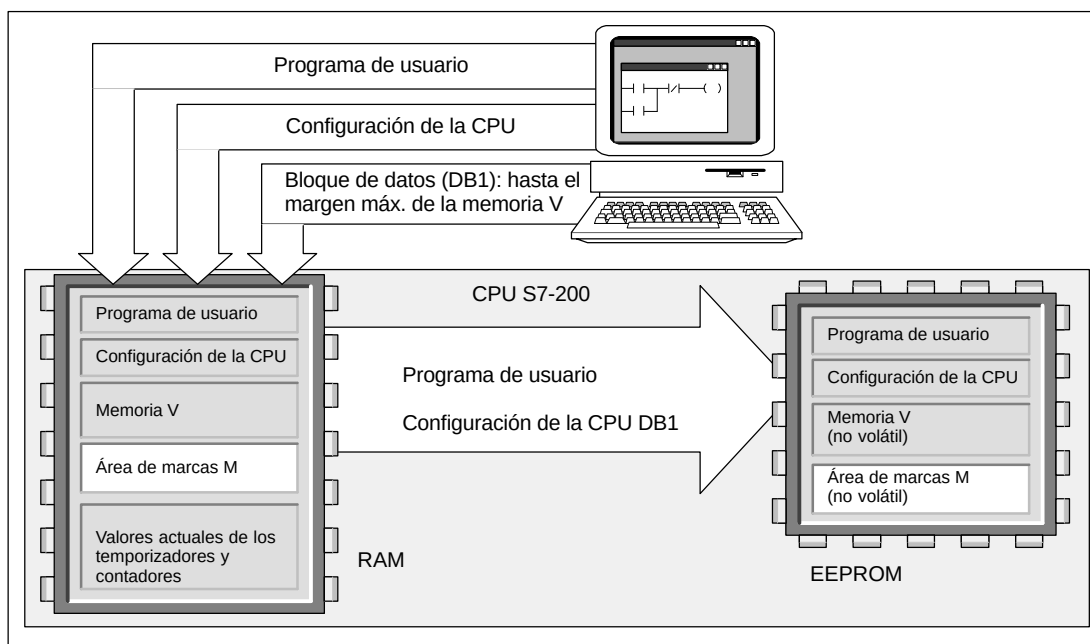


Figura 5-12 Cargar los elementos del proyecto en la CPU

Como muestra la figura 5-13, cuando un proyecto se carga en el PC (o la PG) desde la CPU, la configuración de la CPU se carga en el PC (o la PG) desde la RAM. El programa de usuario y la memoria V no volátil se cargan en la PG/el PC desde la EEPROM.

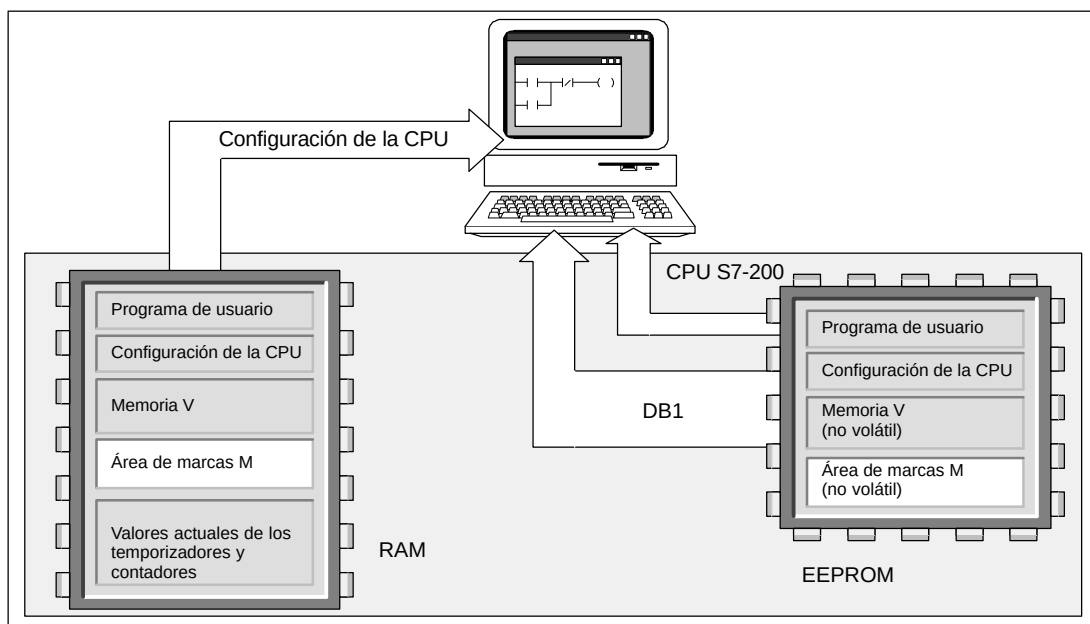


Figura 5-13 Cargar los elementos del proyecto en el PC (o la PG)

Almacenar automáticamente los datos del área de marcas (M) en caso de un corte de alimentación

Si se define que los primeros 14 bytes del área de marcas (MB0 a MB13) sean remanentes, se copiarán automáticamente en la EEPROM cuando se produzca un corte de alimentación de la CPU. Como muestra la figura 5-14, la CPU transfiere dichas áreas remanentes del área de marcas a la EEPROM. En STEP 7-Micro/WIN 32, el ajuste estándar es "off".

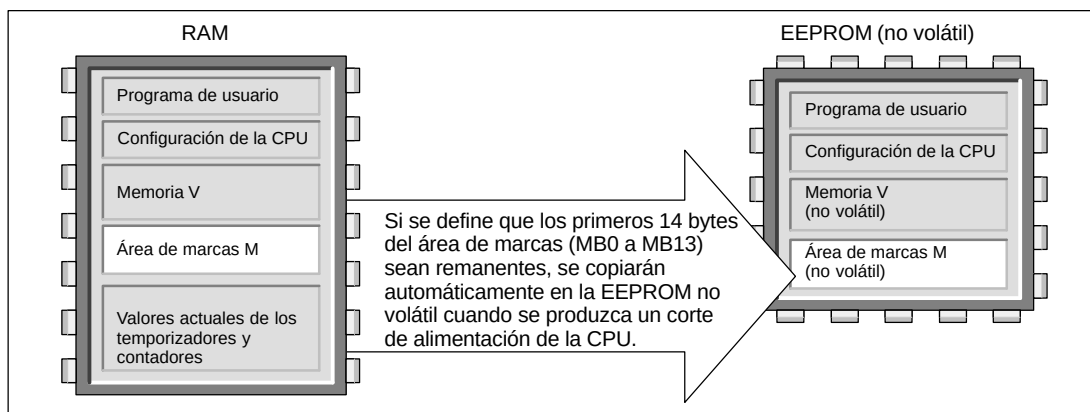


Figura 5-14 Almacenar partes del área de marcas (M) en la EEPROM en caso de un corte de alimentación

Respaldar la memoria al conectar la alimentación

Cuando se conecta la alimentación, la CPU restablece el programa de usuario y la configuración (de la CPU) que se han depositado en la memoria EEPROM (v. fig. 5-15).

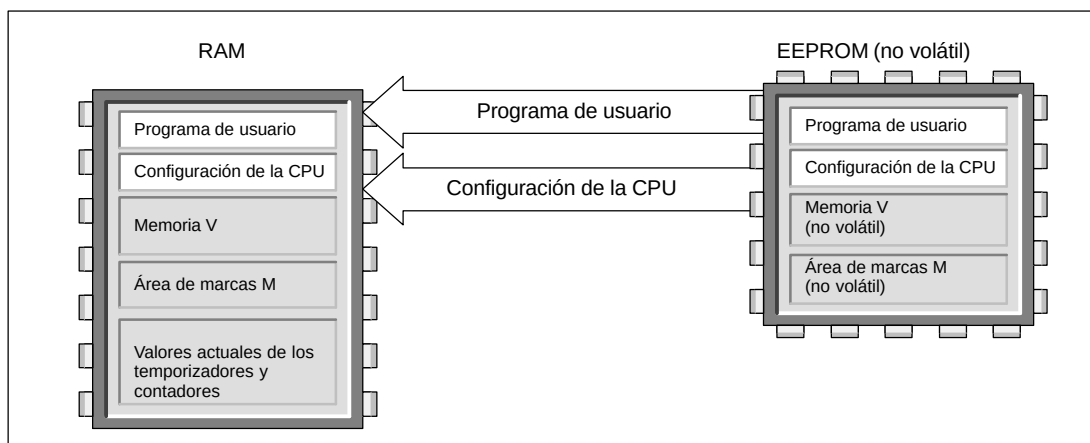


Figura 5-15 Restablecer el programa de usuario y la configuración de la CPU al conectar la alimentación

Cuando se conecta la alimentación, la CPU comprueba la memoria RAM para verificar si el condensador de alto rendimiento ha respaldado los datos almacenados en la RAM. En caso afirmativo, no se modificarán las áreas remanentes de la misma. Como muestra la figura 5-16, las áreas no remanentes de la memoria V se restablecen conforme a la correspondiente área no volátil de la memoria V contenida en la EEPROM.

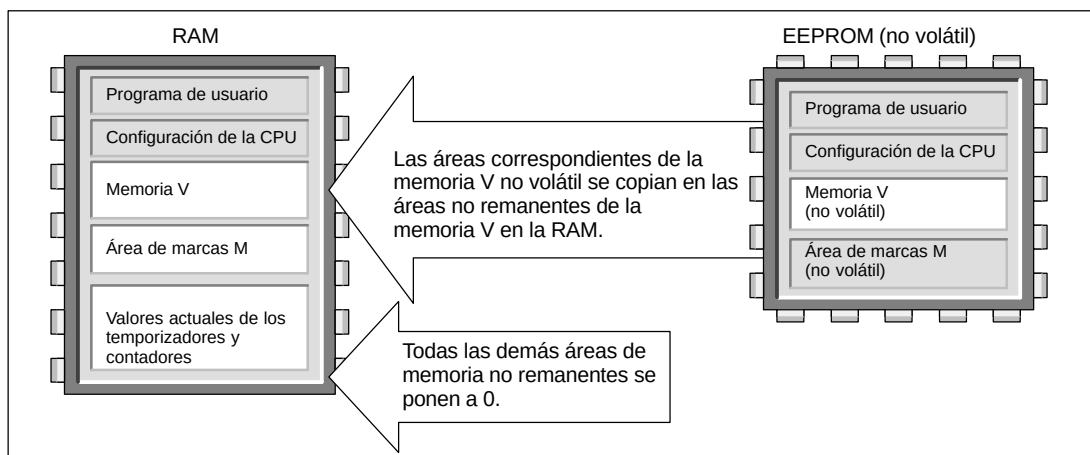


Figura 5-16 Restablecer los datos del programa al conectarse la alimentación (habiéndose respaldado los datos en la RAM)

Si el contenido de la RAM no se ha respaldado (p.ej. en el caso de un corte de alimentación prolongado), la CPU borra la memoria RAM (tanto las áreas remanentes como las no remanentes) y activa la marca Datos remanentes perdidos (SM0.2) en el primer ciclo que le sigue a la puesta en marcha. Como muestra la figura 5-17, los datos almacenados en la EEPROM no volátil se copian entonces en la memoria RAM.

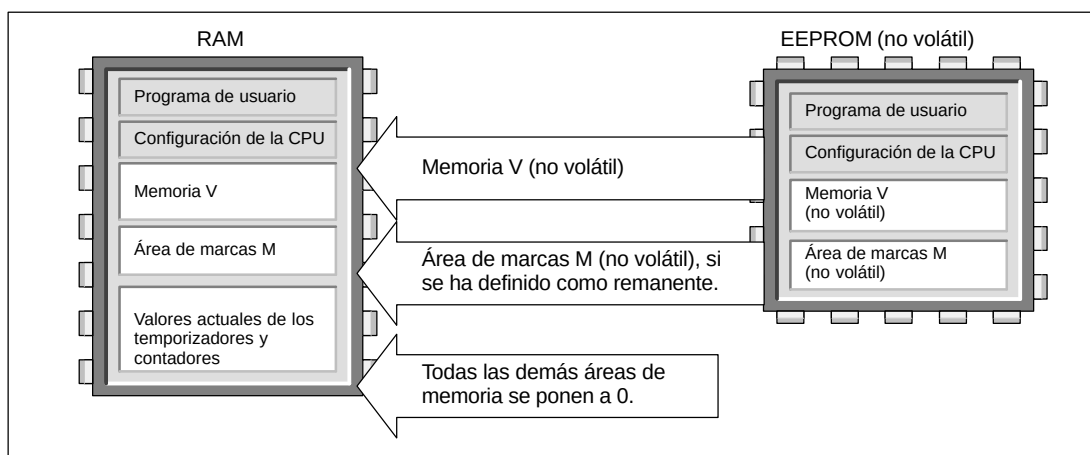


Figura 5-17 Restablecer los datos del programa al conectarse la alimentación (sin haberse respaldado los datos en la RAM)

Definir las áreas remanentes

El usuario puede definir hasta seis áreas remanentes para elegir las áreas de memoria que se deberán respaldar cuando se interrumpa la alimentación (v. figura 5-18). Es posible determinar que sean remanentes los márgenes de direcciones en las áreas de memoria V, M, C y T. En el caso de los temporizadores, sólo es posible respaldar los de retardo a la conexión memorizado (TONR). En STEP 7-Micro/WIN 32, el ajuste estándar es que los primeros 14 bytes de la memoria M no sean remanentes. Este ajuste desactiva la operación de almacenamiento que, por lo general, se efectúa cuando se desconecta la CPU.

Nota

Sólo se pueden respaldar los valores actuales de los temporizadores y contadores. Los bits de los temporizadores y de los contadores no son remanentes.

Para definir las áreas remanentes, elija el comando de menú **Ver > Bloque de sistema** y haga clic en la ficha "Áreas remanentes". La figura 5-18 muestra el cuadro de diálogo donde se definen dichas áreas. Para visualizar las áreas remanentes estándar de la CPU, haga clic en el botón **Estándar**.

	Área de datos	Offset	Nº de elementos	
Área 0:	VB	0	5120	Borrar
Área 1:	VB	0	0	Borrar
Área 2:	T	0	32	Borrar
Área 3:	T	64	32	Borrar
Área 4:	C	0	256	Borrar
Área 5:	MB	14	18	Borrar

Para que los parámetros de configuración tengan efecto, es preciso cargarlos previamente en la CPU.

No todos los tipos de CPUs asisten todas las opciones del bloque de sistema. Pulse F1 para visualizar las opciones que soportan las distintas CPUs.

Aceptar Cancelar

Figura 5-18 Configurar las áreas remanentes de la memoria de la CPU

5.4 Utilizar el programa para guardar datos en la memoria no volátil

Cualquier valor (byte, palabra o palabra doble) almacenado en la memoria de variables (memoria V) se puede guardar en la memoria EEPROM. Esta función se puede utilizar para almacenar un valor en cualquier dirección de la memoria V no volátil.

Por lo general, la operación de guardar en EEPROM prolonga el tiempo de ciclo en hasta 5 ms. Si en esta operación se escribe un valor en el área no volátil de la memoria V de la EEPROM no volátil, se sobrescribirá el valor anterior de dicha dirección.

Nota

La operación de guardar en EEPROM no actualiza los datos contenidos en el cartucho de memoria.

Copiar la memoria de variables en la EEPROM

El byte de marcas 31 (SMB31) y la palabra de marcas 32 (SMW32) indican a la CPU que copie un valor de la memoria V en el área no volátil de la memoria V de la EEPROM. La figura 5-19 muestra el formato de SMB31 y SMW32. Para programar la CPU con objeto de guardar o escribir un valor determinado en la memoria V, siga los siguientes pasos:

1. Cargue la dirección de la memoria V del valor a almacenar en SMW32.
2. Cargue el tamaño de los datos en SM31.0 y SM31.1 (v. fig. 5-19).
3. Active la marca SM31.7.

Al final de cada ciclo, la CPU comprueba el estado de SM31.7. Si SM31.7 está activada (puesta a 1), el valor indicado se guardará en la EEPROM. La operación se finalizará cuando la CPU desactive SM31.7. No cambie el valor en la memoria V antes de finalizar la operación de guardar.

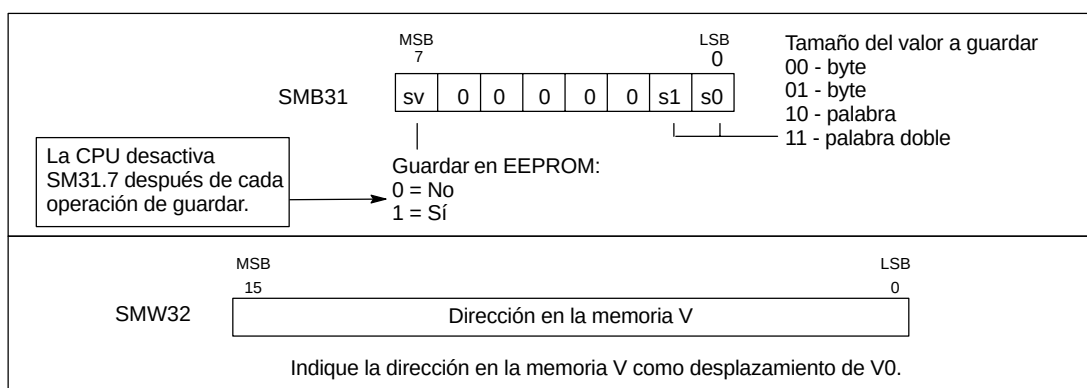


Figura 5-19 Formato de SMB31 y SMW32

Limitar el número de las operaciones de guardar en EEPROM

Puesto que el número de operaciones de guardar en EEPROM es limitado (mín. 100.000, típ. 1.000.000), sólo se deberán almacenar los valores realmente necesarios. De lo contrario, es posible que se sobrecargue la EEPROM y que falle la CPU. Generalmente, las operaciones de guardar se ejecutan sólo cuando se presentan determinados eventos, lo cual no suele ocurrir con frecuencia.

Por ejemplo, si el tiempo de ciclo del S7-200 es de 50 ms y un valor se almacena una vez por ciclo, la EEPROM se llenaría al cabo de 5.000 segundos, es decir, en menos de una hora y media. En cambio, si dicho valor se almacena cada hora, la EEPROM podría utilizarse 11 años como mínimo.

5.5 Guardar el programa en un cartucho de memoria

Las CPUs asisten un cartucho de memoria opcional que permite almacenar el programa en una EEPROM portátil. La CPU guarda los siguientes datos en el cartucho de memoria:

- Programa de usuario
- Datos almacenados en la memoria de variables no volátil de la EEPROM
- Configuración de la CPU (bloque de sistema)

Para obtener más información sobre el cartucho de memoria, consulte el Anexo A.

Copiar en el cartucho de memoria

El programa se puede copiar en el cartucho de memoria desde la RAM sólo si se ha arrancado la CPU, si ésta se encuentra en modo STOP y si dicho cartucho está insertado.

Cuidado

Las descargas electroestáticas pueden deteriorar el cartucho de memoria o su receptáculo en la CPU.

Cuando utilice el cartucho de memoria, deberá estar en contacto con una superficie conductiva puesta a tierra y/o llevar puesta una pulsera puesta a tierra. Guarde el cartucho en una caja conductiva.

El cartucho de memoria se puede enchufar o extraer estando conectada la alimentación de la CPU. Para enchufarlo, retire la tapa de plástico de la CPU e inserte el cartucho en ésta última. (El cartucho de memoria se ha diseñado de forma que sólo se pueda insertar en un sólo sentido en el receptáculo). Una vez insertado el cartucho, copie el programa como se indica a continuación.

1. Cambie la CPU a modo STOP.
2. Cargue el programa en la CPU si no lo ha hecho todavía.
3. Elija el comando de menú **CPU > Cartucho de memoria** para copiar el programa en el cartucho de memoria. La figura 5-20 muestra los componentes de la memoria de la CPU que se almacenan en dicho cartucho.
4. Extraiga el cartucho de memoria del receptáculo (opcional).

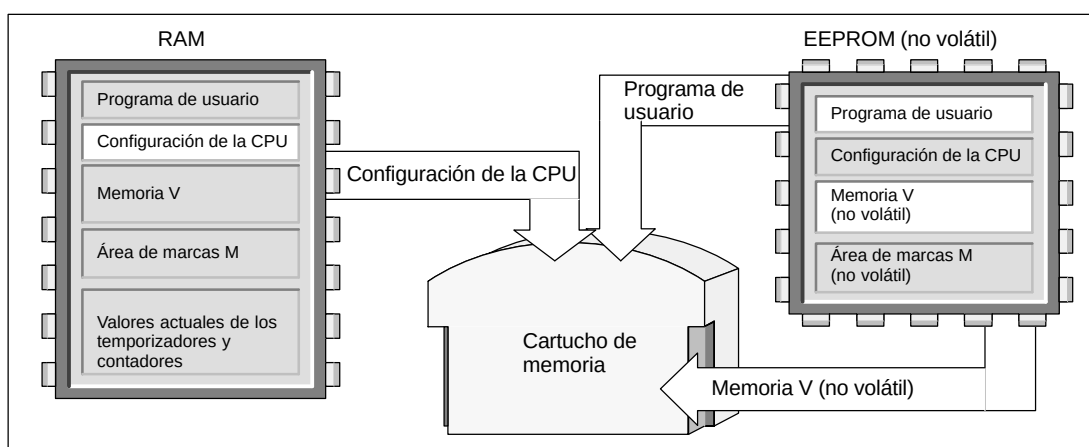


Figura 5-20 Copiar la memoria de la CPU en el cartucho de memoria

Restablecer el programa y la memoria mediante un cartucho de memoria

Para transferir el programa de un cartucho de memoria en la CPU, es preciso desconectar la alimentación de la CPU y conectarla de nuevo con el cartucho insertado. Como muestra la figura 5-21, la CPU ejecuta las siguientes tareas después del arranque (si está insertado el cartucho de memoria):

- Se borra la memoria RAM.
- El contenido del cartucho de memoria se copia en la memoria RAM.
- El programa de usuario, la configuración de la CPU y la memoria V se copian en la EEPROM no volátil.

Nota

Si se conecta la alimentación de la CPU estando insertado un cartucho de memoria vacío o programado con un modelo diferente de CPU, se puede producir un error. Los cartuchos de memoria que se hayan programado en una CPU de un modelo inferior se pueden leer en una CPU de un modelo superior, mas no al contrario. Por ejemplo, los cartuchos de memoria programados en una CPU 221 ó 222 se pueden leer también en una CPU 224. No obstante, los cartuchos de memoria programados en una CPU 224 serán rechazados por una CPU 221 ó 222.

Retire el cartucho de memoria y arranque la CPU de nuevo. Si lo desea, podrá insertar y volver a programar el cartucho.

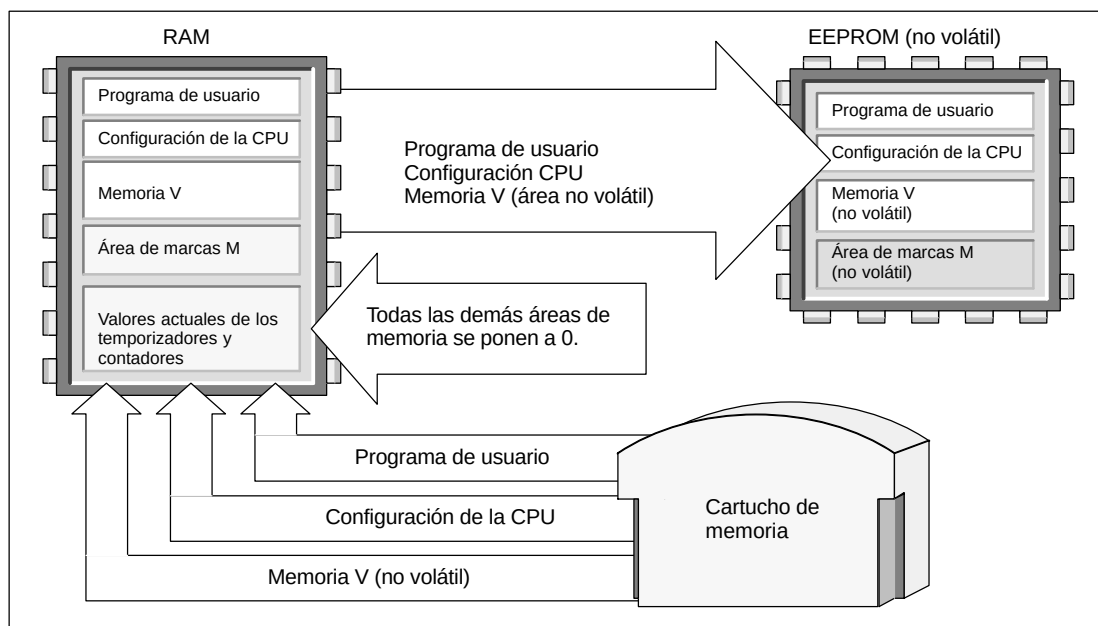


Figura 5-21 Restablecer la memoria durante el arranque (estando insertado un cartucho de memoria)

Configurar las entradas y salidas

El sistema se controla mediante entradas y salidas (E/S). Las entradas vigilan las señales de los dispositivos de campo (p.ej. sensores e interruptores), mientras que las salidas supervisan las bombas, motores u otros aparatos del proceso. Se dispone de entradas y salidas integradas (en la CPU), así como de E/S adicionales (en los módulos de ampliación). Las CPUs S7-200 incorporan además entradas y salidas rápidas.

Índice del capítulo

Apartado	Descripción	Página
6.1	Entradas y salidas integradas y adicionales	6-2
6.2	Utilizar filtros de entrada para suprimir interferencias	6-4
6.3	Capturar impulsos	6-5
6.4	Configurar los estados de señal de las salidas	6-8
6.5	Filtrar entradas analógicas	6-9
6.6	Entradas y salidas rápidas	6-10
6.7	Potenciómetros analógicos	6-13

6.1 Entradas y salidas integradas y adicionales

El sistema se controla mediante entradas y salidas (E/S). Las entradas vigilan las señales de los dispositivos de campo (p.ej. sensores e interruptores), mientras que las salidas supervisan las bombas, motores u otros aparatos del proceso. Se dispone de entradas y salidas integradas (en la CPU), así como de E/S adicionales (en los módulos de ampliación).

- Las CPUs S7-200 disponen de un número determinado de entradas y salidas digitales. Para obtener más información acerca de las E/S integradas en su CPU, consulte las hojas de datos técnicos que se incluyen en el Anexo A.
- Las CPUs 222, 224 y 226 asisten módulos de ampliación con entradas y salidas tanto digitales como analógicas. Para obtener más información acerca de los diversos módulos de ampliación, consulte las hojas de datos técnicos que se incluyen en el Anexo A.

Direccionar las E/S integradas y adicionales

Las entradas y salidas integradas en la unidad central de procesamiento (CPU) tienen direcciones fijas. Para añadir a la CPU entradas y salidas adicionales, se pueden conectar módulos de ampliación a la derecha de la CPU, formando una cadena de E/S. Las direcciones de las E/S de cada módulo vienen determinadas por el tipo de E/S y por la posición del módulo en la cadena, con respecto al anterior módulo de entradas o de salidas del mismo tipo. Por ejemplo, un módulo de salidas no afecta a las direcciones de un módulo de entradas y viceversa. Igualmente, los módulos analógicos no afectan al direccionamiento de los módulos digitales y viceversa.

Los módulos de ampliación digitales reservan siempre un espacio de la imagen del proceso en incrementos de ocho bits (un byte). Si un módulo no dispone de un punto físico para cada bit de cada byte reservado, se pierden estos bits no utilizados y no se pueden asignar a los módulos siguientes en la cadena de E/S. En cuanto a los módulos de entradas, los bits no utilizados en los bytes reservados se ponen a cero cada vez que se actualizan las entradas.

Las direcciones de los módulos de ampliación analógicos se asignan siempre en incrementos de dos puntos. Si un módulo no ofrece E/S físicas para cada uno de dichos puntos, se pierden los mismos y no se pueden asignar a los módulos siguientes en la cadena de E/S.

Ejemplos de E/S integradas y adicionales

Las figuras 6-1 y 6-2 muestran ejemplos de cómo las diferentes configuraciones del hardware afectan a la numeración de las entradas y salidas. Tenga en cuenta que algunas configuraciones tienen espacios entre las direcciones que no se pueden utilizar en el programa.

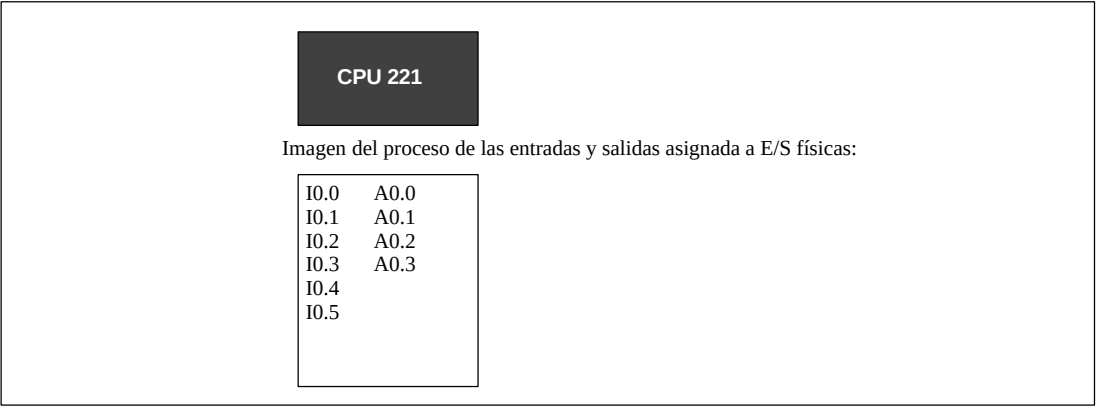


Figura 6-1 Ejemplos de numeración de E/S para una CPU 221

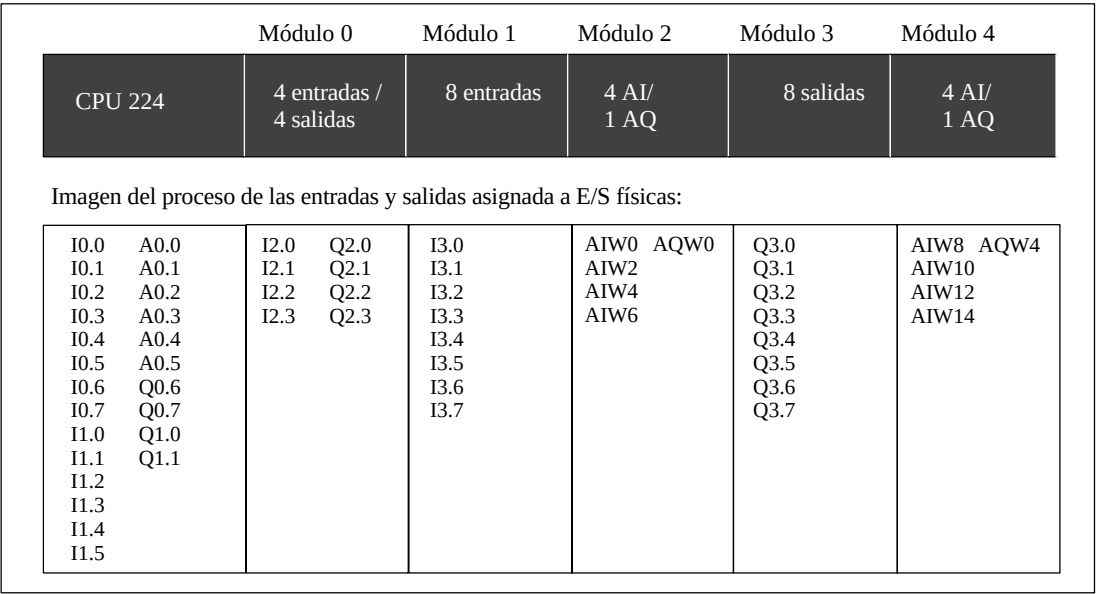


Figura 6-2 Ejemplos de numeración de E/S para una CPU 224

6.2 Utilizar filtros de entrada para suprimir interferencias

Las CPU S7-200 permiten seleccionar un filtro de entrada que define un tiempo de retardo (comprendido entre 0,2 ms y 12,8 ms) para algunas o bien para todas las entradas digitales integradas. (Para obtener más información acerca de su CPU, consulte el Anexo A). Como muestra la figura 6-3, cada indicación del tiempo de retardo se aplica a grupos de cuatro entradas. Dicho retardo permite filtrar en el cableado de entrada las interferencias que pudieran causar cambios accidentales de los estados de las salidas.

El filtro de entrada forma parte de los datos de configuración de la CPU que se cargan y se almacenan en la memoria de la misma. Para configurar los tiempos de retardo del filtro de entrada, elija el comando de menú **Ver > Bloque de sistema** y haga clic en la ficha "Filtros de entrada".

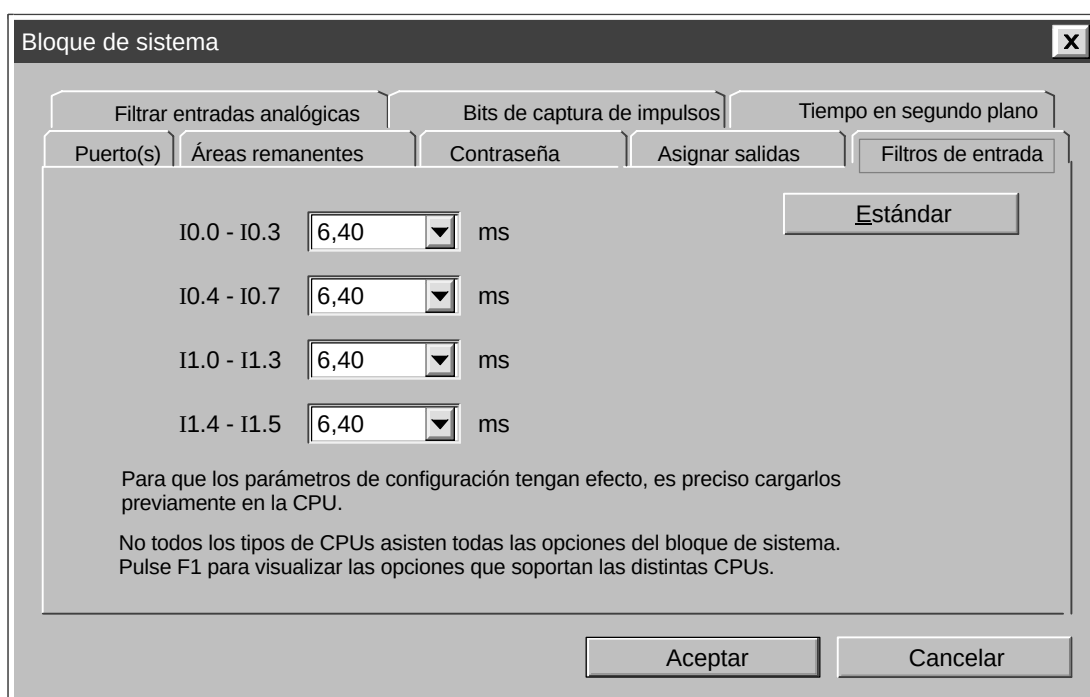


Figura 6-3 Configurar el filtro de entrada para suprimir interferencias

6.3 Capturar impulsos

Las CPUs S7-200 ofrecen una función de captura de impulsos para algunas o todas las entradas digitales integradas. (Para obtener más información acerca de su CPU, consulte el Anexo A). Esta función permite capturar impulsos altos o bajos de tan corta duración que no se registrarían en todos los casos, cuando la CPU lee las entradas digitales al comienzo del ciclo.

La función de captura de impulsos se puede habilitar individualmente para cada una de las entradas digitales integradas. Si se ha habilitado la captura de impulsos en una entrada y se produce un cambio de estado de la misma, dicho cambio se señalará, conservándose hasta que la entrada se actualice al comienzo del siguiente ciclo. De esta forma, un impulso de breve duración se capturará y se conservará hasta que la CPU lea las entradas, garantizando así que el impulso no pase desapercibido. La figura 6-4 muestra el funcionamiento básico de la CPU con y sin captura de impulsos.

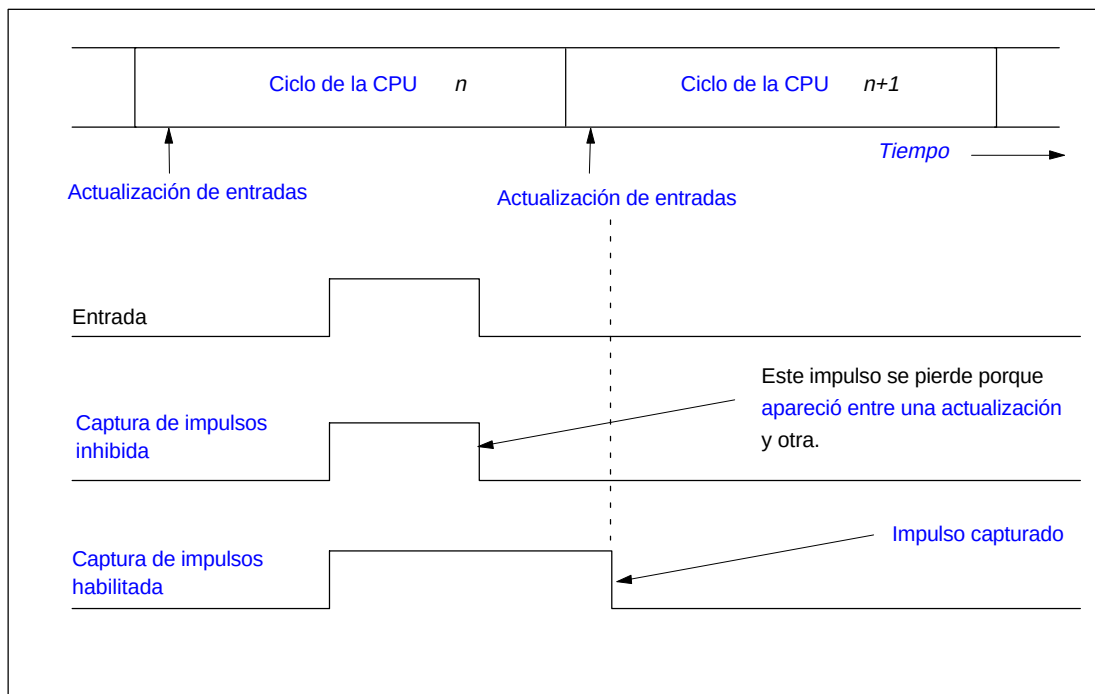


Figura 6-4 Funcionamiento de la CPU con y sin captura de impulsos

Cuando se utilice la función de captura de impulsos, el tiempo del filtro de entrada se deberá ajustar de manera que dicho filtro no elimine el impulso. (La función de captura de impulsos se ejecuta en la entrada tras haber pasado ésta por el filtro).

La figura 6-5 muestra un esquema funcional de una entrada digital.

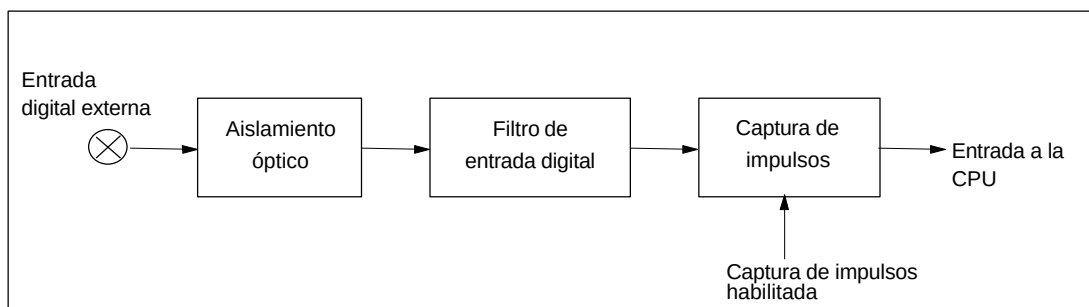


Figura 6-5 Esquema funcional de una entrada digital

La figura 6-6 muestra la reacción de un circuito de captura de impulsos a diversas condiciones de entrada. Si en un ciclo dado se presenta más de un impulso, se leerá únicamente el primer impulso. En caso de que se presenten varios impulsos en un ciclo, se recomienda utilizar la interrupción de E/S descrita en el apartado 9.15.

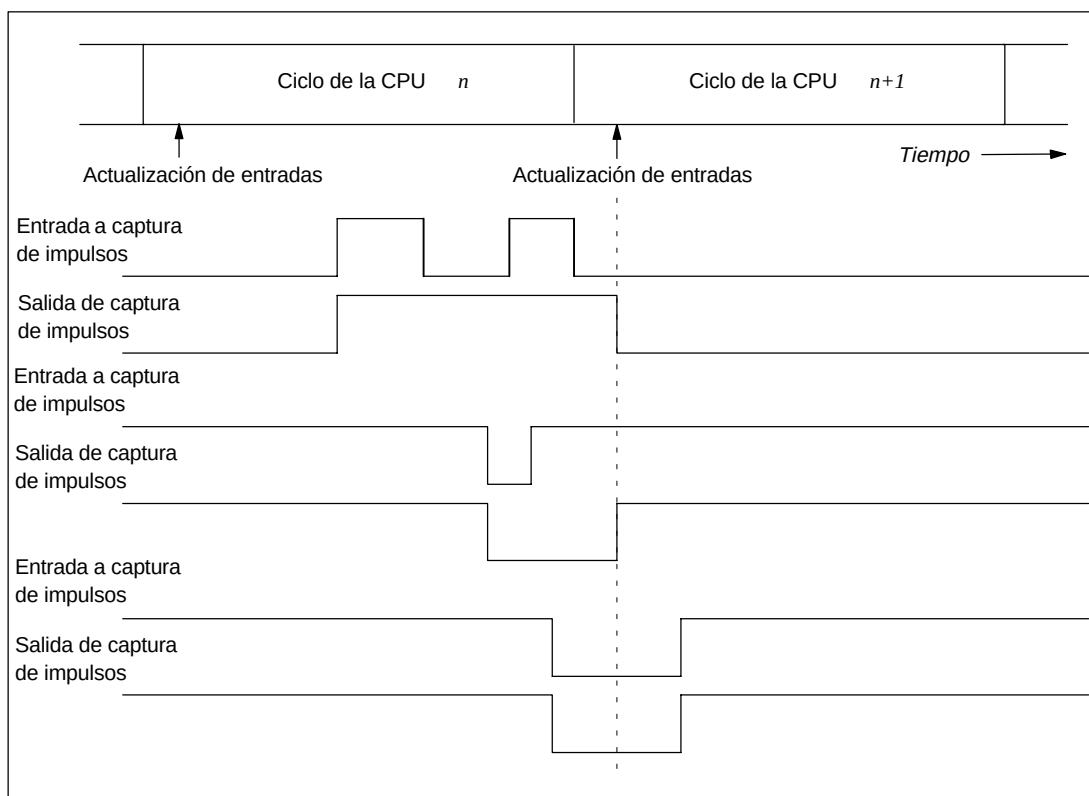


Figura 6-6 Ejemplo de captura de impulsos

Para acceder a la ventana donde se configura la captura de impulsos, elija el comando de menú **Ver > Bloque de sistema** y haga clic en la ficha "Bits de captura de impulsos". La figura 6-7 muestra la ventana para configurar la captura de impulsos. Para habilitar la función de captura de impulsos en una entrada determinada, haga clic en la casilla de verificación correspondiente.

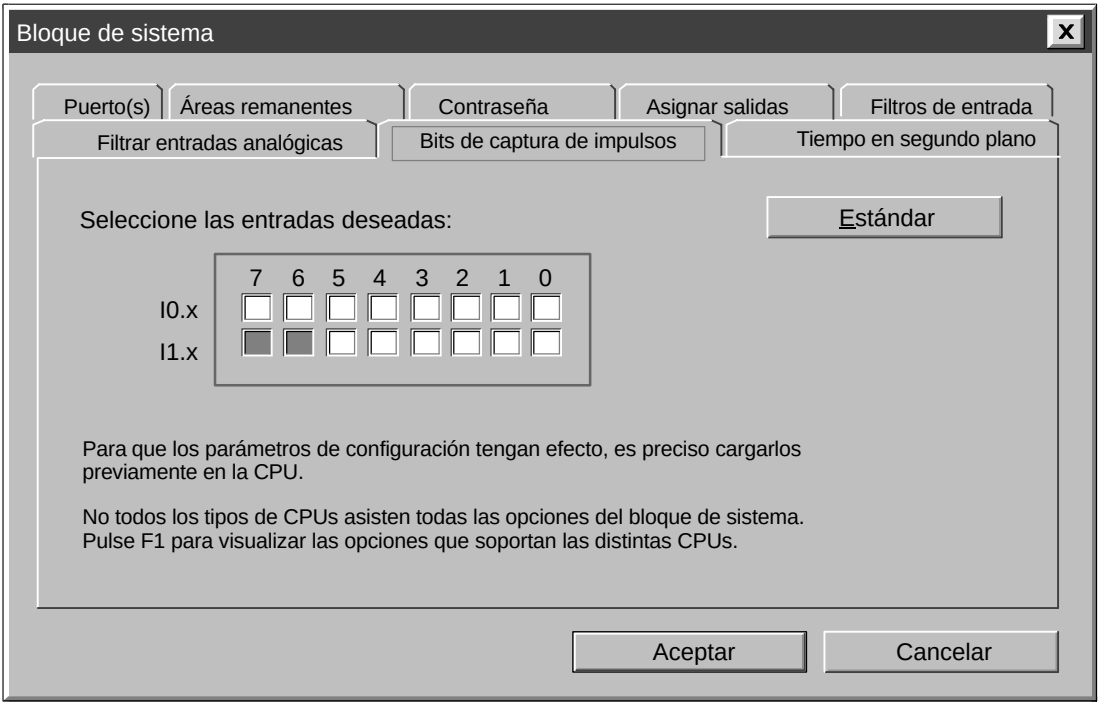


Figura 6-7 Ventana para configurar la captura de impulsos

6.4 Configurar los estados de señal de las salidas

La CPU S7-200 permite elegir si las salidas digitales deben adoptar valores conocidos cuando cambie a modo STOP, o bien congelar las salidas en su último estado antes de dicho cambio.

La tabla de salidas forma parte de los datos de configuración de la CPU que se cargan y se almacenan en la memoria de la misma.

La configuración de estos valores es aplicable sólo a las salidas digitales. Los valores de las salidas analógicas se congelan cuando se produce un cambio a modo STOP. La CPU no actualiza las entradas o salidas analógicas como una función del sistema, ni dispone tampoco de una imagen del proceso para las mismas.

Para acceder al cuadro de diálogo donde se configuran los estados de señal de las salidas, elija el comando de menú **Ver > Bloque de sistema** y haga clic en la ficha "Asignar salidas" (v. fig. 6-8). Hay dos opciones para configurar las salidas:

- Si desea que las salidas se congelen en su último estado, elija la opción "Congelar salidas" y haga clic en "Aceptar".
- Si desea copiar los valores de la tabla en las salidas, introduzca los correspondientes valores. Haga clic en la casilla de verificación correspondiente a cada salida que desee activar (poner a 1) cuando se produzca un cambio de RUN a STOP. Haga clic en "Aceptar" para guardar sus ajustes.

En dicha tabla, todas las salidas están desactivadas (puestas a 0) por defecto.

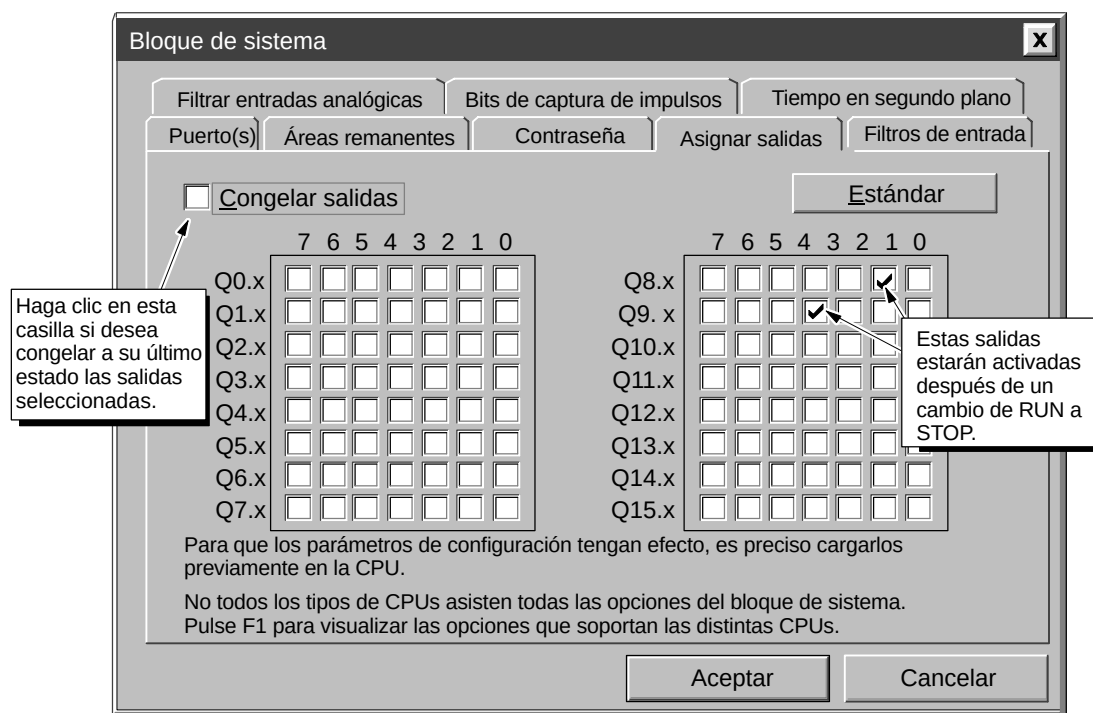


Figura 6-8 Configurar el estado de las salidas

6.5 Filtrar entradas analógicas

En las CPUs 222, 224 y 226, cada una de las entradas analógicas se puede filtrar utilizando el software. El valor filtrado es el valor promedio de la suma de un número estándar de muestreos de la entrada analógica. Los datos de filtración indicados (número de impulsos y tiempo muerto) se aplica a todas las entradas analógicas para las que se habilite dicha función.

El filtro dispone de una función de respuesta rápida para que los cambios considerables se puedan reflejar rápidamente en el valor de filtración. El filtro cambia al último valor de la entrada analógica cuando ésta exceda una determinada diferencia del valor promedio. Dicha diferencia se denomina tiempo muerto, indicándose en contajes del valor digital de la entrada analógica.

Nota

La filtración de entradas analógicas se deberá desactivar en módulos que transfieran informaciones digitales o indicaciones de alarma en las palabras analógicas. Desactive la filtración de entradas analógicas de los módulos RTD, Termopar y AS-Interface Master en el cuadro de diálogo que muestra la figura 6-9.

Para acceder al filtro de entradas analógicas, elija el comando de menú **Ver > Bloque de sistema** y haga clic en la ficha "Filtrar entradas analógicas". Elija las entradas analógicas que desee filtrar y haga clic en "Aceptar" (v. fig. 6-9). La configuración estándar de STEP 7-Micro/WIN 32 está habilitada para que se filtren todas las entradas analógicas.

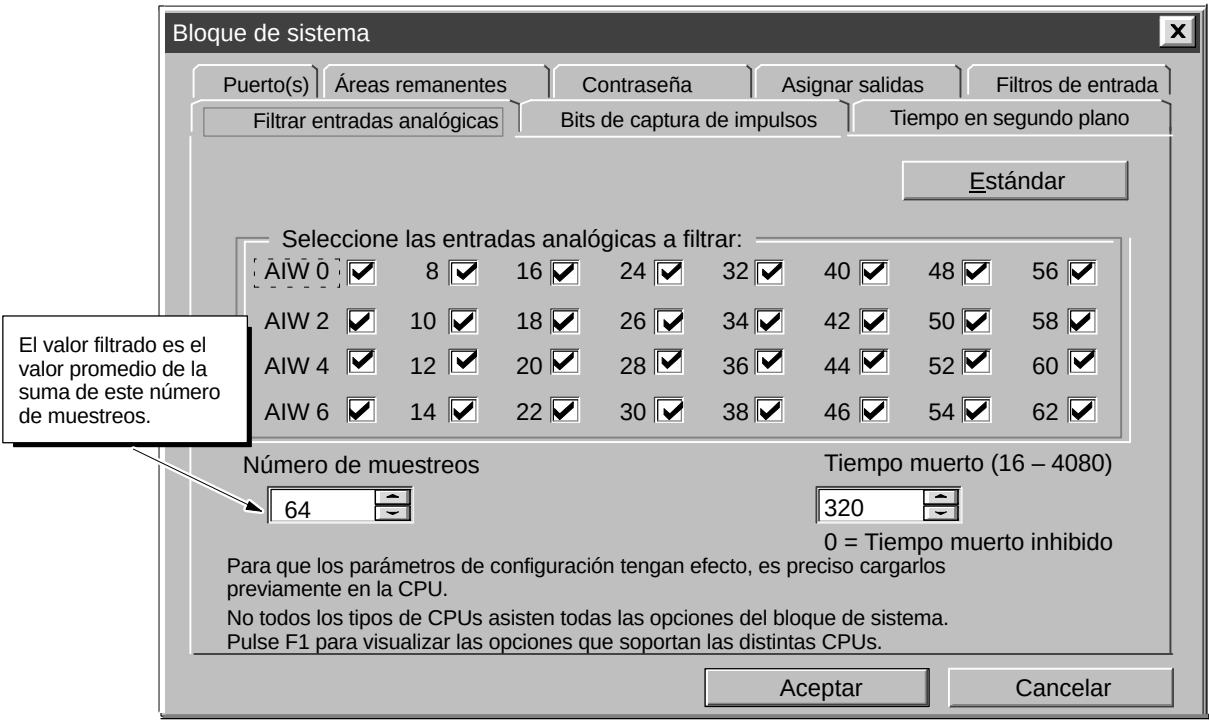


Figura 6-9 Filtrar entradas analógicas

6.6 Entradas y salidas rápidas

Las CPUs S7-200 disponen de entradas y salidas para controlar los eventos rápidos. Para obtener más información acerca de las entradas y salidas rápidas de cada modelo de CPU, consulte las hojas de datos técnicos que se incluyen en el Anexo A.

Contadores rápidos

Las CPUs S7-200 disponen de contadores rápidos integrados que cuentan eventos externos a velocidades de hasta 20 KHz sin influir en el funcionamiento de la CPU. A continuación se describen dichos contadores rápidos:

- HSC0 y HSC4 son contadores versátiles que se pueden configurar para uno de ocho modos de operación diferentes, incluyendo entradas de reloj de fase simple y de dos fases.
- HSC1 y HSC2 son contadores versátiles que se pueden configurar para uno de doce modos de operación diferentes, incluyendo entradas de reloj de fase simple y de dos fases.
- HSC3 y HSC5 son contadores sencillos que tienen sólo un modo de operación (sólo entradas de reloj de fase simple).

La tabla 6-1 muestra los modos de operación asistidos por los contadores rápidos HSC0, HSC3, HSC4 y HSC5. Todas las CPUs S7-200 asisten dichos contadores rápidos.

Tabla 6-1 Contadores rápidos HSC0, HSC3, HSC4, HSC5

Modo	HSC0			HSC3	HSC4			HSC5
	I0.0	I0.1	I0.2	I0.1	I0.3	I0.4	I0.5	I0.4
0	Reloj			Reloj	Reloj			Reloj
1	Reloj		Puesta a 0		Reloj		Puesta a 0	
2								
3	Reloj	Sentido			Reloj	Sentido		
4	Reloj	Sentido	Puesta a 0		Reloj	Sentido	Puesta a 0	
5								
6	Reloj adelante	Reloj atrás			Reloj adelante	Reloj atrás		
7	Reloj adelante	Reloj atrás	Puesta a 0		Reloj adelante	Reloj atrás	Puesta a 0	
8								
9	Fase A	Fase B			Fase A	Fase B		
10	Fase A	Fase B	Puesta a 0		Fase A	Fase B	Puesta a 0	
11								

Como muestra la tabla, si HSC0 se está utilizando en los modos 3 a 10 (reloj y sentido o alguna de las dos fases de reloj), HSC3 no se podrá utilizar, puesto que tanto HSC0 como HSC3 utilizan la entrada I0.1. Esto mismo es aplicable a HSC4 y HSC5 (ambos utilizan I0.4).

Las entradas I0.0 a I0.3 se pueden utilizar no sólo para los contadores rápidos, sino también para ocho eventos de interrupción de flanco. Dichas entradas no se pueden utilizar simultáneamente para interrupciones de flanco y para los contadores rápidos.

Una misma entrada no se puede utilizar para dos funciones diferentes. No obstante, cualquier entrada que no se esté utilizando en el modo actual del contador rápido se puede utilizar para otro fin. Por ejemplo, si HSC0 se está utilizando en modo 2 (que utiliza las entradas I0.0 e I0.2), I0.1 se podrá utilizar para interrupciones de flanco o para HSC3.

La tabla 6-2 muestra los modos de operación asistidos por los contadores rápidos HSC1 y HSC2. Las CPUs 224 y 226 son las únicas que soporta dichos contadores rápidos.

Tabla 6-2 Contadores rápidos HSC1 y HSC2

Modo	HSC1				HSC2			
	I0.6	I0.7	I1.0	I1.1	I1.2	I1.3	I1.4	I1.5
0	Reloj				Reloj			
1	Reloj		Puesta a 0		Reloj		Puesta a 0	
2	Reloj		Puesta a 0	Arranque	Reloj		Puesta a 0	Arranque
3	Reloj	Sentido			Reloj	Sentido		
4	Reloj	Sentido	Puesta a 0		Reloj	Sentido	Puesta a 0	
5	Reloj	Sentido	Puesta a 0	Arranque	Reloj	Sentido	Puesta a 0	Arranque
6	Reloj adelante	Reloj atrás			Reloj adelante	Reloj atrás		
7	Reloj adelante	Reloj atrás	Puesta a 0		Reloj adelante	Reloj atrás	Puesta a 0	
8	Reloj adelante	Reloj atrás	Puesta a 0	Arranque	Reloj adelante	Reloj atrás	Puesta a 0	Arranque
9	Fase A	Fase B			Fase A	Fase B		
10	Fase A	Fase B	Puesta a 0		Fase A	Fase B	Puesta a 0	
11	Fase A	Fase B	Puesta a 0	Arranque	Fase A	Fase B	Puesta a 0	Arranque

Cada contador dispone de entradas que soportan funciones tales como relojes, control del sentido, puesta a 0 y arranque. Los contadores A/B permiten elegir una velocidad simple o cuádruple para el conteo. HSC1 y HSC2 trabajan de forma completamente independiente y sin afectar a otras funciones rápidas. Ambos contadores funcionan a velocidades máximas sin interferirse mutuamente.

Para obtener más información acerca de cómo utilizar los contadores rápidos, consulte el apartado 9.4 (Operaciones con contadores rápidos SIMATIC) en el capítulo 9.

Salidas de impulsos rápidos

Las CPUs S7-200 asisten salidas de impulsos rápidos. Q0.0 y Q0.1 pueden generar trenes de impulsos rápidos (PTO) o controlar la modulación del ancho de impulsos (PWM).

- La función PTO ofrece una salida en cuadratura (con un ancho de impulsos de 50%) para un número determinado de impulsos y un tiempo de ciclo determinado. El número de impulsos puede estar comprendido entre 1 y 4.294.967.295. El tiempo de ciclo se puede indicar en microsegundos o milisegundos, bien sea de 50 μ s a 65.535 μ s, o bien de 2 ms a 65.535 ms. Un número impar de microsegundos o milisegundos (p.ej. 75 ms) causa una distorsión del factor de trabajo relativo. La función Tren de impulsos (PTO) se puede programar para producir un tren de impulsos, o bien un perfil de impulsos compuesto por varios trenes de impulsos. En éste último caso, la función PTO se puede programar para controlar un motor paso a paso utilizando una secuencia simple de aceleración, funcionamiento y desaceleración, o bien secuencias más complicadas. El perfil de impulsos puede comprender hasta 255 segmentos, correspondiendo un segmento a la función de aceleración, funcionamiento o desaceleración.
- La función PWM ofrece un tiempo de ciclo fijo con una salida de ancho de impulsos variable. El tiempo de ciclo y el ancho de impulsos pueden indicarse en incrementos de microsegundos o milisegundos. El tiempo de ciclo puede estar comprendido entre 50 μ s y 65.535 μ s, o bien entre 2 ms y 65.535 ms. El tiempo del ancho de impulsos puede estar comprendido entre 0 μ s y 65.535 μ s, o bien entre 0 ms y 65.535 ms. Si el ancho de impulsos y el tiempo de ciclo son iguales, entonces el factor de trabajo relativo (relación impulso-pausa) será de 100% y la salida se activará continuamente. Si el ancho de impulsos es cero, el factor de trabajo relativo (relación impulso-pausa) será de 0% y se desactivará la salida.

Para obtener más información acerca de la salida de impulsos, consulte el apartado 9.4 (Operaciones con contadores rápidos SIMATIC) en el capítulo 9.

6.7 Potenciómetros analógicos

Los potenciómetros analógicos están ubicados debajo de la tapa de acceso frontal de la CPU. Dichos potenciómetros permiten incrementar o decrementar valores almacenados en los bytes de marcas especiales SMB28 y SMB29. El programa puede utilizar estos valores de sólo lectura para diversas funciones, p.ej. para actualizar el valor actual de un temporizador o de un contador, para introducir o modificar los valores estándar, o bien, para ajustar límites.

SMB28 almacena el valor digital que representa la posición del potenciómetro analógico 0. SMB29 almacena el valor digital que representa la posición del potenciómetro analógico 1. El potenciómetro analógico tiene un margen nominal comprendido entre 0 y 255, así como una capacidad de repetición de ± 2 contajes.

Utilice un destornillador pequeño para ajustar los potenciómetros. Gire el potenciómetro hacia la derecha para incrementar el valor, o bien hacia la izquierda para decrementarlo. La figura 6-10 muestra un programa de ejemplo donde se utiliza el potenciómetro analógico.

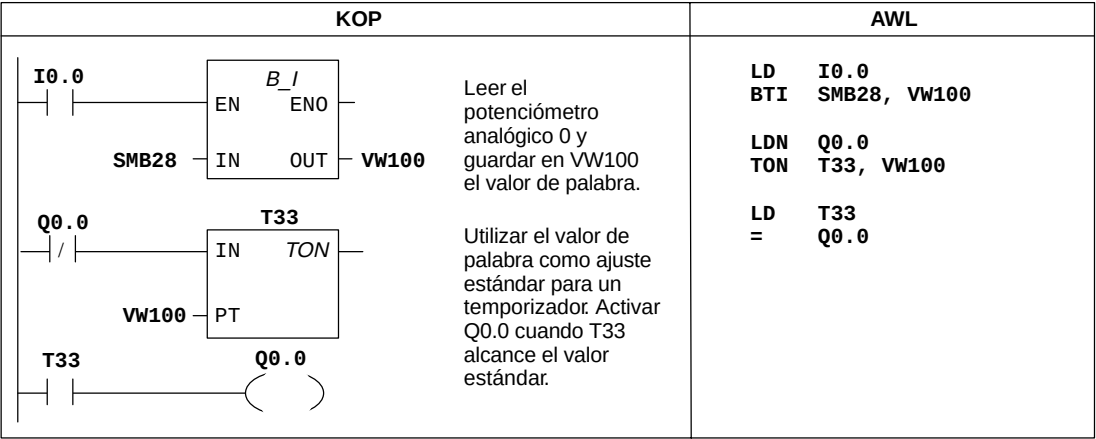


Figura 6-10 Ejemplo de un potenciómetro analógico

Configurar el hardware para la comunicación en redes

7

Este capítulo describe las comunicaciones utilizando la versión 3.0 de STEP 7-Micro/WIN 32. Las versiones anteriores del software ofrecían otras funciones. También se explica cómo configurar el hardware y cómo instalar una red de comunicaciones S7-200.

Índice del capítulo

Apartado	Descripción	Página
7.1	Opciones de comunicación	7-2
7.2	Instalar y desinstalar interfaces de comunicación	7-7
7.3	Seleccionar y cambiar parámetros	7-9
7.4	Comunicación con módems	7-16
7.5	Redes y protocolos	7-27
7.6	Componentes de redes	7-32
7.7	Utilizar el cable PC/PPI con otros dispositivos y en modo Freeport	7-36
7.8	Rendimiento de la red	7-42

7.1 Opciones de comunicación

Las CPUs S7-200 se pueden disponer en diversas configuraciones para soportar la comunicación en redes. El software STEP 7-Micro/WIN 32 se puede instalar en un PC dotado con el sistema operativo Windows 95, Windows 98 o Windows NT, o bien, en una unidad de programación SIMATIC (p.ej. la PG 740). El PC o la PG se pueden utilizar como unidades maestras en cualquiera de las siguientes configuraciones:

- Monomaestro: Un sólo maestro se conecta a uno o varios esclavos (v. fig. 7-1).
- Multimaestro: Un sólo maestro se conecta a uno o varios esclavos y a uno o más maestros (v. fig. 7-2).
- Para usuarios de módems de 11 bits: Un sólo maestro se conecta a uno o varios esclavos. El maestro se conecta a través de módems de 11 bits a una CPU S7-200 esclava o a una red de CPUs S7-200 esclavas.
- Para usuarios de módems de 10 bits: Un sólo maestro se conecta a través de un módem de 10 bits a una CPU S7-200 esclava.

Las figuras 7-1 y 7-2 muestran una configuración con un PC conectado a varias CPUs S7-200. STEP 7-Micro/WIN 32 se ha diseñado para comunicarse con una sola CPU S7-200. No obstante, se puede acceder a cualquier CPU que intervenga en la red. Las CPUs pueden ser tanto maestras como esclavas. El TD 200 es una unidad maestra. Para obtener más información sobre la comunicación en redes, consulte el apartado 7.5.

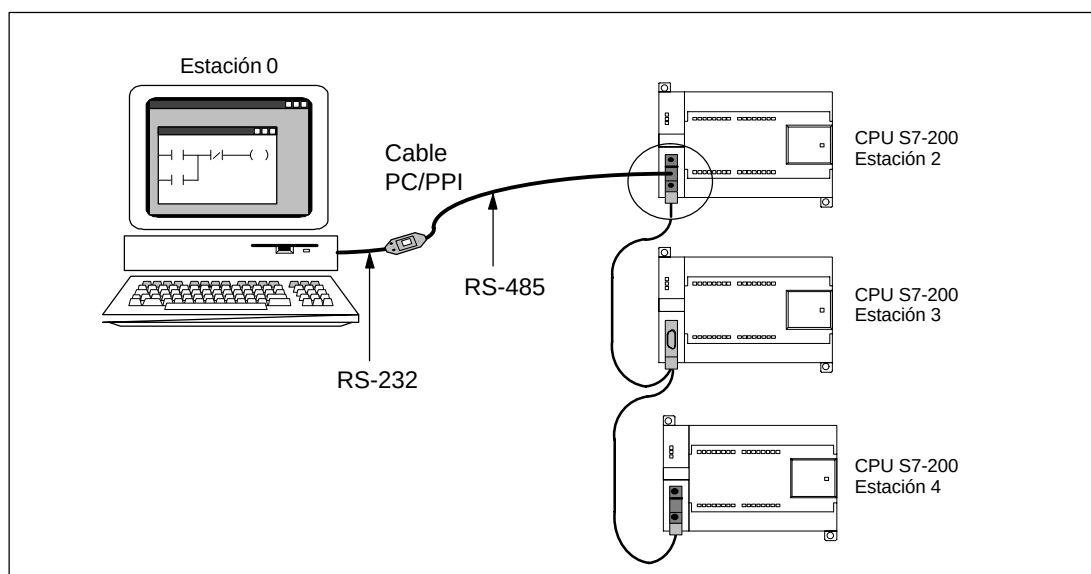


Figura 7-1 Utilizar un cable PC/PPI para la comunicación con varias CPUs S7-200

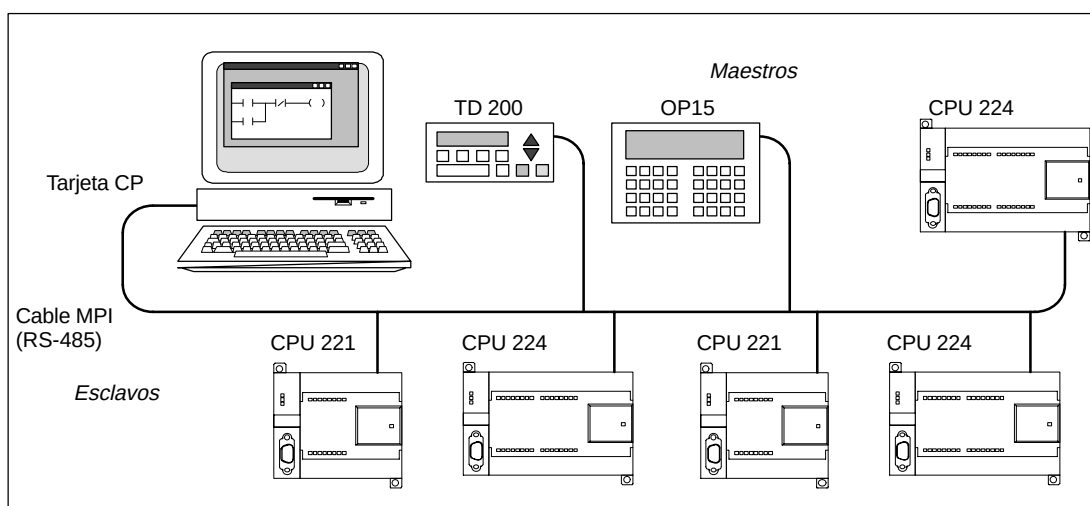


Figura 7-2 Ejemplo de una tarjeta CP con maestros y esclavos

Cómo elegir la configuración de comunicación

La tabla 7-1 muestra las configuraciones de hardware posibles y las velocidades de transferencia que soporta STEP 7-Micro/WIN 32. En la tabla 7-2 se indican las configuraciones asistidas por la CPU S7-200 y el módulo EM 277 PROFIBUS-DP.

Tabla 7-1 Configuraciones de hardware soportadas por STEP 7-Micro/WIN 32

Hardware soportado	Tipo	Velocidad de transferencia soportada	Protocolos soportados
Cable PC/PPI	Conector de cable al puerto COM del PC	9,6 kbit/s 19,2 kbit/s	Protocolo PPI
CP 5511	Tipo II, tarjeta PCMCIA	9,6 kbit/s 19,2 kbit/s 187,5 kbit/s	Protocolos PPI, MPI y PROFIBUS para PCs portátiles formato notebook
CP 5611	Tarjeta PCI (versión 3 o superior)		Protocolos PPI, MPI y PROFIBUS para PCs
MPI	Tarjeta PC ISA integrada en la PG		

Tabla 7-2 Configuraciones soportadas por la CPU S7-200 y el módulo EM 277

Puerto de conexión	Velocidad de transferencia soportada	Cantidad de enlaces lógicos	Protocolos soportados
CPU S7-200			
Puerto 0	9,6 kbit/s	4 por puerto	PPI, MPI y PROFIBUS
Puerto 1	19,2 kbit/s 187,5 kbit/s 187,5 kbit/s		
Módulo EM277 PROFIBUS–DP			
Hasta 2 por cada CPU	9,6 kbit/s a 12 Mbit/s	6 por módulo	MPI y PROFIBUS

Comunicación utilizando tarjetas CP o MPI

Siemens ofrece diversas tarjetas de interface que se pueden incorporar en un PC o en una PG (unidad de programación) SIMATIC. Las tarjetas permiten que el PC o la PG actúen de estaciones maestras en la red. Estas tarjetas contienen componentes de hardware especiales para asistir al PC o a la PG en la gestión de la red multimaestro, soportando diferentes protocolos y diversas velocidades de transferencia (v. tabla 7-1).

La tarjeta y el protocolo específicos se ajustan en el cuadro de diálogo "Ajustar interface PG/PC" accesible desde STEP 7-Micro/WIN 32 (v. apt. 7.3). Utilizando Windows 95, Windows 98 o Windows NT se puede seleccionar cualquier protocolo (PPI, MPI o PROFIBUS) a utilizar con cualquiera de las tarjetas de red.

Todas las tarjetas incorporan un puerto RS-485 sencillo para la conexión a la red PROFIBUS. La tarjeta CP 5511 PCMCIA dispone de un adaptador que incorpora el conector D subminiatura de 9 pines. Uno de los extremos del cable MPI se conecta al puerto RS-485 de la tarjeta y el otro, al conector del puerto de programación (v. fig. 7-2). Para obtener informaciones más detalladas acerca de los procesadores de comunicación, consulte el Catálogo ST 70 1997: *Componentes SIMATIC para la Integración Total en Automatización*.

¿Dónde se configura la comunicación?

La comunicación se puede configurar desde los siguientes puntos en Windows 95, Windows 98 o Windows NT 4.0:

- Durante la fase final de la instalación del software STEP 7-Micro/WIN 32.
- En STEP 7-Micro/WIN 32.

Cómo configurar la comunicación en STEP 7-Micro/WIN 32

En el cuadro de diálogo "Configurar la comunicación" de STEP 7-Micro/WIN 32 se pueden configurar los parámetros de comunicación. Para acceder a dicho cuadro puede optar por uno de los siguientes métodos:

- Elija el comando de menú **Ver > Comunicación**.
- Haga clic en el icono "Comunicación" en la pantalla de STEP 7-Micro/WIN 32 (v. fig. 7-3).

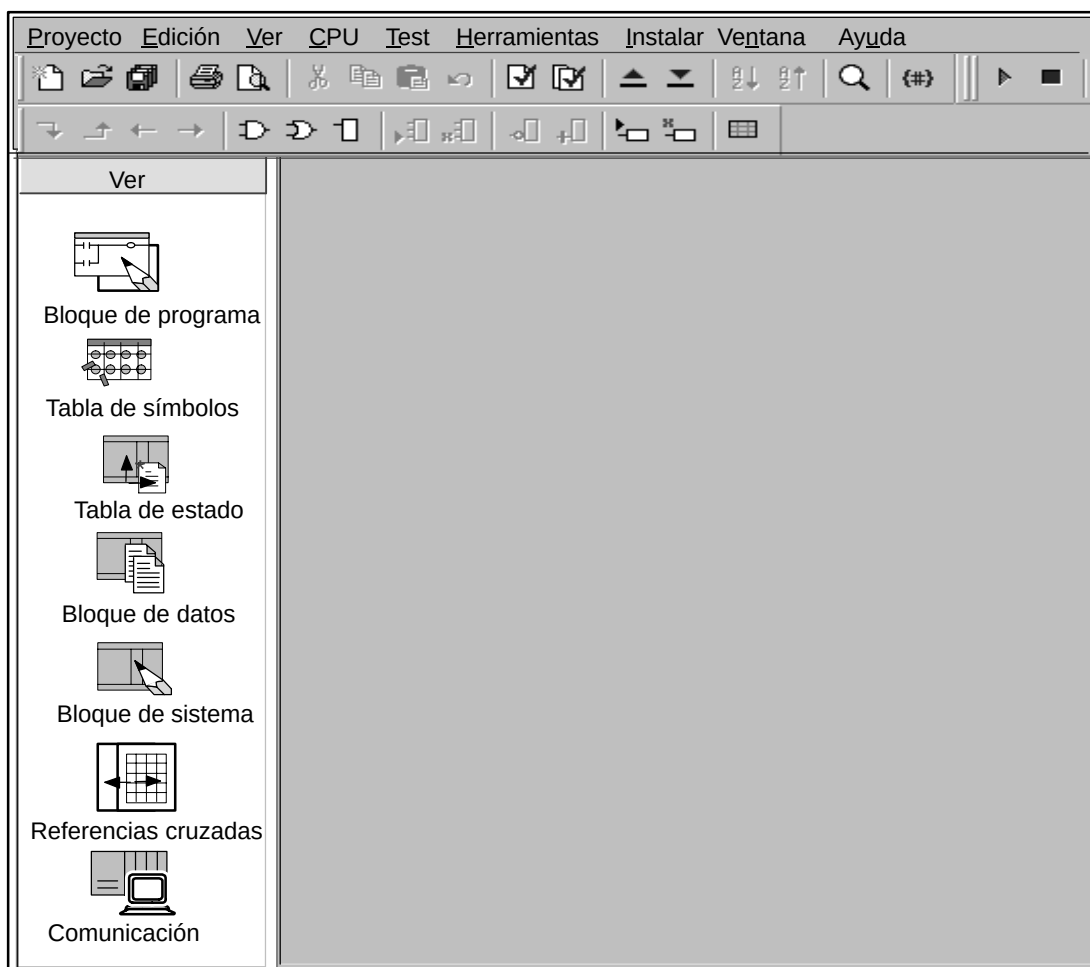


Figura 7-3 Menú Ver de STEP 7-Micro/WIN 32

En el cuadro de diálogo "Configurar la comunicación", haga doble clic en el icono superior que aparece en el lado derecho. Se visualizará el cuadro de diálogo "Ajustar interface PG/PC" (v. fig. 7-4).

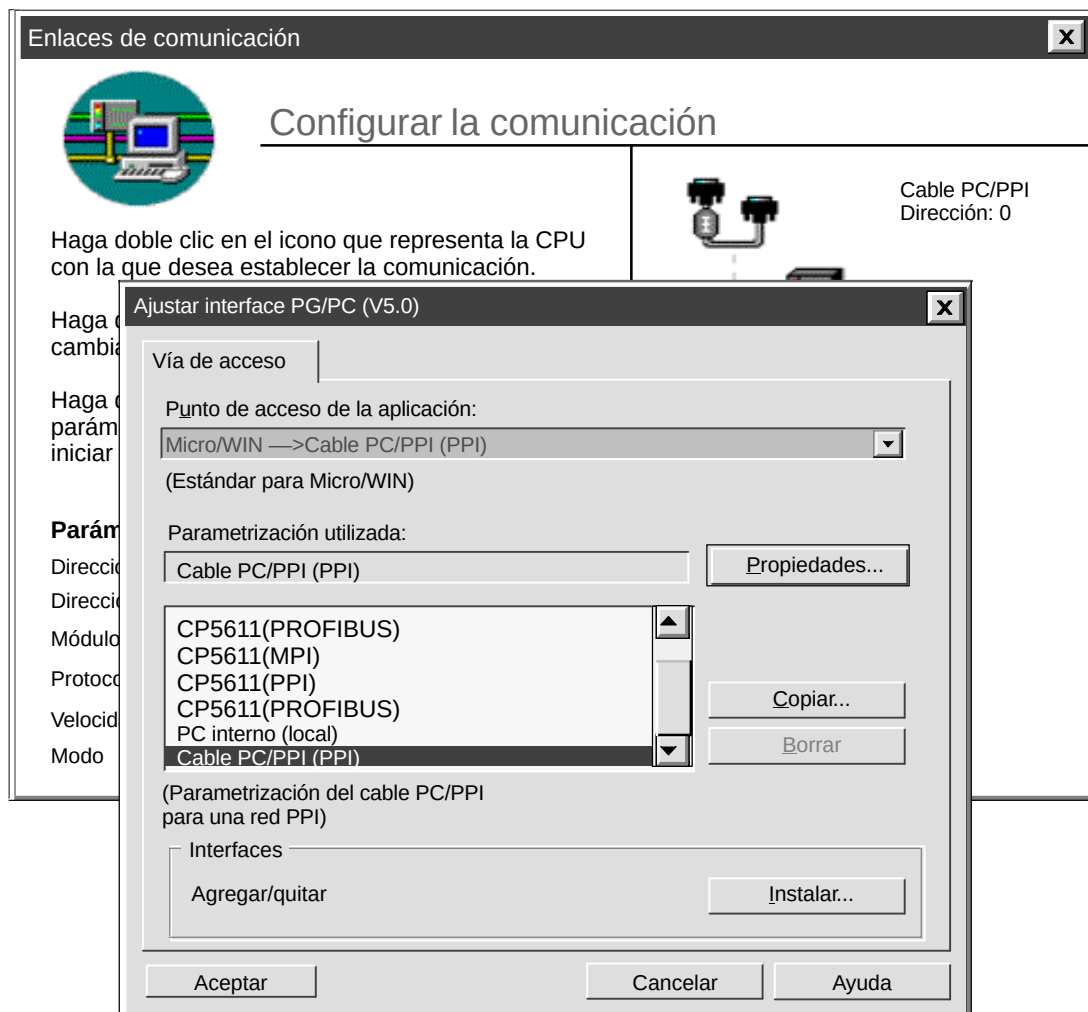


Figura 7-4 Ajustes en el cuadro de diálogo "Interface PG/PC"

7.2 Instalar y desinstalar interfaces de comunicación

Los componentes de hardware de comunicación se pueden instalar o desinstalar en el cuadro de diálogo "Instalar/desinstalar interfaces" que muestra la figura 7-5. En el lado izquierdo del cuadro de diálogo figura una lista de los componentes que no se han instalado todavía. En el lado derecho se visualiza una lista de los componentes instalados actualmente. En caso de utilizar el sistema operativo Windows NT 4.0, el botón "Recursos" se visualizará debajo de ésta última lista.

Instalar componentes de hardware:

Para instalar componentes de hardware, siga los siguientes pasos:

1. En el cuadro de diálogo "Ajustar interface PG/PC" (v. fig. 7-4), haga clic en el botón "Instalar..." para acceder al cuadro de diálogo "Instalar/desinstalar interfaces" que muestra la figura 7-5.
2. En el cuadro de lista "Selección", elija el componente de hardware que desea instalar. En la ventana inferior se visualiza una descripción del componente seleccionado.
3. Haga clic en el botón "Instalar -->".
4. Cuando termine de instalar los componentes de hardware, haga clic en el botón "Cerrar". Aparecerá el cuadro de diálogo "Ajustar interface PG/PC" y los componentes que haya seleccionado se visualizarán en el cuadro de lista "Parametrización utilizada" (v. fig. 7-4).

Desinstalar componentes de hardware:

Para desinstalar componentes de hardware, siga los siguientes pasos:

1. En el cuadro de lista "Instalados" que aparece en el lado derecho, elija los componentes que desea desinstalar.
2. Haga clic en el botón "<-- Desinstalar".
3. Cuando termine de desinstalar los componentes de hardware, haga clic en el botón "Cerrar". Aparecerá el cuadro de diálogo "Ajustar interface PG/PC" y los componentes que haya seleccionado se visualizarán en el cuadro de lista "Parametrización utilizada" (v. fig. 7-4).

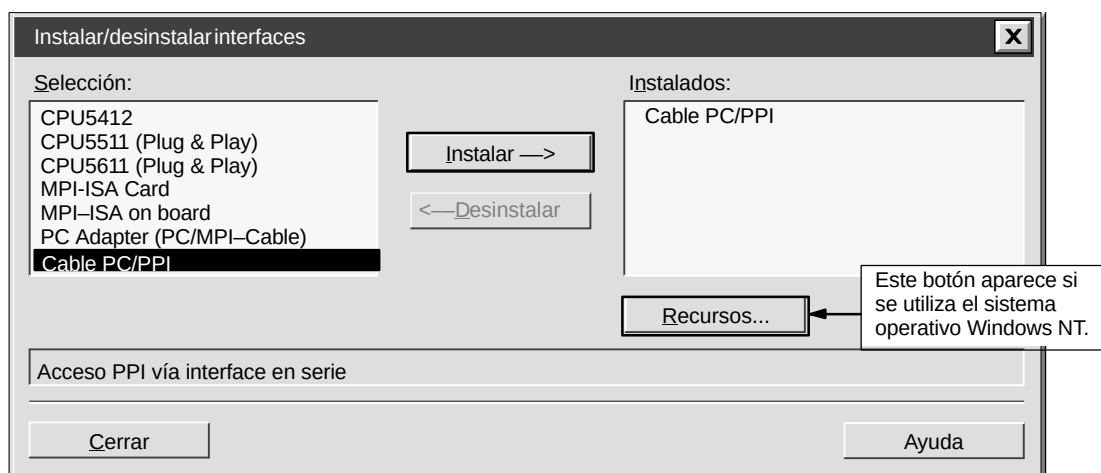


Figura 7-5 Cuadro de diálogo "Instalar/desinstalar interfaces"

Instalación de componentes de hardware en Windows NT

La instalación de componentes de hardware en el sistema operativo Windows NT difiere un poco de la instalación en Windows 95. Aunque para ambos sistemas operativos se utilizan los mismos componentes, la instalación bajo Windows NT exige conocimientos más detallados del hardware a instalar. Windows 95 intenta instalar automáticamente los recursos del sistema, contrariamente a Windows NT que sólo proporciona los valores estándar. Estos valores pueden o no corresponder a la configuración del hardware. No obstante, los parámetros se pueden modificar fácilmente para que concuerden con los ajustes que exige el sistema.

Tras haber instalado un componente de hardware, selecciónelo en el cuadro de lista "Instalados" y haga clic en el botón "Recursos" (v. fig. 7-5). Aparecerá el cuadro de diálogo "Recursos" (v. fig. 7-6). Allí se pueden modificar los ajustes del componente de hardware que se ha instalado. Si el botón aparece atenuado (gris), no es necesario tomar más medidas al respecto.

En caso necesario, consulte el manual del componente de hardware en cuestión para determinar los valores de los parámetros que aparecen en el cuadro de diálogo, dependiendo de los ajustes del hardware. Es posible que deba realizar varios intentos a la hora de elegir la interrupción adecuada para establecer la comunicación.

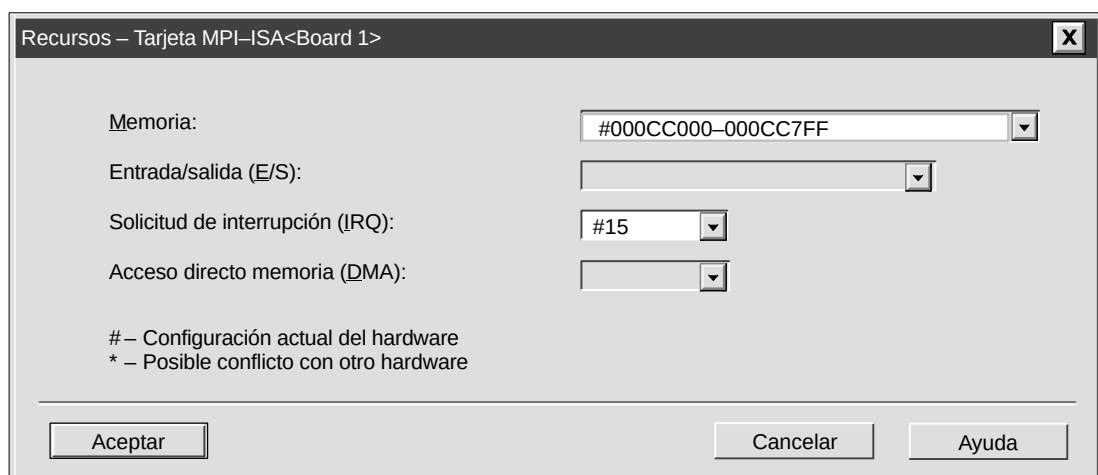


Figura 7-6 Cuadro de diálogo "Recursos" para Windows NT

Nota

Si utiliza Windows NT y un cable PC/PPI, en la red no podrá intervenir ningún otro maestro.

7.3 Seleccionar y cambiar parámetros

Seleccionar y configurar la parametrización correcta

Tras acceder al cuadro de diálogo "Ajustar interface PG/PC", verifique si "Micro/WIN" aparece en el cuadro de lista "Punto de acceso de la aplicación" (v. fig. 7-4). El cuadro de diálogo "Ajustar interface PG/PC" se utiliza en diferentes aplicaciones, como p.ej. STEP 7 y WinCC. Por tanto, puede ser necesario indicar al programa para qué aplicación desea configurar los parámetros.

Tras seleccionar "Micro/WIN" e instalar el hardware, se deben ajustar las propiedades actuales para la comunicación con éste último. Primero que todo es necesario determinar el protocolo a utilizar en la red. Es recomendable utilizar el protocolo PPI para todas las CPUs.

Tras seleccionar el protocolo que desea utilizar, puede elegir la parametrización correcta en el cuadro de lista "Parametrización utilizada" del cuadro de diálogo "Ajustar interface PG/PC". Allí se indican los componentes de hardware instalados, junto con el tipo de protocolo (entre paréntesis). Por ejemplo, una configuración sencilla puede exigir que se utilice un cable PC/PPI para la comunicación con una CPU 222. En este caso se debe seleccionar "Cable PC/PPI(PPI)".

Tras haber elegido la parametrización correcta, debe ajustar los distintos parámetros para la configuración actual. Haga clic en el botón "Propiedades..." del cuadro de diálogo "Ajustar interface PG/PC". Esta acción le conduce a uno de varios cuadros de diálogo posibles, dependiendo de la parametrización que haya seleccionado (v. fig. 7-7). En los apartados siguientes se describe detalladamente cada uno de ellos.

En resumen, para seleccionar la parametrización de un interface, siga los siguientes pasos:

1. En la ficha "Vía de acceso" del cuadro de diálogo "Ajustar interface PG/PC" (v. fig. 7-4), seleccione "Micro/WIN" en el cuadro de lista "Punto de acceso de la aplicación".
2. Verifique si los componentes de hardware están instalados (v. apt. 7.2).
3. Determine el protocolo que desea utilizar. Es recomendable utilizar el protocolo PPI para todas las CPUs.
4. En el cuadro de lista "Parametrización utilizada" del cuadro de diálogo "Ajustar interface PG/PC", elija la configuración correcta.
5. Haga clic en el botón "Propiedades..." del cuadro de diálogo "Ajustar interface PG/PC".

Allí puede efectuar los ajustes conforme a la parametrización elegida.

Ajustar los parámetros del cable PC/PPI (PPI)

Aquí se explica cómo ajustar los parámetros PPI en los sistemas operativos Windows 95, Windows 98 o Windows NT 4.0 para el cable PC/PPI.

Si en el cuadro de diálogo "Ajustar interface PG/PC" está seleccionado el cable PC/PPI (PPI) y se hace clic en el botón "Propiedades...", aparecerá la ficha de propiedades del mismo (v. fig. 7-7).

STEP 7-Micro/WIN 32 utiliza por defecto un protocolo PPI multimaestro para comunicarse con las CPUs S7-200. Este protocolo le permite a STEP 7-Micro/WIN 32 coexistir con otros maestros (TDs 200 y paneles de operador) en una red. Este modo se habilita marcando la casilla de verificación "Red multimaestro" en el cuadro de propiedades del cable PC/PPI accesible desde el cuadro de diálogo "Ajustar interface PG/PC". Windows NT 4.0 no soporta la opción multimaestro.

STEP 7-Micro/WIN 32 también soporta el protocolo PPI con un maestro único. Si se utiliza este protocolo, STEP 7-Micro/WIN 32 supone que es el único maestro en la red, por lo que no coopera para compartir la red con otros maestros. Este protocolo sólo se deberá utilizar al transmitir datos vía módems o en redes con muchas interferencias. El protocolo con un maestro único se selecciona borrando la marca de verificación de la casilla "Red multimaestro" en el cuadro de propiedades del cable PC/PPI accesible desde el cuadro de diálogo "Ajustar interface PG/PC".

Para ajustar los parámetros PPI, siga los siguientes pasos:

1. En el área "Propiedades del equipo" de la ficha "PPI", elija un número en el cuadro "Dirección". Este número indica qué dirección debe tener STEP 7-Micro/WIN 32 en la red de sistemas de automatización. El ajuste estándar para el PC en el que se está ejecutando STEP 7-Micro/WIN 32 es la dirección 0. El ajuste estándar para la primera CPU de la red es la dirección 2. Todo dispositivo (PC, CPU, etc.) que intervenga en la red debe tener una dirección unívoca. No asigne una misma dirección a varios dispositivos.
2. Elija un valor en el cuadro "Timeout". Este valor representa el tiempo durante el cual los drivers de comunicación deben intentar establecer enlaces. El valor estándar debería ser suficiente.
3. Determine si desea que STEP 7-Micro/WIN 32 intervenga en una red multimaestro. Puede dejar marcada la casilla "Red multimaestro", a menos que esté utilizando un módem o Windows NT 4.0. En ese caso, la casilla no se puede marcar, puesto que STEP 7-Micro/WIN 32 no soporta esta funcionalidad.
4. Ajuste la velocidad de transferencia que desea utilizar para la comunicación de STEP 7-Micro/WIN 32 en la red. El cable PPI soporta velocidades de 9,6 kbit/s y 19,2 kbit/s.
5. Elija la dirección de estación más alta. STEP 7-Micro/WIN 32 busca otras estaciones de una red PPI hasta esta dirección como máximo.

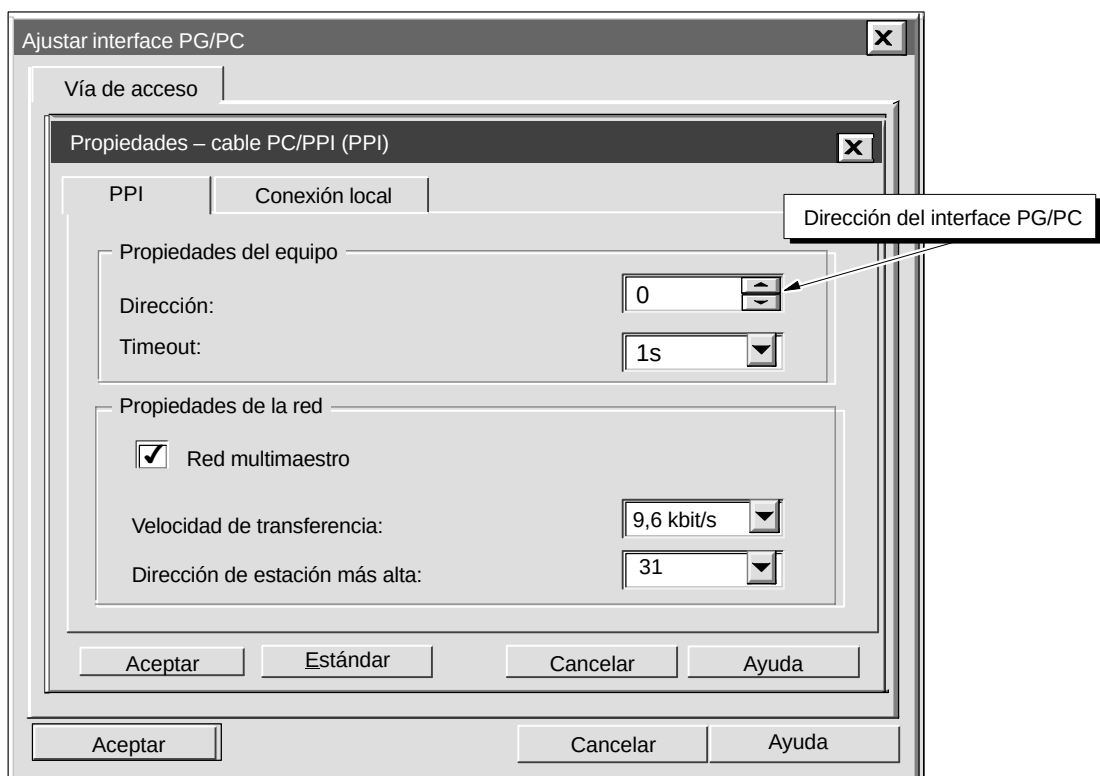


Figura 7-7 Cuadro de diálogo "Propiedades – Cable PC/PPI (PPI)", ficha "PPI"

6. Haga clic en la ficha "Conexión local" (v. fig. 7-8).
7. En la ficha "Conexión local", seleccione el puerto COM al que está conectado el cable PC/PPI. Si utiliza un módem, seleccione el puerto COM al que esté conectado el módem y marque la casilla de verificación "Utilizar módem".
8. Haga clic en el botón "Aceptar" para salir del cuadro de diálogo "Ajustar interface PG/PC".

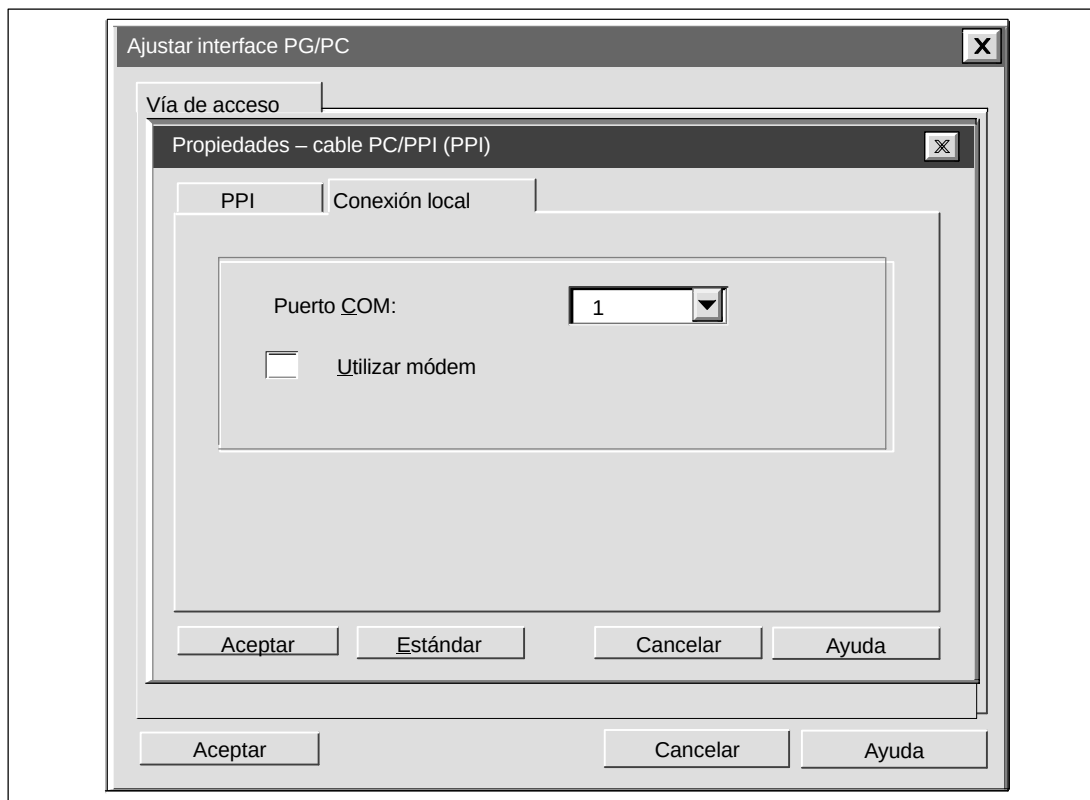


Figura 7-8 Cuadro de diálogo "Propiedades – cable PC/PPI (PPI)", ficha "Conexión local"

Configuraciones posibles al utilizar un PC con una tarjeta MPI o un CP en una red multimaestro

Una tarjeta interface multipunto (tarjeta MPI) o un procesador de comunicaciones (CP) permiten crear numerosas configuraciones. Ambos componentes disponen de un puerto RS-485 sencillo para la conexión a la red mediante un cable MPI. Una estación en la que se ejecute el software de programación STEP 7-Micro/WIN 32 (PC con tarjeta MPI o CP, o bien una unidad de programación SIMATIC) se puede conectar a una red en la que intervengan varios maestros. (Ello es aplicable también al cable PC/PPI si se han habilitado varios maestros). Los maestros pueden ser también paneles de operador y visualizadores de textos (TDs 200). La figura 7-9 muestra una configuración con dos TDs 200 que se han integrado en la red.

Nota

Si se utiliza la parametrización PPI, STEP 7-Micro/WIN 32 no soportará la ejecución simultánea de dos aplicaciones diferentes en una misma tarjeta MPI o CP. Cierre la otra aplicación antes de conectar STEP 7-Micro/WIN 32 a la red a través de la tarjeta MPI o CP.

Esta configuración ofrece las siguientes posibilidades de comunicación:

- STEP 7-Micro/WIN 32 (en la estación 0) puede vigilar el estado de la estación de programación 2, mientras que los visualizadores de textos TD 200 (estaciones 5 y 1) se comunican con las CPUs 224 (estaciones 3 y 4, respectivamente).
- Ambas CPUs 224 se pueden habilitar para que envíen mensajes utilizando operaciones de red (NETR y NETW).
- La estación 3 puede leer datos de y escribir datos en las estaciones 2 (CPU 222) y 4 (CPU 224).
- La estación 4 puede leer datos de y escribir datos en las estaciones 2 (CPU 222) y 3 (CPU 224).

A una sola red se pueden conectar numerosos maestros y esclavos. No obstante, el rendimiento de la misma puede disminuir cuantas más estaciones se incorporen.

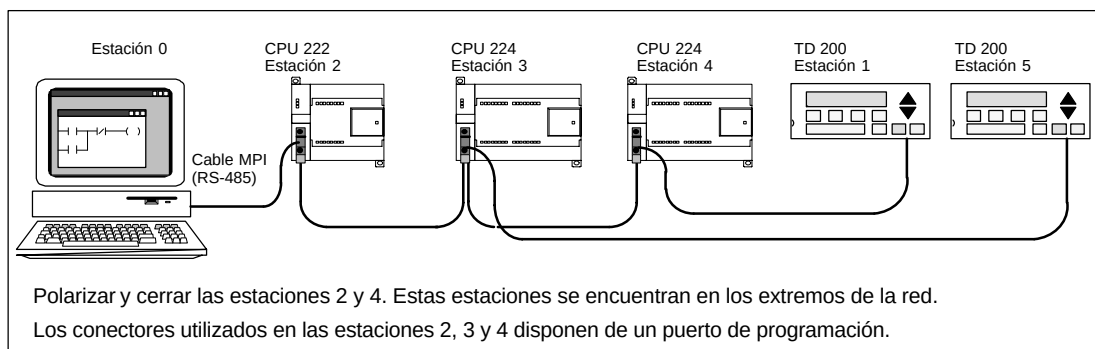


Figura 7-9 Utilizar una tarjeta MPI o un CP para la comunicación con CPUs S7-200

Ajustar los parámetros de las tarjetas CP o MPI (PPI)

Aquí se explica cómo ajustar los parámetros PPI en los sistemas operativos Windows 95, Windows 98 o Windows NT 4.0 para los siguientes componentes de hardware:

- CP 5511
- CP 5611
- MPI

Partiendo del cuadro de diálogo "Ajustar interface PG/PC", si alguna de las tarjetas MPI o CP mencionadas se utiliza junto con el protocolo PPI y se hace clic en el botón "Propiedades...", aparecerá la ficha de propiedades de la tarjeta XXX(PPI), donde "XXX" es el tipo de tarjeta que se ha instalado (p.ej. MPI-ISA) (v. fig. 7-10).

Nota

Utilice el protocolo MPI al comunicarse con una CPU S7-200 215 (por el puerto 1). Para obtener más información sobre la CPU 215 y el protocolo MPI, consulte la versión anterior del *Manual del sistema de automatización S7-200* (referencia: 6ES7-298-8FA01-8BH0).

Para ajustar los parámetros PPI, siga los siguientes pasos:

1. En la ficha "PPI", elija un número en el cuadro "Dirección". Este número indica qué dirección debe tener STEP 7-Micro/WIN 32 en la red de sistemas de automatización.
2. Elija un valor en el cuadro "Timeout". Este valor representa el tiempo durante el cual los drivers de comunicación deben intentar establecer enlaces. El valor estándar debería ser suficiente.
3. Ajuste la velocidad de transferencia que desea utilizar para la comunicación de STEP 7-Micro/WIN 32 en la red.
4. Elija la dirección de estación más alta. STEP 7-Micro/WIN 32 busca otros maestros en la red hasta esta dirección como máximo.
5. Haga clic en el botón "Aceptar" para salir del cuadro de diálogo "Ajustar interface PG/PC".

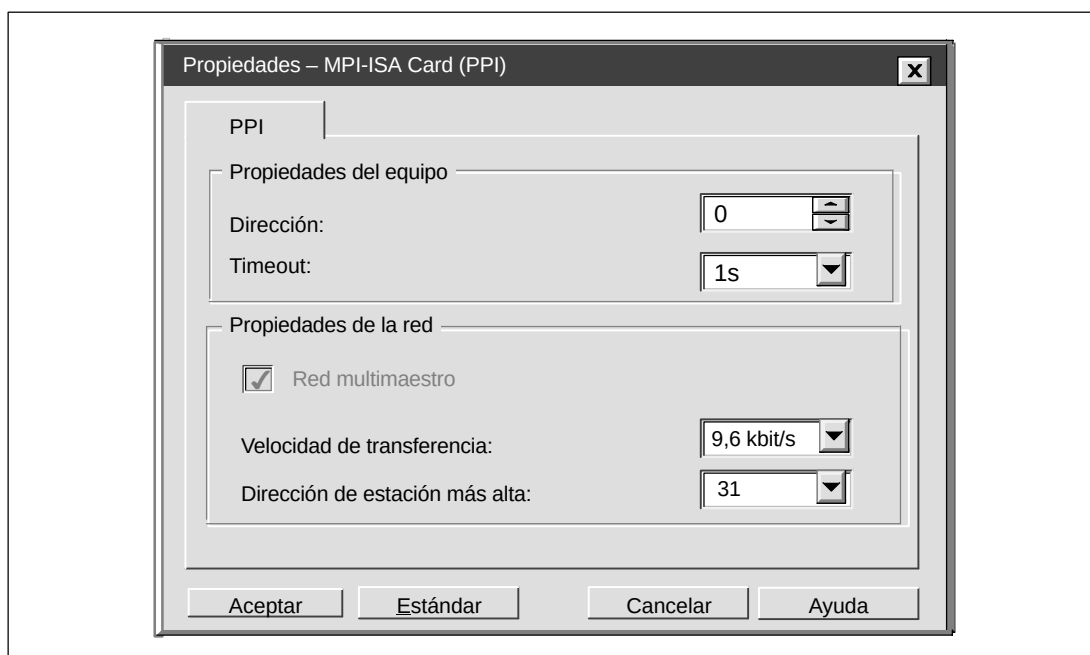


Figura 7-10 Propiedades de la tarjeta MPI-ISA (PPI)

7.4 Comunicación con módems

Ajustar los parámetros de comunicación al utilizar módems

Para ajustar los parámetros de comunicación entre la unidad de programación (PG) o el PC y la CPU al utilizar módems, es preciso utilizar la parametrización del cable PC/PPI. En caso contrario no se dispondrá de la función "Configurar módem". Verifique que la función esté habilitada y ajuste los parámetros de configuración como se indica a continuación:

Nota

STEP 7-Micro/WIN 32 visualiza módems estándar en el cuadro de diálogo "Configurar módem". Estos módems se han comprobado, verificándose que trabajen con STEP 7-Micro/WIN 32 conforme a los ajustes visualizados.

Configurar el módem local:

1. Elija el comando de menú **Ver > Comunicación** (o haga clic en el icono "Comunicación").

En el cuadro de diálogo "Configurar la comunicación", haga doble clic en el icono del cable PC/PPI. Aparecerá el cuadro de diálogo "Ajustar interface PG/PC". Salte al paso 3.

Si en el cuadro de diálogo "Configurar la comunicación" no se visualiza el icono del cable PC/PPI, haga doble clic en el icono del PC o en el icono superior del área derecha.

2. En el cuadro de diálogo "Ajustar interface PG/PC", elija "Cable PC/PPI(PPI)". Si esta selección no figura en el cuadro de lista, será preciso instalarla (v. apt. 7.2).
3. Haga clic en el botón "Propiedades". Se visualizarán las propiedades del cable PC/PPI(PPI) para la CPU y el módem (v. fig. 7-8).
4. En la ventana "Propiedades – Cable PC/PPI(PPI)", haga clic en la ficha "Conexión local".
5. En el área "Puerto COM", verifique que esté marcada la casilla "Utilizar módem". Si la casilla está vacía, haga clic allí para insertar una marca de verificación (v. fig. 7-8).
6. Haga clic en el botón "Aceptar". Aparecerá el cuadro de diálogo "Ajustar interface PG/PC".
7. Haga clic en el botón "Aceptar". Aparecerá el cuadro de diálogo "Configurar la comunicación". Ahora se visualizan dos iconos de módem y un icono "Conectar módem" (v. fig. 7-11).

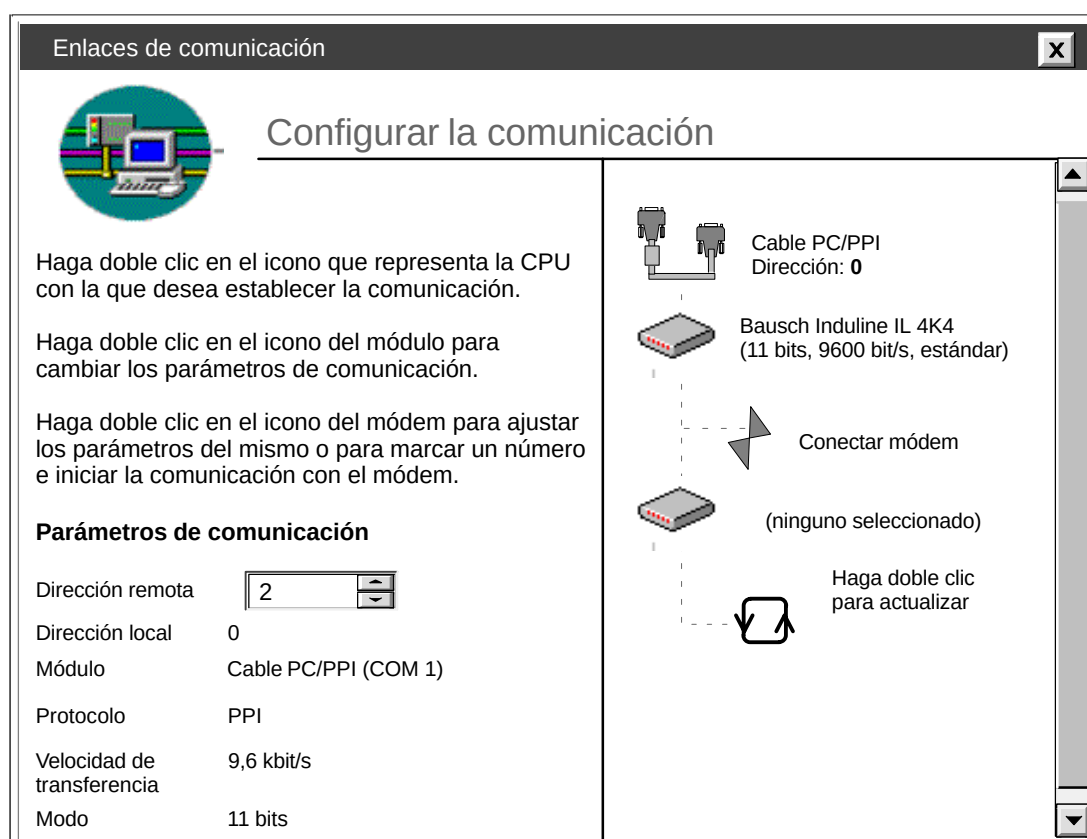


Figura 7-11 Cuadro de diálogo "Configurar la comunicación"

8. En el cuadro de diálogo "Configurar la comunicación", haga doble clic en el primer icono de módem. Aparecerá el cuadro de diálogo "Configurar módem" correspondiente al módem local (v. fig. 7-12).
9. En el área "Módem local", elija su tipo de módem. Si su módem no figura en la lista, haga clic en el botón "Agregar" para configurarlo. Para ello debe conocer los comandos AT del módem. Consulte a este respecto la documentación del módem.
10. En el área "Modo de comunicación", elija el modo deseado (10 u 11 bits). Éste depende de la capacidad del módem. (Los modos de comunicación de 10 bits y de 11 bits se describen más abajo). Los módems local y remoto deben tener el mismo modo de comunicación. Haga clic en el botón "Configurar".



Figura 7-12 Cuadro de diálogo "Configurar módem" para el módem local

11. Aparecerá el cuadro de diálogo "Configurar" (v. fig. 7-13). Si utiliza un módem estándar, el único campo que podrá editar en este cuadro de diálogo es el cuadro "Timeout". El timeout representa el tiempo durante el cual el módem local intenta establecer la comunicación con el módem remoto. Si el tiempo indicado (en segundos) en el cuadro "Timeout" transcurre antes de establecerse la comunicación, fallará el intento de conexión. Si no está utilizando un módem estándar, deberá introducir la cadena de comando AT del módem. Consulte a este respecto la documentación del módem.
12. Si desea comprobar la configuración del módem local, haga clic en el botón "Programa/Test" mientras el módem está conectado a su equipo local (unidad de programación o PC). Así se configura el módem conforme al protocolo y a los ajustes actuales, verificando que el módem acepte los ajustes de configuración. Haga clic en "Aceptar" para regresar al cuadro de diálogo "Configurar la comunicación".
13. Desconecte el módem local y conecte el módem remoto a su equipo local (unidad de programación o PC).

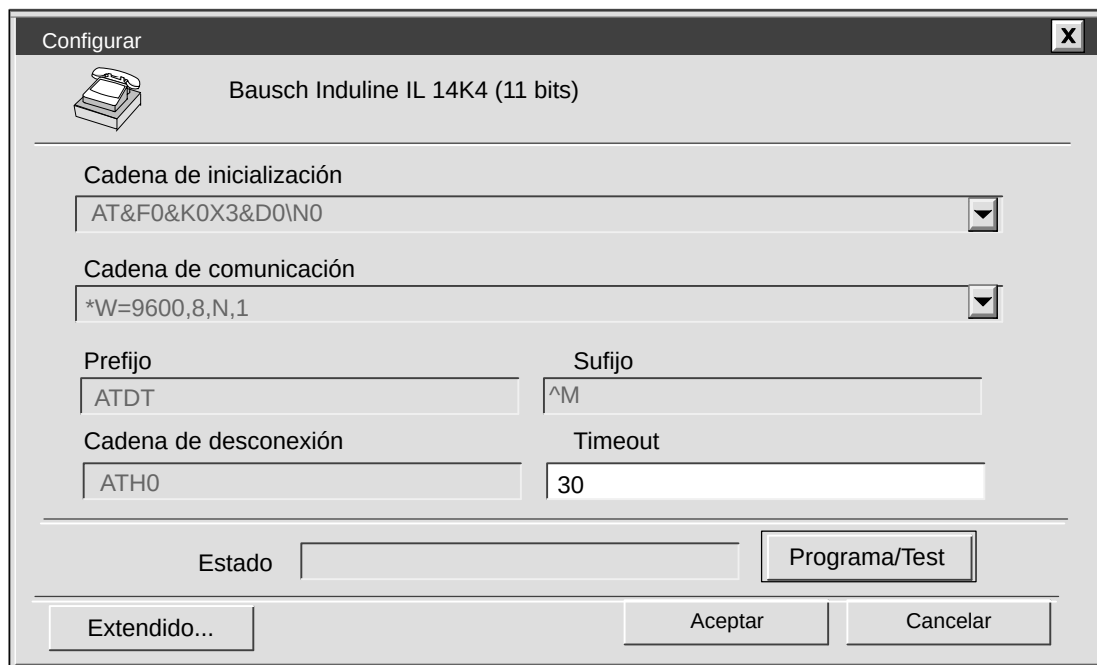


Figura 7-13 Configuración del módem local

Configurar el módem remoto:

1. En el cuadro de diálogo "Configurar la comunicación", haga doble clic en el segundo icono de módem (v. fig. 7-11). Aparecerá el cuadro de diálogo "Configurar módem" correspondiente al módem remoto (v. fig. 7-14).
2. En el área "Módem remoto", elija su tipo de módem. Si su módem no figura en la lista, haga clic en el botón "Agregar" para configurarlo. Para ello debe conocer los comandos AT del módem. Consulte a este respecto la documentación del módem.
3. En el área "Modo de comunicación", elija el modo deseado (10 u 11 bits). Éste depende de la capacidad del módem. (Los modos de comunicación de 10 bits y de 11 bits se describen más abajo). Los módems local y remoto deben tener el mismo modo de comunicación. Haga clic en el botón "Configurar".
4. Aparecerá el cuadro de diálogo "Configurar" (v. fig. 7-15). Si está utilizando un módem estándar no podrá editar ningún campo. Si no está utilizando un módem estándar, deberá introducir la cadena de comando AT del módem. Consulte a este respecto la documentación del módem.
5. Para comprobar la configuración del módem remoto, haga clic en el botón "Programa/Test" mientras el módem está conectado a su equipo local (unidad de programación o PC) para transferir los parámetros a un chip de memoria del módem remoto.
6. Haga clic en el botón "Aceptar". Aparecerá el cuadro de diálogo "Configurar la comunicación".

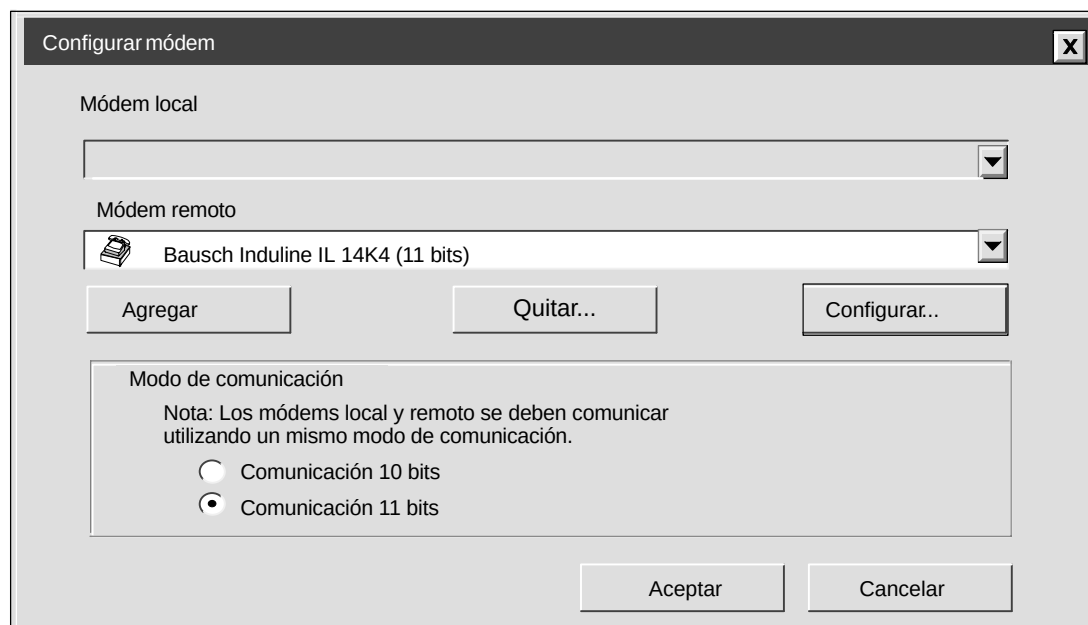


Figura 7-14 Cuadro de diálogo "Configurar módem" para el módem remoto

The screenshot shows a Windows-style dialog box titled "Configurar" (Configure) with a close button (X) in the top right corner. The main title bar also displays a modem icon and the text "Bausch Induline IL 14K4 (11 bits)".

Inside the dialog, there are three text input fields:

- Cadena de inicialización** (Initialization string): Contains the text "AT&F08K0X3&D0\N0".
- Cadena de comunicación** (Communication string): Contains the text "*W=9600,8,N,1".
- Sufijo** (Suffix): Contains the text "&Y0&W0^M".

At the bottom of the dialog, there are five buttons:

- Estado** (Status): A button with a small text box next to it.
- Programa/Test** (Program/Test): A button.
- Extendido...** (Extended...): A button.
- Aceptar** (Accept): A button.
- Cancelar** (Cancel): A button.

Figura 7-15 Configuración del módem remoto

7. Desconecte el módem remoto de su equipo local (unidad de programación o PC).
8. Conecte el módem remoto a su sistema de automatización S7-200.
9. Conecte el módem local a su unidad de programación o PC.

Conectar los módems:

1. Para conectar el módem, haga doble clic en el icono "Conectar módem" en el cuadro de diálogo "Configurar la comunicación". Aparecerá el cuadro de diálogo "Marcar" (v. fig. 7-16).
2. En el cuadro de diálogo "Marcar", introduzca el número de teléfono en el cuadro correspondiente.
3. Para conectar el módem local al módem remoto, haga clic en el botón "Conectar".

Así se finaliza la configuración de los módems.

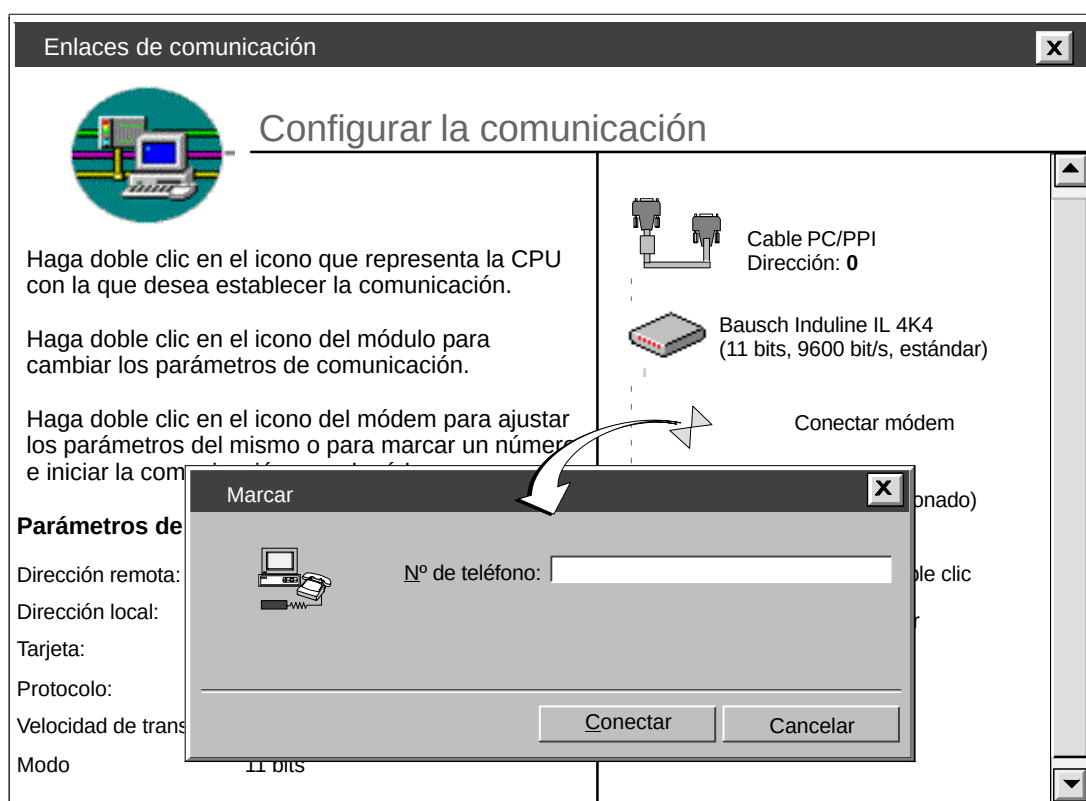


Figura 7-16 Conectar los módems

Utilizar un módem de 10 bits para conectar una CPU S7-200 a un maestro STEP 7-Micro/WIN 32

Utilizando STEP 7-Micro/WIN 32 en un PC con Windows 95, Windows 98 o Windows NT, o usando una unidad de programación SIMATIC (p.ej. la PG 740) como maestro único es posible conectar sólo una CPU S7-200. Se puede utilizar un módem de 10 bits compatible con Hayes para comunicarse con una sola CPU S7-200 remota. La figura 7-17 muestra la comunicación de datos S7-200 utilizando un módem de 10 bits con un cable PC/PPI de 5 interruptores DIP.

Para ello se necesitan los siguientes equipos:

- Una sola CPU S7-200 esclava. Las CPUs 221, 222, 224 y 226 soportan el formato de 10 bits. Los modelos anteriores de CPUs S7-200 no soportan este formato.
- Un cable RS-232 para conectar el PC o la unidad de programación SIMATIC a un módem local full-dúplex de 10 bits.
- Un cable PC/PPI de 5 interruptores DIP (ajustado a la velocidad de transferencia correcta, modo de comunicación de 10 bits y modo DTE) para conectar el módem remoto a la CPU.
- Un adaptador opcional de 9 a 25 pines (si fuera necesario).

Nota

El cable PC/PPI de 4 interruptores DIP no soporta el formato de 10 bits.

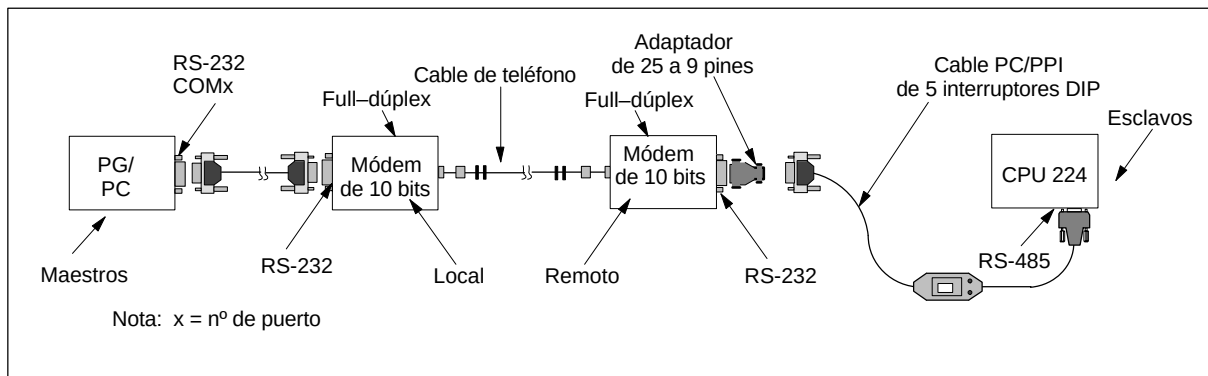


Figura 7-17 Comunicación de datos S7-200 utilizando un módem de 10 bits con un cable PC/PPI de 5 interruptores DIP

Esta configuración sólo permite utilizar un maestro y un esclavo. El sistema de automatización S7-200 exige aquí un bit de inicio, ocho bits de datos, ningún bit de paridad y un bit de parada, comunicación asíncrona y una velocidad de transferencia de 9600/19.200 bit/s. El módem exige los ajustes que figuran en la tabla 7-3. La figura 7-18 muestra la asignación de pines para un adaptador de 25 a 9 pines.

Tabla 7-3 Ajustes necesarios para un módem de 10 bits

Módem	Formato de datos en bits	Velocidad de transferencia entre el módem y el PC	Velocidad de transferencia en el cable	Demás propiedades
10 bits	8 bits de datos	9600 bit/s 19.200 bit/s	9600 bit/s 19.200 bit/s	Ignorar señal DTR
	1 bit de inicio			Sin flujo de control de hardware
	1 bit de parada			Sin flujo de control de software
	sin paridad			

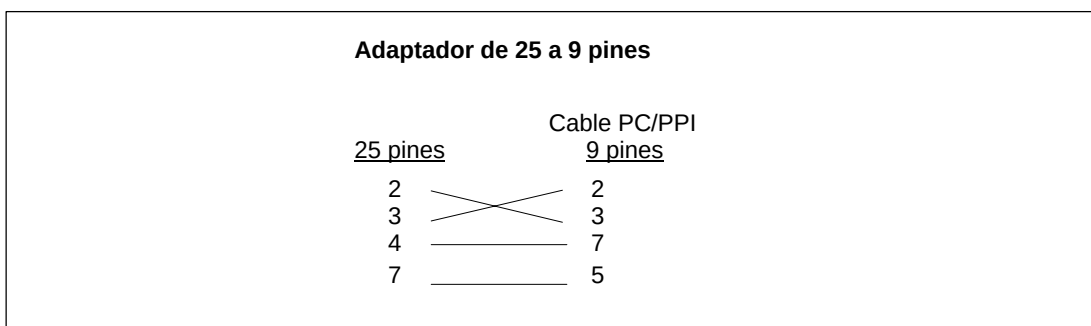


Figura 7-18 Asignación de pines para un adaptador de 25 a 9 pines

Utilizar un módem de 11 bits para conectar una CPU S7-200 a un maestro STEP 7-Micro/WIN 32

Utilizando STEP 7-Micro/WIN 32 en un PC con Windows 95, Windows 98 o Windows NT, o usando una unidad de programación SIMATIC (p.ej. la PG 740) como maestro único es posible conectar una o más CPUs S7-200 (esclavos). La mayoría de los módems no soportan el protocolo de 11 bits.

Dependiendo de si desea conectar sólo una CPU S7-200 o una red de CPUs (v. fig. 7-19), necesitará los siguientes componentes:

- Un cable RS-232 estándar para conectar el PC o la unidad de programación SIMATIC a un módem local full-dúplex de 11 bits.
- Uno de los siguientes cables PC/PPI:
 - Un cable PC/PPI de 5 interruptores DIP (ajustado a la velocidad de transferencia correcta, modo de comunicación de 11 bits y modo DTE) para conectar el módem remoto a la CPU.
 - Un cable PC/PPI de 4 interruptores DIP (ajustado a la velocidad de transferencia correcta) y un adaptador de módem nulo para conectar el módem remoto a la CPU.
- Si hay varias CPUs conectadas al módem remoto se necesitará un conector de puerto de programación Siemens en una red PROFIBUS (la figura 7-24 muestra cómo polarizar y cerrar los cables de interconexión).

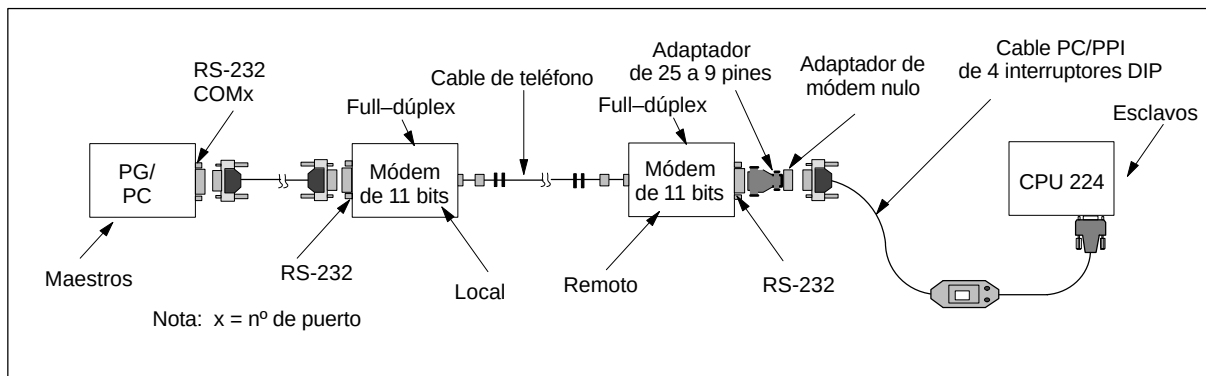


Figura 7-19 Comunicación de datos S7-200 utilizando un módem de 11 bits con un cable PC/PPI de 4 interruptores DIP

Esta configuración sólo permite utilizar un maestro y soporta únicamente el protocolo PPI. Para poder comunicarse por el interface PPI, la CPU S7-200 exige que el módem utilice una cadena de datos de 11 bits. El sistema de automatización S7-200 exige aquí un bit de inicio, ocho bits de datos, un bit de paridad par y un bit de parada, comunicación asíncrona y una velocidad de transferencia de 9600/19.200 bit/s. Numerosos módems no soportan este formato de datos. El módem exige los ajustes que figuran en la tabla 7-4.

La figura 7-20 muestra la asignación de pines para un adaptador de módem nulo y para un adaptador de 25 a 9 pines.

Tabla 7-4 Ajustes necesarios para un módem de 11 bits

Módem	Formato de datos en bits	Velocidad de transferencia entre el módem y el PC	Velocidad de transferencia en el cable	Demás propiedades
11 bits	8 bits de datos	9600 bit/s 19.200 bit/s	9600 bit/s 19.200 bit/s	Ignorar señal DTR
	1 bit de inicio			Sin flujo de control de hardware
	1 bit de parada			Sin flujo de control de software
	1 bit de paridad (par)			

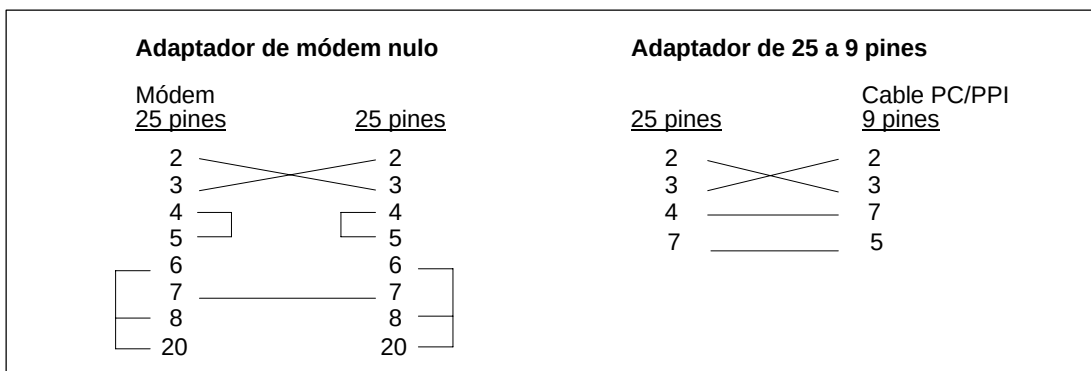


Figura 7-20 Asignación de pines para un adaptador de módem nulo y para un adaptador de 25 a 9 pines

7.5 Redes y protocolos

Maestros

La figura 7-21 muestra una configuración con un PC conectado a varias CPUs S7-200. STEP 7-Micro/WIN 32 se ha diseñado para comunicarse con una sola CPU S7-200. No obstante, se puede acceder a cualquier CPU que intervenga en la red. Las CPUs de la figura 7-21 podrían actuar de esclavas o de maestras. El TD 200 es una unidad maestra.

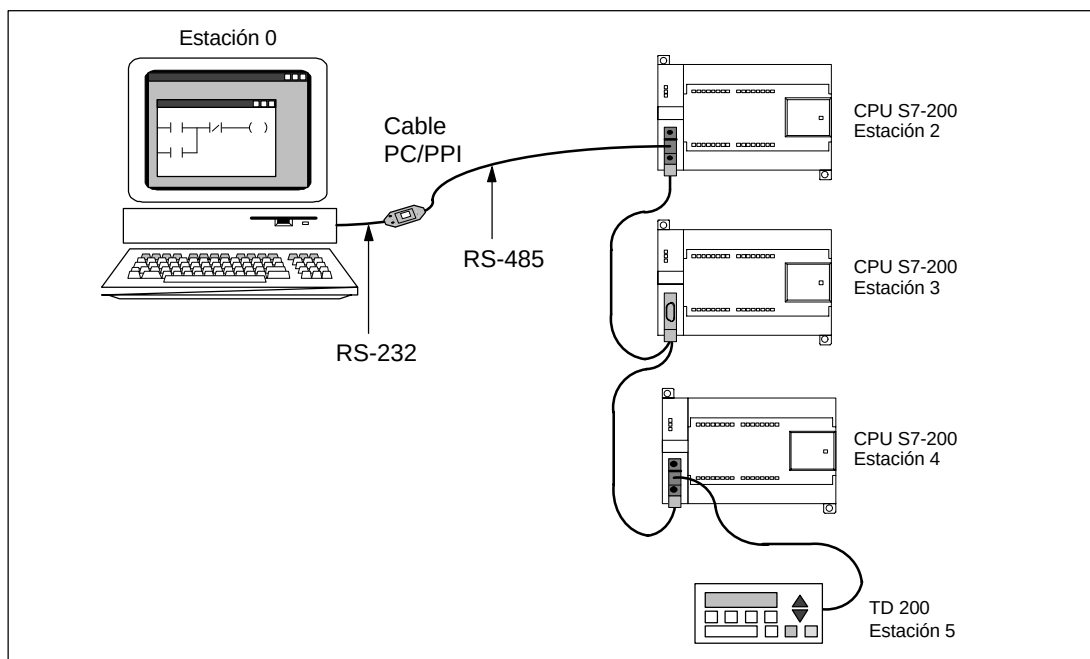


Figura 7-21 Utilizar un cable PC/PPI para la comunicación con varias CPUs S7-200 estando habilitada la opción multimaestro

La figura 7-22 muestra una red más general que incluye varios maestros. Utilizando el módulo EM 277 PROFIBUS-DP es posible obtener velocidades de transferencia más elevadas y un mayor número de enlaces.

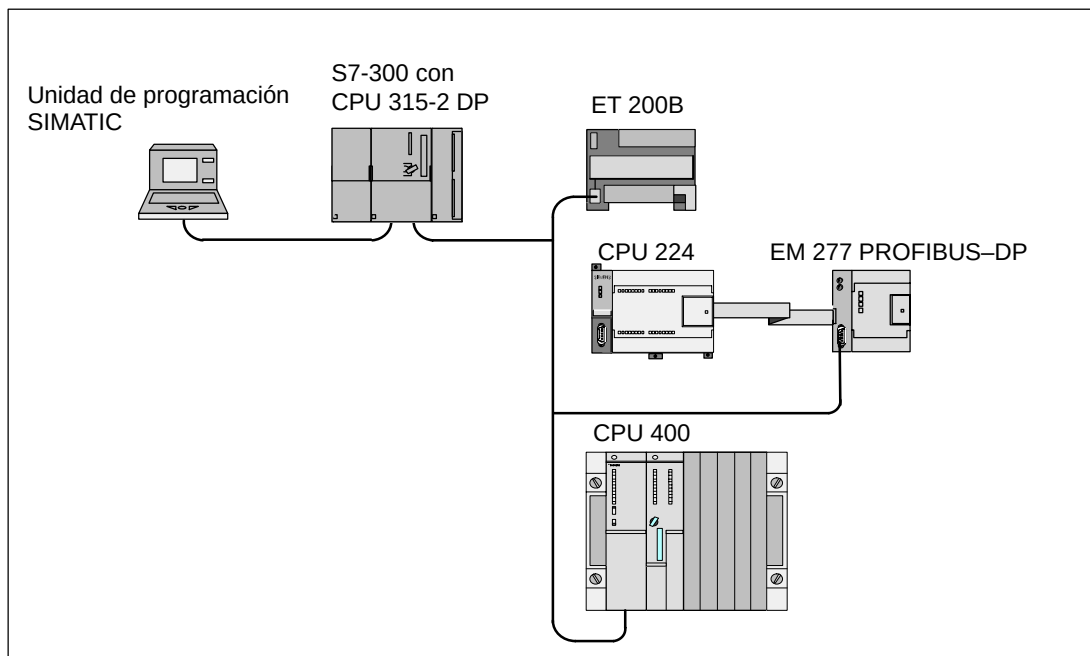


Figura 7-22 Módulo EM 277 PROFIBUS-DP y CPU 224 en una red PROFIBUS

Protocolos de comunicación

Las CPUs S7-200 soportan diversos métodos de comunicación. Dependiendo de la CPU S7-200 utilizada, la red puede asistir uno o varios de los siguientes protocolos de comunicación:

- Interface punto a punto (PPI)
- Interface multipunto (MPI)
- PROFIBUS

Estos protocolos se basan en la intercomunicación de sistemas abiertos (OSI) de la arquitectura de siete capas. Los protocolos PPI y MPI se implementan en una red "token ring" (red de anillo con testigo) conforme al estándar Process Field Bus (PROFIBUS) que se describe en la norma europea EN 50170.

Se trata de protocolos asíncronos de caracteres que utilizan un bit de inicio, ocho bits de datos, un bit de paridad par y un bit de parada. Los bloques de comunicación dependen de los caracteres especiales de inicio y de parada, de las direcciones de estación de fuente y de destino, de la longitud de los bloques y de la suma de verificación para garantizar la integridad de los datos. Los tres protocolos se pueden utilizar simultáneamente en una red sin que interfieran entre sí, con la condición de que usen una misma velocidad de transferencia.

La red PROFIBUS utiliza el estándar RS-485 con cables de par trenzado. Ello permite interconectar hasta 32 dispositivos en un segmento de la red. Los segmentos pueden tener una longitud máxima de 1.200 m, dependiendo de la velocidad de transferencia. Es posible conectar repetidores para poder incorporar más dispositivos en la red o con objeto de utilizar cables más largos. Las redes pueden tener una longitud de hasta 9.600 m, utilizando un total de hasta 9 repetidores (dependiendo de la velocidad de transferencia) (v. tabla 7-7). Para obtener más información sobre el módulo EM 277 PROFIBUS-DP, consulte los datos técnicos en el Anexo A.

Los protocolos prevén dos tipos de dispositivos de red: los maestros y los esclavos. Los maestros pueden enviar una petición a otros dispositivos. En cambio, los esclavos sólo pueden responder a las peticiones de los maestros, sin poder lanzar nunca una petición por su propia cuenta.

Los protocolos soportan 127 direcciones (0 a 126) en una red. Una red puede comprender 32 maestros como máximo. Todos los dispositivos que formen parte de una red deberán tener direcciones unívocas para poder comunicarse entre sí. El ajuste estándar para las unidades de programación SIMATIC y los PCs con STEP 7-Micro/WIN 32 es la dirección "0". Los visualizadores de textos TD 200 y los paneles de operador OP3 y OP7 tienen la dirección predeterminada "1". La dirección estándar de los sistemas de automatización es "2".

Protocolo PPI

PPI es un protocolo maestro/esclavo. Los maestros (otras CPUs, unidades de programación SIMATIC o visualizadores de textos TD 200) envían peticiones a los esclavos y éstos últimos responden. Los esclavos no inician mensajes, sino que esperan a que un maestro les envíe una petición o solicite una respuesta. Todas las CPUs S7-200 actúan de estaciones esclavas en la red.

Estando en modo RUN, algunas CPUs S7-200 pueden actuar de estaciones maestras en la red si se encuentra habilitado el modo maestro PPI en el programa de usuario. (Consulte la descripción de SMB30 en el Anexo C). Una vez habilitado el modo maestro PPI, se podrán enviar mensajes a otras CPUs, usando las operaciones Leer de la red (NETR) y Escribir en la red (NETW). En el apartado 9.15 (Operaciones de comunicación SIMATIC) encontrará una descripción de dichas operaciones. Mientras actúa de estación maestra PPI, la CPU S7-200 sigue respondiendo en calidad de esclava a las peticiones de otros maestros.

El protocolo PPI no limita la cantidad de maestros que pueden comunicarse con una CPU cualquiera que actúe de esclava, pero la red no puede comprender más de 32 maestros.

Protocolo MPI

MPI puede ser un protocolo maestro/maestro, o bien maestro/esclavo. El funcionamiento de este protocolo depende de los equipos utilizados. Si el dispositivo de destino es una CPU S7-300, se establece un enlace maestro/maestro, puesto que todas las CPUs S7-300 son estaciones maestras en la red. Si es una CPU S7-200, se establece un enlace maestro/esclavo, ya que las CPUs S7-200 son unidades esclavas.

El protocolo MPI crea siempre un enlace entre los dos dispositivos intercomunicados. Un enlace es una conexión privada entre los dos dispositivos. Ningún otro maestro puede interferir en un enlace establecido entre dos dispositivos. Un maestro puede establecer un enlace para utilizarlo durante un tiempo breve o indefinido.

Puesto que los enlaces son conexiones privadas entre los dispositivos y utilizan recursos de la CPU, cada CPU puede soportar sólo una cantidad limitada de enlaces. Todas las CPUs S7-200 soportan 4 enlaces en cada uno de los puertos (0 y 1), en tanto que el módulo EM 277 asiste 6 enlaces. Tanto el módulo EM 277 como la CPU S7-200 reservan dos de sus enlaces; uno para una unidad de programación SIMATIC o un PC y el otro para paneles de operador. Los enlaces reservados permiten conectar a la CPU S7-200 o al módulo EM 277 PROFIBUS-DP como mínimo una unidad de programación o un PC, y por lo menos un panel de operador. Los enlaces reservados no pueden ser utilizados por otros maestros (p.ej. CPUs).

Las CPUs S7-300 y S7-400 se pueden comunicar con las CPUs S7-200 estableciendo una conexión a través de los enlaces no reservados de la CPU S7-200 o del módulo EM 277 PROFIBUS-DP. Las CPUs S7-300 y S7-400 pueden leer y escribir datos en las CPUs S7-200, utilizando las operaciones XGET y XPUT (consulte el manual de programación de la CPU S7-300 ó S7-400, respectivamente).

Protocolo PROFIBUS

El protocolo PROFIBUS se ha diseñado para la comunicación rápida con unidades periféricas descentralizadas (E/S remotas). Hay numerosos dispositivos PROFIBUS ofrecidos por diversos fabricantes. Estos dispositivos abarcan desde módulos sencillos de entradas o salidas hasta controladores de motores y sistemas de automatización (autómatas programables).

Por lo general, las redes PROFIBUS tienen un maestro y varios esclavos. La configuración del maestro permite detectar los tipos de esclavos que están conectados, así como sus respectivas direcciones. El maestro inicializa la red y verifica si los esclavos coinciden con la configuración. Continuamente, el maestro escribe los datos de salida en los esclavos y lee de allí los datos de entrada. Una vez que un maestro DP haya configurado correctamente a un esclavo, éste último le pertenecerá. Si hay otro maestro en la red, tendrá apenas un acceso muy limitado a los esclavos del primer maestro.

Para obtener más información sobre el módulo EM 277 PROFIBUS-DP, consulte los datos técnicos en el Anexo A.

Protocolos definidos por el usuario (Freeport)

La comunicación Freeport es un modo de operación que permite al programa de usuario controlar el puerto de comunicación de la CPU S7-200. Con el modo Freeport se pueden implementar protocolos de comunicación definidos por el usuario para crear enlaces con numerosos dispositivos inteligentes.

El programa de usuario controla el funcionamiento del puerto de comunicación utilizando interrupciones de recepción y de transmisión, así como las operaciones Transmitir mensaje (XMT) y Recibir mensaje (RCV). En modo Freeport, el programa de usuario controla por completo el protocolo de comunicación. El modo Freeport se habilita con la marca SMB30 (puerto 0), estando activo únicamente cuando la CPU está en modo RUN. Cuando la CPU retorna a modo STOP, la comunicación Freeport se detiene y el puerto de comunicación vuelve a utilizar el protocolo PPI normal. En el apartado 9.15 (Operaciones de comunicación SIMATIC) encontrará una descripción de las operaciones Transmitir mensaje y Recibir mensaje.

7.6 Componentes de redes

Un sistema de automatización S7-200 se puede conectar a través del puerto de comunicación a un bus de red. A continuación se describen el puerto, los conectores para el bus, el cable de conexión y los repetidores utilizados para ampliar la red.

Puerto de comunicación

Los puertos de comunicación de las CPUs S7-200 son compatibles con el estándar RS-485 mediante un conector D subminiatura de 9 pines conforme al estándar PROFIBUS definido en la norma europea EN 50170. La figura 7-23 muestra el conector que ofrece el enlace físico para el puerto de comunicación y en la tabla 7-5 figuran las asignaciones de pines para los puertos de comunicación. Consulte el Anexo A para obtener más información sobre el módulo EM 277 PROFIBUS-DP.

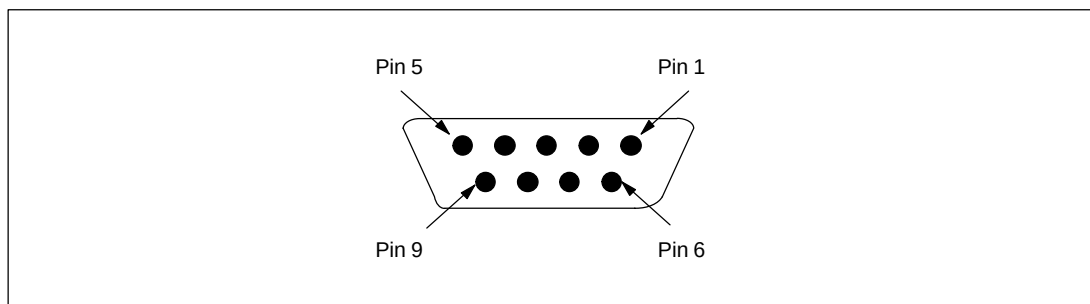


Figura 7-23 Pines del puerto de comunicación de la CPU S7-200

Tabla 7-5 Asignación de pines del puerto de comunicación de la CPU S7-200

Nº de pin	Denominación PROFIBUS	Puerto 0/Puerto 1
1	Blindaje	Tierra
2	Hilo de retorno 24 V	Hilo lógico
3	Señal B RS-485	Señal B RS-485
4	Petición de transmitir	RTS (TTL)
5	Hilo de retorno 5 V	Hilo lógico
6	+5 V	+5 V, 100 Ω resistor en serie
7	+24 V	+24 V
8	Señal A RS-485	Señal A RS-485
9	No aplicable	Selección protocolo de 10 bits (entrada)
Carcasa del enchufe	Blindaje	Tierra

Conectores de bus

Siemens ofrece dos tipos de conectores de bus que permiten conectar fácilmente varios dispositivos a una red. Ambos conectores poseen dos juegos de tornillos para fijar los cables de entrada y salida. Asimismo, disponen de interruptores para polarizar y cerrar la red de forma selectiva. Uno de ellos ofrece sólo un enlace a la CPU, en tanto que el otro agrega un puerto de programación (v. fig. 7-24). En el Anexo E se indican los números de referencia.

El conector que provee un puerto de programación permite añadir a la red una unidad de programación SIMATIC o un panel de operador, sin perturbar ningún enlace existente. Este conector transmite todas las señales de la CPU a través del puerto de programación, adecuándose para conectar dispositivos alimentados por la CPU (p.ej. un TD 200 o un OP3). Los pines de alimentación del conector del puerto de comunicación se pasan por el puerto de programación.



Cuidado

En caso de interconectar equipos con potenciales de referencia diferentes pueden circular corrientes indeseadas por el cable de conexión.

Las corrientes indeseadas pueden causar errores de comunicación o deteriorar los equipos.

Para evitar corrientes indeseadas, asegúrese de que todos los equipos que se deban conectar con un cable de comunicación compartan un circuito de referencia, o bien estén aislados unos de otros. Para obtener más información al respecto, consulte el tema “Reglas de puesta a tierra de referencia de potencial para circuitos aislados” en el apartado 2.3.

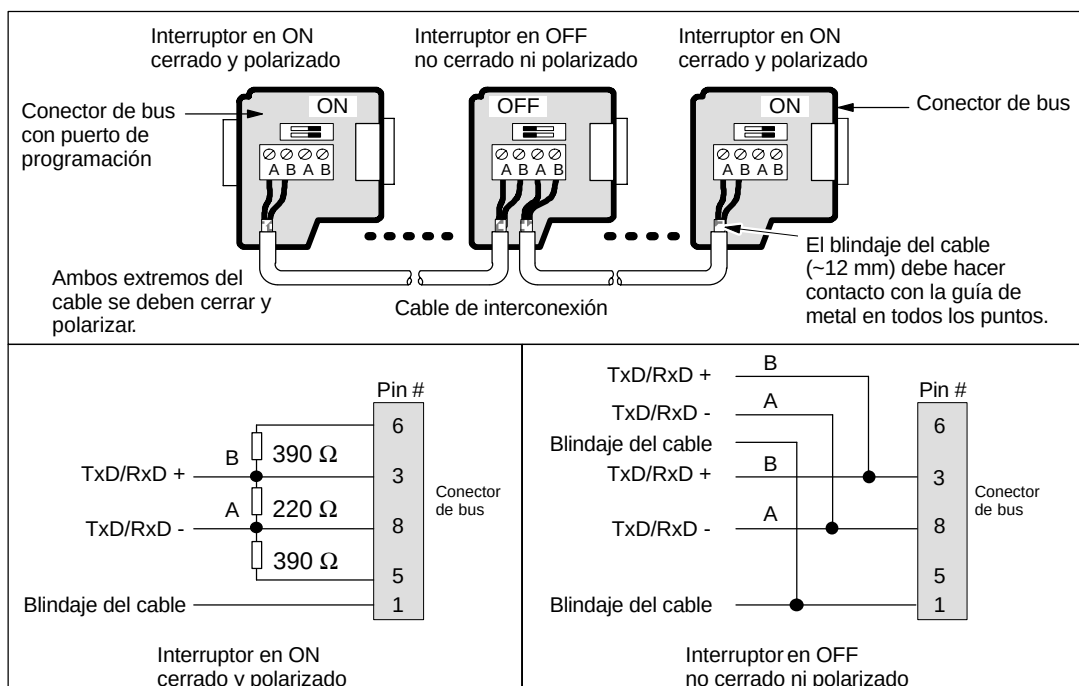


Figura 7-24 Polarizar y cerrar el cable de interconexión

Cable para una red PROFIBUS

En la tabla 7-6 figuran los datos técnicos generales de un cable para una red PROFIBUS. En el Anexo E se indica la referencia de Siemens de cables PROFIBUS que cumplan los requisitos indicados.

Tabla 7-6 Datos técnicos generales de un cable para una red PROFIBUS

Características generales	Datos técnicos
Tipo	Apantallado, con par trenzado
Sección transversal del cable	24 AWG (0,22 mm ²) o superior
Capacidad del cable	< 60 pF/m
Impedancia nominal	100 Ω a 120 Ω

La longitud máxima de un segmento de red PROFIBUS depende de la velocidad de transferencia y del tipo de cable utilizados. En la tabla 7-7 figuran las longitudes máximas de los segmentos para el cable indicado en la tabla 7-6.

Tabla 7-7 Longitud máxima del cable en un segmento de una red PROFIBUS

Velocidad de transferencia	Longitud máxima del cable en un segmento
9,6 kbit/s a 93,75 kbit/s	1.200 m
187,5 kbit/s	1.000 m
500 kbit/s	400 m
1 Mbit/s a 1,5 Mbit/s	200 m
3 Mbit/s a 12 Mbit/s	100 m

Repetidores

Siemens ofrece repetidores para interconectar segmentos de redes PROFIBUS (v. fig. 7-25). Utilizando repetidores es posible ampliar la longitud total de la red, añadir dispositivos a la misma y/o aislar diferentes segmentos de la red. El protocolo PROFIBUS soporta máximo 32 dispositivos en un segmento de red de hasta 1.200 m a una velocidad de transferencia de 9.600 bit/s. Cada repetidor permite añadir 32 dispositivos adicionales a la red y así ampliarla 1.200 m con una velocidad de transferencia de 9.600 bit/s. En una red se pueden utilizar 9 repetidores como máximo, pero la longitud total de la red no puede exceder 9.600 metros. Cada repetidor permite polarizar y cerrar el segmento de red en cuestión. En el Anexo E se indican los números de referencia.

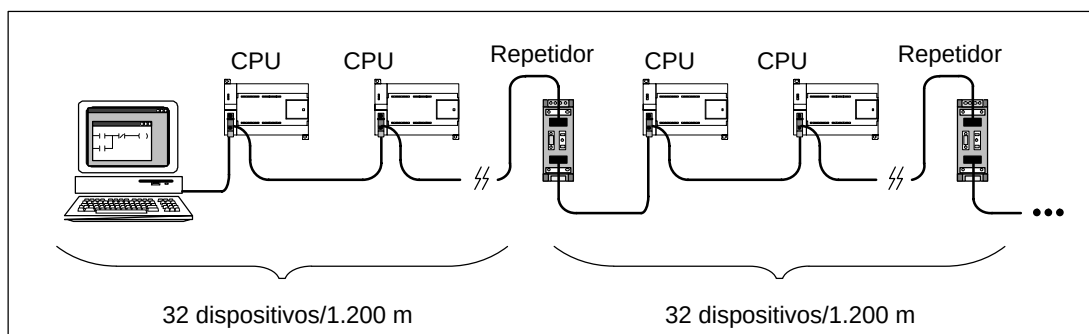


Figura 7-25 Red con repetidores

7.7 Utilizar el cable PC/PPI con otros dispositivos y en modo Freeport

El cable PC/PPI y el modo Freeport se pueden utilizar para conectar las CPUs S7-200 a numerosos dispositivos compatibles con el estándar RS-232.

Hay dos tipos de cables PC/PPI:

- Un cable PC/PPI aislado con un puerto RS-232 que tiene 5 interruptores DIP para ajustar la velocidad de transferencia y configurar otros parámetros más (v. fig. 7-27). Los datos técnicos del cable PC/PPI se indican en el Anexo A.
- Un cable PC/PPI no aislado con un puerto RS-232 que tiene 4 interruptores DIP para ajustar la velocidad de transferencia. Los datos técnicos del cable PC/PPI no aislado se indican en la versión anterior del *Manual del sistema de automatización S7-200* (referencia: 6ES7-298-8FA01-8BH0). Ya no se dispone del cable PC/PPI no aislado, habiéndose sustituido por el cable PC/PPI aislado.

Ambos cables PC/PPI soportan velocidades de transferencia comprendidas entre 600 bit/s y 38.400 bit/s. Utilice los interruptores DIP dispuestos en la carcasa del cable PC/PPI para configurar la velocidad de transferencia correcta. La tabla 7-8 muestra las velocidades de transferencia y las posiciones de los interruptores DIP.

Tabla 7-8 Posición de los interruptores DIP en el cable PC/PPI para seleccionar la velocidad de transferencia

Velocidad de transferencia (en bit/s)	Interruptor DIP (1 = arriba)
38400	000
19200	001
9600	010
4800	011
2400	100
1200	101
600	110

El cable PC/PPI se encuentra en modo de transmisión cuando los datos se envían del puerto RS-232 al RS-485. En cambio, se encuentra en modo de recepción al estar inactivo, o bien cuando los datos se transmiten del puerto RS-485 al RS-232. El cable cambia inmediatamente de modo de recepción a transmisión cuando detecta caracteres en el canal de transmisión del RS-232. El cable cambia nuevamente a modo de recepción cuando el canal de transmisión del RS-232 está inactivo durante el tiempo de inversión del cable. Este tiempo depende de la velocidad de transferencia seleccionada con los interruptores DIP del cable (v. tabla 7-9).

Si el cable PC/PPI se utiliza en un sistema que use también el modo Freeport, el tiempo de inversión se deberá tener en cuenta en el programa de usuario de la CPU S7-200 en las siguientes situaciones:

- La CPU S7-200 responde a los mensajes que envía el dispositivo RS-232.

Tras recibir una petición del dispositivo RS-232, la transmisión de una respuesta de la CPU S7-200 se deberá retardar por un período mayor o igual al tiempo de inversión del cable.

- El dispositivo RS-232 responde a los mensajes que envía la CPU S7-200.

Tras recibir una respuesta del dispositivo RS-232, la transmisión de la siguiente petición de la CPU S7-200 se deberá retardar por un período mayor o igual al tiempo de inversión del cable.

En ambos casos, el tiempo de retardo es suficiente para que el cable PC/PPI pueda cambiar de modo de transmisión a modo de recepción, enviando entonces los datos del puerto RS-485 al RS-232.

Tabla 7-9 Tiempo de inversión del cable PC/PPI (cambio de transmisión a recepción)

Velocidad de transferencia (en bit/s)	Tiempo de inversión (en milisegundos)
38400	0.5
19200	1
9600	2
4800	4
2400	7
1200	14
600	28

Utilizar un módem con un cable PC/PPI de 5 interruptores

El cable PC/PPI de 5 interruptores DIP se puede utilizar para conectar el puerto de comunicación RS-232 de un módem a una CPU S7-200. Por lo general, los módems utilizan las señales de control RS-232 (tales como RTS, CTS y DTR) para que un PC pueda controlar el módem. El cable PC/PPI no vigila ninguna de estas señales pero ofrece RTS en modo DTE. Si se utiliza un módem con un cable PC/PPI, el módem se deberá configurar para que funcione sin estas señales. Como mínimo, el módem se debe configurar de manera que ignore la señal DTR. Consulte el manual del módem para determinar los comandos necesarios para configurarlo.

Para el puerto RS-232 del cable PC/PPI de 5 interruptores se puede ajustar el modo DCE (equipo de comunicación de datos), o bien el modo DTE (equipo terminal de datos). Las únicas señales presentes en el puerto son las de transmitir datos, petición de transmitir, recibir datos y tierra. El cable PC/PPI de 5 interruptores no usa ni emite la señal CTS (preparado para transmitir). En las tablas 7-10 y 7-11 se indica la asignación de los pines del cable PC/PPI.

Un módem es un equipo de comunicación de datos (DCE). Al conectar un cable PC/PPI a un módem, el puerto RS-232 del cable PC/PPI se deberá configurar en modo DTE (equipo terminal de datos), conforme a la selección efectuada con el 5º interruptor DIP del cable. Así se evita la necesidad de utilizar un adaptador de módem nulo entre el cable PC/PPI y el módem. Sin embargo, puede precisarse un adaptador de 9 a 25 pines (dependiendo del conector del módem). La figura 7-26 muestra una configuración típica y la asignación de pines de un adaptador de 25 a 9 pines.

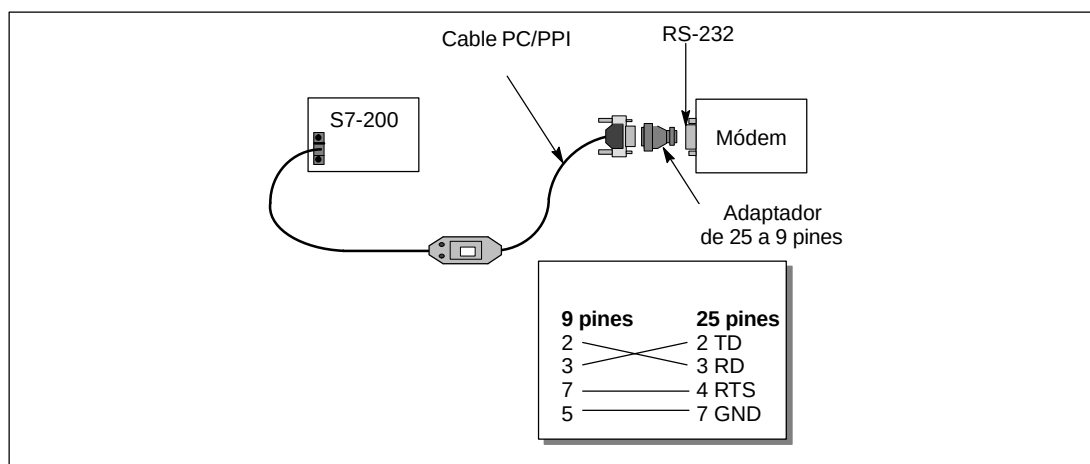


Figura 7-26 Asignación de pines para un cable PC/PPI de 5 interruptores DIP con un módem

Para ajustar el modo DCE (equipo de comunicación de datos), es preciso colocar el 5º interruptor en la posición "0" (= abajo). Para ajustar el modo DTE (equipo terminal de datos), es preciso colocar el 5º interruptor en la posición "1" (= arriba). La tabla 7-10 muestra los números de los pines y las funciones del puerto RS-485 a RS-232 del cable PC/PPI en modo DTE. La tabla 7-11 muestra los números de los pines y las funciones del puerto RS-485 a RS-232 del cable PC/PPI en modo DCE. Hay que tener en cuenta que el cable PC/PPI sólo envía RTS (peticiones de transmitir) si está en modo DTE.

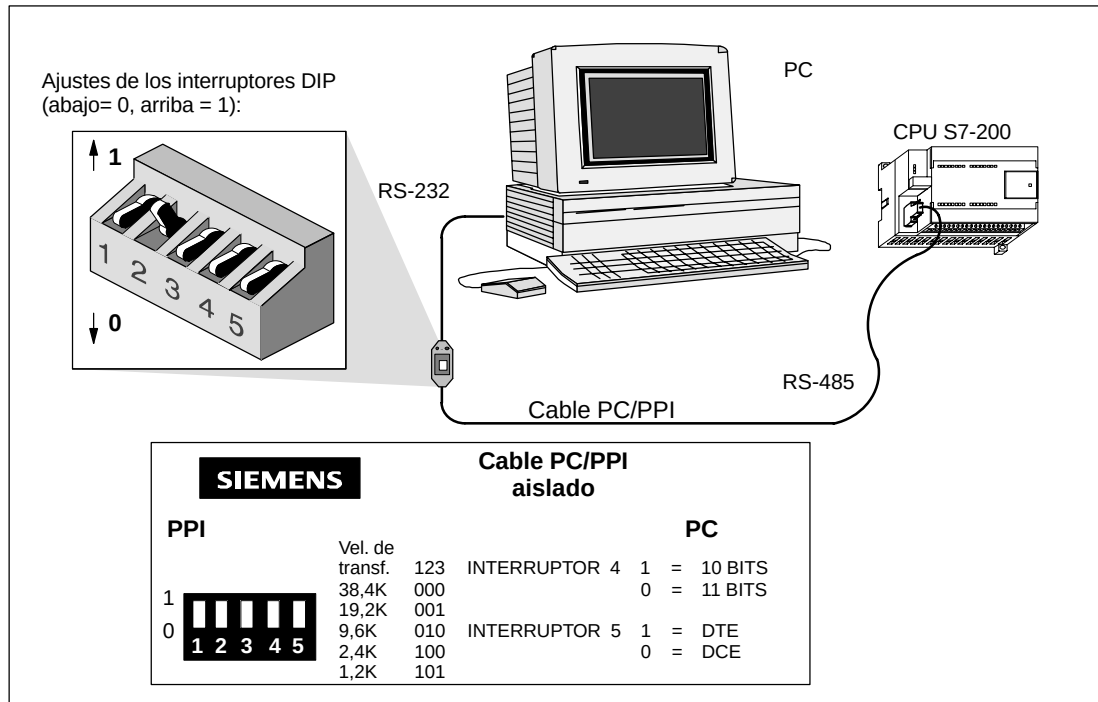


Figura 7-27 Comunicación con una CPU en modo PPI

El 4º interruptor DIP del cable PC/PPI le indica a la CPU S7-200 si debe utilizar el protocolo de 10 bits o el protocolo PPI normal de 11 bits. Este interruptor se usa únicamente cuando la CPU se está comunicando con STEP 7-Micro/WIN. Si STEP 7-Micro/WIN no se utiliza junto con un módem, el 4º interruptor DIP deberá permanecer en el ajuste correspondiente al protocolo de 11 bits para garantizar el funcionamiento correcto con otros equipos.

Tabla 7-10 Asignación de pines para un conector de RS-485 a RS-232 DTE¹

Asignación de pines del conector RS-485		Asignación de pines del conector RS-232 DTE ¹	
Nº de pin	Descripción de la señal	Nº de pin	Descripción de la señal
1	Tierra (RS-485)	1	Data Carrier Detect (DCD) (no utilizado)
2	Hilo de retorno 24 V (tierra RS-485)	2	Receive Data (RD) (entrada al cable PC/PPI)
3	Señal B (Rx/D/TxD+)	3	Transmit Data (TD) (salida del cable PC/PPI)
4	RTS (nivel TTL)	4	Data Terminal Ready (DTR) (no utilizado)
5	Tierra (RS-485)	5	Tierra (RS-232)
6	+5 V (con resistor en serie de 100 Ω)	6	Data Set Ready (DSR) (no utilizado)
7	Alimentación 24 V	7	Request To Send (RTS) (salida del cable PC/PPI)
8	Señal A (Rx/D/TxD-)	8	Clear To Send (CTS) (no utilizado)
9	Selección de protocolo	9	Ring Indicator (RI) (no utilizado)

¹Para los módems se necesita un adaptador de hembra a macho y un adaptador de 9 a 25 pines.

Tabla 7-11 Asignación de pines para un conector de RS-485 a RS-232 DCE

Asignación de pines del conector RS-485		Asignación de pines del conector RS-232 DCE	
Nº de pin	Descripción de la señal	Nº de pin	Descripción de la señal
1	Tierra (RS-485)	1	Data Carrier Detect (DCD) (no utilizado)
2	Hilo de retorno 24 V (tierra RS-485)	2	Receive Data (RD) (salida del cable PC/PPI)
3	Señal B (Rx/D/TxD+)	3	Transmit Data (TD) (entrada al cable PC/PPI)
4	RTS (nivel TTL)	4	Data Terminal Ready (DTR) (no utilizado)
5	Tierra (RS-485)	5	Tierra (RS-232)
6	+5 V (con resistor en serie de 100 Ω)	6	Data Set Ready (DSR) (no utilizado)
7	Alimentación 24 V	7	Request To Send (RTS) (no utilizado)
8	Señal A (Rx/D/TxD-)	8	Clear To Send (CTS) (no utilizado)
9	Selección de protocolo	9	Ring Indicator (RI) (no utilizado)

Utilizar un módem con un cable PC/PPI de 4 interruptores

El cable PC/PPI de 4 interruptores DIP se puede utilizar para conectar el puerto de comunicación RS-232 de un módem a una CPU S7-200. Por lo general, los módems utilizan las señales de control RS-232 (tales como RTS, CTS y DTR) para que un PC pueda controlar el módem. Este cable PC/PPI no utiliza ninguna de estas señales. Por tanto, si un módem se utiliza con un cable PC/PPI de 4 interruptores DIP, el módem se deberá configurar para que no utilice ninguna de estas señales. Como mínimo, se deberán ignorar las señales RTS y DTR. Consulte el manual del módem para determinar los comandos necesarios para configurarlo.

Un módem es un equipo de comunicación de datos (DCE). El puerto RS-232 del cable PC/PPI de 4 interruptores también se considera un DCE. Al conectarse dos dispositivos de una misma clase (ambos DCE), los pines para transmitir y recibir datos se deberán invertir utilizando para ello un adaptador de módem nulo. La figura 7-28 muestra una configuración típica y la asignación de pines de un adaptador de módem nulo.

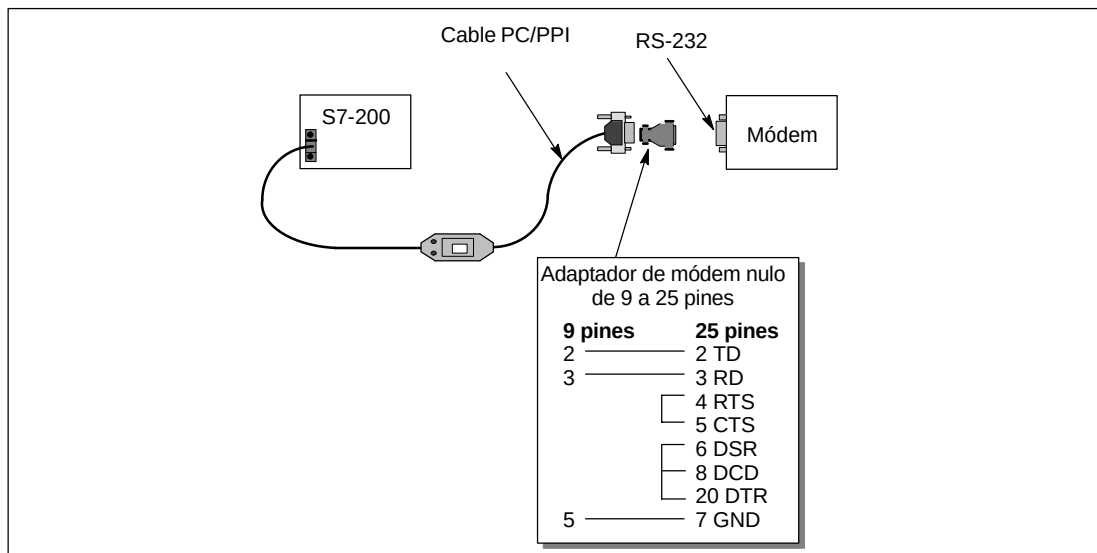


Figura 7-28 Módem de 11 bits con unadaptador de módem nulo combinado con un adaptador de 9 a 25 pines

7.8 Rendimiento de la red

Optimizar el rendimiento de la red

Los dos factores con mayor efecto en el rendimiento de la red son la velocidad de transferencia y el número de maestros. El rendimiento óptimo de la red se logra utilizando la velocidad de transferencia máxima soportada por todos los dispositivos. Si el número de maestros se reduce a un mínimo, aumenta también el rendimiento de la red. Cada maestro de la red prolonga el tiempo de procesamiento en la red. Por tanto, el tiempo se acortará cuanto menor sea el número de maestros.

Los siguientes factores influyen también en el rendimiento de la red:

- Las direcciones elegidas para los maestros y esclavos.
- El factor de actualización GAP.
- La dirección de estación más alta.

Las direcciones de los maestros se deberán elegir de forma secuencial, evitando huecos entre las mismas. Si hay un hueco (GAP) entre las direcciones de los maestros, éstos comprueban continuamente las direcciones del GAP para averiguar si hay otro maestro que desee conectarse online. Esta comprobación aumenta el tiempo de procesamiento de la red. Si no existe un hueco entre las direcciones de los maestros, la comprobación no se efectúa, por lo que se minimiza el tiempo de procesamiento.

Las direcciones de los esclavos se pueden ajustar a cualquier valor sin que ello influya en el rendimiento de la red, a menos que los esclavos se encuentren entre los maestros. En este último caso aumentaría también el tiempo de procesamiento de la red como si existieran huecos entre las direcciones de los maestros.

Las CPUs S7-200 se pueden configurar para que comprueben sólo periódicamente si hay huecos entre las direcciones. Para ello, en STEP 7-Micro/WIN 32 se ajusta el factor de actualización GAP cuando se configura el correspondiente puerto de la CPU. El factor de actualización GAP le indica a la CPU la frecuencia con la que debe comprobar el hueco de direcciones para determinar si hay otros maestros. Si se elige "1" como factor de actualización GAP, la CPU comprobará el hueco de direcciones cada vez que tenga el testigo en su poder. Si se elige "2", la CPU comprobará el hueco cada 2 veces que tenga el testigo en su poder. Ajustándose un factor de actualización GAP más elevado se reduce el tiempo de procesamiento en la red si hay huecos entre las direcciones de los maestros. Si no existen huecos, el factor de actualización GAP no tendrá efecto alguno en el rendimiento. Si se ajusta un factor de actualización GAP elevado pueden producirse grandes demoras cuando se desee incorporar nuevos maestros a la red, puesto que las direcciones se comprueban con menor frecuencia. El factor de actualización GAP se utiliza únicamente cuando una CPU actúa de maestro PPI.

La dirección de estación más alta es el valor donde un maestro debe buscar a otro. Ajustándose este valor se limita el hueco de direcciones que el último maestro (la dirección más alta) debe comprobar en la red. Limitando el tamaño del hueco de direcciones se reduce el tiempo necesario para buscar e incorporar en la red a un nuevo maestro. La dirección de estación más alta no tiene efecto sobre las direcciones de los esclavos. Los maestros pueden comunicarse con esclavos cuyas direcciones sean superiores a la dirección de estación más alta. Ésta última se utiliza sólo cuando una CPU actúa de maestro PPI. La dirección de estación más alta se puede ajustar en STEP 7-Micro/WIN 32 al configurar el puerto de la CPU.

Por regla general, se deberá ajustar en todos los maestros un mismo valor para la dirección de estación más alta. Esta dirección debería ser mayor o igual a la dirección más alta de los maestros. El ajuste estándar de la dirección de estación más alta en las CPUs S7-200 es "31".

Rotación del testigo

En una red con "token passing" (paso de testigo), la estación que tiene el testigo en su poder es la única que puede iniciar la comunicación. Por tanto, un importante factor en una red con "token passing" es el tiempo de rotación del testigo. Éste es el tiempo que el testigo necesita para recorrer el anillo lógico, o sea, para circular por todos los maestros ("token holders") que lo constituyen. El ejemplo de la figura 7-29 muestra el funcionamiento de una red multimaestro.

La red de la figura 7-29 comprende cuatro CPUs S7-200, teniendo cada una de ellas su propio TD 200. Dos CPUs 224 recopilan datos de las demás CPUs.

Nota

El ejemplo indicado se basa en la configuración que muestra la figura 7-29. Esta configuración incluye visualizadores de textos TD 200. Las CPUs 224 utilizan operaciones NETR y NETW. Las fórmulas para calcular el tiempo de posesión y de rotación del testigo que muestra la figura 7-30 se basan también en esta configuración.

El software COM PROFIBUS permite analizar el rendimiento de la red.

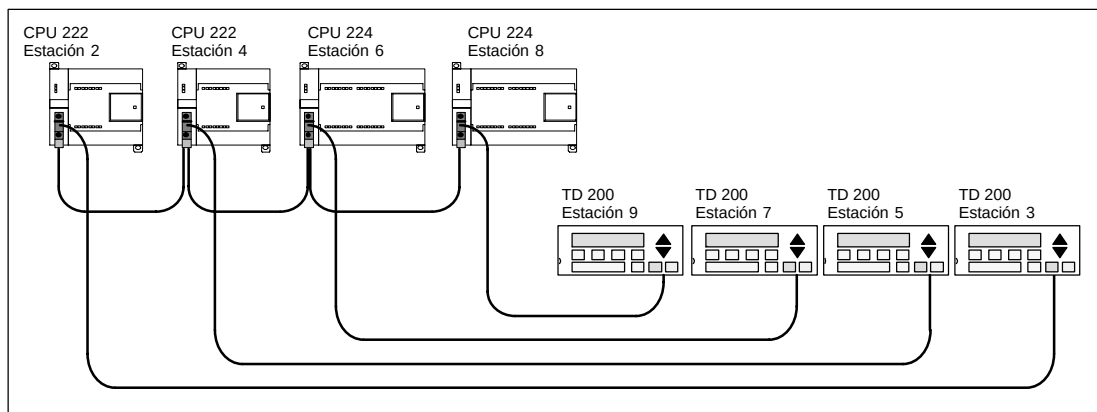


Figura 7-29 Ejemplo de una red con "token passing"

En esta configuración, un TD 200 (estación 3) se comunica con una CPU 222 (estación 2), otro TD 200 (estación 5) se comunica con la otra CPU 222 (estación 4), etc. Además, una CPU 224 (estación 6) envía mensajes a las estaciones 2, 4 y 8, y la otra CPU 224 (estación 8) envía mensajes a las estaciones 2, 4 y 6. Esta red comprende seis estaciones maestras (los cuatro TDs 200 y las dos CPUs 224), así como dos estaciones esclavas (las dos CPUs 222).

Enviar mensajes

Para que un maestro pueda enviar un mensaje deberá tener el testigo en su poder. Por ejemplo: cuando la estación 3 tiene el testigo en su poder, envía una petición a la estación 2 y pasa el testigo a la estación 5. La estación 5 envía una petición a la estación 4 y pasa el testigo a la estación 6. La estación 6 envía un mensaje a las estaciones 2, 4 u 8 y pasa el testigo a la estación 7. Este proceso de enviar un mensaje y pasar el testigo continúa por el anillo lógico de la estación 3 a la estación 5, a la estación 6, a la estación 7, a la estación 8, a la estación 9 y de allí retorna finalmente a la estación 3. El testigo debe recorrer todo el anillo lógico para que un maestro pueda enviar una petición de información. En un anillo lógico compuesto por seis estaciones que envían una petición para leer o escribir un valor de doble palabra (cuatro bytes de datos) cada vez que tienen el testigo en su poder, el tiempo de rotación del mismo será de unos 900 milisegundos a una velocidad de transferencia de 9.600 bit/s. Si aumenta el número de bytes de datos a los que se debe acceder por mensaje o si se incorporan más estaciones, se prolongará el tiempo de rotación del testigo.

Tiempo de rotación del testigo

Éste depende del tiempo que cada estación tiene el testigo en su poder. El tiempo de rotación del testigo en redes S7-200 multimaestro se puede determinar sumando los tiempos de posesión del testigo por parte de cada maestro. Si se ha habilitado el modo maestro PPI (en el protocolo PPI de la red en cuestión), es posible enviar mensajes a otras CPUs utilizando las operaciones Leer de la red (NETR) y Escribir en la red (NETW) con la CPU. En el apartado 9.15 (Operaciones de comunicación SIMATIC) se describen dichas operaciones. Si envía mensajes utilizando las operaciones NETR y NETW, puede utilizar la fórmula que muestra la figura 7-30 para calcular el tiempo aproximado de rotación del testigo, dando por supuesto que:

- Cada estación envía una petición cuando tiene el testigo en su poder.
- La petición es una operación de lectura o de escritura a direcciones consecutivas de datos.
- No hay conflictos de acceso al único búfer de comunicación de la CPU.
- Ninguna CPU tiene un tiempo de ciclo superior a aprox. 10 ms.

Tiempo de posesión del testigo (T_{pos}) = (tiempo necesario 128 + n caract. datos) * 11 bits/caract. * 1/vel. transf.	
Tiempo de rotación del testigo (T_{rot}) = T_{pos} del maestro 1 + T_{pos} del maestro 2 + ... + T_{pos} del maestro m	
siendo n el número de caracteres de datos (bytes) y m el número de maestros	
Conforme al ejemplo indicado arriba, el tiempo de rotación se calcula de la siguiente forma si el tiempo de posesión del testigo es igual en los seis maestros:	
T (tiempo de posesión del testigo)	= (128 + 4 caracteres) * 11 bits/carácter * 1/9.600 "bit times"/s = 151,25 ms/maestro
T (tiempo de rotación del testigo)	= 151,25 ms/maestro * 6 maestros = 907,5 ms
(Un "bit time" equivale a la duración de un período de señal).	

Figura 7-30 Fórmulas para determinar los tiempos de posesión y de rotación del testigo utilizando las operaciones NETR y NETW

Tiempo de rotación del testigo en función del número de estaciones

Las tablas 7-12, 7-13 y 7-14 muestran el tiempo de rotación del testigo en función del número de estaciones y del volumen de datos a transferir a 9,6 kbit/s, 19,2 kbit/s y 187,5 kbit/s, respectivamente. Estos tiempos son válidos utilizando las operaciones Leer de la red (NETR) y Escribir en la red (NETW) con la CPU u otros maestros.

Tabla 7-12 Tiempo de rotación del testigo en función del número de estaciones y del volumen de datos a 9,6 kbit/s

Bytes transferidos por estación a 9,6 kbit/s	Número de estaciones (indicaciones de tiempo en segundos)								
	2 estaciones	3 estaciones	4 estaciones	5 estaciones	6 estaciones	7 estaciones	8 estaciones	9 estaciones	10 estaciones
1	0,30	0,44	0,59	0,74	0,89	1,03	1,18	1,33	1,48
2	0,30	0,45	0,60	0,74	0,89	1,04	1,19	1,34	1,49
3	0,30	0,45	0,60	0,75	0,90	1,05	1,20	1,35	1,50
4	0,30	0,45	0,61	0,76	0,91	1,06	1,21	1,36	1,51
5	0,30	0,46	0,61	0,76	0,91	1,07	1,22	1,37	1,52
6	0,31	0,46	0,61	0,77	0,92	1,07	1,23	1,38	1,54
7	0,31	0,46	0,62	0,77	0,93	1,08	1,24	1,39	1,55
8	0,31	0,47	0,62	0,78	0,94	1,09	1,25	1,40	1,56
9	0,31	0,47	0,63	0,78	0,94	1,10	1,26	1,41	1,57
10	0,32	0,47	0,63	0,79	0,95	1,11	1,27	1,42	1,58
11	0,32	0,48	0,64	0,80	0,96	1,11	1,27	1,43	1,59
12	0,32	0,48	0,64	0,80	0,96	1,12	1,28	1,44	1,60
13	0,32	0,48	0,65	0,81	0,97	1,13	1,29	1,45	1,62
14	0,33	0,49	0,65	0,81	0,98	1,14	1,30	1,46	1,63
15	0,33	0,49	0,66	0,82	0,98	1,15	1,31	1,47	1,64
16	0,33	0,50	0,66	0,83	0,99	1,16	1,32	1,49	1,65

Tabla 7-13 Tiempo de rotación del testigo en función del número de estaciones y del volumen de datos a 19,2 kbit/s

Bytes transferidos por estación a 19,2 kbit/s	Número de estaciones (indicaciones de tiempo en segundos)								
	2 estaciones	3 estaciones	4 estaciones	5 estaciones	6 estaciones	7 estaciones	8 estaciones	9 estaciones	10 estaciones
1	0,15	0,22	0,30	0,37	0,44	0,52	0,59	0,67	0,74
2	0,15	0,22	0,30	0,37	0,45	0,52	0,60	0,67	0,74
3	0,15	0,23	0,30	0,38	0,45	0,53	0,60	0,68	0,75
4	0,15	0,23	0,30	0,38	0,45	0,53	0,61	0,68	0,76
5	0,15	0,23	0,30	0,38	0,46	0,53	0,61	0,69	0,76
6	0,15	0,23	0,31	0,38	0,46	0,54	0,61	0,69	0,77
7	0,15	0,23	0,31	0,39	0,46	0,54	0,62	0,70	0,77
8	0,16	0,23	0,31	0,39	0,47	0,55	0,62	0,70	0,78
9	0,16	0,24	0,31	0,39	0,47	0,55	0,63	0,71	0,78
10	0,16	0,24	0,32	0,40	0,47	0,55	0,63	0,71	0,79
11	0,16	0,24	0,32	0,40	0,48	0,56	0,64	0,72	0,80
12	0,16	0,24	0,32	0,40	0,48	0,56	0,64	0,72	0,80
13	0,16	0,24	0,32	0,40	0,48	0,57	0,65	0,73	0,81
14	0,16	0,24	0,33	0,41	0,49	0,57	0,65	0,73	0,81
15	0,16	0,25	0,33	0,41	0,49	0,57	0,66	0,74	0,82
16	0,17	0,25	0,33	0,41	0,50	0,58	0,66	0,74	0,83

Tabla 7-14 Tiempo de rotación del testigo en función del número de estaciones y del volumen de datos a 187,5 kbit/s

Bytes transferidos por estación a 187,5 kbit/s	Número de estaciones (indicaciones de tiempo en milisegundos)								
	2 estaciones	3 estaciones	4 estaciones	5 estaciones	6 estaciones	7 estaciones	8 estaciones	9 estaciones	10 estaciones
1	8,68	13,02	17,37	21,71	26,05	30,39	34,73	39,07	43,41
2	8,80	13,20	17,60	22,00	26,40	30,80	35,20	39,60	44,00
3	8,92	13,38	17,83	22,29	26,75	31,21	35,67	40,13	44,59
4	9,03	13,55	18,07	22,59	27,10	31,62	36,14	40,66	45,17
5	9,15	13,73	18,30	22,88	27,46	32,03	36,61	41,18	45,76
6	9,27	13,90	18,54	23,17	27,81	32,44	37,08	41,71	46,35
7	9,39	14,08	18,77	23,47	28,16	32,85	37,55	42,24	46,93
8	9,50	14,26	19,01	23,76	28,51	33,26	38,02	42,77	47,52
9	9,62	14,43	19,24	24,05	28,86	33,67	38,49	43,30	48,11
10	9,74	14,61	19,48	24,35	29,22	34,09	38,95	43,82	48,69
11	9,86	14,78	19,71	24,64	29,57	34,50	39,42	44,35	49,28
12	9,97	14,96	19,95	24,93	29,92	34,91	39,89	44,88	49,87
13	10,09	15,14	20,18	25,23	30,27	35,32	40,36	45,41	50,45
14	10,21	15,31	20,42	25,52	30,62	35,73	40,83	45,84	51,04
15	10,33	15,49	20,65	25,81	30,98	36,14	41,30	46,46	51,63

Convenciones para las operaciones S7-200

8

En el presente capítulo se utilizan las siguientes convenciones para representar las operaciones en los lenguajes de programación KOP (esquema de contactos), FUP (diagrama de funciones) y AWL (lista de instrucciones), indicándose también las CPUs que soportan la correspondiente operación.

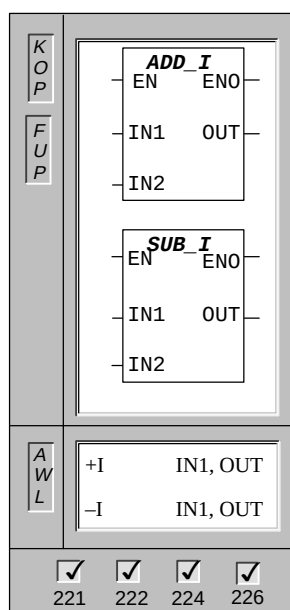
Índice del capítulo

Apartado	Descripción	Página
8.1	Conceptos y convenciones para programar con STEP 7-Micro/WIN 32	8-2
8.2	Márgenes válidos para las CPUs S7-200	8-7

8.1 Conceptos y convenciones para programar con STEP 7-Micro/WIN 32

El siguiente diagrama muestra el formato de las operaciones STEP 7-Micro/WIN 32 utilizado en este capítulo. Los componentes del formato de la operación se describen a continuación del diagrama.

Sumar y restar enteros de 16 bits



Las operaciones **Sumar enteros de 16 bits** y **Restar enteros de 16 bits** suman/restan dos enteros de 16 bits, arrojando un resultado de 16 bits (OUT).

En KOP y FUP: $IN1 + IN2 = OUT$
 $IN1 - IN2 = OUT$

En AWL: $IN1 + OUT = OUT$
 $OUT - IN1 = OUT$

Condiciones de error que ponen ENO a 0: SM1.1 (desbordamiento), SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales: SM1.0 (cero); SM1.1 (desbordamiento); SM1.2 (negativo)

Entradas/salidas	Operandos	Tipos de datos
IN1, IN2	VW, IW, QW, MW, SW, SMW, LW, AIW, T, C, AC, constante, *VD, *AC, *LD	INT
OUT	VW, IW, QW, MW, SW, SMW, LW, T, C, AC, *VD, *AC, *LD	INT

Título de la operación o del grupo de operaciones: En el presente ejemplo, el título es **Sumar enteros de 16 bits y Restar enteros de 16 bits**.

Figura correspondiente a la operación STEP 7-Micro/WIN 32: La figura que aparece debajo del título de la operación muestra los elementos KOP y FUP de la operación, así como (en las operaciones SIMATIC) la nemotécnica y los operandos AWL. En algunos casos, la figura de las operaciones KOP y FUP es idéntica, mostrando un solo cuadro que contiene tanto la figura correspondiente a KOP como a FUP (como en el presente ejemplo). La nemotécnica y los operandos AWL aparecen siempre en un cuadro por separado.

En el ejemplo, los cuadros KOP/FUP tienen tres entradas (que se muestran siempre en el lado izquierdo del cuadro) y dos salidas (siempre en el lado derecho). En KOP hay dos tipos básicos de entradas y salidas (E/S). El primer tipo de E/S es una entrada o salida de circulación de corriente.

En KOP, de forma similar a los escalones de un diagrama lógico de escalera de relés, hay una barra de alimentación a la izquierda que está energizada. Los contactos cerrados permiten que la corriente circule por ellos hasta el siguiente elemento, en tanto que los contactos abiertos bloquean el flujo de energía. Cualquier elemento KOP que se puede conectar a la barra de alimentación izquierda o derecha, o bien a un contacto, tiene una entrada y/o una salida de circulación de corriente.

En el lenguaje de programación FUP (SIMATIC), que no utiliza las barras de alimentación izquierda ni derecha, el término “circulación de corriente” se utiliza para expresar el concepto análogo del flujo de señales por los bloques lógicos FUP. El recorrido “1” lógico por los elementos FUP se denomina circulación de corriente.

En KOP, una entrada o salida de circulación de corriente no se utiliza más que para el flujo de señales, no pudiendo asignarse a ningún operando. En FUP, el origen de una entrada de circulación de corriente y el destino de una salida de circulación de corriente se pueden asignar directamente a un operando.

Adicionalmente a la circulación de corriente, numerosas operaciones KOP/FUP/AWL (si bien no todas) tienen uno o más operandos de entrada y de salida. Los parámetros permitidos para dichos operandos se indican en la tabla de entradas/salidas que aparece debajo de la figura KOP/FUP/AWL.

Tipo de CPU: Las casillas seleccionadas en el lado inferior de la figura de la operación indican cuáles CPUs la soportan. En el presente ejemplo, la operación se puede utilizar con las CPUs 221, 222, 224 y 226.

Descripción de la operación: El texto que aparece a la derecha de la figura de la operación en la página 8-2 describe su funcionamiento. En algunos casos, la operación se describe por separado para cada uno de los lenguajes de programación. En otros, hay una sola descripción aplicable a los tres lenguajes. Tenga en cuenta que la terminología IEC difiere considerablemente de la terminología SIMATIC (tanto en cuanto a los nombres de las operaciones como al de los lenguajes de programación). Por ejemplo, en SIMATIC existe la operación Contar adelante (CTU), en tanto que en IEC se hace referencia al bloque funcional Contador ascendente (CTU). Además, en SIMATIC se habla del lenguaje KOP (Esquema de contactos) que equivale en IEC al lenguaje LD (Diagrama de escalera). Igualmente, el lenguaje FUP (Diagrama de funciones) de SIMATIC se denomina FBD (Diagrama de bloques funcionales) en IEC.

Condiciones de error que ponen ENO a 0: Si las operaciones KOP/FUP tienen una salida ENO (salida de habilitación), se indican aquí las condiciones de error que ponen ENO a cero.

Marcas especiales afectadas: Si la operación afecta a ciertas marcas especiales como parte normal de la ejecución de la misma, se indican aquí las marcas en cuestión y la forma en que se ven afectadas.

Tabla de operandos: Debajo de la figura KOP/FUP/AWL aparece una tabla en la que se indican los operandos permitidos para cada una de las entradas y salidas, así como los tipos de datos de cada uno de los operandos. Los márgenes de memoria de los operandos correspondientes a cada CPU figuran en la tabla 8-3.

Los operandos y los tipos de datos EN/ENO no figuran en la tabla de operandos de la operación, puesto que son **idénticos para todas las operaciones KOP y FUP**. La tabla 8-1 muestra los operandos y tipos de datos EN/ENO para KOP y FUP, siendo aplicables a todas las operaciones KOP y FUP descritas en el presente manual.

Tabla 8-1 Operandos y tipos de datos EN/ENO para KOP y FUP

Editor	Entradas/salidas	Operandos	Tipos de datos
KOP	EN	Circulación de corriente	BOOL
	ENO	Circulación de corriente	BOOL
FUP	EN	I, Q, M, S, SM, T, C, V, L, circulación de corriente	BOOL
	ENO	I, Q, M, S, SM, T, C, V, L, circulación de corriente	BOOL

Convenciones generales para programar en SIMATIC

Segmento ("Network"): En KOP, el programa se divide en segmentos denominados "networks". Un segmento es una red organizada, compuesta por contactos, bobinas y cuadros que se interconectan para conformar un circuito completo entre las barras de alimentación izquierda y derecha (no se permiten los cortocircuitos, ni los circuitos abiertos, ni la circulación de corriente inversa). STEP 7-Micro/WIN 32 ofrece la posibilidad de crear comentarios para cada uno de los segmentos del programa KOP.

El lenguaje FUP utiliza el concepto de segmentos para subdividir y comentar el programa. Los programas AWL no utilizan segmentos. Sin embargo, la palabra clave NETWORK se puede utilizar para estructurar el programa. En este caso, el programa se puede convertir a KOP o a FUP.

Unidades de organización del programa: En KOP, FUP o AWL, un programa comprende una parte obligatoria (como mínimo), pudiendo incluir también otras secciones opcionales. La parte obligatoria es el programa principal. Las secciones opcionales pueden incluir una o más subrutinas y/o rutinas de interrupción. Para desplazarse por las unidades de organización del programa, basta con seleccionar o hacer clic en la correspondiente ficha en STEP 7-Micro/WIN 32.

Definición de EN/ENO: EN (entrada de habilitación) es una entrada booleana para los cuadros KOP y FUP. Para que la operación se pueda ejecutar, el estado de señal de dicha entrada deberá ser "1" (ON). En AWL, las operaciones no tienen una entrada EN, pero el valor del nivel superior de la pila deberá ser un "1" lógico para poder ejecutar la correspondiente operación AWL.

ENO (salida de habilitación) es una salida booleana para los cuadros KOP y FUP. Si el estado de señal de la entrada EN es "1" y el cuadro ejecuta la función sin que se presenten errores, la salida ENO conducirá corriente al siguiente elemento. Si se detecta un error en la ejecución del cuadro, la circulación de corriente se detendrá en el cuadro que ha generado el error.

En AWL (SIMATIC) no existe la salida ENO, pero las operaciones AWL correspondientes a las funciones KOP y FUP con salidas ENO activan un bit ENO especial. A dicho bit se accede mediante la operación AWL AENO (Y=ENO), pudiendo utilizarse para generar el mismo efecto que el bit ENO de un cuadro.

Entradas condicionadas e incondicionadas: En KOP y FUP, un cuadro o una bobina que dependa de la circulación de corriente aparecerá sin conexión a ningún elemento a la izquierda. Una bobina o un cuadro que no dependa de la circulación de corriente se mostrará con una conexión directa a la barra de alimentación izquierda. La figura 8-1 muestra las entradas condicionadas e incondicionadas.

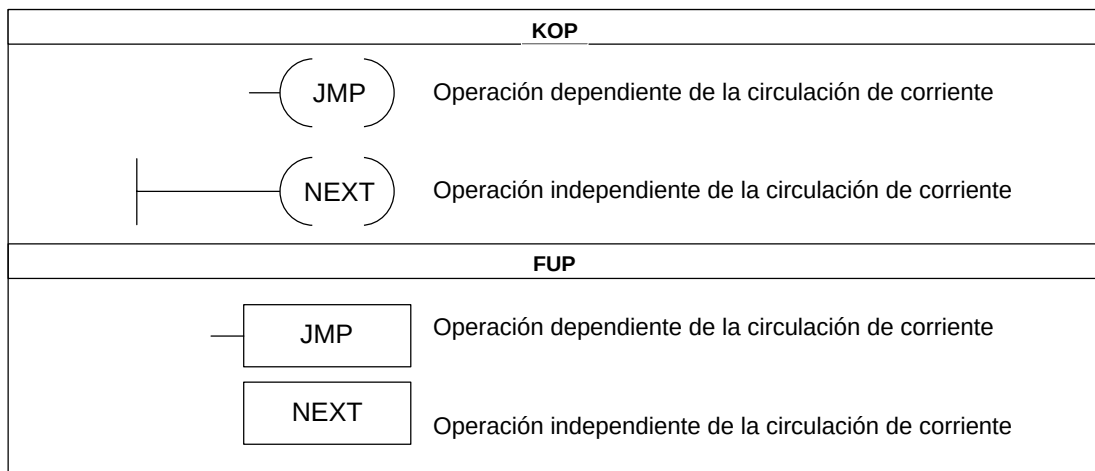


Figura 8-1 Diagrama KOP/FUP de entradas condicionadas e incondicionadas

Operaciones sin salidas: Los cuadros que no se puedan conectar en cascada se representan sin salidas booleanas. Estos cuadros incluyen las llamadas a subrutinas, JMP, CRET, etc. También hay bobinas KOP que sólo se pueden disponer en la barra de alimentación izquierda, incluyendo LBL, NEXT, SCR, SCRE, etc. Éstas se representan en FUP en forma de cuadros con entradas sin meta y sin salidas.

Operaciones de comparación: Las operaciones de comparación FUP (SIMATIC), LD (IEC) y FBD (IEC) se representan mediante cuadros, aunque la operación se ejecute en forma de contacto.

La operación de comparación se ejecutará sin tener en cuenta el estado de señal. Si el estado es "0" (FALSO), el estado de señal de la salida también será "0" (FALSO). Si el estado de señal es "1" (VERDADERO), la salida se activará dependiendo del resultado de la comparación.

Convenciones de STEP 7-Micro/WIN 32: En STEP 7-Micro/WIN 32 rigen las siguientes convenciones en todos los editores:

- Si un símbolo aparece en letras mayúsculas (ABC), significa que se trata de un símbolo global.
- Si un nombre simbólico (p.ej. #var1) va antecedido de un signo de número (#), significa que se trata de un símbolo local.
- El símbolo % identifica una dirección directa.
- El símbolo de operando "?" ó "???" indica que el operando se debe configurar.

Convenciones en KOP: En el editor KOP, las teclas de función F4, F6 y F9 se pueden utilizar para acceder a los contactos, los cuadros y las bobinas. Las siguientes convenciones rigen en el editor KOP:

- El símbolo "—>>" es un circuito abierto o una conexión necesaria para la circulación de corriente.
- El símbolo  indica que la salida es una conexión opcional para la circulación de corriente en una operación que se puede disponer en cascada o conectar en serie.
- Los símbolos "<<" o ">>" indican que se puede utilizar bien sea un valor, o bien la circulación de corriente.
- Un contacto conectado a la barra de alimentación indica que la operación no depende de la circulación de corriente (v. fig. 8-1).

- La condición lógica NOT del estado del operando o la corriente que circula por la entrada se representa mediante un pequeño círculo en la entrada de una operación FUP. En la figura 8-2, Q0.0 es igual al NOT de I0.0 AND I0.1.

Convenciones en FUP: En el editor FUP, las teclas de función F4, F6 y F9 se pueden utilizar para acceder a las operaciones AND y OR, así como a las operaciones con cuadros. Las siguientes convenciones rigen en el editor FUP:

- El símbolo “—>” en un operando EN es un indicador de circulación de corriente o de operando. También puede representar un circuito abierto o una conexión necesaria para la circulación de corriente.
- El símbolo —>| indica que la salida es una conexión opcional para la circulación de corriente en una operación que se puede disponer en cascada o conectar en serie.
- Símbolo de negación: La condición lógica NOT (la condición invertida) del operando o la corriente se representa mediante un pequeño círculo en la entrada. En la figura 8-2, Q0.0 es igual al NOT de I0.0 AND I0.1. Los símbolos de negación sólo son válidos para las señales booleanas, que se pueden indicar en forma de parámetros o de circulación de corriente.

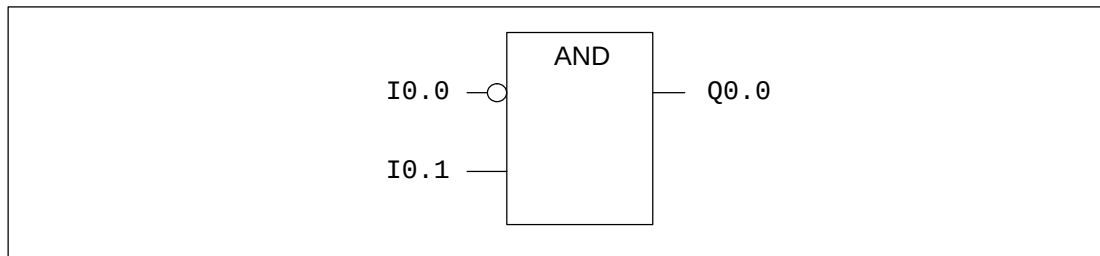


Figura 8-2 Diagrama FUP de la condición lógica NOT

- Indicadores directos: La condición directa de un operando booleano se muestra mediante la línea vertical en la entrada de una operación FUP (v. fig. 8-3). El indicador directo causa una lectura directa de la entrada física indicada. Los indicadores directos sólo son aplicables a las entradas físicas.

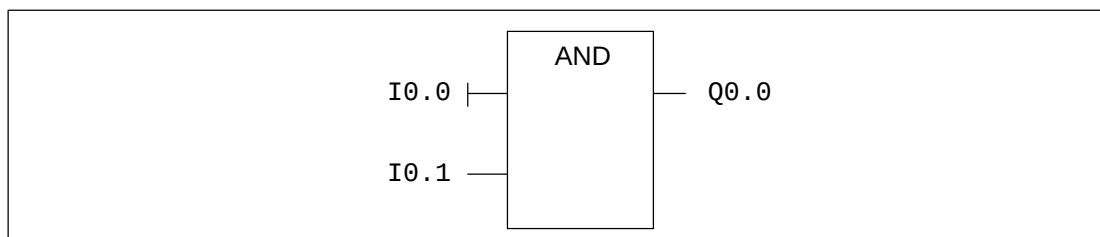


Figura 8-3 Diagrama FUP de la condición directa

- Tecla TAB: La tecla TAB desplaza el cursor de una entrada a otra. La entrada seleccionada actualmente se destaca en rojo. El movimiento es circular, comenzando en la primera entrada y finalizando en la salida.
- Cuadro sin entradas ni salidas: Un cuadro sin entradas ni salidas indica que la operación no depende de la circulación de corriente (v. fig. 8-1).
- Cantidad de operandos: La cantidad de operandos se puede incrementar hasta 32 entradas en el caso de las operaciones AND y OR. Para agregar y quitar operandos, utilice las teclas “+” y “-” del teclado, respectivamente.

8.2 Márgenes válidos para las CPUs S7-200

Tabla 8-2 Resumen de los márgenes de memoria y funciones de las CPUs S7-200

Descripción	CPU 221	CPU 222	CPU 224	CPU 226
Tamaño del programa de usuario	2K palabras	2K palabras	4K palabras	4K palabras
Tamaño de los datos de usuario	1K palabras	1K palabras	2,5K palabras	2,5K palabras
Imagen del proceso de las entradas	I0.0 a I15.7	I0.0 a I15.7	I0.0 a I15.7	I0.0 a I15.7
Imagen del proceso de las salidas	Q0.0 a Q15.7	Q0.0 a Q15.7	Q0.0 a Q15.7	Q0.0 a Q15.7
Entradas analógicas (sólo lectura)	—	AIW0 a AIW30	AIW0 a AIW62	AIW0 a AIW62
Salidas analógicas (sólo escritura)	—	AQW0 a AQW30	AQW0 a AQW62	AQW0 a AQW62
Memoria de variables (V) ¹	VB0.0 a VB2047.7	VB0.0 a VB2047.7	VB0.0 a VB5119.7	VB0.0 a VB5119.7
Memoria local (L) ²	LB0.0 a LB63.7	LB0.0 a LB63.7	LB0.0 a LB63.7	LB0.0 a LB63.7
Área de marcas (M)	M0.0 a M31.7	M0.0 a M31.7	M0.0 a M31.7	M0.0 a M31.7
Marcas especiales (SM) Sólo lectura	SM0.0 a SM179.7 SM0.0 a SM29.7	SM0.0 a SM299.7 SM0.0 a SM29.7	SM0.0 a SM299.7 SM0.0 a SM29.7	SM0.0 a SM299.7 SM0.0 a SM29.7
Temporizadores	256 (T0 a T255)	256 (T0 a T255)	256 (T0 a T255)	256 (T0 a T255)
Retardo a la conexión memorizado 1 ms	T0, T64	T0, T64	T0, T64	T0, T64
Retardo a la conexión memorizado 10 ms	T1 a T4, T65 a T68	T1 a T4, T65 a T68	T1 a T4, T65 a T68	T1 a T4, T65 a T68
Retardo a la conexión memorizado 100 ms	T5 a T31, T69 a T95	T5 a T31, T69 a T95	T5 a T31, T69 a T95	T5 a T31, T69 a T95
Retardo a la con./descon. 1 ms	T32, T96	T32, T96	T32, T96	T32, T96
Retardo a la con./descon. 10 ms	T33 a T36, T97 a T100	T33 a T36, T97 a T100	T33 a T36, T97 a T100	T33 a T36, T97 a T100
Retardo a la con./descon. 100 ms	T37 a T63, T101 a T255	T37 a T63, T101 a T255	T37 a T63, T101 a T255	T37 a T63, T101 a T255
Contadores	C0 a C255	C0 a C255	C0 a C255	C0 a C255
Contadores rápidos	HC0, HC3, HC4, HC5	HC0, HC3, HC4, HC5	HC0 a HC5	HC0 a HC5
Relés de control secuencial (S)	S0.0 a S31.7	S0.0 a S31.7	S0.0 a S31.7	S0.0 a S31.7
Acumuladores	AC0 a AC3	AC0 a AC3	AC0 a AC3	AC0 a AC3
Salto a metas	0 a 255	0 a 255	0 a 255	0 a 255
Llamadas a subrutinas	0 a 63	0 a 63	0 a 63	0 a 63
Rutinas de interrupción	0 a 127	0 a 127	0 a 127	0 a 127
Lazos PID	0 a 7	0 a 7	0 a 7	0 a 7
Puerto	Puerto 0	Puerto 0	Puerto 0	Puerto 0, puerto 1

¹ Todo el contenido de la memoria V se puede guardar en la memoria no volátil.

² STEP 7-Micro/WIN 32 (versión 3.0 o posterior) reserva LB60 a LB63.

Tabla 8-3 Áreas de operandos de las CPUs S7-200

Tipo de acceso	CPU 221	CPU 222	CPU 224, CPU 226
Bit (byte.bit)	V 0.0 a 2047.7 I 0.0 a 15.7 Q 0.0 a 15.7 M 0.0 a 31.7 SM 0.0 a 179.7 S 0.0 a 31.7 T 0 a 255 C 0 a 255 L 0.0 a 63.7	V 0.0 a 2047.7 I 0.0 a 15.7 Q 0.0 a 15.7 M 0.0 a 31.7 SM 0.0 a 299.7 S 0.0 a 31.7 T 0 a 255 C 0 a 255 L 0.0 a 63.7	V 0.0 a 5119.7 I 0.0 a 15.7 Q 0.0 a 15.7 M 0.0 a 31.7 SM 0.0 a 299.7 S 0.0 a 31.7 T 0 a 255 C 0 a 255 L 0.0 a 63.7
Byte	VB 0 a 2047 IB 0 a 15 QB 0 a 15 MB 0 a 31 SMB 0 a 179 SB 0 a 31 LB 0 a 63 AC 0 a 3 Constante	VB 0 a 2047 IB 0 a 15 QB 0 a 15 MB 0 a 31 SMB 0 a 179 SB 0 a 31 LB 0 a 63 AC 0 a 3 Constante	VB 0 a 5119 IB 0 a 15 QB 0 a 15 MB 0 a 31 SMB 0 a 179 SB 0 a 31 LB 0 a 63 AC 0 a 3 Constante
Palabra	VW 0 a 2046 IW 0 a 14 QW 0 a 14 MW 0 a 30 SMW 0 a 178 SW 0 a 30 T 0 a 255 C 0 a 255 LW 0 a 62 AC 0 a 3 Constante	VW 0 a 2046 IW 0 a 14 QW 0 a 14 MW 0 a 30 SMW 0 a 178 SW 0 a 30 T 0 a 255 C 0 a 255 LW 0 a 62 AC 0 a 3 AIW 0 a 30 AQW 0 a 30 Constante	VW 0 a 5118 IW 0 a 14 QW 0 a 14 MW 0 a 30 SMW 0 a 178 SW 0 a 30 T 0 a 255 C 0 a 255 LW 0 a 62 AC 0 a 3 AIW 0 a 62 AQW 0 a 62 Constante
Palabra doble	VD 0 a 2044 ID 0 a 12 QD 0 a 12 MD 0 a 28 SMD 0 a 176 SD 0 a 28 LD 0 a 60 AC 0 a 3 HC 0, 3, 4, 5 Constante	VD 0 a 2044 ID 0 a 12 QD 0 a 12 MD 0 a 28 SMD 0 a 176 SD 0 a 28 LD 0 a 60 AC 0 a 3 HC 0, 3, 4, 5 Constante	VD 0 a 5116 ID 0 a 12 QD 0 a 12 MD 0 a 28 SMD 0 a 176 SD 0 a 28 LD 0 a 60 AC 0 a 3 HC 0 a 5 Constante

Operaciones SIMATIC

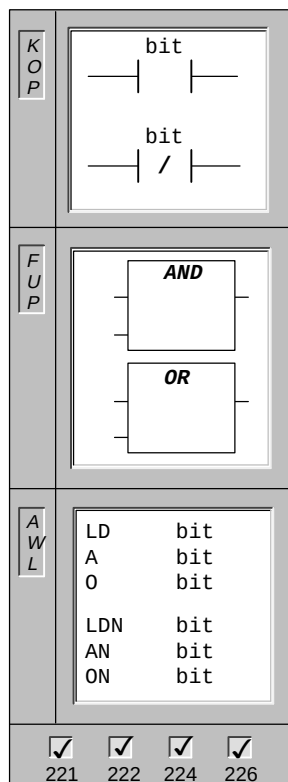
El presente capítulo describe el juego de operaciones SIMATIC para los sistemas de automatización S7-200.

Índice del capítulo

Apartado	Descripción	Página
9.1	Operaciones lógicas con bits (SIMATIC)	9-2
9.2	Operaciones de comparación (SIMATIC)	9-10
9.3	Operaciones de temporización (SIMATIC)	9-15
9.4	Operaciones con contadores (SIMATIC)	9-23
9.5	Operaciones de reloj (SIMATIC)	9-71
9.6	Operaciones aritméticas con enteros (SIMATIC)	9-73
9.7	Operaciones aritméticas con números reales (SIMATIC)	9-82
9.8	Operaciones con funciones numéricas (SIMATIC)	9-85
9.9	Operaciones de transferencia (SIMATIC)	9-102
9.10	Operaciones de tabla (SIMATIC)	9-107
9.11	Operaciones lógicas (SIMATIC)	9-114
9.12	Operaciones de desplazamiento y rotación (SIMATIC)	9-120
9.13	Operaciones de conversión (SIMATIC)	9-130
9.14	Operaciones de control del programa (SIMATIC)	9-145
9.15	Operaciones de interrupción y comunicación (SIMATIC)	9-169
9.16	Operaciones lógicas de pilas (SIMATIC)	9-197

9.1 Operaciones lógicas con bits (SIMATIC)

Contactos estándar



Estas operaciones leen el valor direccionado de la memoria o de la imagen del proceso si el tipo de datos es I o Q. Para los cuadros AND y OR se pueden utilizar siete entradas como máximo.

El **Contacto normalmente abierto** se cierra (ON) si el bit es igual a 1.

El **Contacto normalmente cerrado** se cierra (ON) si el bit es igual a 0.

En KOP, las operaciones Contacto normalmente abierto y Contacto normalmente cerrado se representan mediante contactos.

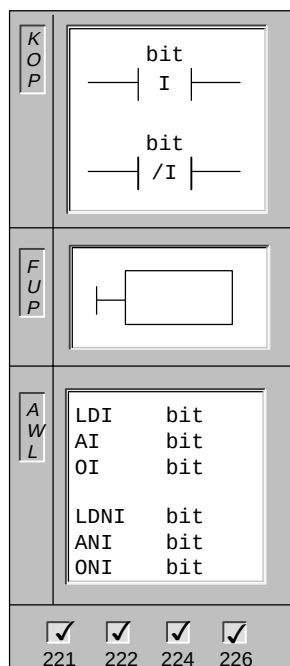
En FUP, los contactos normalmente abiertos se representan mediante cuadros AND/OR. Estas operaciones sirven para manipular señales booleanas de la misma forma que los contactos KOP. Los contactos cerrados también se representan mediante cuadros. Una operación Contacto normalmente cerrado se realiza situando el símbolo de negación en la raíz de la señal de entrada. La cantidad de entradas de los cuadros AND y OR se puede incrementar a siete como máximo.

En AWL, el contacto normalmente abierto se representa con las operaciones **Cargar, Y y O**. Estas operaciones cargan el valor binario del bit de dirección en el nivel superior de la pila y lo combinan mediante Y u O.

En AWL, el contacto normalmente cerrado se representa con las operaciones **Cargar valor negado, Y-NO y O-NO**. Estas operaciones cargan el valor binario invertido del bit de la dirección en el nivel superior de la pila y lo combinan mediante Y u O.

Entradas/salidas	Operandos	Tipos de datos
Bit (KOP, AWL)	I, Q, M, SM, T, C, V, S, L	BOOL
Entradas (FUP)	I, Q, M, SM, T, C, V, S, L, circulación de corriente	BOOL
Salida (FUP)	I, Q, M, SM, T, C, V, S, L, circulación de corriente	BOOL

Contactos directos



Estas operaciones leen el valor de la entrada física al ejecutarse la operación, pero la imagen del proceso no se actualiza.

El **Contacto abierto directo** se cierra (se activa) si la entrada física (bit) es 1.

El **Contacto cerrado directo** se cierra (se activa) si la entrada física (bit) es 0.

En KOP, las operaciones Contacto abierto directo y Contacto cerrado directo se representan mediante contactos.

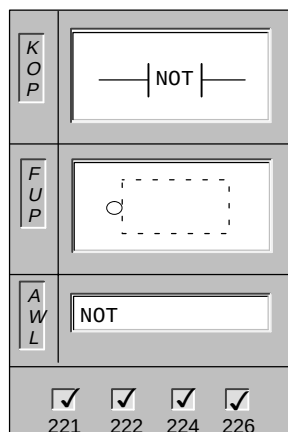
En FUP, la operación Contacto abierto directo se representa mediante un corchete delante del operando. El corchete puede faltar si se usa la circulación de corriente. Esta operación sirve para manipular señales físicas de la misma forma que los contactos KOP.

En FUP, la operación Contacto cerrado directo se representa también mediante un corchete y el símbolo de negación delante del operando. El corchete puede faltar si se usa la circulación de corriente. La operación Contacto normalmente cerrado se realiza situando el símbolo de negación en la raíz de la señal de entrada.

En AWL, el contacto abierto directo se representa con las operaciones **Cargar directamente, Y directa y O directa**. Estas operaciones cargan directamente el valor de la entrada física en el nivel superior de la pila y lo combinan mediante Y u O.

En AWL, el contacto cerrado directo se representa con las operaciones **Cargar valor negado directamente, Y-NO directa y O-NO directa** (ONI). Estas operaciones cargan directamente el valor binario negado de la entrada física en el nivel superior de la pila y lo combinan mediante Y u O.

Entradas/salidas	Operandos	Tipos de datos
Bit (KOP, AWL)	I	BOOL
Entrada (FUP)	I	BOOL

NOT

El contacto **NOT** cambia el estado de la entrada de circulación de corriente. La corriente se detiene al alcanzar el contacto NOT. Si no logra alcanzar el contacto, entonces hace circular la corriente.

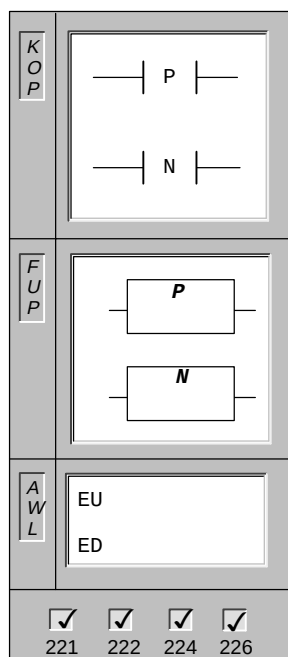
En KOP, la operación **NOT** se representa en forma de contacto.

En FUP, la operación **NOT** utiliza el símbolo gráfico de negación con entradas booleanas de cuadro.

En AWL, la operación **Invertir primer valor** (NOT) invierte el primer valor de la pila de 0 a 1, o bien de 1 a 0.

Operandos: Ninguno

Tipos de datos: Ninguno

Detectar flanco positivo y negativo

El contacto **Detectar flanco positivo** permite que la corriente circule durante un ciclo cada vez que se produce un cambio de 0 a 1 (de "off" a "on").

El contacto **Detectar flanco negativo** permite que la corriente circule durante un ciclo cada vez que se produce un cambio de señal de 1 a 0 (de "on" a "off").

En KOP, las operaciones Detectar flanco positivo y negativo se representan mediante contactos.

En FUP, estas operaciones se representan mediante los cuadros POS y NEG.

En AWL, la transición positiva se representa con la operación **Detectar flanco positivo**. Cuando se detecta un cambio de señal de 0 a 1 en el primer valor de la pila, éste se pone a 1. En caso contrario, se pone a 0.

En AWL, la transición negativa se representa con la operación **Detectar flanco negativo**. Cuando se detecta un cambio de señal de 1 a 0 en el primer valor de la pila, éste se pone a 1. En caso contrario, se pone a 0.

Entradas/salidas	Operandos	Tipos de datos
IN (FUP)	I, Q, M, SM, T, C, V, S, L, circulación de corriente	BOOL
OUT (FUP)	I, Q, M, SM, T, C, V, S, L, circulación de corriente	BOOL

Ejemplos de operaciones con contactos

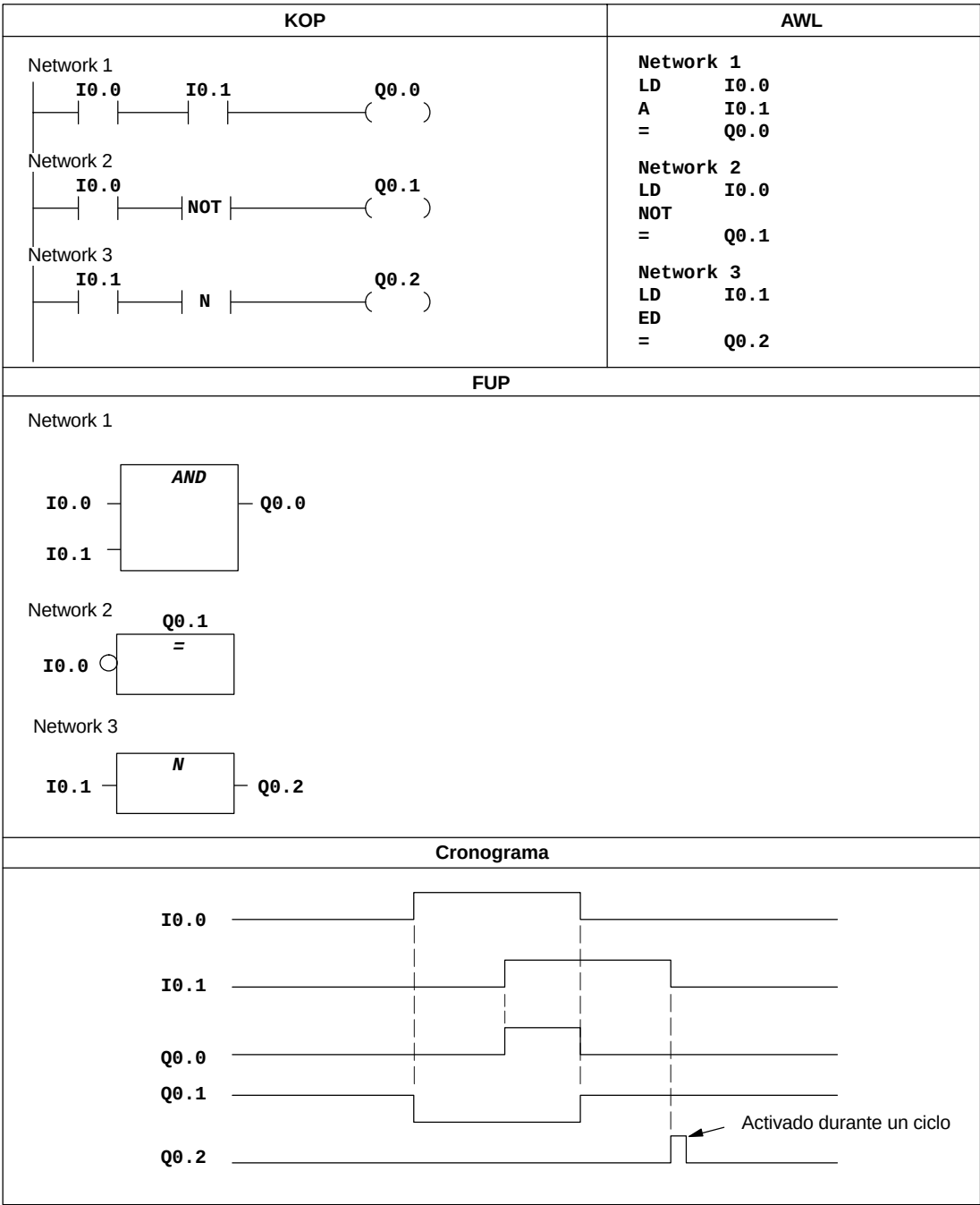
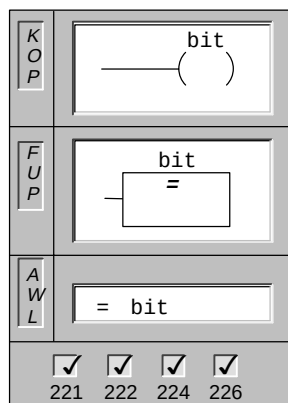


Figura 9-1 Ejemplos de operaciones con contactos booleanos para KOP, AWL y FUP (SIMATIC)

Asignar



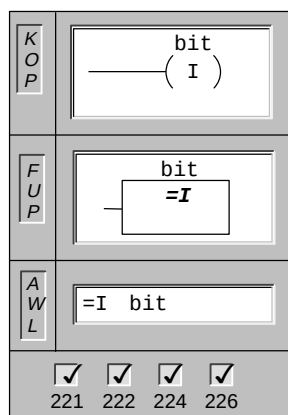
Cuando se ejecuta la operación **Asignar**, el bit de salida se activa en la imagen del proceso.

Cuando la operación Asignar se ejecuta en KOP y FUP, el bit indicado se ajusta de forma equivalente a la circulación de la corriente.

En AWL, la operación Asignar copia el primer valor de la pila en el bit indicado.

Entradas/salidas	Operandos	Tipos de datos
Bit	I, Q, M, SM, T, C, V, S, L	BOOL
Entrada (KOP)	Circulación de corriente	BOOL
Entrada (FUP)	I, Q, M, SM, T, C, V, S, L, circulación de corriente	BOOL

Asignar directamente



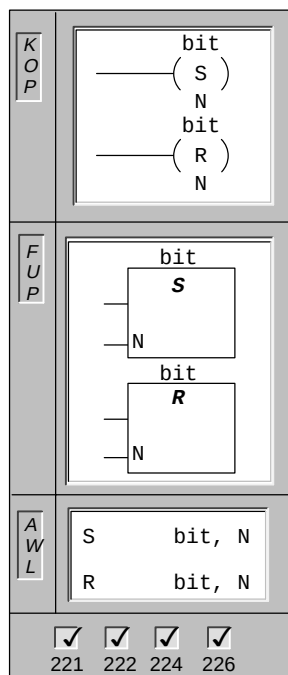
Cuando se ejecuta la operación **Asignar directamente**, la entrada física (bit u OUT) se ajusta de forma equivalente a la circulación de la corriente.

La "I" indica que la operación se ejecuta directamente. El nuevo valor se escribe entonces tanto en la salida física como en la correspondiente dirección de la imagen del proceso. En cambio, en las operaciones no directas, el nuevo valor se escribe sólo en la imagen del proceso.

En AWL, la operación Asignar directamente copia el primer valor de la pila directamente en la salida física indicada (bit).

Entradas/salidas	Operandos	Tipos de datos
Bit	Q	BOOL
Entrada (KOP)	Circulación de corriente	BOOL
Entrada (FUP)	I, Q, M, SM, T, C, V, S, L, circulación de corriente	BOOL

Poner a 1, Poner a 0 (N bits)



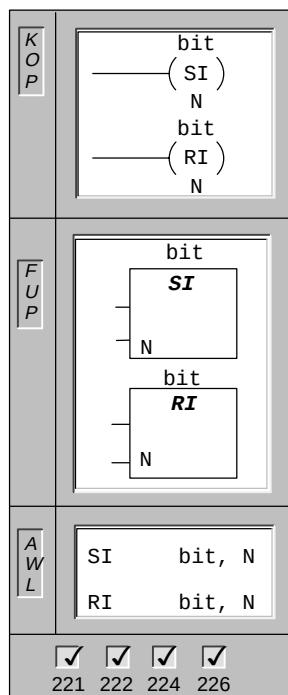
Cuando se ejecutan las operaciones **Poner a 1** y **Poner a 0**, se activa (se pone a 1) o se desactiva (se pone a 0) el número indicado de salidas (N) a partir del valor indicado por el bit o por el parámetro OUT.

El margen de entradas y/o salidas que se pueden activar o desactivar está comprendido entre 1 y 255. Con la operación Poner a 0, si el bit indicado es un bit T (bit de temporización) o un bit C (bit de conteo), se desactivará el bit de temporización/conteo y se borrará el valor actual del temporizador/contador.

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto), 0091 (operando fuera de área)

Entradas/salidas	Operandos	Tipos de datos
Bit	I, Q, M, SM, T, C, V, S, L	BOOL
N	VB, IB, QB, MB, SMB, SB, LB, AC, constante, *VD, *AC, *LD	BYTE

Poner a 1 directamente, Poner a 0 directamente (N bits)



Cuando se ejecutan las operaciones **Poner a 1 directamente** y **Poner a 0 directamente** se activa (se pone a 1) o se desactiva (se pone a 0) directamente el número indicado de salidas físicas (N) a partir del bit o de OUT.

El margen de entradas y/o salidas que se pueden activar o desactivar está comprendido entre 1 y 128.

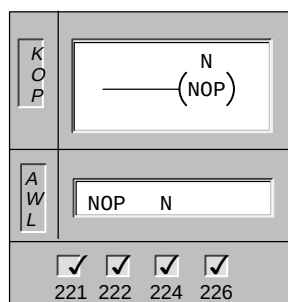
La "I" indica que la operación se ejecuta directamente. Al ejecutarse ésta, el nuevo valor se escribe tanto en la salida física como en la correspondiente dirección de la imagen del proceso. En cambio, en las operaciones no directas, el nuevo valor se escribe sólo en la imagen del proceso.

Condiciones de error que ponen ENO a 0:

SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto), 0091 (operando fuera de área)

Entradas/salidas	Operandos	Tipos de datos
Bit	Q	BOOL
N	VB, IB, QB, MB, SMB, SB, LB, AC, constante, *VD, *AC, *LD	BYTE

Operación nula



La **Operación nula** no tiene efecto alguno sobre la ejecución del programa. Esta operación no está disponible en FUP. El operando N es un número comprendido entre 0 y 255.

Operandos: N: Constante (0 a 255)

Tipos de datos: BYTE

Ejemplos de las operaciones Asignar, Poner a 1 y Poner a 0

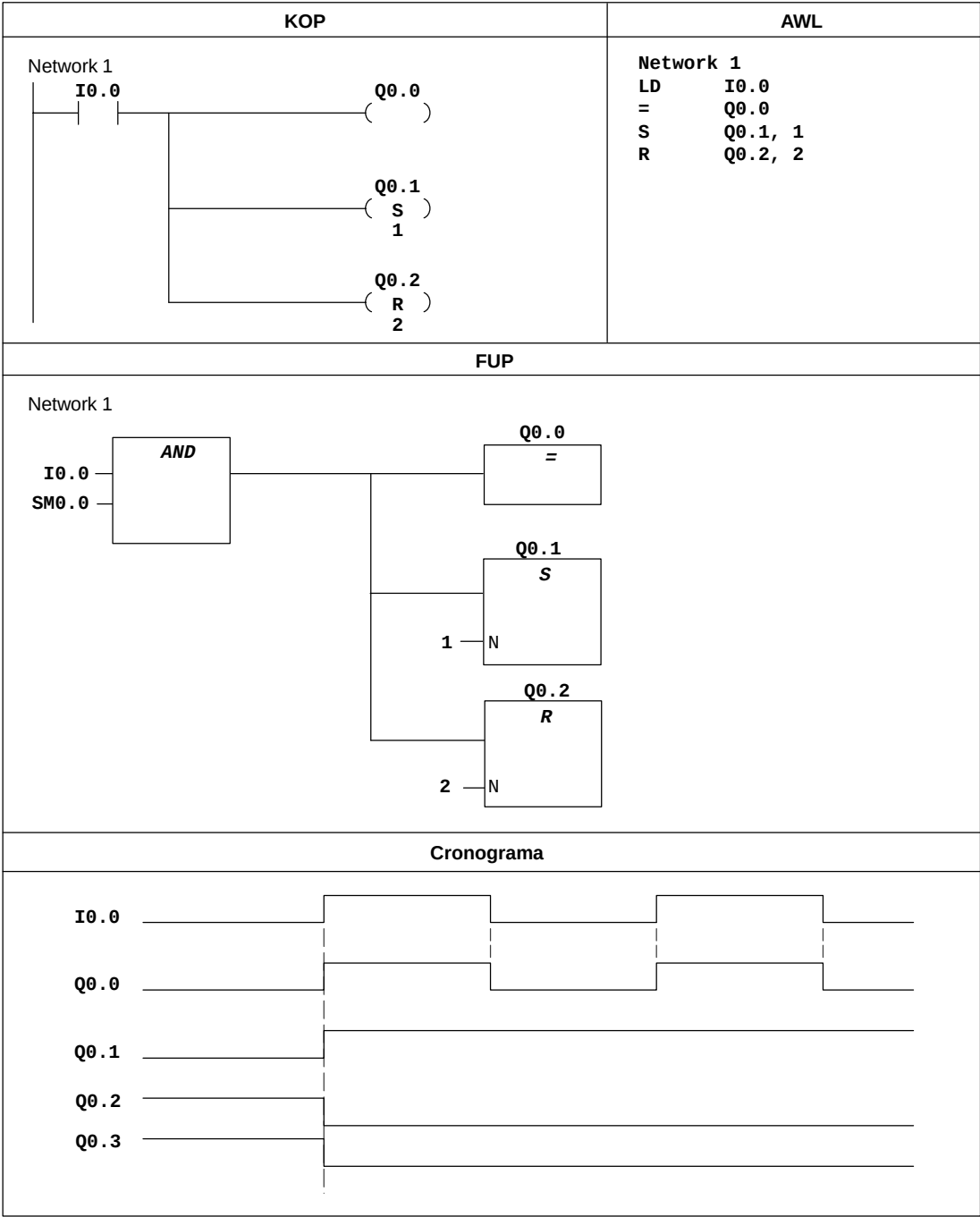
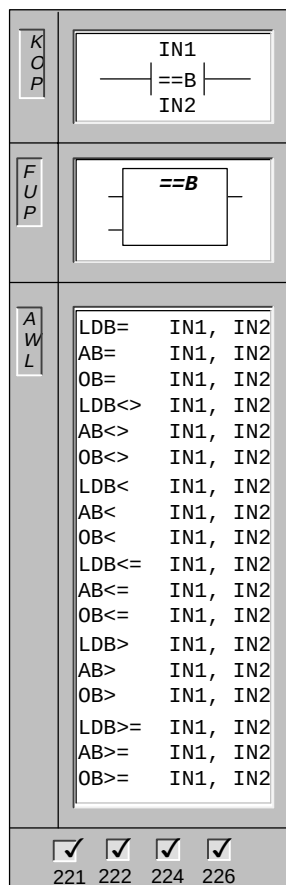


Figura 9-2 Ejemplos de las operaciones Asignar, Poner a 1 y Poner a 0 en KOP, AWL y FUP (SIMATIC)

9.2 Operaciones de comparación (SIMATIC)

Comparar byte



La operación **Comparar byte** se utiliza para comparar dos valores: IN1 e IN2. Las comparaciones incluyen: IN1 = IN2, IN1 >= IN2, IN1 <= IN2, IN1 > IN2, IN1 < IN2, o IN1 <> IN2.

Las comparaciones de bytes no llevan signo.

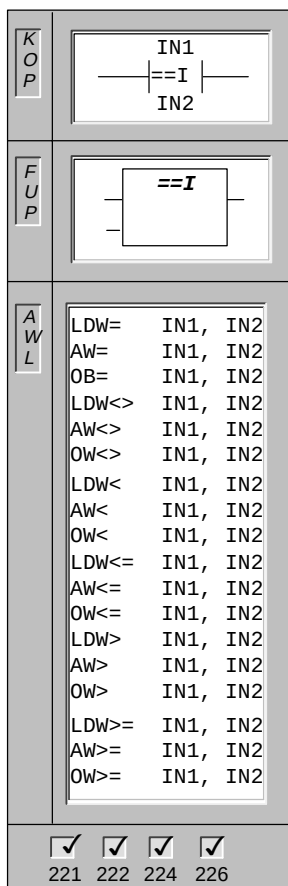
En KOP, el contacto se activa si la comparación es verdadera.

En FUP, la salida se activa si la comparación es verdadera.

En AWL, las operaciones cargan un "1" en el nivel superior de la pila si la comparación es verdadera.

Entradas/salidas	Operandos	Tipos de datos
Entradas	IB, QB, MB, SMB, VB, SB, LB, AC, constante, *VD, *AC, *LD	BYTE
Salidas (FUP)	I, Q, M, SM, T, C, V, S, L, circulación de corriente	BOOL

Comparar entero



La operación **Comparar entero** se utiliza para comparar dos valores: IN1 e IN2. Las comparaciones incluyen: IN1 = IN2, IN1 >= IN2, IN1 <= IN2, IN1 > IN2, IN1 < IN2, o IN1 <> IN2.

Las comparaciones de enteros llevan signo (16#7FFF > 16#8000).

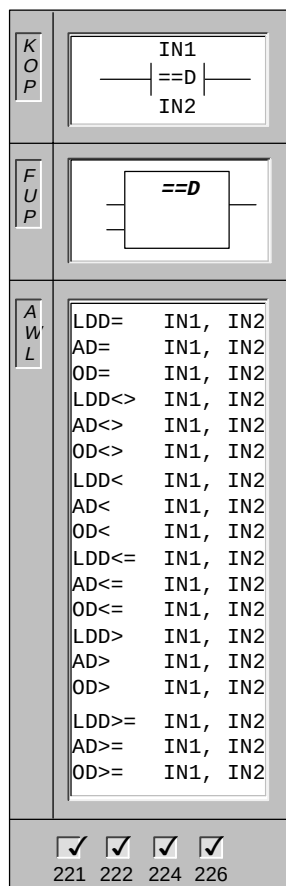
En KOP, el contacto se activa si la comparación es verdadera.

En FUP, la salida se activa si la comparación es verdadera.

En AWL, las operaciones cargan un "1" en el nivel superior de la pila y combinan el valor "1" con el primer valor de la pila mediante Y u O cuando la comparación es verdadera.

Entradas/salidas	Operandos	Tipos de datos
Entradas	IW, QW, MW, SW, SMW, T, C, VW, LW, AIW, AC, constante, *VD, *AC, *LD	INT
Salidas (FUP)	I, Q, M, SM, T, C, V, S, L, circulación de corriente	BOOL

Comparar palabra doble



La operación **Comparar palabra doble** se utiliza para comparar dos valores: IN1 e IN2. Las comparaciones incluyen: IN1 = IN2, IN1 >= IN2, IN1 <= IN2, IN1 > IN2, IN1 < IN2, o IN1 <> IN2.

Las comparaciones de palabras dobles llevan signo (16#7FFFFFFF > 16#80000000).

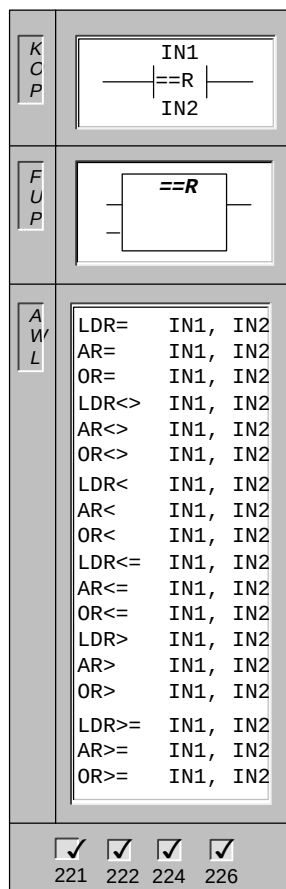
En KOP, el contacto se activa si la comparación es verdadera.

En FUP, la salida se activa si la comparación es verdadera.

En AWL, las operaciones cargan un "1" en el nivel superior de la pila y combinan el valor "1" con el primer valor de la pila mediante Y u O cuando la comparación es verdadera.

Entradas/salidas	Operandos	Tipos de datos
Entradas	ID, QD, MD, SD, SMD, VD, LD, HC, AC, constante, *VD, *AC, *LD	DINT
Salidas (FUP)	I, Q, M, SM, T, C, V, S, L, circulación de corriente	BOOL

Comparar real



La operación **Comparar real** se utiliza para comparar dos valores: IN1 e IN2. Las comparaciones incluyen: IN1 = IN2, IN1 >= IN2, IN1 <= IN2, IN1 > IN2, IN1 < IN2, o IN1 <> IN2.

Las comparaciones de números reales llevan signo.

En KOP, el contacto se activa si la comparación es verdadera.

En FUP, la salida se activa si la comparación es verdadera.

En AWL, las operaciones cargan un "1" en el nivel superior de la pila y combinan el valor "1" con el primer valor de la pila mediante Y u O cuando la comparación es verdadera.

Entradas/salidas	Operandos	Tipos de datos
Entradas	ID, QD, MD,SD, SMD, VD, LD, AC, constante, *VD, *AC, *LD	REAL
Salidas (FUP)	I, Q, M, SM, T, C, V, S, L, circulación de corriente	BOOL

Ejemplos de operaciones de comparación

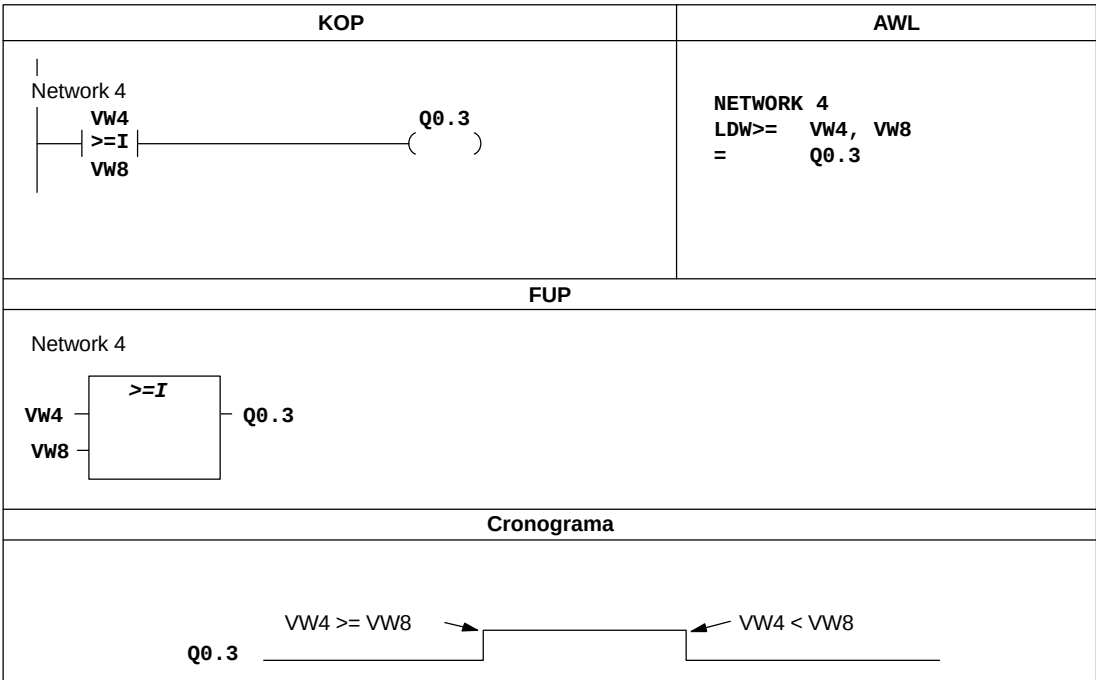
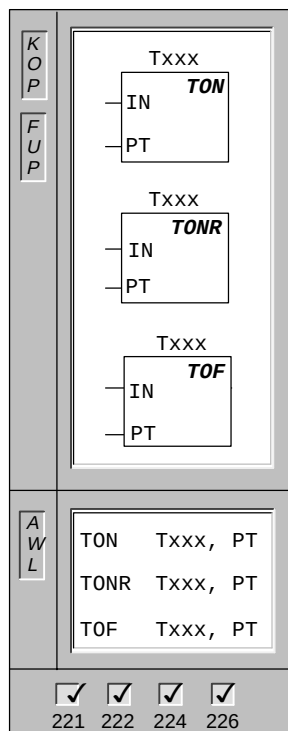


Figura 9-3 Ejemplos de operaciones de comparación en KOP, AWL y FUP (SIMATIC)

9.3 Operaciones de temporización (SIMATIC)

Temporizador de retardo a la conexión, Temporizador de retardo a la conexión memorizado, Temporizador de retardo a la desconexión



Las operaciones **Temporizador de retardo a la conexión** y **Temporizador de retardo a la conexión memorizado** cuentan el tiempo al estar activada (ON) la entrada de habilitación. Si el valor actual (Txxx) es mayor o igual al valor de preselección (PT), se activa el bit de temporización (bit T).

Cuando la entrada de habilitación está desconectada (OFF), el valor actual se borra en el caso del temporizador de retardo a la conexión. En cambio, se conserva en el temporizador de retardo a la conexión memorizado. Éste último sirve para acumular varios períodos de tiempo de la entrada en ON. Para borrar el valor actual del temporizador de retardo a la conexión memorizado se utiliza la operación Poner a 0 (R).

Tanto el temporizador de retardo a la conexión como el temporizador de retardo a la conexión memorizado continúan contando tras haberse alcanzado el valor de preselección y paran de contar al alcanzar el valor máximo de 32767.

El **Temporizador de retardo a la desconexión** se utiliza para retardar la puesta a 0 (OFF) de una salida durante un período determinado tras haberse desactivado (OFF) una entrada. Cuando la entrada de habilitación se activa (ON), el bit de temporización se activa (ON) inmediatamente y el valor actual se pone a 0. Cuando la entrada se desactiva (OFF), el temporizador cuenta hasta que el tiempo transcurrido alcance el valor de preselección. Una vez alcanzado éste, el bit de temporización se desactiva (OFF) y el valor actual detiene el conteo. Si la entrada está desactivada (OFF) durante un tiempo inferior al valor de preselección, el bit de temporización permanece activado (ON). Para que la operación TOF comience a contar se debe producir un cambio de ON a OFF.

Si un temporizador TOF se encuentra dentro de una sección SCR y ésta se encuentra desactivada, el valor actual se pone a 0, el bit de temporización se desactiva (OFF) y el valor actual no cuenta.

Entradas/salidas	Operandos	Tipos de datos
Txxx	Constante	WORD
IN (KOP)	Circulación de corriente	BOOL
IN (FUP)	I, Q, M, SM, T, C, V, S, L, circulación de corriente	BOOL
PT	VW, IW, QW, MW, SW, SMW, LW, AIW, T, C, AC, constante, *VD, *AC, *LD	INT

Se dispone de temporizadores TON, TONR y TOF con tres resoluciones. La resolución viene determinada por el número del temporizador (v. tabla 9-1). El valor actual resulta del valor de conteo multiplicado por la base de tiempo. Por ejemplo, el valor de conteo 50 en un temporizador de 10 ms equivale a 500 ms.

Tabla 9-1 Temporizadores y sus resoluciones

Tipo de temporizador	Resolución en milisegundos (ms)	Valor máximo en segundos (s)	Nº de temporizador
TONR (memorizado)	1 ms	32,767 s (0,546 min.)	T0, T64
	10 ms	327,67 s (0,546 min.)	T1 a T4, T65 a T68
	100 ms	3276,7 s (0,546 min.)	T5 a T31, T69 a T95
TON, TOF (no memorizados)	1 ms	32,767 s (0,546 min.)	T32, T96
	10 ms	327,67 s (0,546 min.)	T33 a T36, T97 a T100
	100 ms	3276,7 s (0,546 min.)	T37 a T63, T101 a T255

Nota

No se pueden compartir números iguales para los temporizadores TOF y TON. Por ejemplo, no puede haber tanto un TON T32 como un TOF T32.

Operaciones de temporización del S7-200

Es posible utilizar temporizadores para implementar funciones controladas por tiempo. El juego de operaciones S7-200 ofrece tres tipos de temporizadores como se indica a continuación. La tabla 9-2 muestra las acciones de los diferentes temporizadores.

- Temporizador de retardo a la conexión (TON) para temporizar un solo intervalo.
- Temporizador de retardo a la conexión memorizado (TONR) para acumular varios intervalos temporizados.
- Temporizador de retardo a la desconexión (TOF) para ampliar el tiempo después de un cambio a "falso" para enfriar un motor tras haber sido desconectado

Tabla 9-2 Acciones de los temporizadores

Tipo de temporizador	Actual $\geq$ Preselección	Entrada de habilitación ON	Entrada de habilitación OFF	Alimentación/ primer ciclo
TON	Bit de temporización ON. El valor actual continúa contando hasta 32.767.	El valor actual cuenta el tiempo.	Bit de temporización OFF. Valor actual = 0	Bit de temporización OFF, Valor actual = 0
TONR	Bit de temporización ON. El valor actual continúa contando hasta 32.767.	El valor actual cuenta el tiempo.	El bit de temporización y el valor actual conservan el último estado.	Bit de temporización OFF. El valor actual se puede conservar. ¹
TOF	Bit de temporización OFF. Valor actual = valor de preselección, se detiene el conteo.	Bit de temporización ON. Valor actual = 0	El temporizador cuenta tras un cambio de ON a OFF.	Bit de temporización OFF, Valor actual = 0
¹ El valor actual del temporizador de retardo a la conexión memorizado se selecciona para que quede memorizado cuando se interrumpa la alimentación. Consulte el apartado 5.3 para obtener información sobre el respaldo de la memoria de la CPU S7-200.				

Nota

La operación Poner a 0 (R) sirve para inicializar cualquier temporizador. Esta operación arroja los siguientes resultados:

Bit de temporización = OFF.
Valor actual = 0

El temporizador TONR sólo se puede inicializar mediante la operación Poner a 0.

Tras inicializarse un temporizador TOF, la entrada de habilitación debe cambiar de ON a OFF para poder reanunciar el temporizador.

A continuación se explican las actividades de los temporizadores con diferentes resoluciones.

Resolución de 1 milisegundo

Los temporizadores con resolución de 1 ms cuentan el número de intervalos de 1 ms que han transcurrido desde que se habilitó el temporizador activo de 1 ms. La temporización arranca al comenzarse a ejecutar la operación. No obstante, los temporizadores de 1 ms se actualizan (tanto el bit de temporización como el valor actual) cada milisegundo de forma asíncrona al ciclo. En otras palabras, el bit de temporización y el valor actual se actualizan varias veces en un ciclo que dure más de 1 ms.

La operación de temporización se utiliza para activar e inicializar el temporizador o, en el caso del temporizador de retardo a la conexión memorizado (TONR), para desactivarlo.

Puesto que el temporizador puede arrancar en cualquier momento durante un milisegundo, el valor de preselección se debe ajustar a un intervalo de tiempo que exceda el intervalo mínimo deseado. Por ejemplo, para garantizar un intervalo mínimo de 56 ms utilizando un temporizador de 1 ms, es preciso ajustar el valor de preselección a 57.

Resolución de 10 milisegundos

Los temporizadores con resolución de 10 ms cuentan el número de intervalos de 10 ms que han transcurrido desde que se habilitó el temporizador activo de 10 ms. La temporización arranca al comenzarse a ejecutar la operación. No obstante, los temporizadores de 10 ms se actualizan al comienzo de cada ciclo (en otras palabras, el valor actual y el bit de temporización permanecen constantes durante el ciclo), sumando el número acumulado de intervalos de 10 ms (desde el comienzo del ciclo anterior) al valor actual del temporizador activo.

Puesto que el temporizador puede arrancar en cualquier momento durante un intervalo de 10 ms, el valor de preselección se debe ajustar a un intervalo de tiempo que exceda el intervalo mínimo deseado. Por ejemplo, para garantizar un intervalo mínimo de 140 ms utilizando un temporizador de 10 ms, es preciso ajustar el valor de preselección a 15.

Resolución de 100 milisegundos

Los temporizadores con resolución de 100 ms cuentan el número de intervalos de 100 ms que han transcurrido desde la última vez que se actualizó el temporizador activo de 100 ms. Estos temporizadores se actualizan sumando el valor acumulado de intervalos de 100 ms (desde el ciclo anterior) al valor actual del temporizador cuando se ejecuta la operación del mismo.

El valor actual de un temporizador de 100 ms se actualiza únicamente si se ha ejecutado la operación correspondiente. Por consiguiente, si un temporizador de 100 ms está habilitado, pero la correspondiente operación no se ejecuta en cada ciclo, no se actualizará el valor actual de ese temporizador y disminuirá el tiempo. Por otra parte, si se ejecuta una misma operación con un temporizador de 100 ms varias veces en un ciclo, el valor de 100 ms acumulado se sumará también varias veces al valor actual del temporizador, con lo cual se prolonga el tiempo. Debido a ello, es recomendable utilizar los temporizadores con una resolución de 100 ms sólo cuando se ejecute exactamente una operación de temporización en cada ciclo.

Puesto que el temporizador puede arrancar en cualquier momento durante un intervalo de 100 ms, el valor de preselección se debe ajustar a un intervalo de tiempo que exceda el intervalo mínimo deseado. Por ejemplo, para garantizar un intervalo mínimo de 2.100 ms utilizando un temporizador de 100 ms, es preciso ajustar el valor de preselección a 22.

Actualizar el valor actual de un temporizador

El efecto de las diferentes maneras de actualizar el valor actual de los temporizadores depende de cómo se utilicen los mismos. Consideremos p.ej. la operación de temporización que muestra la figura 9-4.

- Si se utiliza un temporizador con una resolución de 1 ms (1), Q0.0 se activa durante un ciclo, siempre que se actualice el valor actual del temporizador tras ejecutarse el contacto normalmente cerrado T32 y antes de ejecutarse el contacto normalmente abierto T32.
- Si se utiliza un temporizador con una resolución de 10 ms (2), Q0.0 no se activará nunca, porque el bit de temporización T33 permanece activado desde el principio del ciclo hasta que se ejecute el cuadro del temporizador. Una vez ejecutado éste, se ponen a 0 el valor actual del temporizador y su bit T. Tras ejecutarse el contacto normalmente abierto T33, se desactivarán T33 y Q0.0.
- Si se utiliza un temporizador con una resolución de 100 ms (3), Q0.0 se activa durante un ciclo, siempre que el valor actual del temporizador alcance el valor de preselección.

Si en vez del bit de temporización se utiliza el contacto normalmente cerrado Q0.0 como entrada de habilitación para el cuadro del temporizador, la salida Q0.0 quedará activada durante un ciclo cada vez que el valor del temporizador alcance el valor de preselección.

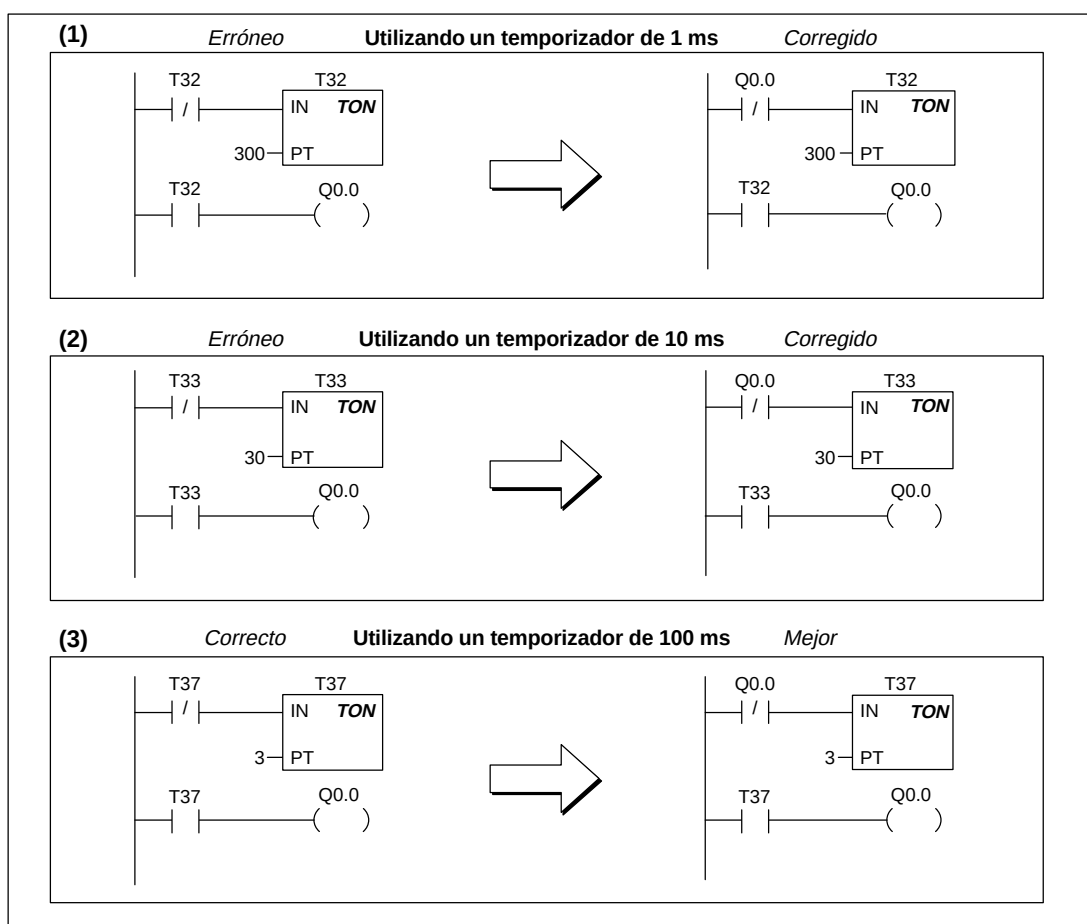


Figura 9-4 Ejemplo del redisparo automático de un temporizador

Ejemplo de un temporizador de retardo a la conexión

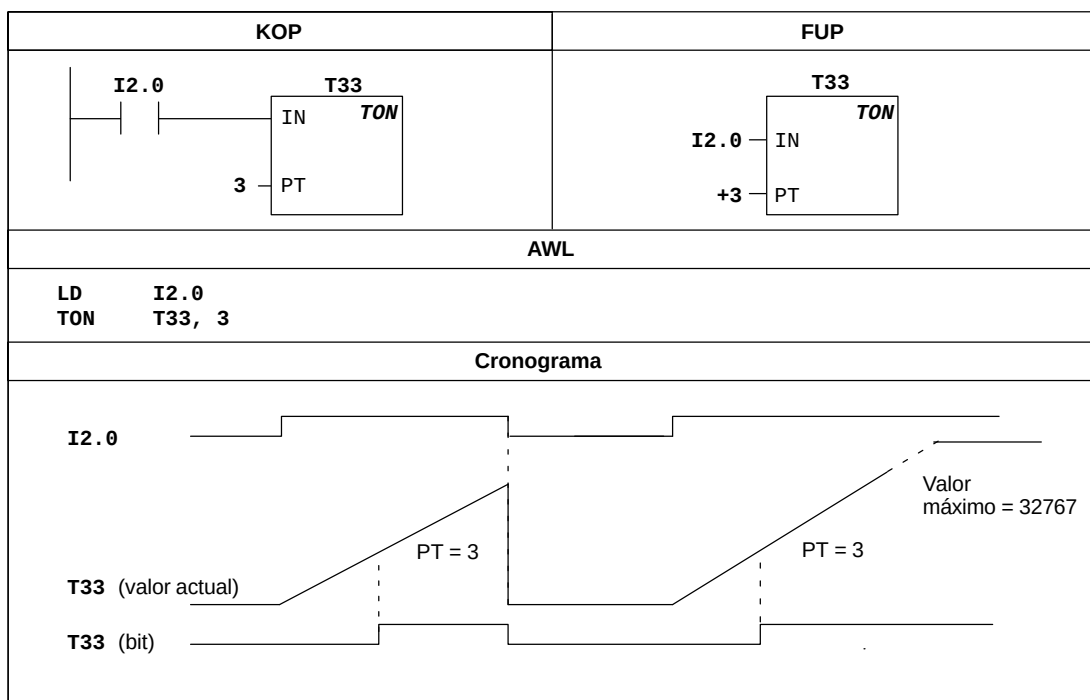


Figura 9-5 Ejemplo de un temporizador de retardo a la conexión en KOP, FUP y AWL (SIMATIC)

Ejemplo de un temporizador de retardo a la conexión memorizado

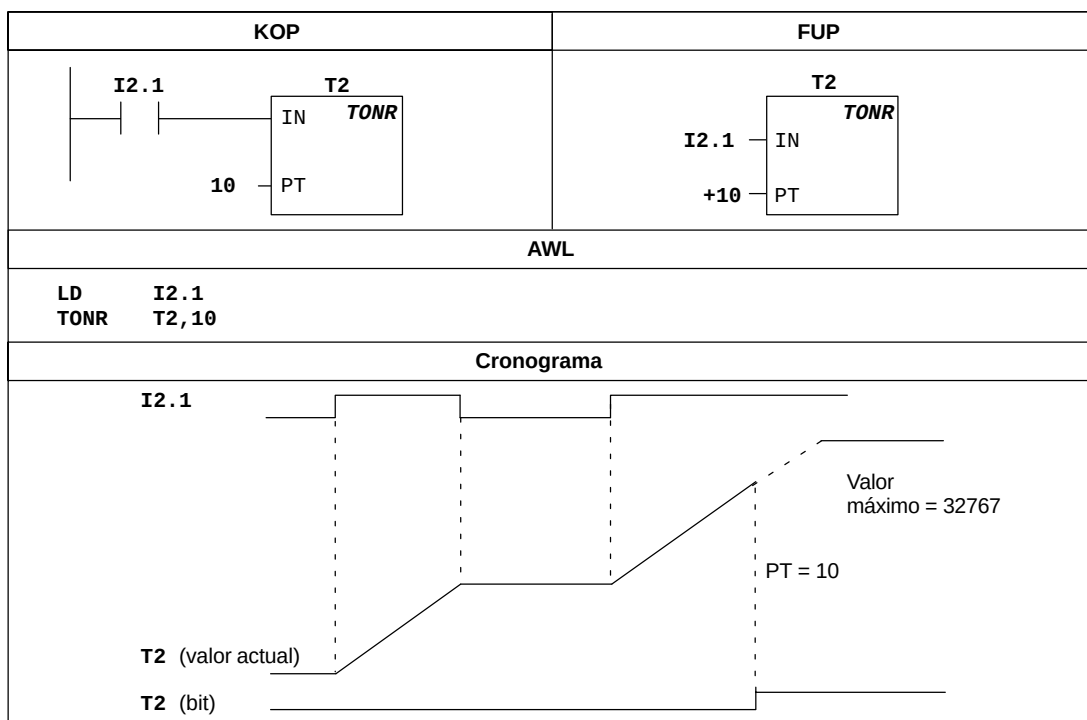


Figura 9-6 Ejemplo de un temporizador de retardo a la conexión memorizado en KOP, FUP y AWL (SIMATIC)

Ejemplo de un temporizador de retardo a la desconexión

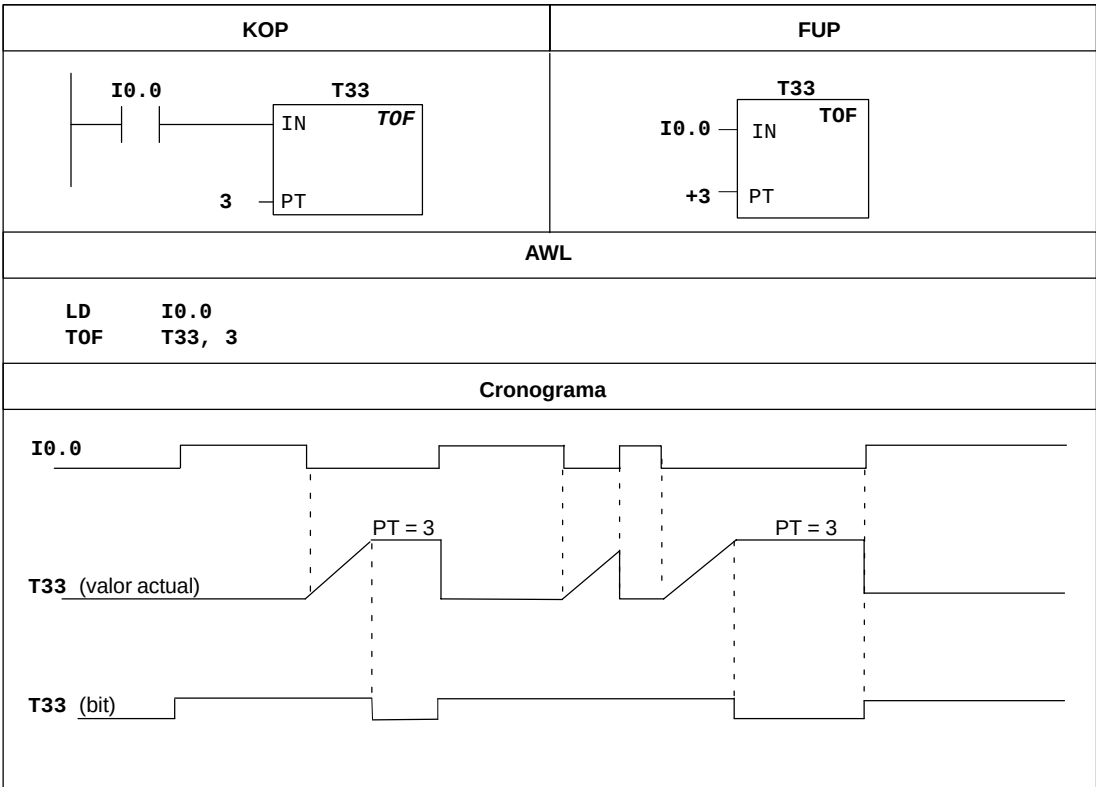
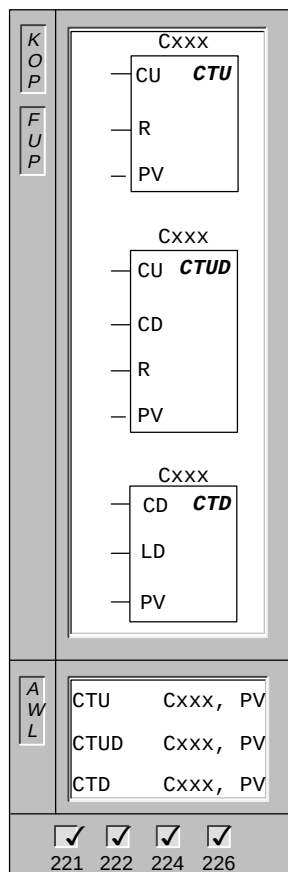


Figura 9-7 Ejemplo de un temporizador de retardo a la desconexión en KOP, FUP y AWL (SIMATIC)

9.4 Operaciones con contadores (SIMATIC)

Contar adelante, Contar adelante/atrás, Contar atrás



La operación **Contar adelante** empieza a contar hasta el valor máximo cuando se produce un flanco positivo en la entrada de contaje adelante (CU). Si el valor actual (Cxxx) es mayor o igual al valor de preselección (PV), se activa el bit de contaje (Cxxx). El contador se inicializa al activarse la entrada de desactivación (R) y para de contar cuando alcanza PV.

La operación **Contar adelante/atrás** empieza a contar adelante cuando se produce un flanco positivo en la entrada de contaje adelante (CU). Por el contrario, empieza a contar atrás cuando se produce un flanco positivo en la entrada de contaje atrás (CD). Si el valor actual (Cxxx) es mayor o igual al valor de preselección (PV), se activa el bit de contaje (Cxxx). El contador se inicializa al activarse la entrada de desactivación (R).

La operación **Contar atrás** empieza a contar atrás desde el valor de preselección cuando se produce un flanco positivo en la entrada de contaje atrás (CD). Si el valor actual es igual a cero, se activa el bit de contaje (Cxxx). El contador desactiva el bit de contaje (Cxxx) y carga el valor actual con el valor de preselección (PV) cuando se activa la entrada de carga (LD). El contador atrás se detiene al alcanzar el valor cero.

Márgenes de contaje: Cxxx=C0 hasta C255

En AWL, la entrada de desactivación CTU es el primer valor de la pila y la entrada de contaje adelante se carga en el segundo nivel de la pila.

En AWL, la entrada de desactivación CTUD es el primer valor de la pila, la entrada de contaje atrás se carga en el segundo nivel de la pila y la entrada de contaje adelante, en el tercer nivel.

En AWL, la entrada de carga CTD es el primer nivel de la pila y la entrada de contaje atrás es el valor cargado en el segundo nivel de la pila.

Entradas/salidas	Operandos	Tipos de datos
Cxxx	Constante	WORD
CU, CD, LD, R (KOP)	Circulación de corriente	BOOL
CU, CD, R, LD (FUP)	I, Q, M, SM, T, C, V, S, L, circulación de corriente	BOOL
PV	VW, IW, QW, MW, SMW, LW, SW, AIW, AC, T, C, constante, *VD, *AC, *LD	INT

Descripción de las operaciones de contaje del S7-200

La operación Contar adelante (CTU) empieza a contar adelante a partir del valor actual cuando se produce un flanco positivo en la entrada de contaje adelante. El contador se inicializa cuando se activa la entrada de desactivación o al ejecutarse la operación Poner a 0. El contador para de contar cuando se alcanza el valor máximo (32.767).

La operación Contar adelante/atrás (CTUD) empieza a contar adelante cuando se produce un flanco positivo en la entrada de contaje adelante, y empieza a contar atrás cuando se produce un flanco positivo en la entrada de contaje atrás. El contador se inicializa cuando se activa la entrada de desactivación o al ejecutarse la operación Poner a 0. Cuando se alcanza el valor máximo (32.767), el siguiente flanco positivo en la entrada de contaje adelante invertirá el contaje hasta alcanzar el valor mínimo (–32.768). Igualmente, cuando se alcanza el valor mínimo (–32.768), el siguiente flanco positivo en la entrada de contaje atrás invertirá el contaje hasta alcanzar el valor máximo (32.767).

Los contadores Contar adelante y Contar adelante/atrás tienen un valor actual que almacena el valor de contaje actual. También disponen de un valor de preselección (PV) que se compara con el valor actual cuando se ejecuta la operación de contaje. Si el valor actual es mayor o igual al valor de preselección, se activa el bit de contaje (bit C). En caso contrario, el bit se desactiva.

La operación Contar atrás empieza a contar atrás a partir del valor actual cuando se produce un flanco negativo en la entrada de contaje adelante. El contador desactiva el bit de contaje y carga el valor actual con el valor de preselección cuando se activa la entrada de carga. El contador se detiene al alcanzar el valor cero y el bit de contaje (bit C) se activa.

Cuando se inicializa un contador con la operación Poner a 0, se desactivan tanto el bit de contaje como el valor actual del contador. El número del contador se debe utilizar para direccionar tanto el valor actual como el bit C del contador.

Nota

Puesto que cada contador dispone sólo de un valor actual, no se podrá asignar un mismo número a varios contadores. (Los contadores Contar adelante, Contar adelante/atrás y Contar atrás acceden a un mismo valor actual).

Ejemplos de una operación de contaje

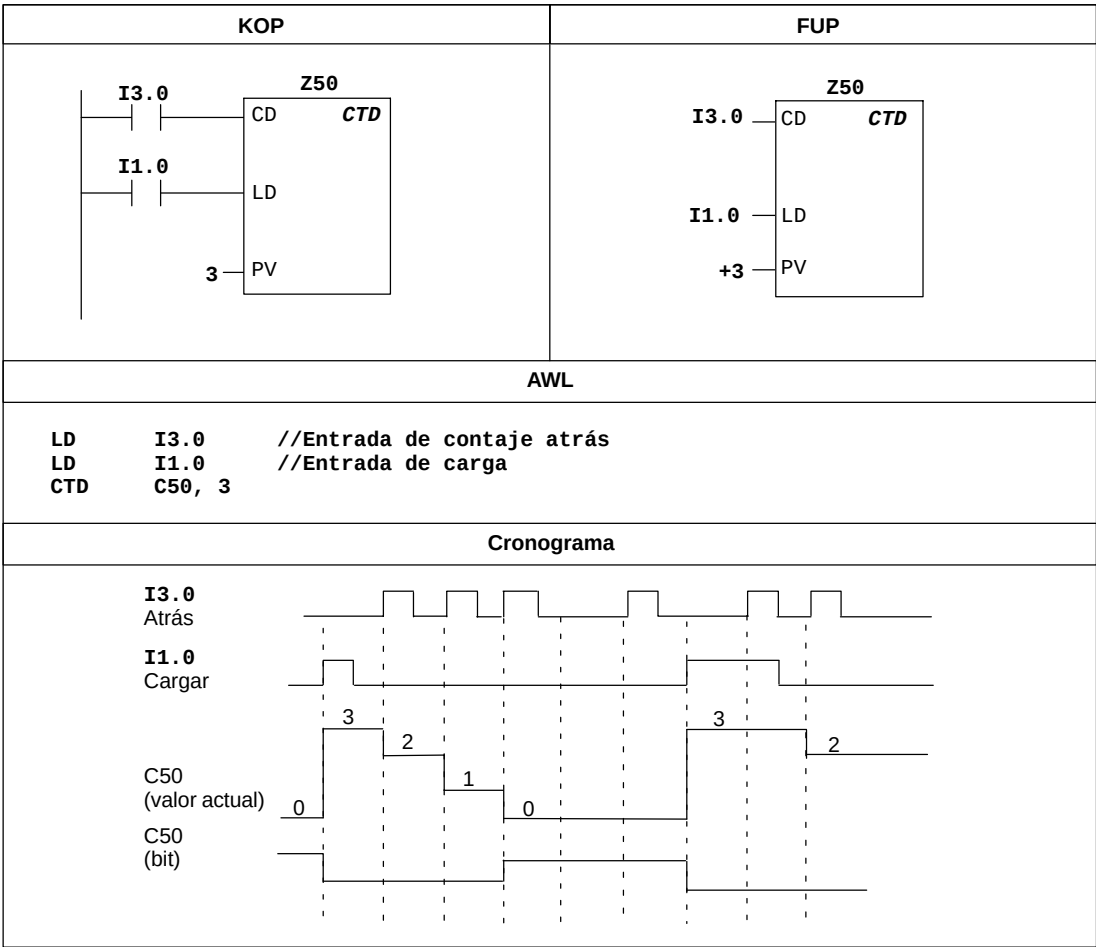


Figura 9-8 Ejemplo de una operación de contaje atrás en KOP, FUP y AWL (SIMATIC)

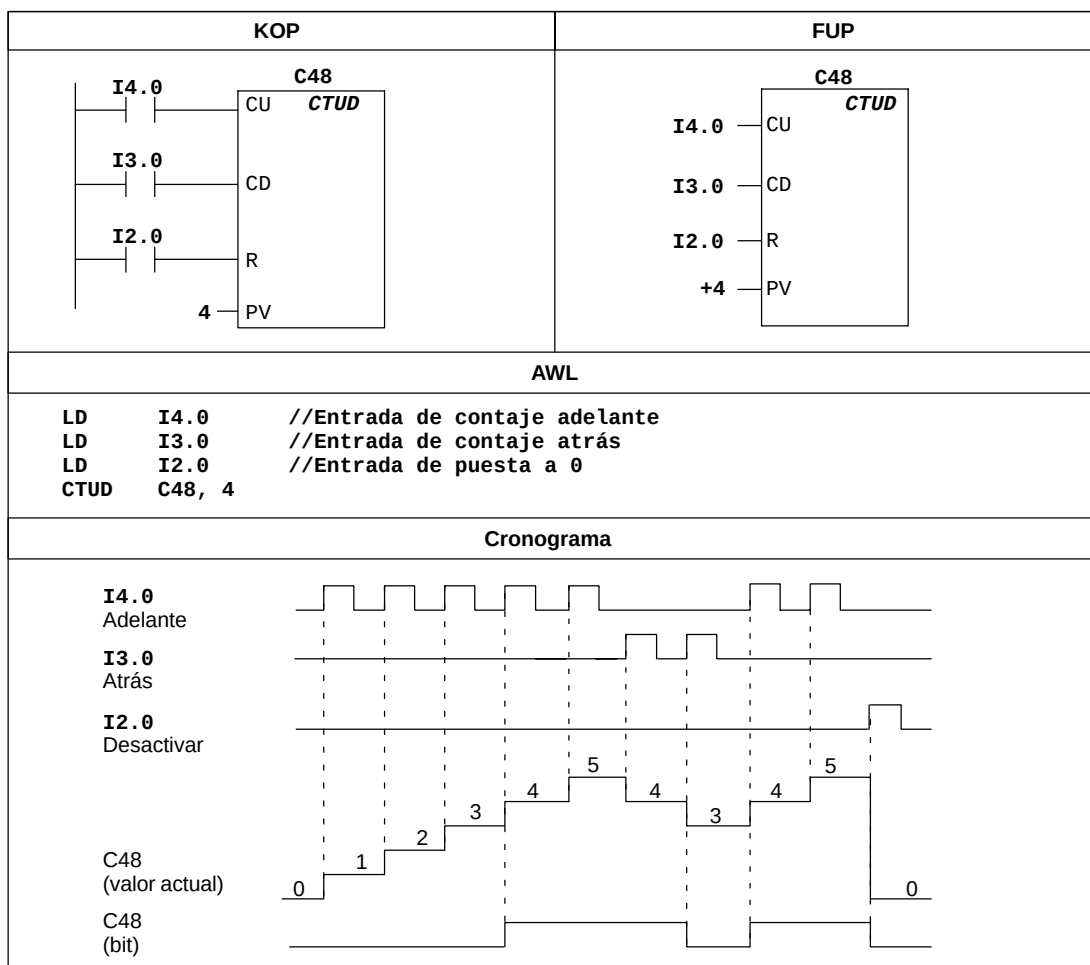
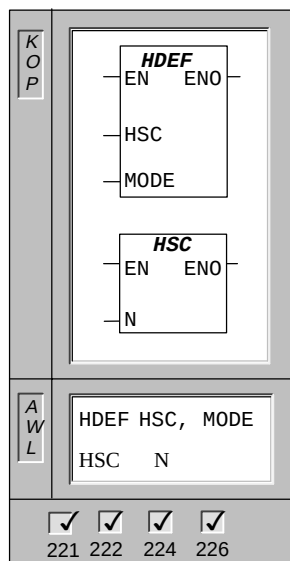


Figura 9-9 Ejemplo de una operación de conteje adelante/atrás en KOP, FUP y AWL (SIMATIC)

Definir modo para contador rápido, Activar contador rápido



La operación **Definir modo para contador rápido** asigna un modo (MODE) al contador rápido direccionado (HSC). Consulte la tabla 9-5 (pág. 9-33).

La operación **Activar contador rápido** configura y controla el funcionamiento del contador rápido direccionado, basándose en el estado de las marcas especiales del mismo. El parámetro N indica el número del contador rápido.

Las CPUs 221 y 222 no soportan los contadores HSC1 y HSC2.

Por cada contador rápido puede utilizarse sólo un cuadro HDEF.

HDEF: Condiciones de error que ponen ENO a 0:

SM4.3 (tiempo de ejecución), 0003 (conflicto de entradas), 0004 (operación no válida en interrupción), 000A (redefinición de HSC)

HSC: Condiciones de error que ponen ENO a 0:

SM4.3 (tiempo de ejecución), 0001 (HSC antes de HDEF), 0005 (operaciones HSC/PLS simultáneas)

Entradas/salidas	Operandos	Tipos de datos
HSC	Constante	BYTE
MODE	Constante	BYTE
N	Constante	WORD

Descripción de las operaciones con contadores rápidos

Los contadores rápidos cuentan eventos que se ejecutan más deprisa de lo que es posible controlarlos en los ciclos de la CPU. Estos contadores se pueden configurar como máximo para 12 modos de operación diferentes. La tabla 9-5 muestra los modos de los contadores. La frecuencia máxima de un contador rápido depende del tipo de CPU. Para obtener más información acerca de la CPU utilizada, consulte el Anexo A.

Todos los contadores disponen de entradas que soportan funciones tales como relojes, control del sentido, puesta a 0 y arranque. Para los contadores de dos fases, ambos relojes pueden funcionar a máxima frecuencia. En el caso de los contadores A/B, se puede elegir entre una velocidad máxima de conteo simple (1x) o cuádruple (4x). Todos los contadores funcionan a velocidades máximas sin interferirse mutuamente.

Utilización de los contadores rápidos

Los contadores rápidos se utilizan habitualmente como accionamiento para temporizadores que funcionan impulsados por un árbol que gira a un régimen constante y provisto de un encoder incremental. Éste último provee un número determinado de valores de conteo por giro, así como un impulso de puesta a 0 una vez por giro. El reloj (o relojes) y el impulso de puesta a 0 del encoder suministran las entradas para el contador rápido. El primero de los valores predeterminados se carga en el contador y las salidas deseadas se activan para el intervalo de tiempo en que el valor actual del contador es menor que el valor predeterminado. El contador se ajusta para que una interrupción se active cuando el conteo actual sea igual al predeterminado o cuando el contador se ponga a 0.

Cuando el valor actual es igual al predeterminado y se presenta un evento de interrupción, entonces se carga un nuevo valor predeterminado y se activa el siguiente estado de señal para las salidas. Si se produce un evento de interrupción porque el contador se ha inicializado, entonces se ajusta el primer valor predeterminado y los primeros estados de las salidas, repitiéndose el ciclo.

Puesto que las interrupciones se producen a una velocidad muy inferior a la de los contadores rápidos, es posible implementar un control preciso de las operaciones rápidas con un impacto relativamente bajo en el ciclo total del sistema de automatización. La posibilidad de asociar interrupciones a rutinas de interrupción permite cargar nuevos valores predeterminados en una rutina de interrupción separada, lo cual simplifica el control del estado, obteniéndose además un programa muy rectilíneo y fácil de leer. Obviamente, todos los eventos de interrupción se pueden ejecutar también en una sola rutina de interrupción. Para obtener más información a este respecto, consulte el apartado 9.15.

Descripción de los cronogramas de los contadores rápidos

Los siguientes cronogramas (figura 9-10 hasta figura 9-16) muestran cómo cada contador funciona conforme a su categoría. El funcionamiento de las entradas de puesta a 0 y de arranque se representa en dos diagramas por separado, siendo aplicable a todos los modos que utilizan dichas entradas. En los diagramas de las entradas de puesta a 0 y de arranque se ha programado la actividad alta para ambas entradas.

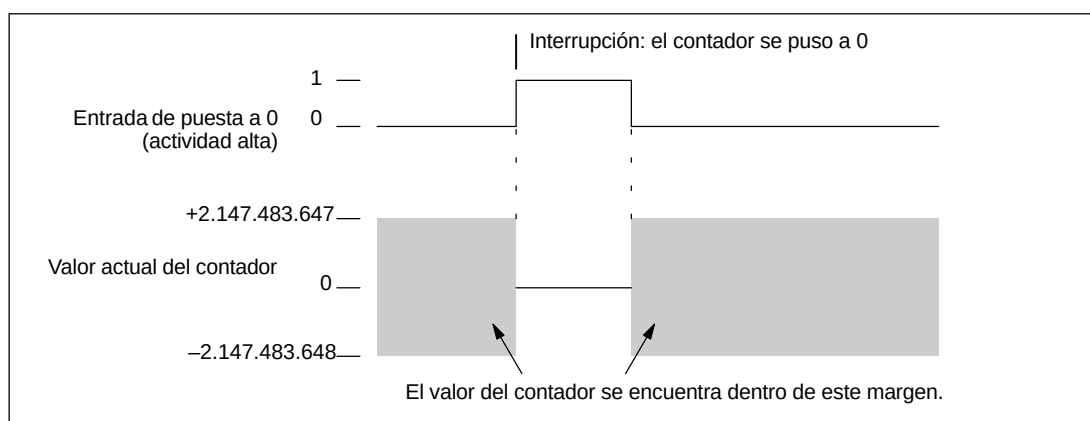


Figura 9-10 Ejemplo del funcionamiento con puesta a 0 y sin arranque

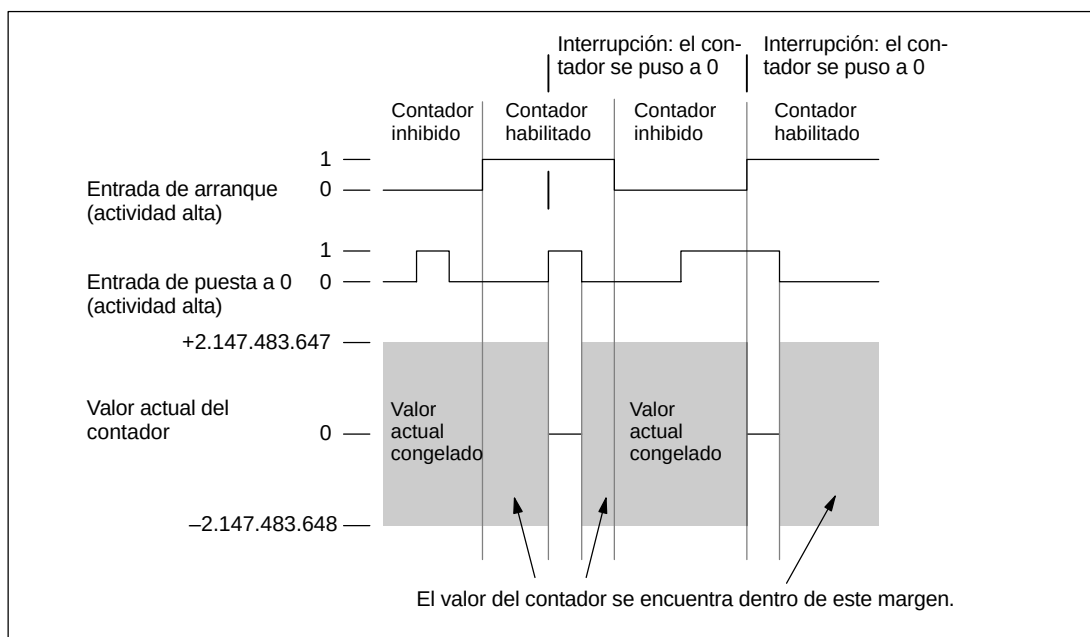


Figura 9-11 Ejemplo del funcionamiento con puesta a 0 y arranque

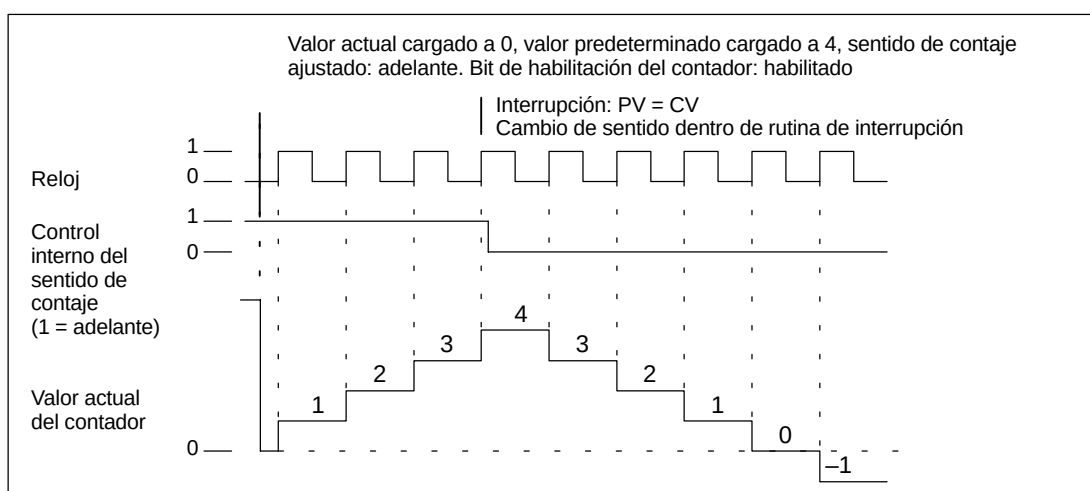


Figura 9-12 Ejemplo del funcionamiento de los modos 0, 1 ó 2

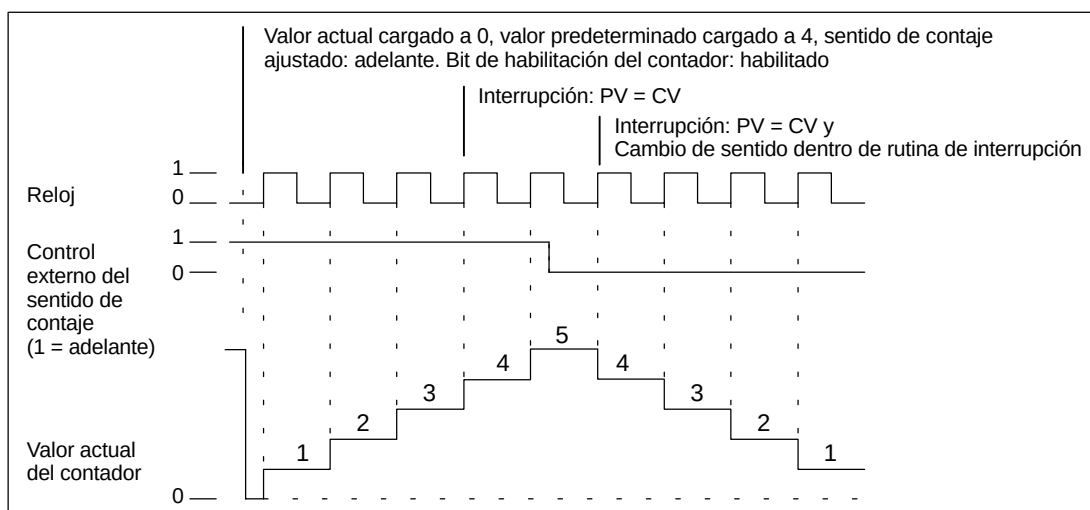


Figura 9-13 Ejemplo del funcionamiento de modos 3, 4 ó 5

Si se utilizan los modos de conteo 6, 7 u 8 y ocurre un flanco positivo tanto en la entrada de reloj de conteo adelante como en la de conteo atrás en menos de 0,3 microsegundos de diferencia, puede ser que el contador rápido considere simultáneos ambos eventos. En este caso, el valor actual permanecerá inalterado y tampoco cambiará el sentido de conteo. Si entre el flanco positivo de la entrada de conteo adelante y el flanco positivo de la entrada de conteo atrás transcurren más de 0,3 microsegundos, el contador rápido recibirá ambos eventos por separado. En ninguno de ambos casos se produce un error (v. figuras 9-14, 9-15 y 9-16).

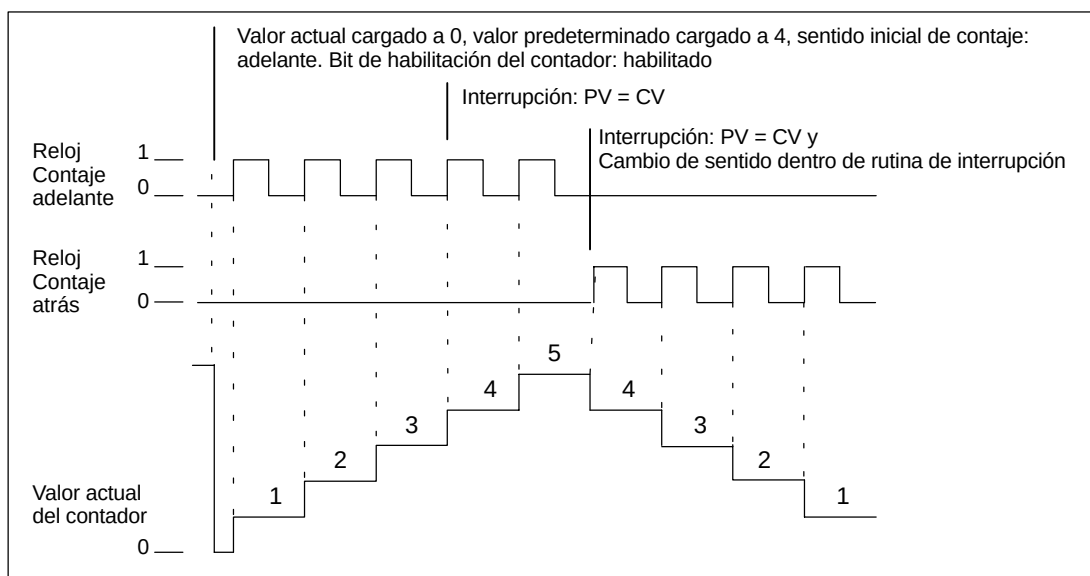


Figura 9-14 Ejemplo del funcionamiento de los modos 6, 7 u 8

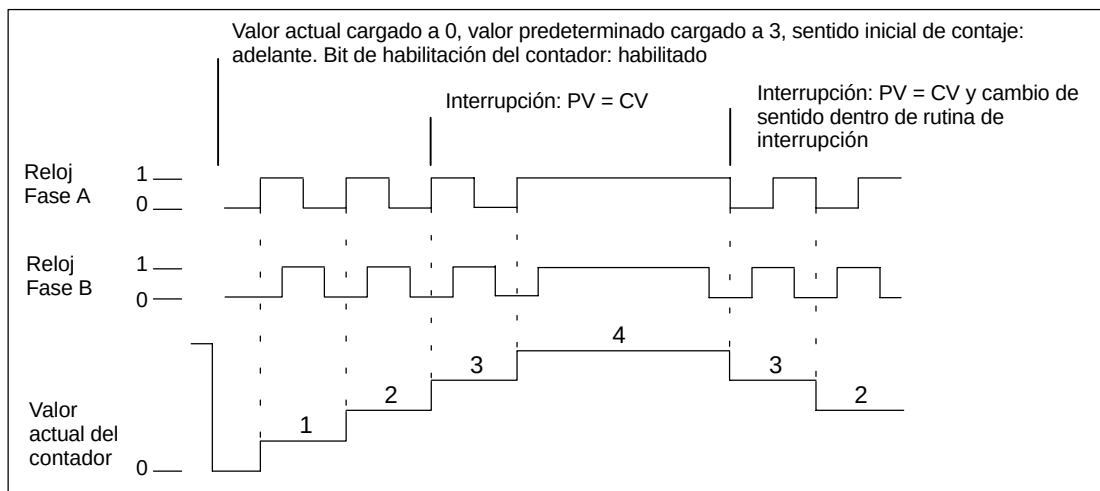


Figura 9-15 Ejemplo del funcionamiento de los modos 9, 10 u 11 (contador A/B, velocidad simple)

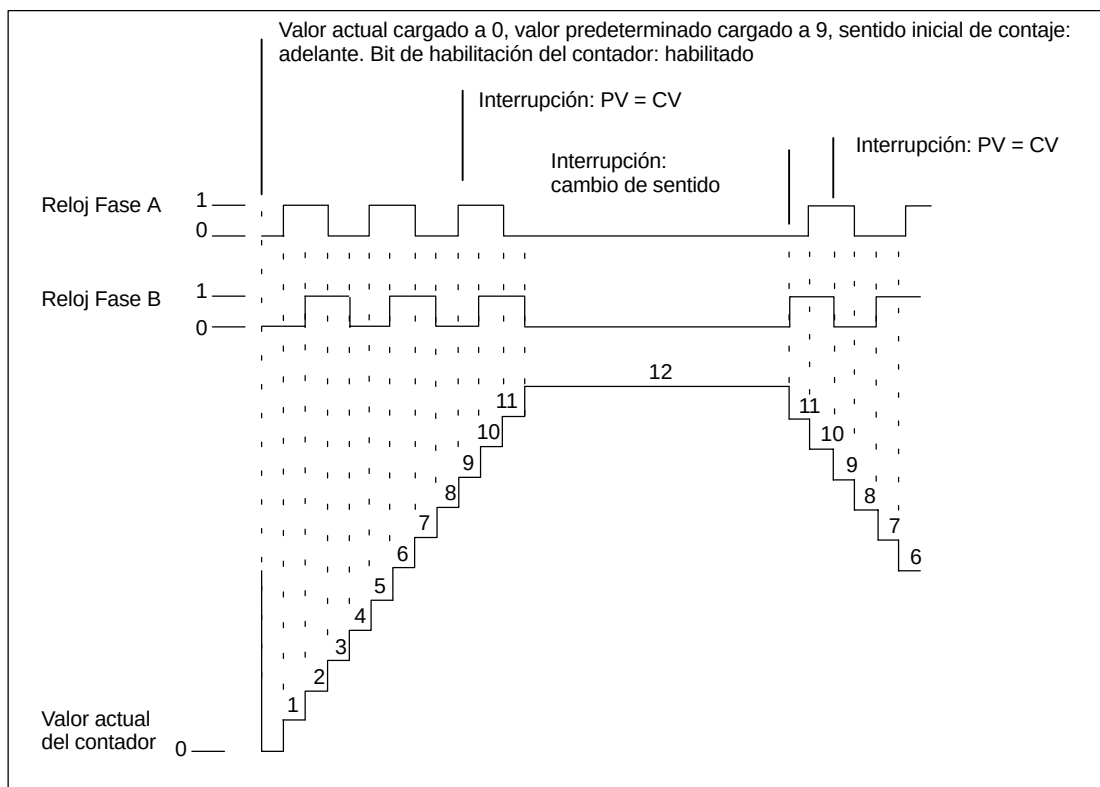


Figura 9-16 Ejemplo del funcionamiento de los modos 9, 10 u 11 (contador A/B, velocidad cuádruple)

Conectar el cableado de las entradas de los contadores rápidos

La tabla 9-3 muestra las entradas correspondientes al reloj, el control del sentido, la puesta a 0 y las funciones de arranque de los contadores rápidos. Estas funciones de entrada y los modos de operación de los contadores rápidos se describen en las tablas 9-5 a 9-10.

Tabla 9-3 Entradas para los contadores rápidos

Contador rápido	Entradas utilizadas
HSC0	I0.0, I0.1, 0.2
HSC1	I0.6, I0.7, I1.0, I1.1
HSC2	I1.2, I1.3, I1.4, I1.5
HSC3	I0.1
HSC4	I0.3, I0.4, I0.5
HSC5	I0.4

Como muestra el área sombreada de la tabla 9-4, la asignación de entradas de algunos contadores rápidos se solapa con las interrupciones de flanco. Una misma entrada no se puede utilizar para dos funciones diferentes. No obstante, cualquier entrada que no se esté utilizando en el modo actual del contador rápido se puede usar para otro fin. Por ejemplo, si HSC0 se está utilizando en modo 2 (que utiliza las entradas I0.0 e I0.2), I0.1 se podrá utilizar para interrupciones de flanco o para HSC3.

Si se utiliza un modo de HSC0 que no use la entrada I0.1, ésta se podrá emplear para HSC3 o para interrupciones de flanco. De forma similar, si I0.2 no se utiliza en el modo de HSC0 seleccionado, la entrada estará disponible para interrupciones de flanco. Asimismo, si I0.4 no se usa en el modo de HSC4 seleccionado, la entrada se podrá utilizar para HSC5. Es preciso tener en cuenta que todos los modos de HSC0 utilizan siempre I0.0 y que todos los de HSC4 usan siempre I0.3. Por tanto, dichas entradas nunca estarán disponibles para otros fines cuando se estén utilizando estos contadores.

Tabla 9-4 Asignación de entradas para los contadores rápidos y las interrupciones de flanco

Entrada (I)														
Elemento	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	1.0	1.1	1.2	1.3	1.4	1.5
HSC0	x	x	x											
HSC1							x	x	x	x				
HSC2											x	x	x	x
HSC3		x												
HSC4				x	x	x								
HSC5					x									
Interrupciones de flanco	x	x	x	x										

Tabla 9-5 Modos de operación de HSC0 (CPU 221, CPU 222, CPU 224 y CPU 226)

HSC0					
Modo	Descripción	I0.0	I0.1	I0.2	
0	Contador adelante/atrás de fase simple con control interno del sentido de contaje.	Reloj			
1	SM37.3 = 0, contaje atrás SM37.3 = 1, contaje adelante			Puesta a 0	
3	Contador adelante/atrás de fase simple con control externo del sentido de contaje.	Reloj	Sentido		
4	I0.1 = 0, contaje atrás I0.1 = 1, contaje adelante			Puesta a 0	
6	Contador de dos fases con entradas de reloj para contaje adelante/atrás	Reloj (adelante)	Reloj (atrás)		
7				Puesta a 0	
9	Contador A/B, la fase A está desfasada 90 grados respecto a B en sentido horario; la fase B está desfasada 90 grados respecto a A en sentido antihorario.	Reloj (fase A)	Reloj (fase B)		
10				Puesta a 0	

Tabla 9-6 Modos de operación de HSC1 (CPU 224 y CPU 226)

HSC1					
Modo	Descripción	I0.6	I 0.7	I1.0	I1.1
0	Contador adelante/atrás de fase simple con control interno del sentido de contaje. SM47.3 = 0, contaje atrás SM47.3 = 1, contaje adelante	Reloj			
1				Puesta a 0	
2					Arranque
3	Contador adelante/atrás de fase simple con control externo del sentido de contaje. I0.7 = 0, contaje atrás I0.7 = 1, contaje adelante	Reloj	Sentido		
4				Puesta a 0	
5					Arranque
6	Contador de dos fases con entradas de reloj para contaje adelante/atrás	Reloj (adelante)	Reloj (atrás)		
7				Puesta a 0	
8					Arranque
9	Contador A/B, la fase A está desfasada 90 grados respecto a B en sentido horario; la fase B está desfasada 90 grados respecto a A en sentido antihorario.	Reloj (fase A)	Reloj (fase B)		
10				Puesta a 0	
11					Arranque

Tabla 9-7 Modos de operación de HSC2 (CPU 224 y CPU 226)

HSC2					
Modo	Descripción	I1.2	I1.3	I1.4	I1.5
0	Contador adelante/atrás de fase simple con control interno del sentido de contaje. SM 57.3 = 0, contaje atrás SM 57.3 = 1, contaje adelante	Reloj			
1				Puesta a 0	
2					Arranque
3	Contador adelante/atrás de fase simple con control externo del sentido de contaje. I1.3 = 0, contaje atrás I1.3 = 1, contaje adelante	Reloj	Sentido		
4				Puesta a 0	
5					Arranque
6	Contador de dos fases con entradas de reloj para contaje adelante/atrás	Reloj (adelante)	Reloj (atrás)		
7				Puesta a 0	
8					Arranque
9	Contador A/B, la fase A está desfasada 90 grados respecto a B en sentido horario; la fase B está desfasada 90 grados respecto a A en sentido antihorario.	Reloj (fase A)	Reloj (fase B)		
10				Puesta a 0	
11					Arranque

Tabla 9-8 Modos de operación de HSC3 (CPU 221, CPU 222, CPU 224 y CPU 226)

HSC3					
Modo	Descripción	I0.1			
0	Contador adelante/atrás de fase simple con control interno del sentido de contaje. SM137.3 = 0, contaje atrás SM137.3 = 1, contaje adelante	Reloj			

Tabla 9-9 Modos de operación de HSC4 (CPU 221, CPU 222, CPU 224 y CPU 226)

HSC4					
Modo	Descripción	I0.3	I0.4	I0.5	
0	Contador adelante/atrás de fase simple con control interno del sentido de contaje. SM147.3 = 0, contaje atrás SM147.3 = 1, contaje adelante	Reloj			
1				Puesta a 0	
3	Contador adelante/atrás de fase simple con control externo del sentido de contaje. I0.4 = 0, contaje atrás I0.4 = 1, contaje adelante	Reloj	Sentido		
4				Puesta a 0	
6	Contador de dos fases con entradas de reloj para contaje adelante/atrás	Reloj (adelante)	Reloj (atrás)		
7				Puesta a 0	
9	Contador A/B, la fase A está desfasada 90 grados respecto a B en sentido horario, la fase B está desfasada 90 grados respecto a A en sentido antihorario.	Reloj (fase A)	Reloj (fase B)		
10				Puesta a 0	

Tabla 9-10 Modos de operación de HSC5 (CPU 221, CPU 222, CPU 224 y CPU 226)

HSC5					
Modo	Descripción	I0.4			
0	Contador adelante/atrás de fase simple con control interno del sentido de contaje. SM157.3 = 0, contaje atrás SM157.3 = 1, contaje adelante	Reloj			

Direccionamiento de los contadores rápidos (HC)

Para acceder al valor de conteo del contador rápido, se indica la dirección del mismo (utilizando el identificador HC) y el número del contador (p.ej. HC0). El valor actual del contador rápido es de sólo lectura, pudiéndose acceder al mismo sólo en formato de palabra doble (32 bits), como muestra la figura 9-17.

Formato: **HC[número del contador rápido]** **HC 2**

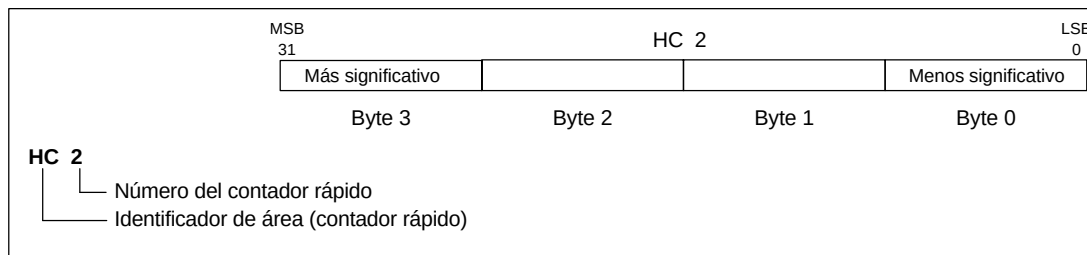


Figura 9-17 Acceso a los valores actuales del contador rápido

Descripción de los diferentes contadores rápidos

Todos los contadores funcionan de la misma manera en el mismo modo de operación. Como muestra la tabla 9-5, hay cuatro tipos básicos de contadores. Es preciso tener en cuenta que no todos los contadores soportan todos los modos. Todos los contadores se pueden utilizar sin entrada de puesta a 0 ni de arranque, con entrada de puesta a 0 pero sin entrada de arranque, o bien, con entrada de puesta a 0 y de arranque.

Activando la entrada de puesta a 0 se borra el valor actual del contador hasta que vuelve a ser desactivada. Al activarse la entrada de arranque se habilita el contador. Si se desactiva esta entrada se mantiene el valor actual del contador, ignorándose los eventos de reloj. Si se activa la entrada de puesta a 0 mientras está desactivada la entrada del arranque, se ignorará la activación de la entrada de puesta a 0, con lo que no se modificará el valor actual. Si la entrada de arranque se activa mientras está activada la entrada de puesta a 0, el valor actual se borrará.

Antes de poder utilizar un contador rápido es preciso elegir su modo de operación. Para ello se utiliza la operación HDEF (Definir modo para contador rápido). HDEF establece la conexión entre un contador rápido (HSCx) y el modo de conteo. Por cada contador sólo se puede ejecutar una operación HDEF. Un contador rápido se define utilizando la marca del primer ciclo SM0.1 (este bit se activa sólo en el primer ciclo y se desactiva posteriormente) para llamar a la subrutina que contiene la operación HDEF.

Elegir el nivel de actividad y el modo de conteo simple o cuádruple

Cuatro contadores tienen tres marcas de control que se utilizan para configurar el estado activo de las entradas de puesta a 0 y de arranque, así como para seleccionar la velocidad simple o cuádruple (ésto sólo en los contadores A/B). Estas marcas están depositadas en el byte de control del respectivo contador y se emplean solamente cuando se ejecuta la operación HDEF. La tabla 9-11 muestra las marcas.

Antes de poder ejecutar la operación HDEF es preciso ajustar las marcas de control de HSC1 y HSC2 al estado deseado. De lo contrario, el contador adoptará la configuración predeterminada del modo de conteo elegido. El ajuste estándar de las entradas de puesta a 0 y de arranque es de actividad alta, y la velocidad de conteo es la cuádruple (es decir, la frecuencia del reloj de entrada multiplicada por cuatro). Una vez ejecutada la operación HDEF, ya no se podrá modificar el ajuste de los contadores, a menos que la CPU se cambie a modo STOP.

Tabla 9-11 Nivel de actividad de las entradas de puesta a 0 y de arranque, marcas para elegir la velocidad simple o cuádruple

HSC0	HSC1	HSC2	HSC4	Descripción (sólo cuando se ejecuta HDEF)
SM37.0	SM47.0	SM57.0	SM147.0	Bit de control para nivel de actividad de la entrada de puesta a 0: 0 = actividad alta; 1 = actividad baja
—	SM47.1	SM57.1	—	Bit de control para nivel de actividad de la entrada de arranque: 0 = actividad alta; 1 = actividad baja
SM37.2	SM47.2	SM57.2	SM147.2	Velocidad de conteo de los contadores A/B: 0 = velocidad cuádruple; 1 = velocidad simple

Byte de control

Una vez definido el contador y el modo de conteo se deben programar los parámetros dinámicos del mismo. Todos los contadores rápidos disponen de un byte que los habilita o inhibe, fijando el sentido de control (sólo en los modos 0, 1 y 2). El byte de control determina asimismo el sentido de conteo inicial para todos los modos restantes, así como el valor actual y el valor predeterminado que se cargarán. El byte de control, los valores actuales asignados y los valores predeterminados se comprueban al ejecutarse la operación HSC. La tabla 9-12 describe cada una de las marcas del byte de control.

Tabla 9-12 Marcas de control de HSC0, HSC1 y HSC2

HSC0	HSC1	HSC2	HSC3	HSC4	HSC5	Descripción
SM37.3	SM47.3	SM57.3	SM137.3	SM147.3	SM157.3	Bit de control para el sentido de conteo: 0 = conteo atrás; 1 = conteo adelante
SM37.4	SM47.4	SM57.4	SM137.4	SM147.4	SM157.4	Escribir el sentido de conteo en el contador rápido: 0 = no actualizar; 1 = actualizar el sentido de conteo
SM37.5	SM47.5	SM57.5	SM137.5	SM147.5	SM157.5	Escribir el nuevo valor predeterminado en el contador rápido: 0 = no actualizar; 1 = actualizar el valor predeterminado
SM37.6	SM47.6	SM57.6	SM137.6	SM147.6	SM157.6	Escribir el nuevo valor actual en el contador rápido: 0 = no actualizar; 1 = actualizar el valor actual
SM37.7	SM47.7	SM57.7	SM137.7	SM147.7	SM157.7	Habilitar el contador rápido: 0 = inhibir el contador rápido; 1 = habilitar el contador rápido

Ajustar los valores actuales y predeterminados

Todos los contadores rápidos disponen de un valor actual y de un valor predeterminado de 32 bits. Ambos son valores enteros con signo. Para cargar un nuevo valor actual o predeterminado en el contador rápido es preciso activar el byte de control y los bytes de las marcas especiales que contienen los valores actuales y/o predeterminados. Después se ejecuta la operación HSC para transferir los nuevos valores al contador rápido. La tabla 9-13 describe los bytes de marcas especiales que contienen los nuevos valores y los valores predeterminados.

Además de los bytes de control y de los bytes que contienen los nuevos valores predeterminados, también se puede leer el valor actual de cada uno de los contadores rápidos, utilizando el tipo de datos HC (valor actual del contador rápido) seguido del número de contador (0, 1, 2, 3, 4 o 5). Ello permite acceder directamente al valor actual para operaciones de lectura. Por el contrario, este valor sólo se puede escribir utilizando la operación HSC que se describe más arriba.

Tabla 9-13 Valores actuales y predeterminado de los contadores HSC0, HSC1, HSC2, HSC3, HSC4 y HSC5

Valor a cargar	HSC0	HSC1	HSC2	HSC3	HSC4	HSC5
Nuevo actual	SMD38	SMD48	SMD58	SMD138	SMD148	SMD158
Nuevo valor predeterminado	SMD42	SMD52	SMD62	SMD142	SMD152	SMD162

Byte de estado

Todos los contadores rápidos disponen de un byte para marcas de estado. Éstas indican el sentido de conteo actual y si el valor actual es igual o mayor que el valor predeterminado. La tabla 9-14 muestra las marcas de estado de los contadores rápidos.

Tabla 9-14 Marcas de estado de los contadores HSC0, HSC1, HSC2, HSC3, HSC4 y HSC5

HSC0	HSC1	HSC2	HSC3	HSC4	HSC5	Descripción
SM36.0	SM46.0	SM56.0	SM136.0	SM146.0	SM156.0	No utilizado
SM36.1	SM46.1	SM56.1	SM136.1	SM146.1	SM156.1	No utilizado
SM36.2	SM46.2	SM56.2	SM136.2	SM146.2	SM156.2	No utilizado
SM36.3	SM46.3	SM56.3	SM136.3	SM146.3	SM156.3	No utilizado
SM36.4	SM46.4	SM56.4	SM136.4	SM146.4	SM156.4	No utilizado
SM36.5	SM46.5	SM56.5	SM136.5	SM146.5	SM156.5	Bit de estado del sentido de conteo actual: 0 = conteo atrás; 1 = conteo adelante
SM36.6	SM46.6	SM56.6	SM136.6	SM146.6	SM156.6	El valor actual es igual al bit de estado del valor predeterminado: 0 = diferente, 1 = igual
SM36.7	SM46.7	SM56.7	SM136.7	SM146.7	SM156.7	El valor actual es mayor que el bit de estado del valor predeterminado: 0 = menor o igual, 1 = mayor que

Nota

Las marcas de estado son válidas únicamente mientras se está ejecutando la rutina de interrupción para el contador rápido. El estado del contador rápido se supervisa con objeto de habilitar las interrupciones para los eventos que puedan afectar a la operación que se está ejecutando.

Interrupciones de los contadores rápidos

Todos los modos de los contadores soportan una interrupción si el valor actual es igual al valor predeterminado. Los modos de los contadores que utilizan una entrada de puesta a 0 externa soportan una interrupción que se ejecuta cuando se activa dicha entrada. Todos los modos de conteo (con excepción de los modos 0, 1 y 2) soportan una interrupción que se ejecuta cuando se produce un cambio del sentido de conteo. Cada una de estas condiciones puede habilitarse o inhibirse por separado. Para obtener más información acerca de la utilización de interrupciones, consulte el apartado 9.15.

Nota

Cuando esté utilizando la entrada de puesta a 0 externa, no intente cargar un nuevo valor actual ni tampoco inhibir y habilitar luego el contador rápido desde la rutina de interrupción asociada a ese evento. En tal caso podría producirse un error fatal.

A continuación se describen las secuencias de inicialización y de operación de los contadores rápidos para facilitar la comprensión de su funcionamiento. En las siguientes descripciones se ha utilizado el contador HSC1 a título de ejemplo. En las explicaciones relativas a la inicialización se supone que el sistema de automatización S7-200 está en modo RUN y que, por consiguiente, la marca del primer ciclo es verdadera. En otro caso se deberá tener en cuenta que la operación HDEF sólo puede ejecutarse una vez por cada contador rápido, después de haber cambiado a modo RUN. Si la operación HDEF se ejecuta por segunda vez para un contador rápido se producirá un error de tiempo de ejecución. Los ajustes del contador permanecerán entonces tal y como se configuraron con la primera operación HDEF que se ejecutó para el contador en cuestión.

Modos de inicialización 0, 1 ó 2

Para inicializar HSC1 como contador adelante/atrás de fase simple con control interno del sentido de conteo (modos 0, 1 ó 2):

1. Con la marca del primer ciclo, llame a una subrutina para ejecutar la inicialización. Puesto que se utiliza una llamada a subrutina, los siguientes ciclos ya no llaman a la misma, con lo cual se acorta el tiempo de ciclo y el programa queda mejor estructurado.
2. Cargue la marca SMB47 en la subrutina de inicialización conforme a la operación de control deseada. Por ejemplo:
 SMB47 = 16#F8 Resultados:
 Se habilita el contador.
 Se escribe un nuevo valor actual.
 Se escribe un nuevo valor predeterminado.
 Se ajusta el sentido de conteo adelante.
 Se ajusta la actividad alta de las entradas de arranque y de puesta a 0.
3. Ejecute la operación HDEF con la entrada HSC puesta a 1 y la entrada MODE a 0 para puesta a 0 o arranque no externos, o bien a 1 para puesta a 0 externa y sin arranque, o bien a 2 para puesta a 0 y arranque externos.
4. Cargue el valor actual deseado en SMD48 (valor de palabra doble) (cargue 0 para borrar la marca).
5. Cargue el valor predeterminado deseado en SMD52 (valor de palabra doble).
6. Para averiguar si el valor actual es igual al predeterminado, programe una interrupción asociando el evento de interrupción CV = PV (evento 13) a una rutina de interrupción. Para obtener más información acerca de la utilización de interrupciones, consulte el apartado 9.15.
7. Para poder detectar una puesta a 0 externa, programe una interrupción asociando el evento de interrupción Puesta a 0 externa (evento 15) a una rutina de interrupción.
8. Ejecute la operación Habilitar todos los eventos de interrupción (ENI) para habilitar las interrupciones.
9. Ejecute la operación HSC para que el sistema de automatización S7-200 programe el contador HSC1.
10. Finalice la subrutina.

Modos de inicialización 3, 4 ó 5

Para inicializar HSC1 como contador adelante/atrás de fase simple con control externo del sentido de conteo (modos 3, 4 ó 5):

1. Con la marca del primer ciclo, llame a una subrutina para ejecutar la inicialización.
Puesto que se utiliza una llamada a subrutina, los siguientes ciclos ya no llaman a la misma, con lo cual se acorta el tiempo de ciclo y el programa queda mejor estructurado.
2. Cargue la marca SMB47 en la subrutina de inicialización conforme a la operación de control deseada. Por ejemplo:
SMB47 = 16#F8 Resultados:
Se habilita el contador.
Se escribe un nuevo valor actual.
Se escribe un nuevo valor predeterminado.
Se ajusta el sentido inicial de conteo adelante.
Se ajusta la actividad alta de las entradas de arranque y de puesta a 0.
3. Ejecute la operación HDEF con la entrada HSC puesta a 1 y la entrada MODE a 3 para puesta a 0 o arranque no externos, o bien a 4 para puesta a 0 externa y sin arranque, o bien a 5 para puesta a 0 y arranque externos.
4. Cargue el valor actual deseado en SMD48 (valor de palabra doble) (cargue 0 para borrar la marca).
5. Cargue el valor predeterminado deseado en SMD52 (valor de palabra doble).
6. Para averiguar si el valor actual es igual al predeterminado, programe una interrupción asociando el evento de interrupción CV = PV (evento 13) a una rutina de interrupción. Para obtener más información acerca de la utilización de interrupciones, consulte el apartado 9.15.
7. Para poder detectar un cambio del sentido de conteo, programe una interrupción asociando el evento de interrupción de cambio de sentido (evento 14) a una rutina de interrupción.
8. Para poder detectar una puesta a 0 externa, programe una interrupción asociando el evento de interrupción Puesta a 0 externa (evento 15) a una rutina de interrupción.
9. Ejecute la operación Habilitar todos los eventos de interrupción (ENI) para habilitar las interrupciones.
10. Ejecute la operación HSC para que el sistema de automatización S7-200 programe el contador HSC1.
11. Finalice la subrutina.

Modos de inicialización 6, 7 u 8

Para inicializar HSC1 como contador adelante/atrás de dos fases con relojes adelante/atrás (modos 6, 7 u 8):

1. Con la marca del primer ciclo, llame a una subrutina para ejecutar la inicialización. Puesto que se utiliza una llamada a subrutina, los siguientes ciclos ya no llaman a la misma, con lo cual se acorta el tiempo de ciclo y el programa queda mejor estructurado.
2. Cargue la marca SMB47 en la subrutina de inicialización conforme a la operación de control deseada. Por ejemplo:
 SMB47 = 16#F8 Resultados:
 Se habilita el contador.
 Se escribe un nuevo valor actual.
 Se escribe un nuevo valor predeterminado.
 Se ajusta el sentido inicial de conteo adelante.
 Se ajusta la actividad alta de las entradas de arranque y de puesta a 0.
3. Ejecute la operación HDEF con la entrada HSC puesta a 1 y la entrada MODE a 6 para puesta a 0 o arranque no externos, o bien a 7 para puesta a 0 externa y sin arranque, o bien a 8 para puesta a 0 y arranque externos.
4. Cargue el valor actual deseado en SMD48 (valor de palabra doble) (cargue 0 para borrar la marca).
5. Cargue el valor predeterminado deseado en SMD52 (valor de palabra doble).
6. Para averiguar si el valor actual es igual al predeterminado, programe una interrupción asociando el evento de interrupción CV = PV (evento 13) a una rutina de interrupción. Para obtener más información acerca de la utilización de interrupciones, consulte el apartado 9.15.
7. Para poder detectar un cambio del sentido de conteo, programe una interrupción asociando el evento de interrupción de cambio de sentido (evento 14) a una rutina de interrupción.
8. Para poder detectar una puesta a 0 externa, programe una interrupción asociando el evento de interrupción Puesta a 0 externa (evento 15) a una rutina de interrupción.
9. Ejecute la operación Habilitar todos los eventos de interrupción (ENI) para habilitar las interrupciones.
10. Ejecute la operación HSC para que el sistema de automatización S7-200 programe el contador HSC1.
11. Finalice la subrutina.

Modos de inicialización 9, 10 u 11

Para inicializar HSC1 como contador A/B (modos 9, 10 u 11):

1. Con la marca del primer ciclo, llame a una subrutina para ejecutar la inicialización. Puesto que se utiliza una llamada a subrutina, los siguientes ciclos ya no llaman a la misma, con lo cual se acorta el tiempo de ciclo y el programa queda mejor estructurado.
2. Cargue la marca SMB47 en la subrutina de inicialización conforme a la operación de control deseada.

Ejemplo (frecuencia simple):

SMB47 = 16#FC Resultados:

Se habilita el contador.
Se escribe un nuevo valor actual.
Se escribe un nuevo valor predeterminado.
Se ajusta el sentido inicial de conteo adelante.
Se ajusta la actividad alta de las entradas de arranque y de puesta a 0.

Ejemplo (frecuencia cuádruple):

SMB47 = 16#F8 Resultados:

Se habilita el contador.
Se escribe un nuevo valor actual.
Se escribe un nuevo valor predeterminado.
Se ajusta el sentido inicial de conteo adelante.
Se ajusta la actividad alta de las entradas de arranque y de puesta a 0.

3. Ejecute la operación HDEF con la entrada HSC puesta a 1 y la entrada MODE a 9 para puesta a 0 o arranque no externos, o bien a 10 para puesta a 0 externa y sin arranque, o bien a 11 para puesta a 0 y arranque externos.
4. Cargue el valor actual deseado en SMD48 (valor de palabra doble) (cargue 0 para borrar la marca).
5. Cargue el valor predeterminado deseado en SMD52 (valor de palabra doble).
6. Para averiguar si el valor actual es igual al predeterminado, programe una interrupción asociando el evento de interrupción CV = PV (evento 13) a una rutina de interrupción. Para obtener más información acerca de la utilización de interrupciones, consulte el apartado 9.15.
7. Para poder detectar un cambio del sentido de conteo, programe una interrupción asociando el evento de interrupción de cambio de sentido (evento 14) a una rutina de interrupción.
8. Para poder detectar una puesta a 0 externa, programe una interrupción asociando el evento de interrupción Puesta a 0 externa (evento 15) a una rutina de interrupción.
9. Ejecute la operación Habilitar todos los eventos de interrupción (ENI) para habilitar las interrupciones.
10. Ejecute la operación HSC para que el sistema de automatización S7-200 programe el contador HSC1.
11. Finalice la subrutina.

Cambio de sentido en los modos 0, 1 ó 2

Para configurar el cambio de sentido de HSC1 como contador de fase simple con control interno del sentido de conteo (modos 0,1 ó 2):

1. Cargue SMB47 para escribir la dirección deseada:
SMB47 = 16#90 Habilita el contador.
Ajusta el sentido de conteo atrás.
SMB47 = 16#98 Habilita el contador.
Ajusta el sentido de conteo adelante.
2. Ejecute la operación HSC para que el sistema de automatización S7-200 programe el contador HSC1.

Cargar nuevo valor actual (cualquier modo)

Si se modifica el valor actual, el contador se inhibirá automáticamente. Mientras está inhibido el contador, no cuenta ni tampoco se generan interrupciones.

Para modificar el valor actual del contador HSC1 (cualquier modo):

1. Cargue SMB47 para escribir el valor actual deseado:
SMB47 = 16#C0 Habilita el contador.
Escribe el nuevo valor actual.
2. Cargue el valor actual deseado en SMD48 (valor de palabra doble) (cargue 0 para borrar la marca).
3. Ejecute la operación HSC para que el sistema de automatización S7-200 programe el contador HSC1.

Cargar nuevo valor predeterminado (cualquier modo)

Para modificar el valor predeterminado de HSC1 (cualquier modo):

1. Cargue SMB47 para escribir el valor predeterminado deseado:
SMB47 = 16#A0 Habilita el contador.
Escribe el nuevo valor predeterminado.
2. Cargue el valor predeterminado deseado en SMD52 (valor de palabra doble).
3. Ejecute la operación HSC para que el sistema de automatización S7-200 programe el contador HSC1.

Inhibir un contador rápido (cualquier modo)

Para inhibir el contador rápido HSC1 (cualquier modo):

1. Cargue SMB47 para inhibir el contador:
SMB47 = 16#00 Inhibe el contador.
2. Ejecute la operación HSC para inhibir el contador.

Arriba se describe cómo modificar de forma individual el sentido de contaje, el valor actual o el valor predeterminado. No obstante, también se pueden cambiar todos o sólo algunos de estos ajustes en ese mismo orden, definiendo el valor de SMB47 de forma apropiada y ejecutando luego la operación HSC.

Ejemplo de un contador rápido

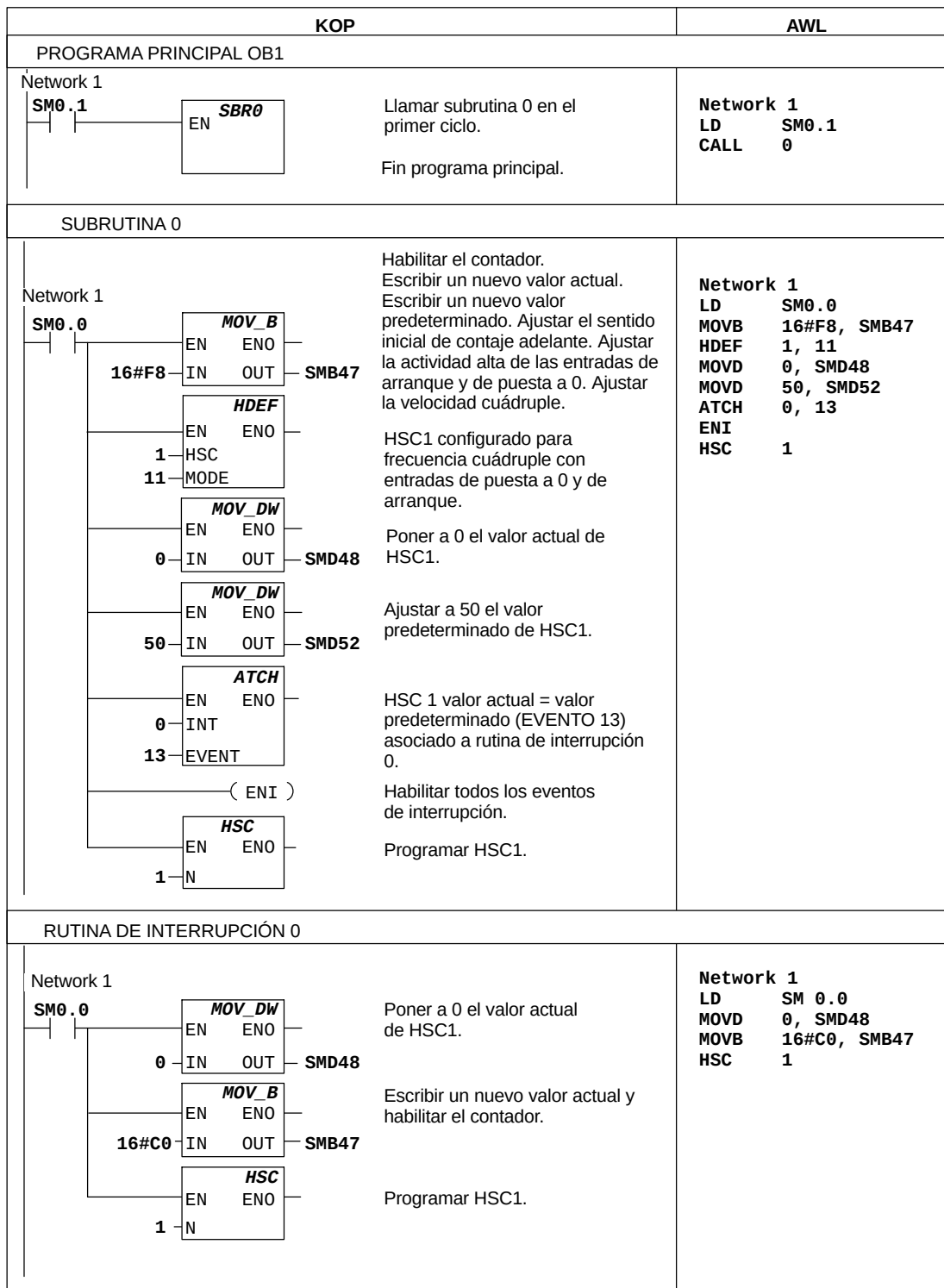


Figura 9-18 Ejemplo de inicialización de HSC1 en KOP y AWL (SIMATIC)

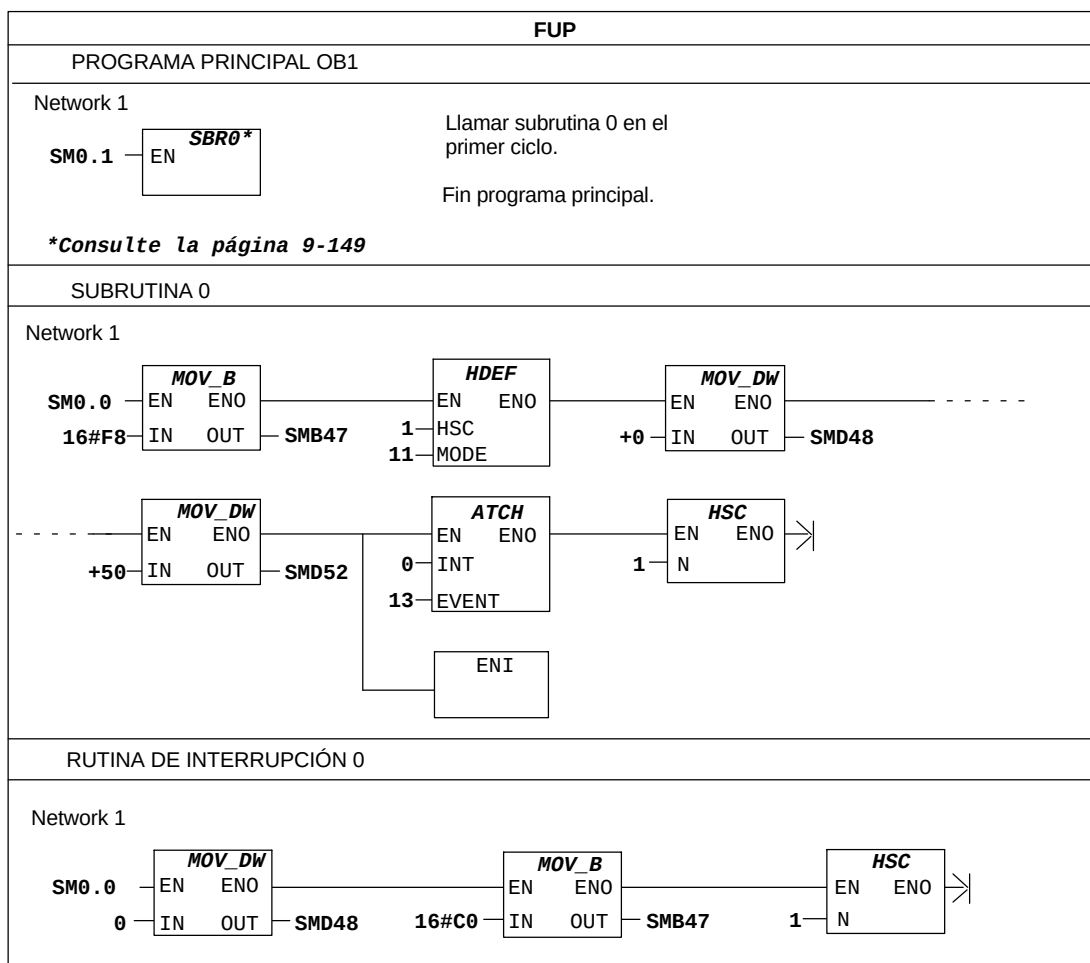
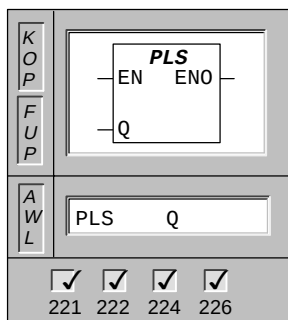


Figura 9-19 Ejemplo de inicialización de HSC1 en FUP (SIMATIC)

Salida de impulsos



La operación **Salida de impulsos** examina las marcas especiales de la salida de impulsos (Q0.0 ó Q0.1). A continuación se llama la operación de impulsos definida por las marcas especiales.

Operandos: Q Constante (0 ó 1)

Tipos de datos: WORD

Márgenes de salida de impulsos Q0.0 hasta Q0.1

Descripción de las operaciones rápidas de salida del S7-200

Todas las CPUs disponen de dos generadores PTO/PWM para producir trenes de impulsos rápidos y formas de onda moduladas por ancho de impulsos. Uno de estos generadores está asignado a la salida digital Q0.0 y, el otro, a la salida digital Q0.1.

Los generadores PTO/PWM y la imagen del proceso comparten el uso de las salidas Q0.0 y Q0.1. Cuando está activa una función PTO o PWM en Q0.0 ó Q0.1, el generador PTO/PWM controla la salida, inhibiéndose el uso normal de la misma. La forma de onda de la salida no se ve afectada por el estado de la imagen del proceso, ni por el valor forzado de la salida o la ejecución de las operaciones directas de salida. Si el generador PTO/PWM está desactivado, el control de la salida retorna a la imagen del proceso. La imagen del proceso determina los estados inicial y final de la forma de onda de la salida, para que la forma de onda comience y termine en un nivel alto o bajo.

Nota

Se recomienda ajustar a 0 la imagen del proceso de Q0.0 y Q0.1 antes de habilitar las operaciones PTO o PWM.

La función Tren de impulsos (PTO) ofrece una onda cuadrada (con un factor de trabajo relativo de 50%), pudiendo el usuario controlar el tiempo de ciclo y el número de impulsos. La función Modulación del ancho de impulsos (PWM) ofrece una salida continua con un factor de trabajo relativo variable, pudiendo el usuario controlar el tiempo de ciclo y el ancho de impulsos.

Todos los generadores PTO/PWM tienen un byte de control (8 bits), un valor de tiempo de ciclo y un valor de ancho de impulsos (ambos son valores de 16 bits sin signo), así como un valor de conteo de impulsos (valor de 32 bits sin signo). Estos valores están almacenados en determinadas direcciones del área de marcas especiales (SM). Una vez que se disponga de las direcciones de marcas especiales para seleccionar la función deseada, ésta se invoca ejecutando la operación Salida de impulsos (PLS). Con esta operación, la CPU S7-200 lee las direcciones de las marcas especiales (SM) y programa el generador PTO/PWM.

Para cambiar las características de una forma de onda PTO o PWM es preciso modificar las direcciones deseadas en el área SM (incluyendo el byte de control), ejecutando luego la operación PLS.

La generación de una forma de onda PTO o PWM se puede inhibir en cualquier momento, poniendo a cero el bit de habilitación PTO/PWM del byte de control (SM67.7 ó SM77.7) y ejecutando luego la operación PLS.

Nota

El ajuste estándar de los bits de control, del tiempo de ciclo, del ancho de impulsos y de los valores de conteo es 0.

Nota

Las salidas PTO/PWM requieren una carga mínima de 10% de la carga nominal para poder garantizar transiciones idóneas de "off" a "on", y viceversa.

Modulación del ancho de impulsos (PWM)

La función PWM ofrece una salida con un factor de trabajo relativo variable. El tiempo de ciclo y el ancho de impulsos pueden indicarse con una base de tiempo en microsegundos o milisegundos. El tiempo de ciclo se puede especificar en microsegundos (de 50 a 65.535 microsegundos) o en milisegundos (de 2 a 65.535 milisegundos). El tiempo de ancho de impulsos puede estar comprendido entre 0 y 65.535 microsegundos ó 0 y 65.535 milisegundos. Si se indica un valor del ancho de impulsos mayor o igual al valor del tiempo de ciclo, el factor de trabajo relativo de la forma de onda será de 100% y la salida estará activada continuamente. Si el ancho de impulsos es 0, el factor de trabajo relativo de la forma de onda será de 0% y se desactivará la salida. Si se indica un tiempo de ciclo inferior a dos unidades de tiempo, el tiempo de ciclo se predeterminará en dos unidades de tiempo.

Hay dos maneras diferentes de cambiar las características de una forma de onda PWM: con una actualización síncrona o asíncrona.

- Actualización síncrona: Si no es necesario cambiar la base de tiempo se puede ejecutar una actualización síncrona. En este caso, el cambio de la forma de onda se efectúa en el límite de un ciclo, ofreciendo una transición suave.
- Actualización asíncrona: En una función PWM típica, el ancho de impulsos se modifica, permaneciendo constante el tiempo de ciclo. Por tanto, no es necesario cambiar la base de tiempo. No obstante, si es preciso modificar la base de tiempo del generador PTO/PWM se utiliza una actualización asíncrona. Ésta inhibe el generador PTO/PWM momentáneamente, de forma asíncrona a la forma de onda PWM. Ello puede provocar fluctuaciones no deseadas en el dispositivo controlado. Por tanto, se recomienda la actualización síncrona de las formas de ondas PWM. Elija una base de tiempo que se adecúe para todos los valores previstos para el tiempo de ciclo.

La marca del método de actualización PWM (SM67.4 ó SM77.4) en el byte de control se utiliza para especificar el tipo de actualización. La operación PLS se debe ejecutar para invocar los cambios. Es preciso tener en cuenta que si se modifica la base de tiempo se efectuará en todo caso una actualización asíncrona, sin importar el estado de esta marca especial.

Tren de impulsos (PTO)

La función PTO genera un tren de impulsos de onda cuadrada (con un factor de trabajo relativo de 50%) con un número determinado de impulsos. El tiempo de ciclo puede indicarse en incrementos de microsegundos o milisegundos. El tiempo de ciclo se puede especificar en microsegundos (de 50 a 65.535 microsegundos) o en milisegundos (de 2 a 65.535 milisegundos). Si el tiempo de ciclo es un número impar, se distorsionará levemente el factor de trabajo relativo. El número de impulsos puede estar comprendido entre 1 y 4.294.967.295.

Si se indica un tiempo de ciclo inferior a dos unidades de tiempo, el tiempo de ciclo se predeterminará en dos unidades de tiempo. Si se indica un conteo de impulsos de 0, se adoptará un valor predeterminado de un impulso.

El bit de inactividad PTO (SM66.7 ó SM76.7) en el byte de estado indica si se ha completado el tren de impulsos programado. Además, tras finalizarse un tren de impulsos se puede llamar a una rutina de interrupción (consulte el apartado 9.15 para obtener más información acerca de las operaciones de interrupción y comunicación). Si se utiliza el pipelining multisegmento, la rutina de interrupción se invocará tras completarse la tabla de perfiles (v. "Pipelining multisegmento").

La operación PTO permite efectuar un "pipelining", es decir, concatenar o canalizar los trenes de impulsos. Tras haberse finalizado el tren de impulsos activo, comienza inmediatamente la salida de un nuevo tren de impulsos. Así se garantiza la continuidad de los trenes de impulsos subsiguientes.

El pipelining se puede efectuar de dos maneras: en un solo segmento o en varios segmentos.

Pipelining monosegmento En este modo, el usuario debe actualizar las direcciones de las marcas especiales para el siguiente tren de impulsos. Tras haber arrancado el segmento PTO inicial, el usuario debe modificar inmediatamente las direcciones de las marcas especiales conforme a lo requerido para la segunda forma de onda, y ejecutar nuevamente la operación PLS. Los atributos del segundo tren de impulsos se conservarán en una pipeline hasta finalizar el primer tren de impulsos. En la pipeline sólo se puede almacenar un registro. Tras haber finalizado el primer tren de impulsos, comenzará la salida de la segunda forma de onda y la pipeline quedará libre para acoger un nuevo tren de impulsos. Este proceso se puede repetir entonces para ajustar las características del siguiente tren de impulsos.

Las transiciones entre los trenes de impulsos serán suaves, con excepción de las siguientes situaciones:

- Si se cambia la base de tiempo.
- Si el tren de impulsos activo se termina antes de que la ejecución de la operación PLS capture el ajuste de un nuevo tren de impulsos.

Si se intenta cargar la pipeline mientras está llena, se activará la marca de desbordamiento PTO en el registro de estado (SM66.6 ó SM76.6). Al pasar a RUN, la marca se vuelve a poner a 0. Para poder detectar los desbordamientos siguientes, la marca se deberá poner a 0 manualmente tras haberse detectado un desbordamiento.

Pipelining multisegmento En este modo, la CPU lee automáticamente las características de cada tren de impulsos en una tabla de perfiles almacenada en la memoria de variables (memoria V). Las únicas marcas especiales utilizadas en este modo son el byte de control y el byte de estado. Para seleccionar el pipelining multisegmento es preciso cargar el offset inicial de la memoria V de la tabla de perfiles (SMW168 ó SMW178). La base de tiempo se puede indicar en microsegundos o milisegundos, pero la selección será aplicable a todos los valores de tiempo de ciclo en la tabla de perfiles, no pudiendo modificarse durante la ejecución del perfil. El pipelining multisegmento se puede arrancar ejecutando la operación PLS.

Todos los registros de segmentos tienen una longitud de 8 bytes, comprendiendo un valor de tiempo de ciclo de 16 bits, un valor delta de tiempo de ciclo de 16 bits y un valor de conteo de impulsos de 32 bits.

La tabla 9-15 muestra el formato de la tabla de perfiles. Una función adicional disponible en el pipelining multisegmento PTO es la posibilidad de prolongar o acortar automáticamente el tiempo de ciclo en cada impulso. El tiempo de ciclo se prolonga o se acorta programando en su campo delta un valor positivo o negativo, respectivamente. Este tiempo permanecerá inalterado si se programa el valor 0.

Si se indica un valor delta que produzca un tiempo de ciclo no válido al cabo de un número de impulsos, ocurrirá una condición de desbordamiento aritmética. La función PTO terminará y la salida retornará al control de la imagen del proceso. Además, se activará la marca especial de error de cálculo delta en el byte de estado (SM66.4 ó SM76.4).

Si el usuario interrumpe un perfil PTO que se esté ejecutando, se activará la marca especial de interrupción manual en el byte de estado (SM66.5 ó SM76.5).

Mientras se está ejecutando el perfil PTO, el número del segmento activo actualmente se indica en SMB166 (ó SMB176).

Tabla 9-15 Formato de la tabla de perfiles para la función PTO multisegmento

Offset (en bytes) desde el comienzo de la tabla	Número de segmento	Descripción de los registros de la tabla
0		Número de segmentos (1 a 255); el valor 0 genera un error no fatal. No se genera una salida PTO
1	#1	Tiempo de ciclo inicial (2 a 65535 unidades de la base de tiempo)
3		Valor delta del tiempo de ciclo por impulso (valor con signo) (–32768 a 32767 unidades de la base de tiempo)
5		Contaje de impulsos (1 a 4294967295)
9	2#	Tiempo de ciclo inicial (2 a 65535 unidades de la base de tiempo)
11		Valor delta del tiempo de ciclo por impulso (valor con signo) (–32768 a 32767 unidades de la base de tiempo)
13		Contaje de impulsos (1 a 4294967295)
:	:	:
:	:	:

Calcular los valores de la tabla de perfiles

El pipelining multisegmento que ofrecen los generadores PTO/PWM se puede utilizar para numerosas aplicaciones, en particular para el control de motores paso a paso.

El ejemplo de la figura 9-20 muestra cómo determinar los valores de la tabla de perfiles necesarios para generar una forma de onda de salida con objeto de acelerar un motor paso a paso, permitir que funcione a una velocidad constante y desacelerarlo luego.

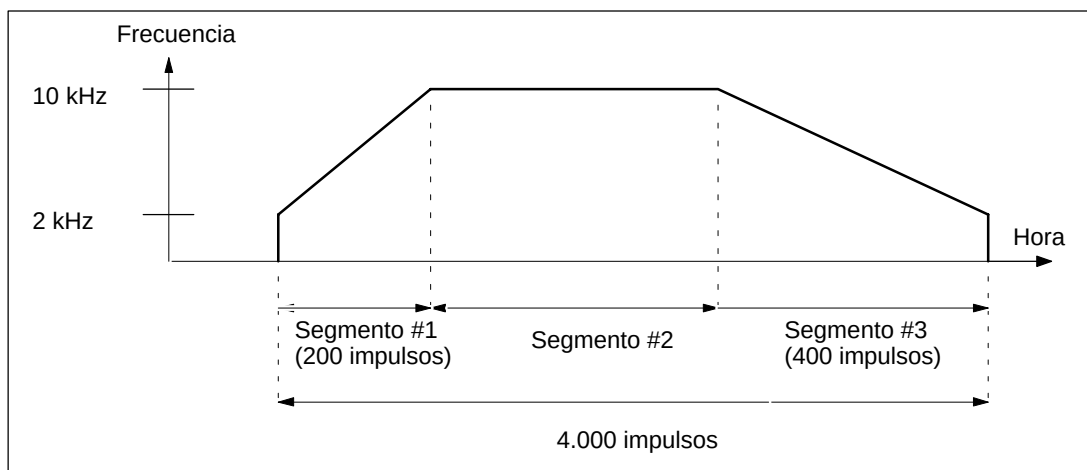


Figura 9-20 Diagrama frecuencia/tiempo de una aplicación de ejemplo para un motor paso a paso

En el presente ejemplo se supone que se necesitan 4.000 impulsos para alcanzar el número deseado de revoluciones del motor. La frecuencia de impulsos inicial y final es de 2 kHz, en tanto que la frecuencia máxima de impulsos es de 10 kHz. Puesto que los valores de la tabla de perfiles se expresan en términos de período (tiempo de ciclo) y no de frecuencia, es preciso convertir los valores de frecuencia dados en valores de tiempo de ciclo. Por tanto, el tiempo de ciclo inicial y final es de 500 μ s, en tanto que el tiempo de ciclo correspondiente a la frecuencia máxima es de 100 μ s.

Durante la etapa de aceleración del perfil de salida, la frecuencia máxima de impulsos se desea alcanzar en aproximadamente 200 impulsos. Asimismo, se supone que la etapa de desaceleración del perfil se debe finalizar en unos 400 impulsos.

En el ejemplo que muestra la figura 9-20, una fórmula sencilla (que figura abajo) sirve para calcular el valor delta del tiempo de ciclo utilizado por el generador PTO/PWM para ajustar el tiempo de ciclo de cada impulso:

Valor delta del tiempo de ciclo de un segmento dado = $|ECT - ICT| / Q$

donde, ECT = tiempo de ciclo final de este segmento

ICT = tiempo de ciclo inicial de este segmento

Q = cantidad de impulsos de este segmento

Utilizando esta fórmula, se calcula que el tiempo de ciclo delta de la etapa de aceleración (o segmento #1) es -2. De forma similar, el tiempo de ciclo delta de la etapa de desaceleración (o segmento #3) es 1. Puesto que el segmento #2 representa la etapa de velocidad constante de la forma de onda de salida, el valor delta del tiempo de ciclo correspondiente a dicho segmento es 0.

Suponiendo que la tabla de perfiles se encuentra almacenada en la memoria V (comenzando en V500), los valores utilizados para generar la forma de onda deseada figuran en la tabla 9-16.

Tabla 9-16 Valores de la tabla de perfiles

Dirección de la memoria V	Valor
VB500	3 (número total de segmentos)
VW501	500 (tiempo de ciclo inicial – segmento #1)
VW503	–2 (tiempo de ciclo inicial – segmento #1)
VD505	200 (número de impulsos – segmento #1)
VW509	100 (tiempo de ciclo inicial – segmento #2)
VW511	0 (tiempo de ciclo delta – segmento #2)
VD513	3400 (número de impulsos – segmento #2)
VW517	100 (tiempo de ciclo inicial – segmento #3)
VW519	1 (tiempo de ciclo delta – segmento #3)
VD521	400 (número de impulsos – segmento #3)

Los valores de esta tabla se pueden depositar en la memoria V utilizando operaciones en el programa. Una alternativa consiste en definir los valores del perfil en el bloque de datos. La figura 9-23 muestra un ejemplo que contiene las operaciones para utilizar la función multi-segmento PTO.

El tiempo de ciclo del último impulso de un segmento no se indica directamente en el perfil, sino que se debe calcular (a menos que el tiempo de ciclo delta sea 0). Para determinar si son aceptables las transiciones entre los segmentos de formas de ondas, es aconsejable conocer el tiempo de ciclo del último impulso de un segmento. La fórmula para calcular este tiempo es la siguiente:

$$\text{Tiempo de ciclo del último impulso del segmento} = \text{ICT} + (\text{DEL} * (\text{Q} - 1))$$

donde, ICT = tiempo de ciclo inicial de este segmento
DEL = valor delta del tiempo de ciclo de este segmento
Q = cantidad de impulsos de este segmento

Aunque el ejemplo simplificado descrito arriba sirve a título de introducción, es posible que en una aplicación real se necesiten perfiles de formas de ondas más complejos. Recuerde que:

- El tiempo de ciclo delta sólo se puede indicar en forma de número entero en microsegundos o milisegundos.
- En cada impulso se modifica el tiempo de ciclo.

El efecto de estos dos puntos es que el cálculo del valor delta del tiempo de ciclo para un determinado segmento puede exigir una aproximación iterativa. Puede ser necesaria cierta flexibilidad en el valor del tiempo de ciclo final o en el número de impulsos para un determinado segmento.

La duración de un segmento de perfil puede ser útil a la hora de determinar los valores correctos para la tabla de perfiles. Esta duración se puede calcular utilizando la fórmula siguiente:

$$\text{Duración del segmento} = Q * (\text{ICT} + ((\text{DEL}/2) * (Q-1)))$$

donde, Q = cantidad de impulsos de este segmento
ICT = tiempo de ciclo inicial de este segmento
DEL = valor delta del tiempo de ciclo de este segmento

Registros de control PTO/PWM

En la tabla 9-17 se describen los registros utilizados para controlar las funciones PTO/PWM. Mediante la tabla 9-18 es posible determinar rápidamente el valor que debe depositarse en el registro de control PTO/PWM para solicitar la operación deseada. Es preciso utilizar SMB67 para PTO/PWM 0 y SMB77 para PTO/PWM 1. Si se desea cargar un nuevo valor de conteo (SMD72 o SMD82), ancho de impulsos (SMW70 o SMW80) o tiempo de ciclo (SMW68 o SMW78), es necesario cargar tanto estos valores como el registro de control antes de ejecutar la operación PLS. Si desea utilizar la función PTO multisegmento, antes de ejecutar la operación PLS también deberá cargar el offset inicial (SMW168 o SMW178) de la tabla de perfiles y los valores de ésta.

Tabla 9-17 Registros de control PTO /PWM

Q0.0	Q0.1	Bits de estado
SM66.4	SM76.4	Interrupción anormal del perfil PTO debido a error de cálculo delta 0 = sin error; 1 = interrupción
SM66.5	SM76.5	Interrupción anormal del perfil PTO causada por el usuario 0 = sin interrupción; 1 = interrupción
SM66.6	SM76.6	Desbordamiento positivo/negativo PTO 0 = sin desbordamiento; 1 = desbordamiento positivo/negativo
SM66.7	SM76.7	PTO en vacío 0 = ejecución; 1 = PTO en vacío
Q0.0	Q0.1	Bits de control
SM67.0	SM77.0	Actualizar tiempo de ciclo PTO/PWM 0 = no actualizar; 1 = actualizar tiempo de ciclo
SM67.1	SM77.1	Actualizar tiempo de ancho de impulsos PWM 0 = no actualizar; 1 = actualizar ancho de impulsos
SM67.2	SM77.2	Actualizar valor de conteo de impulsos PTO 0 = no actualizar; 1 = actualizar valor de conteo de impulsos
SM67.3	SM77.3	Elegir base de tiempo PTO/PWM 0 = 1 µs/ciclo; 1 = 1 ms/ciclo
SM67.4	SM77.4	Método de actualización PWM: 0 = actualización asíncrona, 1 = actualización síncrona
SM67.5	SM77.5	Función PTO: 0 = función monosegmento; 1 = función multisegmento
SM67.6	SM77.6	Elegir modo PTO/PWM 0 = elige PTO; 1 = elige PWM
SM67.7	SM77.7	Habilitar PTO/PWM 0 = inhibe PTO/PWM; 1 = habilita PTO/PWM
Q0.0	Q0.1	Otros registros PTO/PWM
SMW68	SMW78	Valor del tiempo de ciclo PTO/PWM (margen: 2 a 65535)
SMW70	SMW80	Valor del ancho de impulsos PWM (margen: 0 a 65535)
SMD72	SMD82	Valor del conteo de impulsos PTO (margen: 1 a 4294967295)
SMB166	SMB176	Número del segmento que se está ejecutando (sólo se utiliza en la función multisegmento PTO)
SMW168	SMW178	Dirección inicial de la tabla de perfiles, expresada en forma de offset (en bytes) a partir de V0 (se utiliza sólo en la función multisegmento PTO)

Tabla 9-18 Referencias del byte de control PTO/PWM

Registro de control (valor hexadecimal)	Resultado de la operación PLS							
	Habilitar	Modo	Función PTO	Método de actualización PWM	Base de tiempo	Contaje de impulsos	Ancho de impulsos	Tiempo de ciclo
16#81	Sí	PTO	Monosegmento		1 µs/ciclo			Cargar
16#84	Sí	PTO	Monosegmento		1 µs/ciclo	Cargar		
16#85	Sí	PTO	Monosegmento		1 µs/ciclo	Cargar		Cargar
16#89	Sí	PTO	Monosegmento		1 ms/ciclo			Cargar
16#8C	Sí	PTO	Monosegmento		1 ms/ciclo	Cargar		
16#8D	Sí	PTO	Monosegmento		1 ms/ciclo	Cargar		Cargar
16#A0	Sí	PTO	Multisegmento		1 µs/ciclo			
16#A8	Sí	PTO	Multisegmento		1 ms/ciclo			
16#D1	Sí	PWM		Síncrona	1 µs/ciclo			Cargar
16#D2	Sí	PWM		Síncrona	1 µs/ciclo		Cargar	
16#D3	Sí	PWM		Síncrona	1 µs/ciclo		Cargar	Cargar
16#D9	Sí	PWM		Síncrona	1 ms/ciclo			Cargar
16#DA	Sí	PWM		Síncrona	1 ms/ciclo		Cargar	
16#DB	Sí	PWM		Síncrona	1 ms/ciclo		Cargar	Cargar

Inicialización y secuencias de las funciones PTO/PWM

Para comprender mejor el funcionamiento de PTO y PWM, se describe seguidamente su inicialización paso a paso, así como las correspondientes operaciones. En las descripciones se utiliza la salida de impulsos Q0.0 a título de ejemplo. En las explicaciones acerca de la inicialización se supone que el sistema de automatización S7-200 se encuentra en modo RUN y que, por consiguiente, la marca del primer ciclo es verdadera. En otro caso o si se debe inicializar nuevamente la función PTO/PWM, es preciso llamar a la rutina de inicialización usando una condición diferente a la marca del primer ciclo.

Inicializar la función PWM

Siga los pasos siguientes para inicializar la función PWM para Q0.0:

1. Con la marca del primer ciclo (SM0.1), inicialice la salida y llame a la subrutina necesaria para ejecutar la inicialización. Puesto que se utiliza una llamada a subrutina, los siguientes ciclos ya no llaman a la misma. Así se acorta el tiempo de ciclo y el programa queda mejor estructurado.
2. En la subrutina de inicialización, cargue 16#D3 para PWM en SMB67 para incrementar en microsegundos (o cargue 16#DB para PWM si desea incrementar en milisegundos). Estos valores ajustan el byte de control para habilitar la operación PTO/PWM, seleccionan la operación PWM, deciden si se incrementa en micro o milisegundos y ajustan los valores para actualizar el ancho de impulsos y el tiempo de ciclo.
3. Cargue el tiempo de ciclo deseado en SMW68 (valor de palabra).
4. Cargue el ancho de impulsos deseado en SMW70 (valor de palabra).
5. Ejecute la operación PLS para que el S7-200 programe el generador PTO/PWM.
6. Cargue el valor 16#D2 en SM67 para incrementar en microsegundos (o 16#DA si desea incrementar en milisegundos). Así se precarga un nuevo valor del byte de control para los posteriores cambios del ancho de impulsos.
7. Finalice la subrutina.

Cambiar el ancho de impulsos para las salidas PWM

Siga los pasos siguientes para cambiar el ancho de impulsos para salidas PWM en una subrutina: (Se supone que SMB67 se ha precargado con un valor de 16#D2 ó 16#DA).

1. Llame a una subrutina para cargar el ancho de impulsos deseado en SMW70 (valor de palabra).
2. Ejecute la operación PLS para que el S7-200 programe el generador PTO/PWM.
3. Finalice la subrutina.

Inicializar la función PTO monosegmento

Siga los pasos siguientes para inicializar la función PTO:

1. Con la marca del primer ciclo (SM0.1), inicialice la salida y llame a la subrutina necesaria para ejecutar la inicialización. Así se acorta el tiempo de ciclo y el programa queda mejor estructurado.
2. En la subrutina de inicialización, cargue 16#85 para PTO en SMB67 para incrementar en microsegundos (o cargue 16#8D para PTO si desea incrementar en milisegundos). Estos valores ajustan el byte de control para habilitar la operación PTO/PWM, seleccionan la operación PTO, deciden si se incrementa en micro o milisegundos y ajustan los valores para actualizar el ancho de impulsos y el tiempo de ciclo.
3. Cargue el tiempo de ciclo deseado en SMW68 (valor de palabra).
4. Cargue el tiempo de ciclo deseado en SMD72 (valor de palabra doble).
5. Este paso es opcional. Si desea ejecutar una operación asociada en cuanto termine la operación Tren de impulsos, puede programar una interrupción asociando el evento Fin del tren de impulsos (clase de interrupción 19) a una rutina de interrupción (mediante la operación ATCH) y ejecutando la operación Habilitar todos los eventos de interrupción (ENI). Para obtener más información acerca de la utilización de interrupciones, consulte el apartado 9.15.
6. Ejecute la operación PLS para que el S7-200 programe el generador PTO/PWM.
7. Finalice la subrutina.

Cambiar el tiempo de ciclo PTO en la función monosegmento

Siga los pasos siguientes para cambiar el tiempo de ciclo PTO en una subrutina o en una rutina de interrupción al utilizar la función PTO monosegmento:

1. Cargue 16#81 para PTO en SMB67 para incrementar en microsegundos (o cargue 16#89 para PTO si desea incrementar en milisegundos). Estos valores ajustan el byte de control para habilitar la operación PTO/PWM, seleccionan la operación PTO, deciden si se incrementa en micro o milisegundos y ajustan los valores para actualizar el tiempo de ciclo.
2. Cargue el tiempo de ciclo deseado en SMW68 (valor de palabra).
3. Ejecute la operación PLS para que el S7-200 programe el generador PTO/PWM. Si se está ejecutando un PTO, la CPU debe finalizarlo antes de que pueda comenzar la salida de la forma de onda PTO con el tiempo de ciclo actualizado.
4. Finalice la subrutina o la rutina de interrupción.

Cambiar el valor de conteo de impulsos PTO en la función monosegmento

Siga los pasos siguientes para cambiar el valor de conteo de impulsos PTO en una subrutina o en una rutina de interrupción al utilizar la función PTO monosegmento:

1. Cargue 16#84 para PTO en SMB67 para incrementar en microsegundos (o cargue 16#8C para PTO si desea incrementar en milisegundos). Estos valores ajustan el byte de control para habilitar la operación PTO/PWM, seleccionan la operación PTO, deciden si se incrementa en micro o milisegundos y ajustan los valores para actualizar el valor de conteo de impulsos.
2. Cargue el tiempo de ciclo deseado en SMD72 (valor de palabra doble).
3. Ejecute la operación PLS para que el S7-200 programe el generador PTO/PWM. Si se está ejecutando un PTO, la CPU debe finalizarlo antes de que pueda comenzar la salida de la forma de onda PTO con el tiempo de ciclo actualizado.
4. Finalice la subrutina o la rutina de interrupción.

Cambiar el tiempo de ciclo PTO y el conteo de impulsos en la función monosegmento

Siga los pasos siguientes para cambiar el tiempo de ciclo PTO y el conteo de impulsos en una subrutina o en una rutina de interrupción al utilizar la función PTO monosegmento:

1. Cargue 16#85 para PTO en SMB67 para incrementar en microsegundos (o cargue 16#8D para PTO si desea incrementar en milisegundos). Estos valores ajustan el byte de control para habilitar la operación PTO/PWM, seleccionan la operación PTO, deciden si se incrementa en micro o milisegundos y ajustan los valores para actualizar el tiempo de ciclo y el ancho de impulsos.
2. Cargue el tiempo de ciclo deseado en SMW68 (valor de palabra).
3. Cargue el tiempo de ciclo deseado en SMD72 (valor de palabra doble).
4. Ejecute la operación PLS para que el S7-200 programe el generador PTO/PWM. Si se está ejecutando un PTO, la CPU debe finalizarlo antes de que pueda comenzar la salida de la forma de onda PTO con el tiempo de ciclo actualizado.
5. Finalice la subrutina o la rutina de interrupción.

Inicializar la función PTO multisegmento

Siga los pasos siguientes para inicializar la función PTO:

1. Con la marca del primer ciclo (SM0.1), inicialice la salida y llame a la subrutina necesaria para ejecutar la inicialización. Así se acorta el tiempo de ciclo y el programa queda mejor estructurado.
2. En la subrutina de inicialización, cargue 16#A0 para PTO en SMB67 para incrementar en microsegundos (o cargue 16#A8 para PTO si desea incrementar en milisegundos). Estos valores ajustan el byte de control para habilitar la operación PTO/PWM, seleccionan la función PTO multisegmento y determinan si se debe incrementar en microsegundos o en milisegundos.
3. Cargue en SMW168 (valor de palabra) el offset inicial en la memoria V de la tabla de perfiles.
4. Ajuste los valores de segmento en la tabla de perfiles. Verifique que el campo "Número de segmento" (el primer byte de la tabla) sea el correcto.
5. Este paso es opcional. Si desea ejecutar una operación asociada en cuanto termine el perfil PTO, puede programar una interrupción asociando el evento Fin del tren de impulsos (clase de interrupción 19) a una rutina de interrupción. Utilice para ello la operación ATCH y ejecute la operación Habilitar todos los eventos de interrupción (ENI). Para obtener más información acerca de la utilización de interrupciones, consulte el apartado 9.15.
6. Ejecute la operación PLS para que el S7-200 programe el generador PTO/PWM.
7. Finalice la subrutina.

Ejemplo de la función Modulación del ancho de impulsos (PWM)

La figura 9-21 muestra un ejemplo de la operación Modulación del ancho de impulsos (PWM).

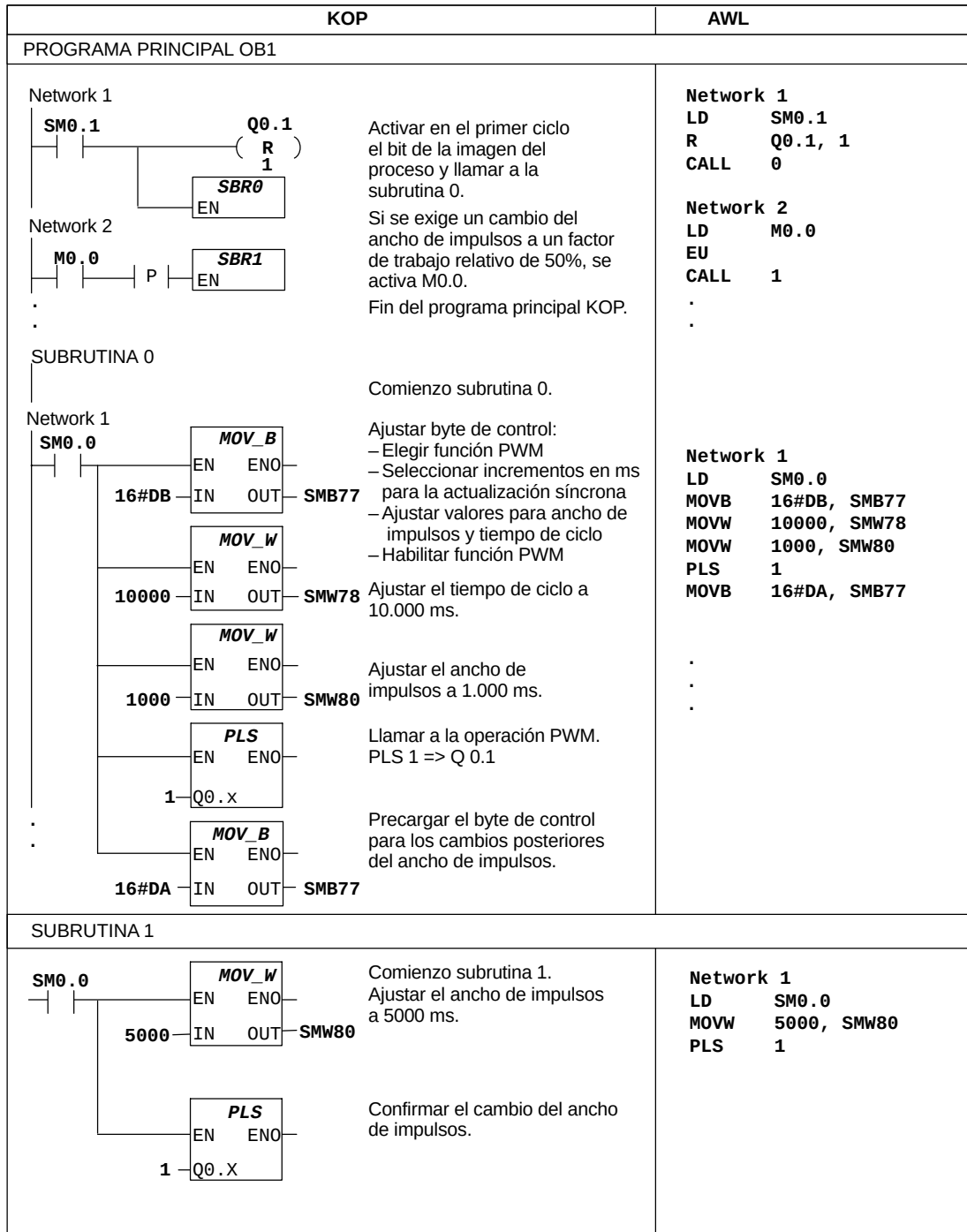


Figura 9-21 Ejemplo de operaciones rápidas de salida con modulación del ancho de impulsos

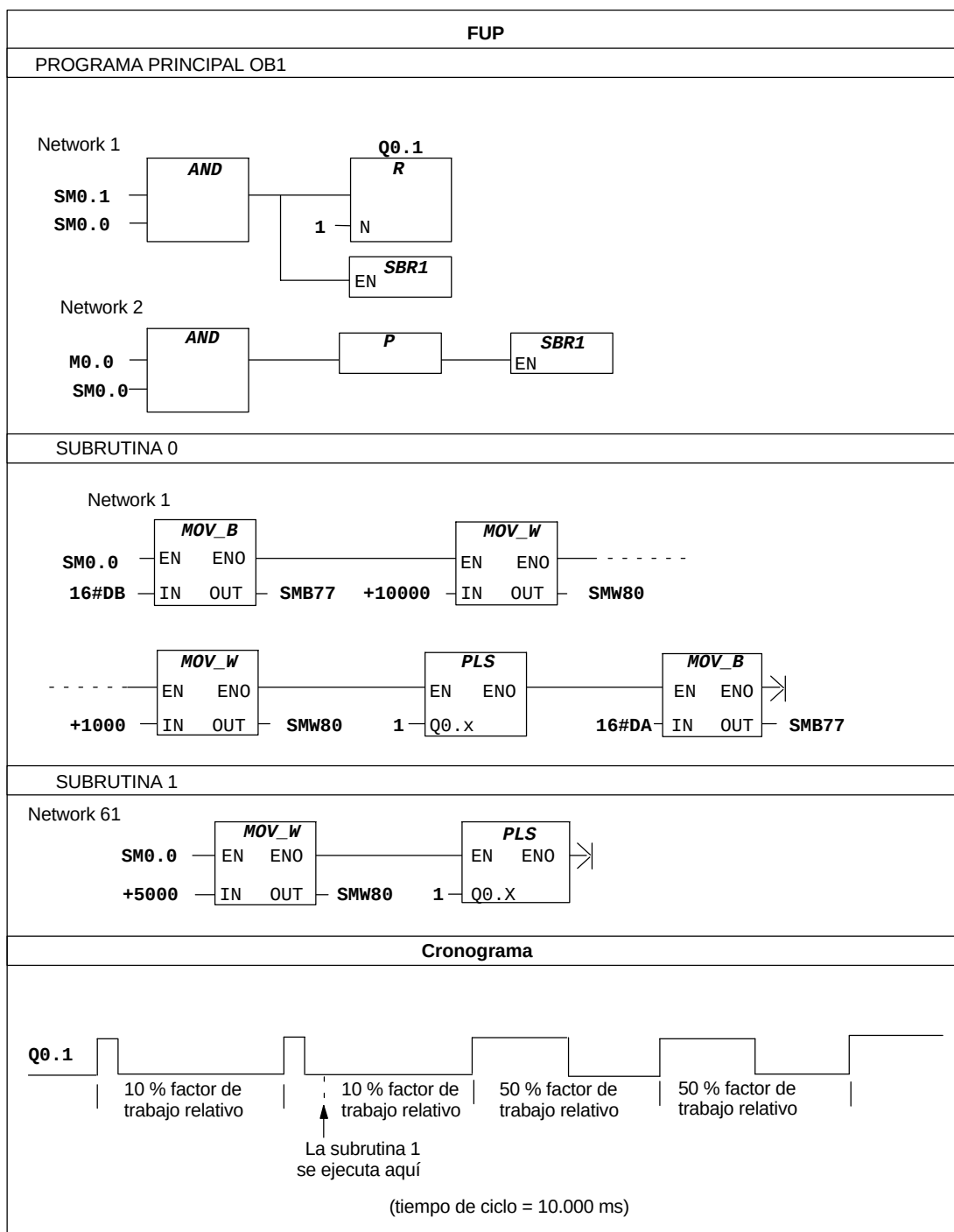


Figura 9-21 Ejemplo de operaciones rápidas de salida con modulación del ancho de impulsos (continuación)

Ejemplo de la función Tren de impulsos en modo monosegmento

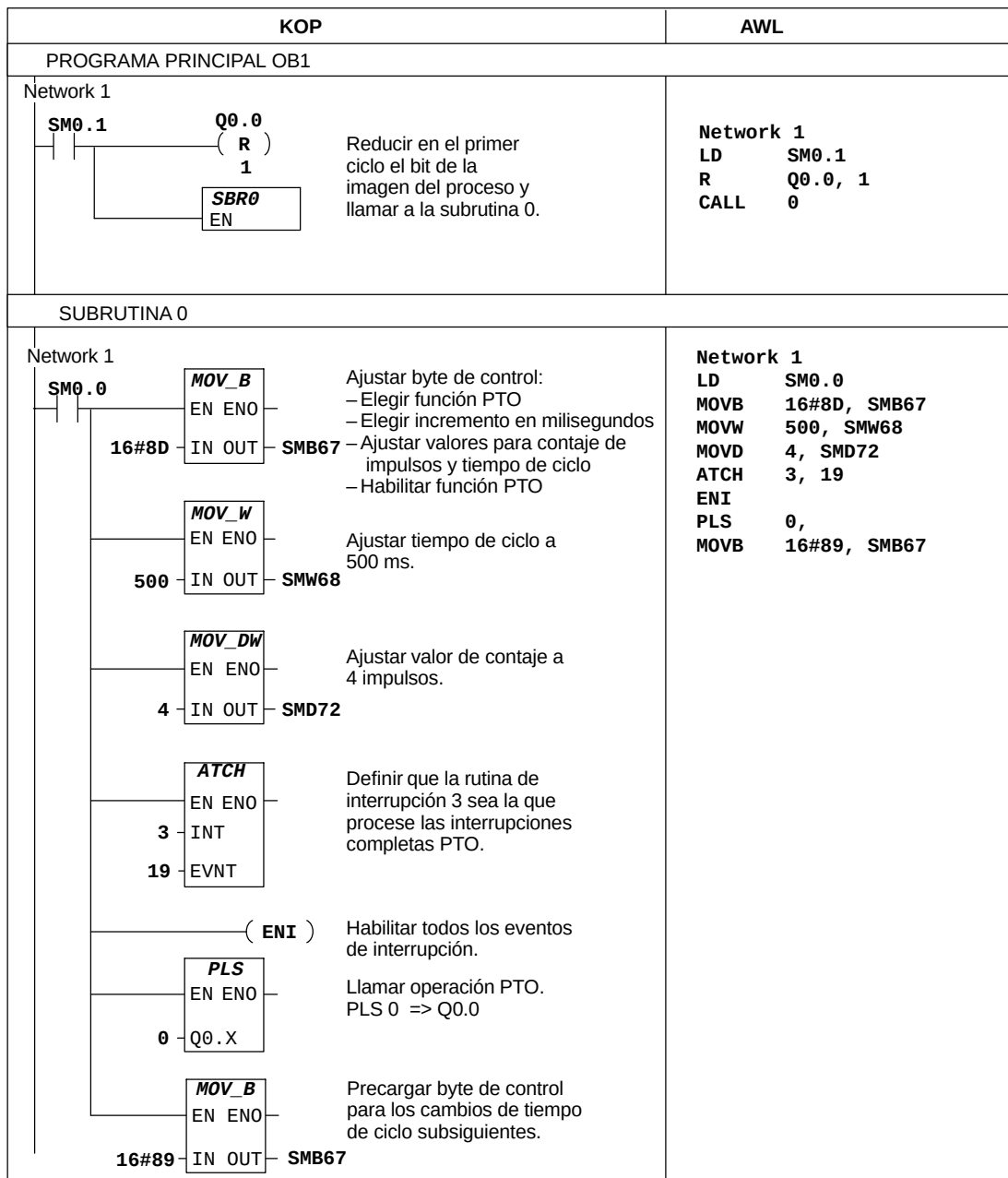


Figura 9-22 Ejemplo de un tren de impulsos utilizando la función monosegmento en el área de marcas especiales

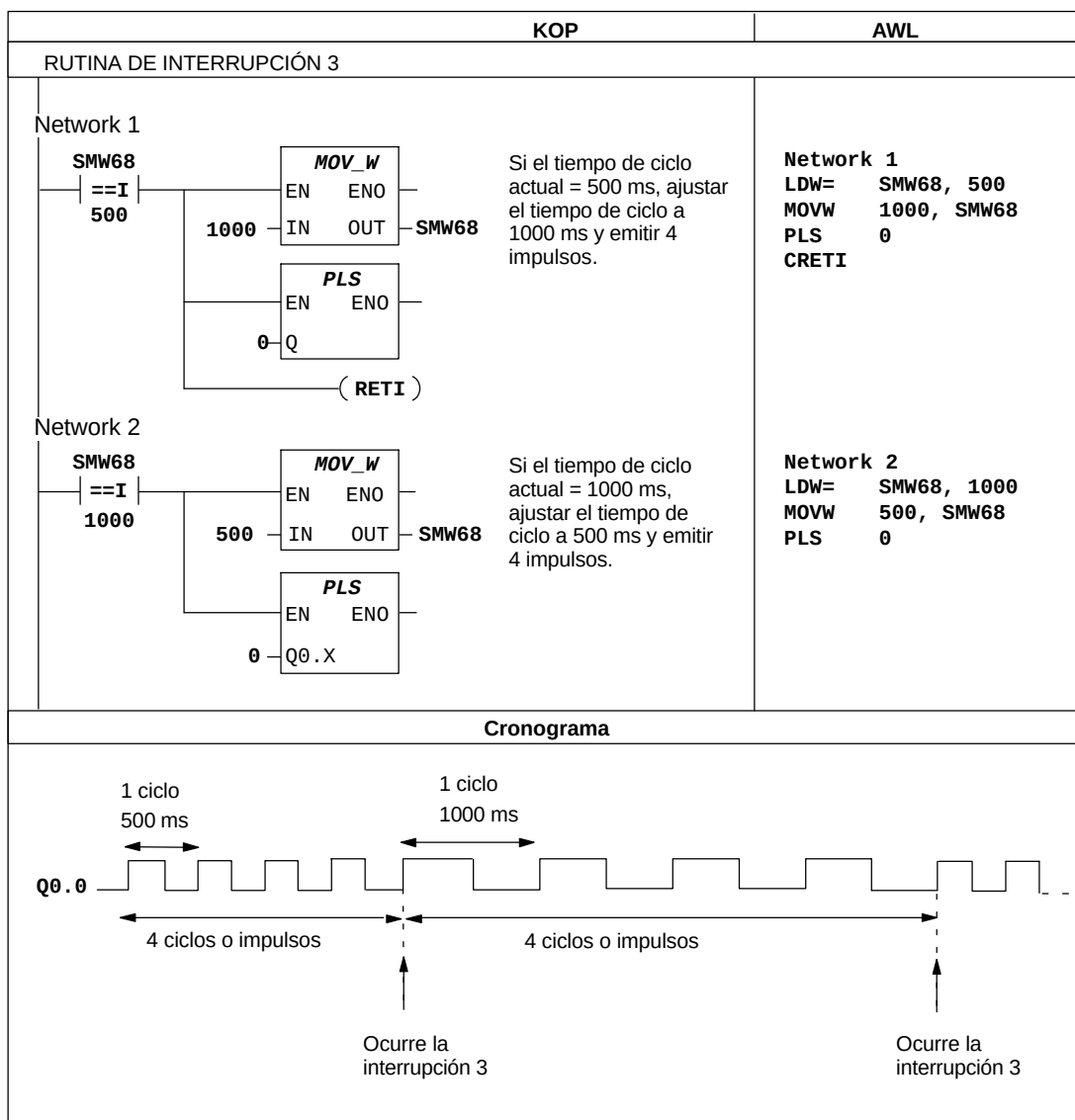


Figura 9-22 Ejemplo de un tren de impulsos utilizando la función monosegmento (continuación)

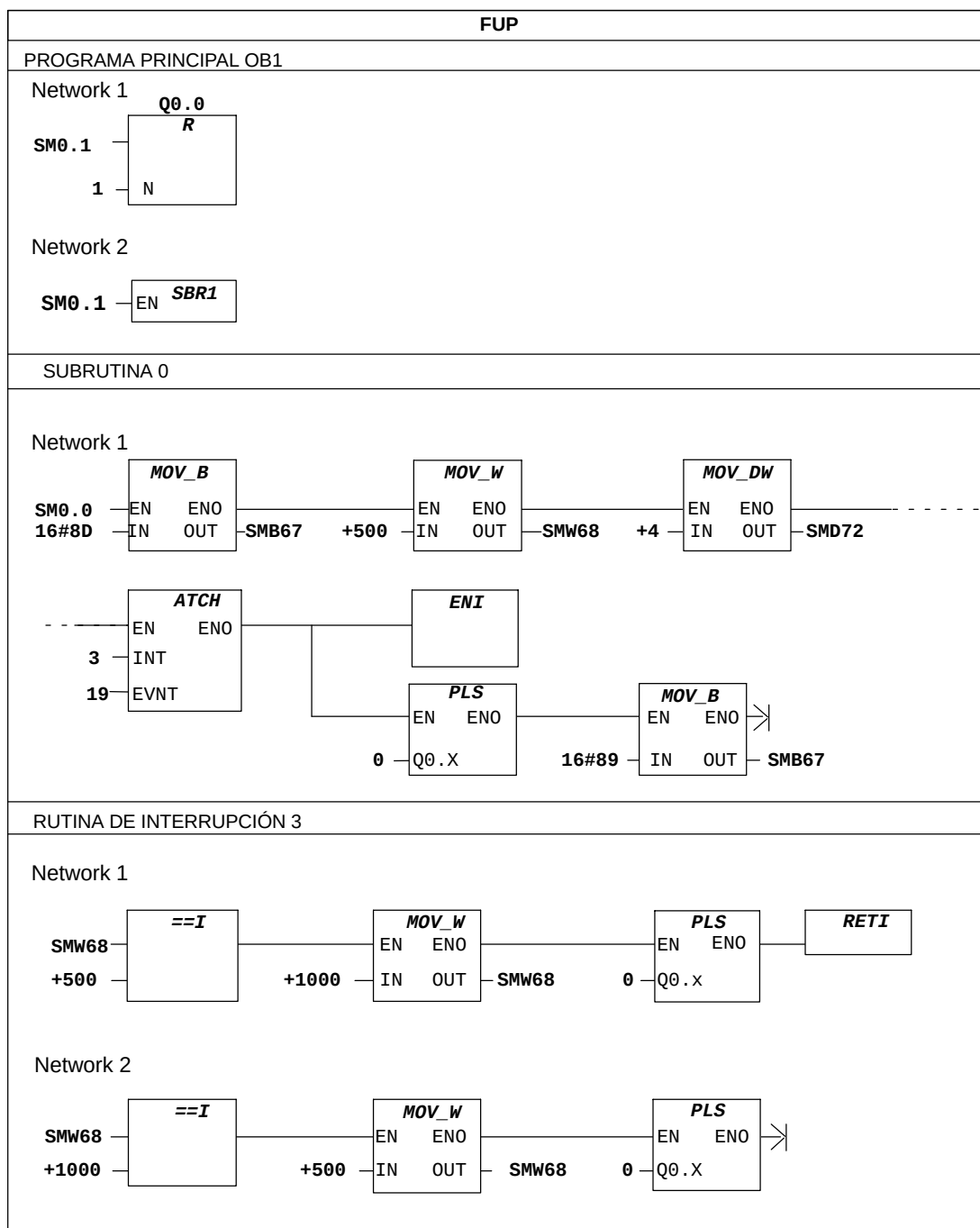


Figura 9-22 Ejemplo de un tren de impulsos utilizando la función monosegmento (continuación)

Ejemplo de la función Tren de impulsos en modo multisegmento

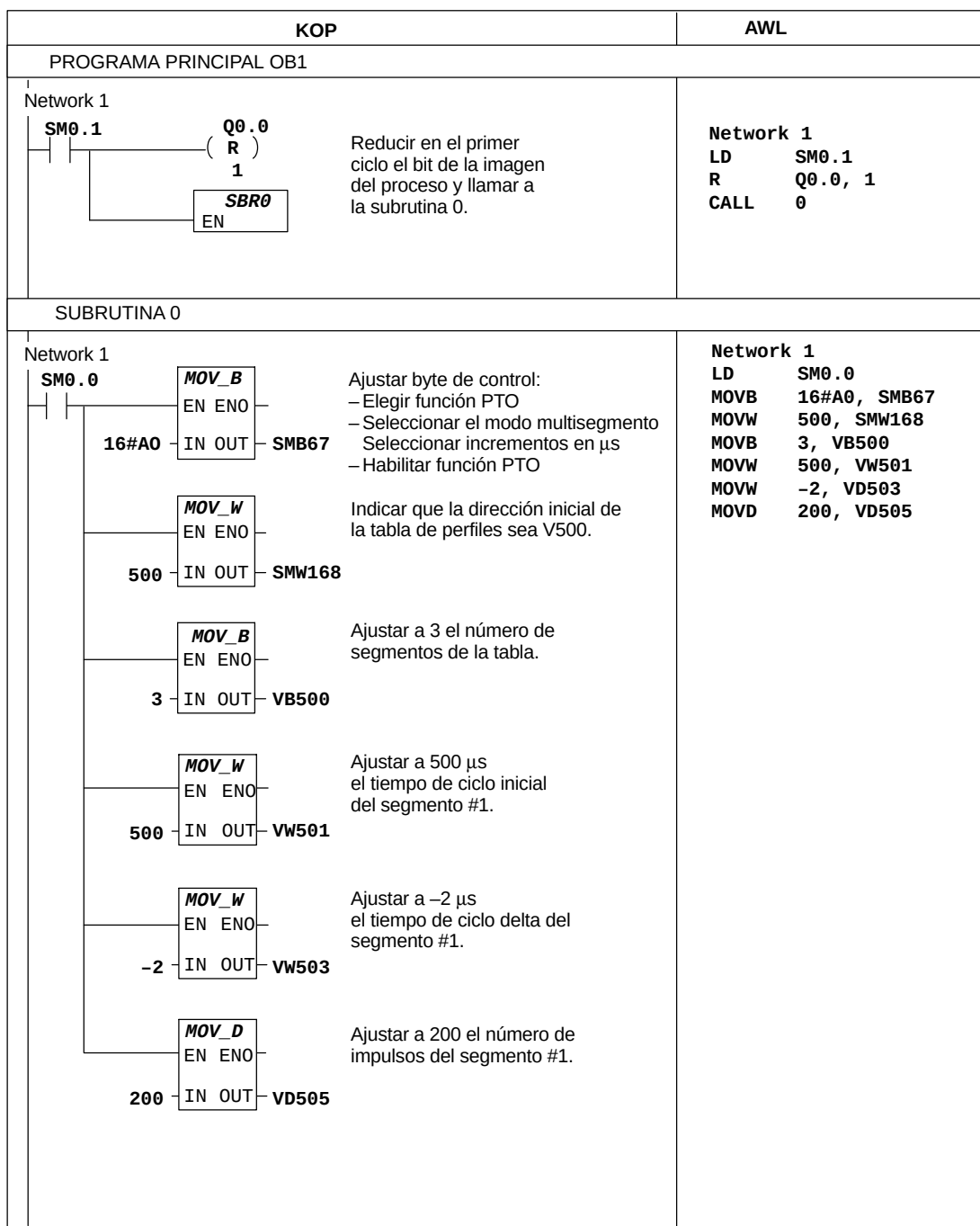


Figura 9-23 Ejemplo de un tren de impulsos utilizando la función multisegmento

KOP			AWL
Network 1 100 — IN OUT — VW509 0 — IN OUT — VW511 3400 — IN OUT — VD513 100 — IN OUT — VW517 1 — IN OUT — VW519 400 — IN OUT — VD521 2 — INT 19 — EVNT (ENI) 0 — Q0.X			MOVW 100, VW509 MOVW 0, VW511 MOVD 3400, VD513 MOVW 100, VW517 MOVW 1, VW519 MOVD 400, VD521 ATCH 2, 19 ENI PLS 0
RUTINA DE INTERRUPCIÓN 0			
Network 1 SM0.0 — (Q0.5) Activar la salida Q0.5 cuando finalice el perfil PTO.			Network 1 LD SM0.0 = Q0.5

Figura 9-23 Ejemplo de un tren de impulsos utilizando la función multisegmento (continuación)

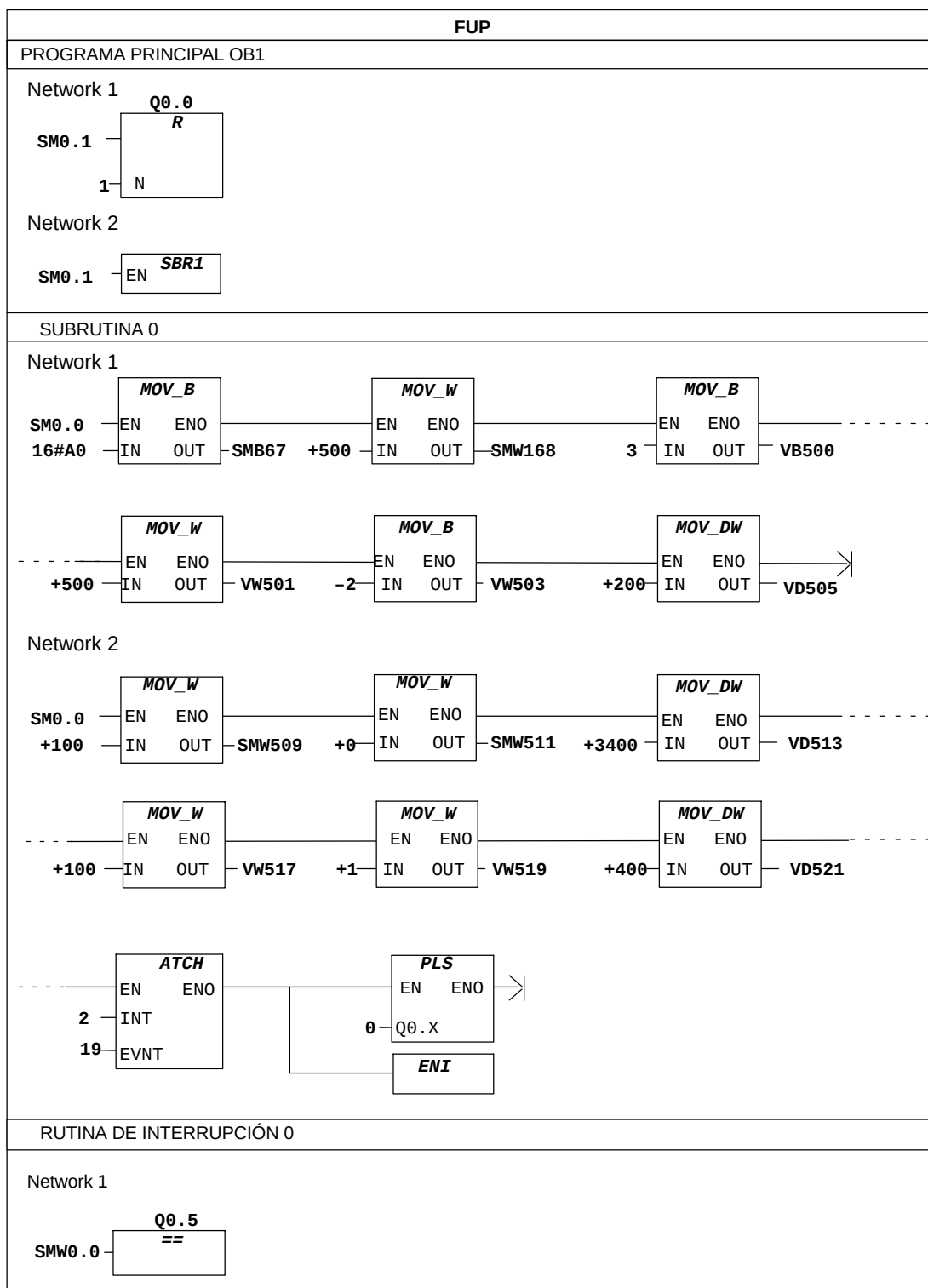
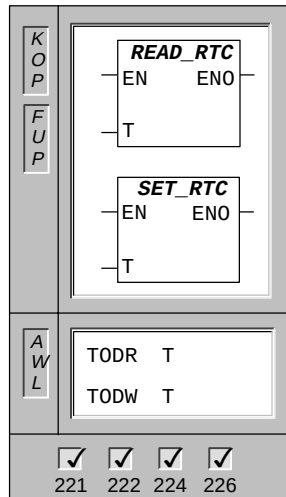


Figura 9-23 Ejemplo de un tren de impulsos utilizando la función multisegmento (continuación)

9.5 Operaciones de reloj (SIMATIC)

Leer reloj de tiempo real, Ajustar reloj de tiempo real



La operación **Leer reloj de tiempo real** lee la hora y fecha actuales del reloj y carga ambas en un búfer de 8 bytes (que comienza en la dirección T).

La operación **Ajustar reloj de tiempo real** escribe en el reloj la hora y fecha actuales que están cargadas en un búfer de 8 bytes (que comienza en la dirección T).

En AWL, las operaciones TODR y TODW se representan mediante las operaciones TODR (Leer reloj de tiempo real) y TODW (Escribir reloj de tiempo real).

TODR: Condiciones de error que ponen ENO a 0:
SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto), 000C (falta cartucho de reloj)

TODW: Condiciones de error que ponen ENO a 0:
SM 4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto), 0007 (error de datos TOD), 000C (falta cartucho de reloj)

Entradas/salidas	Operandos	Tipos de datos
T	VB, IB, QB, MB, SMB, SB, LB, *VD, *AC, *LD	BYTE

La figura 9-24 muestra el formato del búfer de tiempo (T).

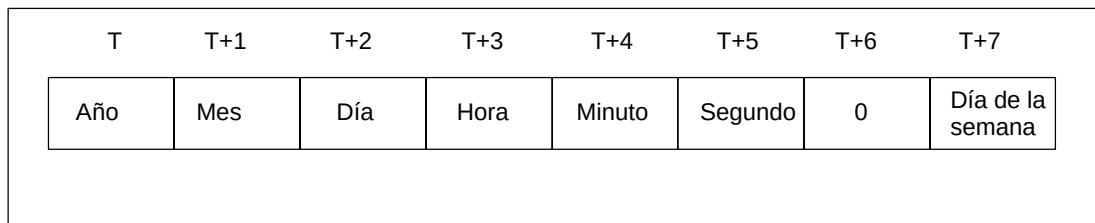


Figura 9-24 Formato del búfer de tiempo

El reloj de tiempo real se inicializa con la siguiente fecha y hora tras un corte de alimentación prolongado o una pérdida de memoria:

Fecha: 01-Ene-90
Hora: 00:00:00
Día de la semana Domingo

El reloj de tiempo real de la CPU S7-200 utiliza sólo los dos dígitos menos significativos para representar el año. Por tanto, el año 2000 se representa como "00".

Todos los valores de la fecha y la hora se deben codificar en BCD (p.ej., 16#97 para el año 1997). Utilice los siguientes formatos de datos:

Año/Mes	aamm	aa	–	0 a 99	mm	–	1 a 12
Día/Hora	ddhh	dd	–	1 a 31	hh	–	0 a 23
Minutos/Segundos	mmss	mm	–	0 a 59	ss	–	0 a 59
Día de la semana	d	d	–	0 a 7	1	=	Domingo
					0	=	desactiva el día de la semana (permanece 0)

Nota

La CPU S7-200 no comprueba si el día de la semana coincide con la fecha. Así puede ocurrir que se acepten fechas no válidas, p.ej. el 30 de febrero. Asegúrese de que los datos introducidos sean correctos.

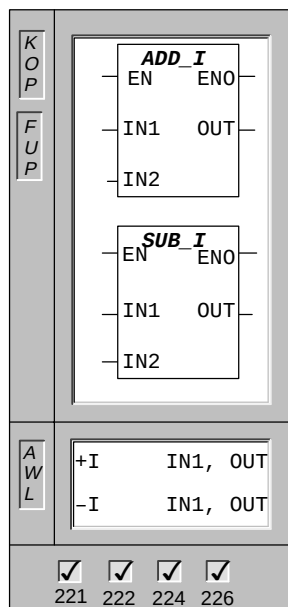
No utilice nunca las operaciones TODR y TODW en el programa principal y en una rutina de interrupción a la vez. Si se está procesando una operación TODR/TODW y se intenta ejecutar simultáneamente otra operación TODR/TODW en una rutina de interrupción, ésta no se procesará. SM4.3 se activa indicando que se intentaron dos accesos simultáneos al reloj (error no fatal 0007).

El sistema de automatización S7-200 no utiliza la información relativa al año de ninguna forma y no es afectado por el cambio de siglo (en el año 2000). No obstante, si en los programas de usuario se utilizan operaciones aritméticas o de comparación con el valor del año, se deberá tener en cuenta la representación de dos dígitos y el cambio de siglo.

Los años bisiestos se tratan correctamente hasta el año 2096.

9.6 Operaciones aritméticas con enteros (SIMATIC)

Sumar y restar enteros de 16 bits



Las operaciones **Sumar enteros de 16 bits** y **Restar enteros de 16 bits** suman/restan dos enteros de 16 bits, arrojando un resultado de 16 bits (OUT).

En KOP y FUP: $IN1 + IN2 = OUT$
 $IN1 - IN2 = OUT$

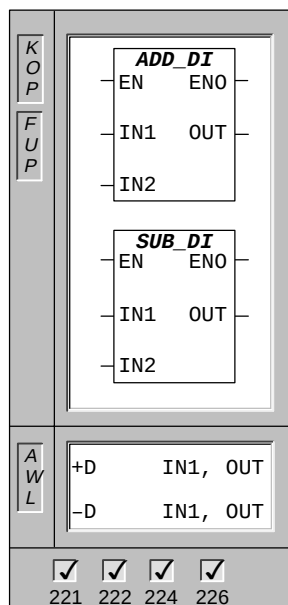
En AWL: $IN1 + OUT = OUT$
 $OUT - IN1 = OUT$

Condiciones de error que ponen ENO a 0:
 SM1.1 (desbordamiento), SM4.3 (tiempo de ejecución),
 0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales:
 SM1.0 (cero); SM1.1 (desbordamiento); SM1.2 (negativo)

Entradas/salidas	Operandos	Tipos de datos
IN1, IN2	VW, IW, QW, MW, SW, SMW, LW, AIW, T, C, AC, constante, *VD, *AC, *LD	INT
OUT	VW, IW, QW, MW, SW, SMW, LW, T, C, AC, *VD, *AC, *LD	INT

Sumar y restar enteros de 32 bits



Las operaciones **Sumar enteros de 32 bits** y **Restar enteros de 32 bits** suman/restan dos enteros de 32 bits, arrojando un resultado de 32 bits (OUT).

En KOP y FUP: $IN1 + IN2 = OUT$
 $IN1 - IN2 = OUT$

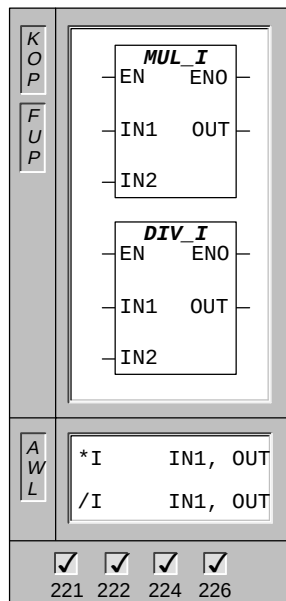
En AWL: $IN1 + OUT = OUT$
 $OUT - IN1 = OUT$

Condiciones de error que ponen ENO a 0:
 SM1.1 (desbordamiento), SM4.3 (tiempo de ejecución),
 0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales:
 SM1.0 (cero); SM1.1 (desbordamiento); SM1.2 (negativo)

Entradas/salidas	Operandos	Tipos de datos
IN1, IN2	VD, ID, QD, MD, SMD, SD, LD, AC, HC, constante, *VD, *AC, *LD	DINT
OUT	VD, ID, QD, MD, SMD, SD, LD, AC, *VD, *AC, *LD	DINT

Multiplicar y dividir enteros de 16 bits



La operación **Multiplicar enteros de 16 bits** multiplica dos números enteros de 16 bits, arrojando un producto de 16 bits.

La operación **Dividir enteros de 16 bits** divide dos números enteros de 16 bits, arrojando un cociente de 16 bits. No se guarda ningún resto.

La marca de desbordamiento se activa si el resultado es mayor que una salida de palabra.

En KOP y FUP: $IN1 * IN2 = OUT$
 $IN1 / IN2 = OUT$

En AWL: $IN1 * OUT = OUT$
 $OUT / IN1 = OUT$

Condiciones de error que ponen ENO a 0:

SM1.1 (desbordamiento), SM1.3 (división por cero),
 SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

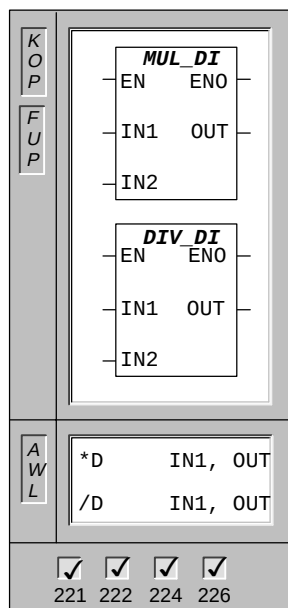
Estas operaciones afectan a las siguientes marcas especiales:
 SM1.0 (cero); SM1.1 (desbordamiento); SM1.2 (negativo);
 SM1.3 (división por cero)

Si SM1.1 (marca de desbordamiento) se activa durante una operación de multiplicación o de división, no se escribe en la salida y todos los demás bits de estado aritméticos se ponen a 0.

Si se activa SM1.3 (división por cero) durante una operación de división, permanecerán inalterados los demás bits aritméticos de estado, así como los operandos de entrada originales. En otro caso, todos los bits aritméticos de estado soportados contendrán el estado válido al finalizar la operación aritmética.

Entradas/salidas	Operandos	Tipos de datos
IN1, IN2	VW, IW, QW, MW, SW, SMW, LW, AIW, T, C, AC, constante, *VD, *AC, *LD	INT
OUT	VW, QW, IW, MW, SW, SMW, LW, T, C, AC, *VD, *LD, *AC	INT

Multiplicar y dividir enteros de 32 bits



La operación **Multiplicar enteros de 32 bits** multiplica dos enteros de 32 bits, arrojando un producto de 32 bits.

La operación **Dividir enteros de 32 bits** divide dos enteros de 32 bits, arrojando un cociente de 32 bits. No se guarda ningún resto.

En KOP y FUP: $IN1 * IN2 = OUT$
 $IN1 / IN2 = OUT$

En AWL: $IN1 * OUT = OUT$
 $OUT / IN1 = OUT$

Condiciones de error que ponen ENO a 0:

SM1.1 (desbordamiento), SM1.3 (división por cero),
 SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

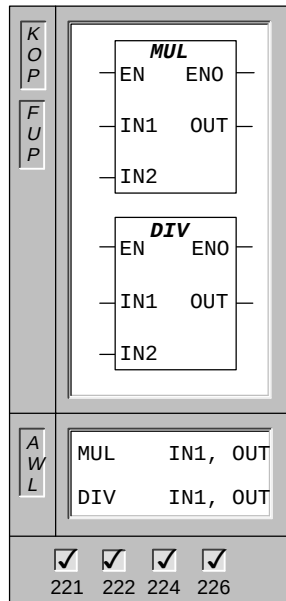
Estas operaciones afectan a las siguientes marcas especiales:
 SM1.0 (cero); SM1.1 (desbordamiento); SM1.2 (negativo);
 SM1.3 (división por cero)

Si SM1.1 (marca de desbordamiento) se activa durante una operación de multiplicación o de división, no se escribe en la salida y todos los demás bits de estado aritméticos se ponen a 0.

Si se activa SM1.3 (división por cero) durante una operación de división, permanecerán inalterados los demás bits aritméticos de estado, así como los operandos de entrada originales. En otro caso, todos los bits aritméticos de estado soportados contendrán el estado válido al finalizar la operación aritmética.

Entradas/salidas	Operandos	Tipos de datos
IN1, IN2	VD, ID, QD, MD, SMD, SD, LD, HC, AC, constante, *VD, *AC, *LD	DINT
OUT	VD, ID, QD, MD, SMD, SD, LD, AC, *VD, *LD, *AC	DINT

Multiplicar y dividir enteros de 16 bits a enteros de 32 bits



La operación **Multiplicar enteros de 16 bits a enteros de 32 bits** multiplica dos números enteros de 16 bits, arrojando un producto de 32 bits.

La operación **Dividir enteros de 16 bits a enteros de 32 bits** divide dos números enteros de 16 bits, arrojando un resultado de 32 bits compuesto de un cociente de 16 bits (los menos significativos) y un resto de 16 bits (los más significativos).

En la operación AWL de multiplicación, la palabra menos significativa (16 bits) del OUT de 32 bits se utiliza como uno de los factores.

En la operación AWL de división, la palabra menos significativa (16 bits) del OUT de 32 bits se utiliza como dividendo.

En KOP y FUP: $IN1 * IN2 = OUT$
 $IN1 / IN2 = OUT$

En AWL: $IN1 * OUT = OUT$
 $OUT / IN1 = OUT$

Condiciones de error que ponen ENO a 0:
 SM1.1 (desbordamiento), SM1.3 (división por cero),
 SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales: SM1.0 (cero);
 SM1.1 (desbordamiento); SM1.2 (negativo); SM1.3 (división por cero)

Si se activa SM1.3 (división por cero) durante una operación de división, permanecerán inalterados los demás bits aritméticos de estado, así como los operandos de entrada originales. En otro caso, todos los bits aritméticos de estado soportados contendrán el estado válido al finalizar la operación aritmética.

Entradas/salidas	Operandos	Tipos de datos
IN1, IN2	VW, IW, QW, MW, SW, SMW, LW, AC, AIW, T, C, constante, *VD, *AC, *LD	INT
OUT	VD, ID, QD, MD, SMD, SD, LD, AC, *VD, *LD, *AC	DINT

Ejemplos de operaciones aritméticas

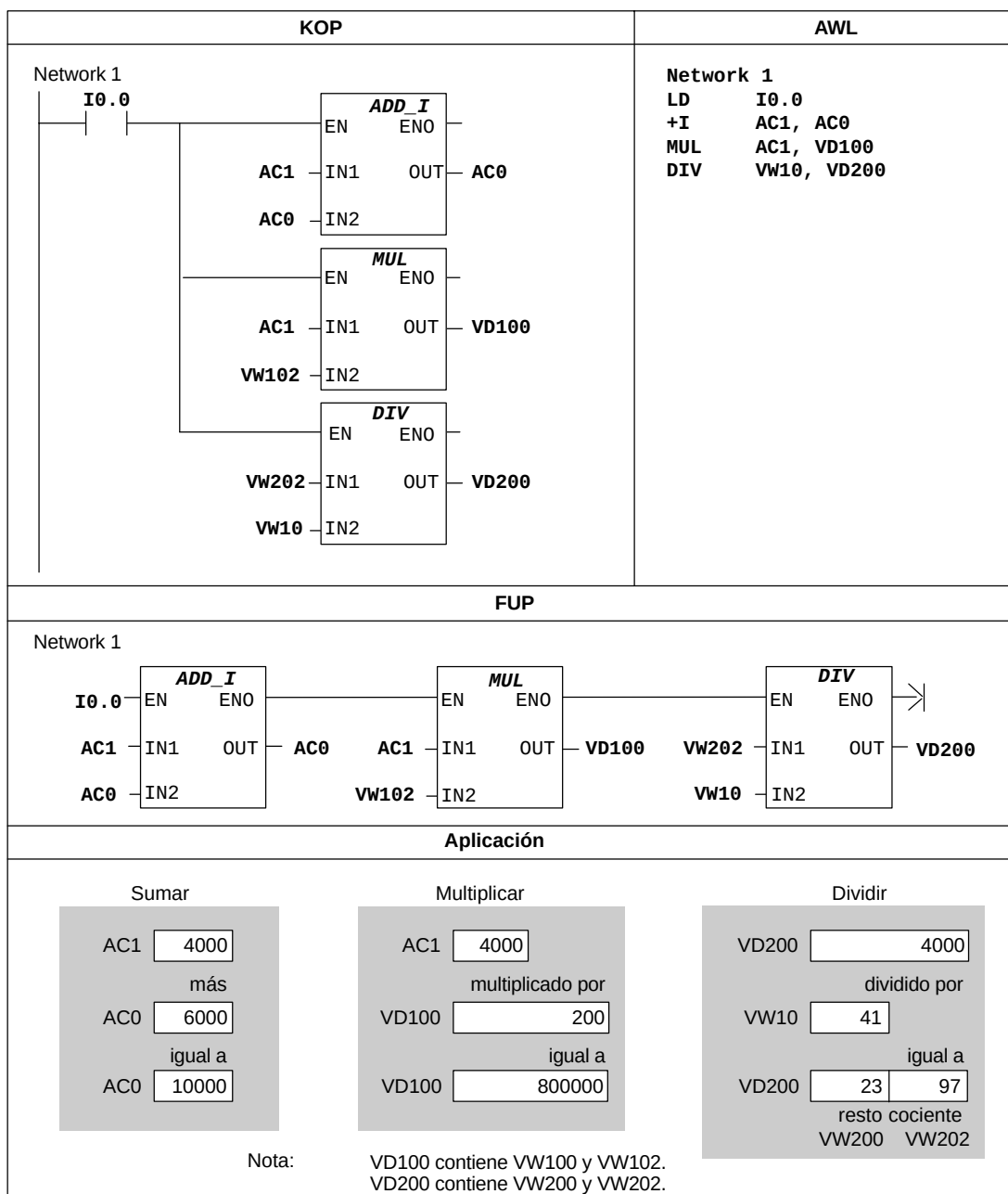
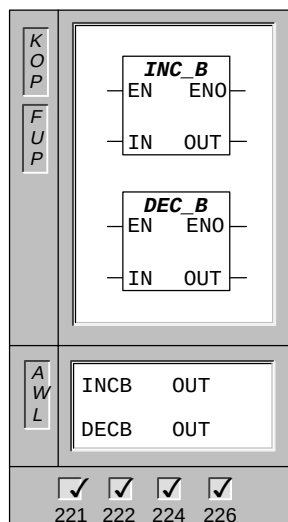


Figura 9-25 Ejemplos de operaciones aritméticas con enteros en KOP, AWL y FUP (SIMATIC)

Incrementar y Decrementar byte



Las operaciones **Incrementar byte** y **Decrementar byte** suman/restan 1 al byte de entrada (IN) y depositan el resultado en la variable indicada por OUT.

Las operaciones Incrementar byte y Decrementar byte no llevan signo.

En KOP y FUP: $IN + 1 = OUT$
 $IN - 1 = OUT$

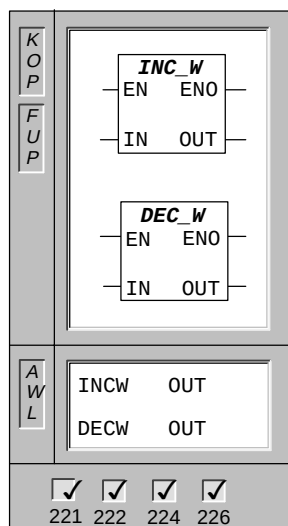
En AWL: $OUT + 1 = OUT$
 $OUT - 1 = OUT$

Condiciones de error que ponen ENO a 0:
 SM1.1 (desbordamiento), SM4.3 (tiempo de ejecución),
 0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales:
 SM1.0 (cero); SM1.1 (desbordamiento)

Entradas/salidas	Operandos	Tipos de datos
IN	VB, IB, QB, MB, SB, SMB, LB, AC, constante, *VD, *AC, *LD	BYTE
OUT	VB, IB, QB, MB, SB, SMB, LB, AC, *VD, *AC, *LD	BYTE

Incrementar y decrementar palabra



Las operaciones **Incrementar palabra** y **Decrementar palabra** suman/restan 1 al valor de la palabra de entrada (IN) y depositan el resultado en OUT.

Las operaciones Incrementar palabra y Decrementar palabra llevan signo ($16\#7FFF > 16\#8000$).

En KOP y FUP: $IN + 1 = OUT$
 $IN - 1 = OUT$

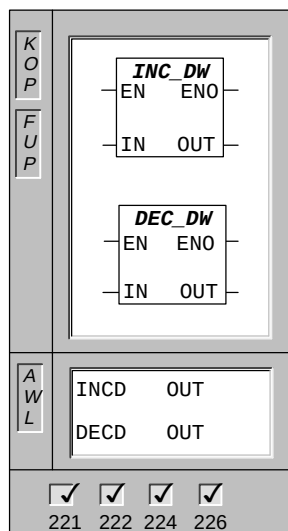
En AWL: $OUT + 1 = OUT$
 $OUT - 1 = OUT$

Condiciones de error que ponen ENO a 0:
 SM1.1 (desbordamiento), SM4.3 (tiempo de ejecución),
 0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales:
 SM1.0 (cero); SM1.1 (desbordamiento); SM1.2 (negativo)

Entradas/salidas	Operandos	Tipos de datos
IN	VW, IW, QW, MW, SW, SMW, AC, AIW, LW, T, C, constante, *VD, *AC, *LD	INT
OUT	VW, IW, QW, MW, SW, SMW, LW, AC, T, C, *VD, *AC, *LD	INT

Incrementar y decrementar palabra doble



Las operaciones **Incrementar palabra doble** y **Decrementar palabra doble** suman/restan 1 al valor de la palabra doble de entrada (IN) y depositan el resultado en OUT.

En KOP y FUP: $IN + 1 = OUT$
 $IN - 1 = OUT$

Las operaciones Incrementar palabra doble y Decrementar palabra doble llevan signo ($16\#7FFFFFFF > 16\#80000000$).

En AWL: $OUT + 1 = OUT$
 $OUT - 1 = OUT$

Condiciones de error que ponen ENO a 0:
 SM1.1 (desbordamiento), SM4.3 (tiempo de ejecución),
 0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales:
 SM1.0 (cero); SM1.1 (desbordamiento); SM1.2 (negativo)

Entradas/salidas	Operandos	Tipos de datos
IN	VD, ID, QD, MD, SD, SMD, LD, AC, HC, constante, *VD, *AC, *LD	DINT
OUT	VD, ID, QD, MD, SD, SMD, LD, AC, *VD, *AC, *LD	DINT

Ejemplos de las operaciones Incrementar y Decrementar

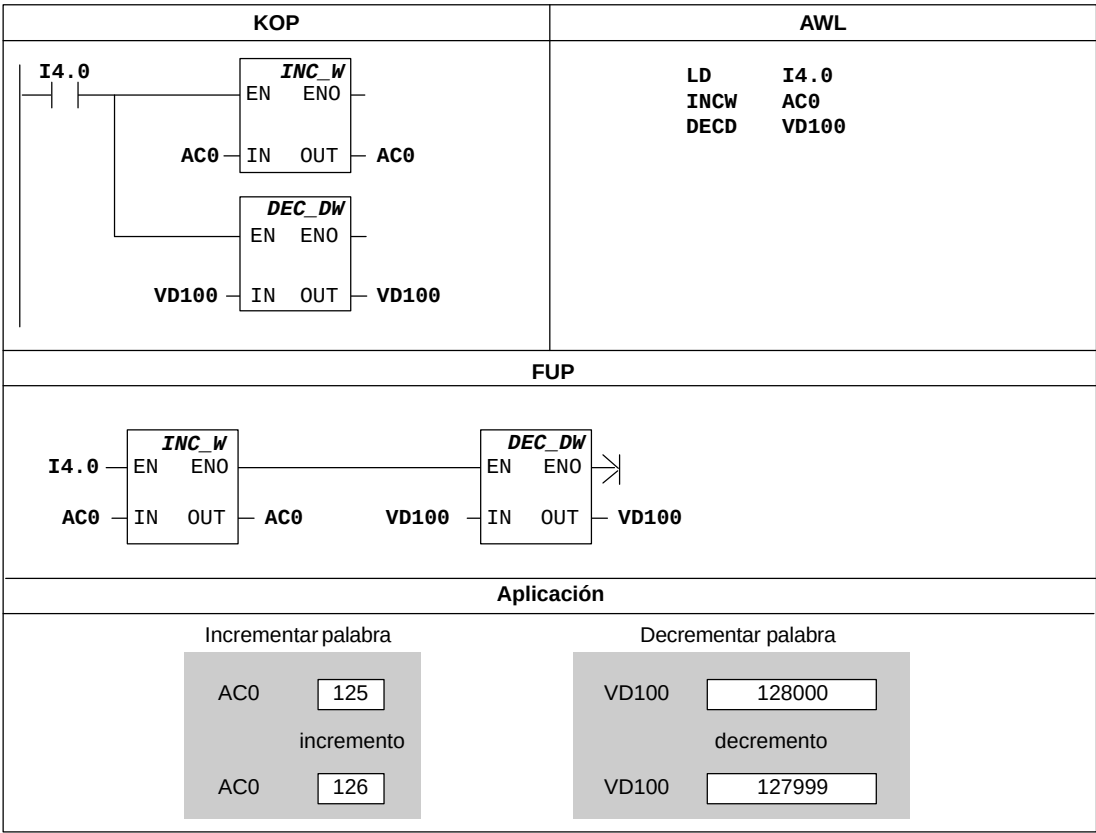
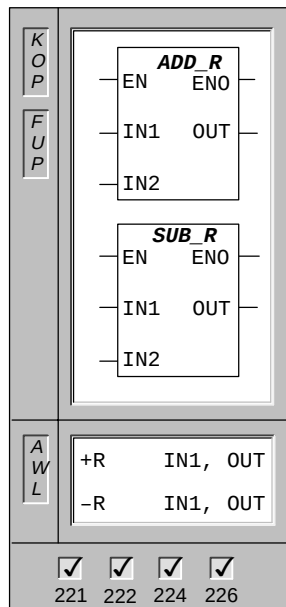


Figura 9-26 Ejemplos de las operaciones Incrementar y Decrementar en KOP, AWL y FUP (SIMATIC)

9.7 Operaciones aritméticas con números reales (SIMATIC)

Sumar y restar reales



Las operaciones **Sumar reales** y **Restar reales** suman/restan dos números reales de 32 bits, dando como resultado un número real de 32 bits (OUT).

En KOP y FUP: $IN1 + IN2 = OUT$
 $IN1 - IN2 = OUT$

En AWL: $IN1 + OUT = OUT$
 $OUT - IN1 = OUT$

Condiciones de error que ponen ENO a 0:
 SM1.1 (desbordamiento), SM4.3 (tiempo de ejecución),
 0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales:
 SM1.0 (cero); SM1.1 (desbordamiento); SM1.2 (negativo)

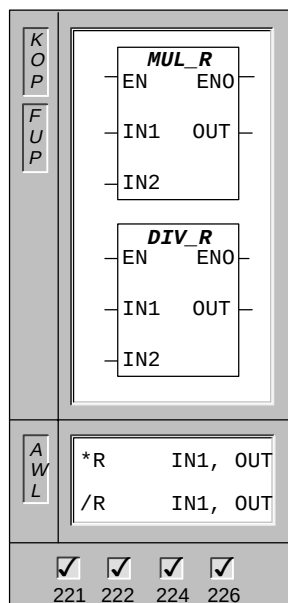
SM1.1 se utiliza para indicar errores de desbordamiento y valores no válidos. Si se activa SM1.1, el estado de SM1.0 y de SM1.2 no será válido y no se alterarán los operandos de entrada originales. Si SM1.1 y SM1.2 no se activan durante una operación de división, la operación aritmética habrá finalizado con un resultado válido, y tanto SM1.0 como SM1.2 contendrán un estado válido.

Entradas/salidas	Operandos	Tipos de datos
IN1, IN2	VD, ID, QD, MD, SD, SMD, AC, LD, constante, *VD, *AC, *LD	REAL
OUT	VD, ID, QD, MD, SD, SMD, AC, LD, *VD, *AC, *LD	REAL

Nota

Los números reales (o números en coma flotante) se representan en el formato descrito en la norma ANSI/IEEE 754-1985 (precisión sencilla). Para obtener más información al respecto, consulte dicha norma.

Multiplicar y dividir reales



La operación **Multiplicar reales** multiplica dos números reales de 32 bits, dando como resultado un número real de 32 bits (OUT).

La operación **Dividir reales** divide dos números reales de 32 bits, dando como resultado un cociente de número real de 32 bits.

En KOP y FUP: $IN1 * IN2 = OUT$
 $IN1 / IN2 = OUT$

En AWL: $IN1 * OUT = OUT$
 $OUT / IN1 = OUT$

Condiciones de error que ponen ENO a 0:
 SM1.1 (desbordamiento), SM1.3 (división por cero),
 SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales:
 SM1.0 (cero); SM1.1 (desbordamiento o valor no válido
 generado durante la operación o parámetro de entrada no
 válido); SM1.2 (negativo); SM1.3 (división por cero)

Si se activa SM1.3 durante una operación de división, permanecerán inalterados los demás bits aritméticos de estado, así como los operandos de entrada originales. SM1.1 se utiliza para indicar errores de desbordamiento y valores no válidos. Si se activa SM1.1, el estado de SM1.0 y de SM1.2 no será válido y no se alterarán los operandos de entrada originales. Si SM1.1 y SM1.3 no se activan (durante una operación de división), la operación aritmética habrá finalizado con un resultado válido, y tanto SM1.0 como SM1.2 contendrán un estado válido.

Entradas/salidas	Operandos	Tipos de datos
IN1, IN2	VD, ID, QD, MD, SMD, SD, LD, AC, constante, *VD, *AC, *LD	REAL
OUT	VD, ID, QD, MD, SMD, SD, LD, AC, *VD, *AC, *LD	REAL

Nota

Los números reales (o números en coma flotante) se representan en el formato descrito en la norma ANSI/IEEE 754-1985 (precisión sencilla). Para obtener más información al respecto, consulte dicha norma.

Ejemplos de operaciones aritméticas

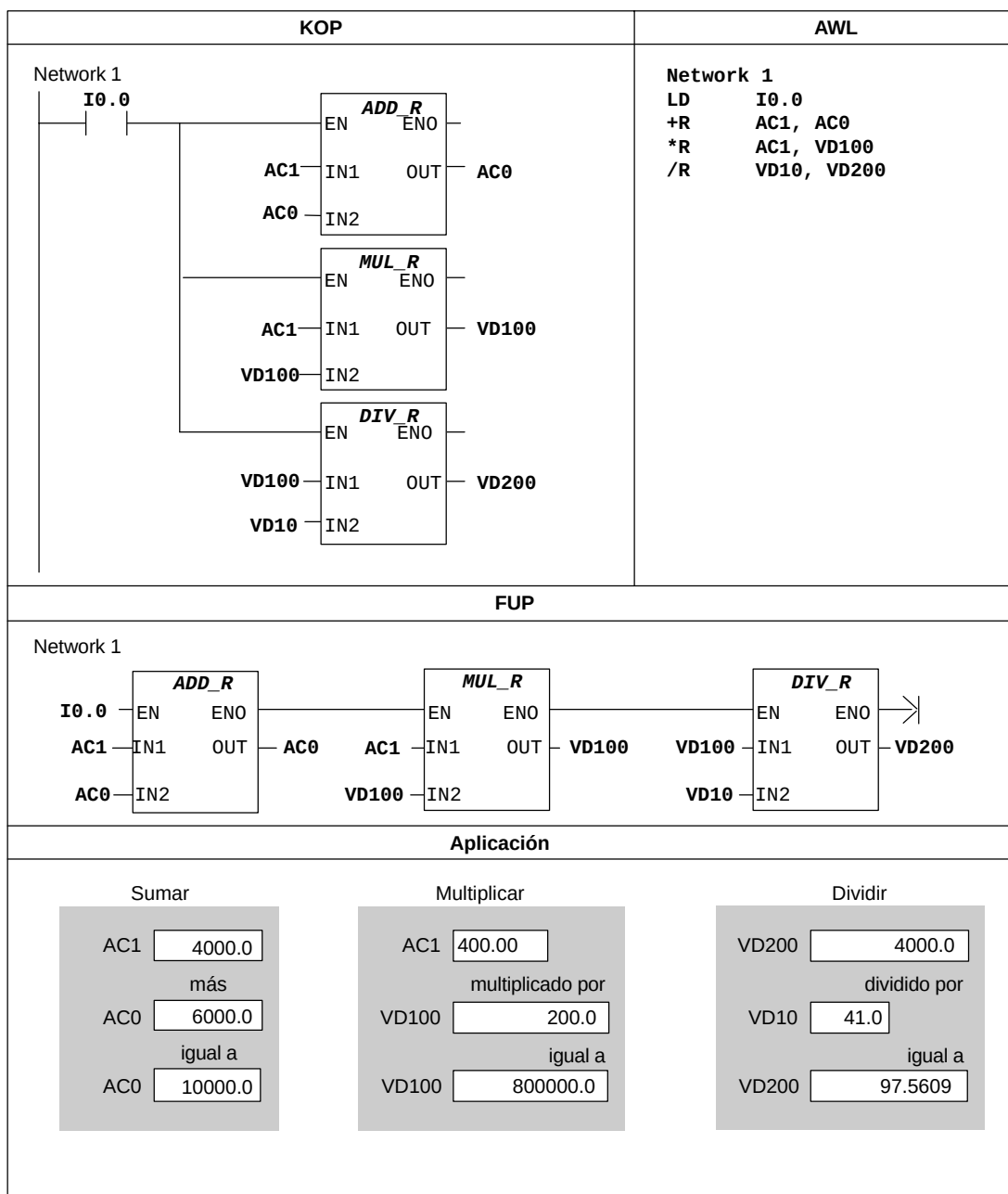
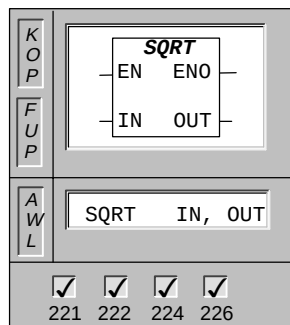


Figura 9-27 Ejemplos de operaciones aritméticas con números reales en KOP, AWL y FUP (SIMATIC)

9.8 Operaciones con funciones numéricas (SIMATIC)

Raíz cuadrada



La operación **Raíz cuadrada** extrae la raíz cuadrada de un número real de 32 bits (IN), dando como resultado un número real de 32 bits (OUT), como muestra la ecuación:

$$\sqrt{\text{IN}} = \text{OUT}$$

Condiciones de error que ponen ENO a 0:

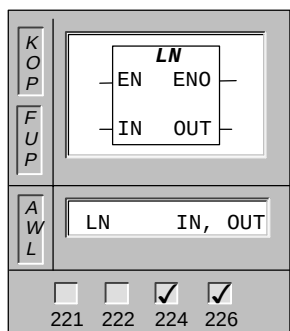
SM1.1 (desbordamiento), SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales: SM1.0 (cero); SM1.1 (desbordamiento); SM1.2 (negativo)

SM1.1 sirve para indicar errores de desbordamiento y valores no válidos. Si se activa SM1.1, el estado de SM1.0 y de SM1.2 no será válido y no se alterarán los operandos de entrada originales. Si SM1.1 y SM1.2 no se activan durante una operación de división, la operación aritmética habrá finalizado con un resultado válido, y tanto SM1.0 como SM1.2 contendrán un estado válido. Para obtener otras raíces, consulte la descripción de la operación Exponencial natural (v. pág. 9-86).

Entradas/salidas	Operandos	Tipos de datos
IN	VD, ID, QD, MD, SMD, SD, LD, AC, constante, *VD, *AC, *LD	REAL
OUT	VD, ID, QD, MD, SMD, SD, LD, AC, *VD, *AC, *LD	REAL

Logaritmo natural



La operación **Logaritmo natural** calcula el logaritmo natural del valor de IN y deposita el resultado en OUT. Para obtener la base 10 del logaritmo natural, utilice la operación Dividir reales (DIV_R (/R)), con objeto de dividir el logaritmo natural por 2,302585 (que es aproximadamente el logaritmo natural de 10).

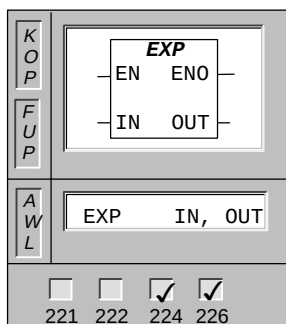
Condiciones de error que ponen ENO a 0:

SM1.1 (desbordamiento), 0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales: SM1.0 (cero); SM1.1 (desbordamiento); SM1.2 (negativo); SM4.3 (división por cero)

Entradas/salidas	Operandos	Tipos de datos
IN	VD, ID, QD, MD, SMD, SD, LD, AC, constante, *VD, *AC, *LD	REAL
OUT	VD, ID, QD, MD, SMD, SD, LD, AC, *VD, *AC, *LD	REAL

Exponencial natural



La operación **Exponencial natural** ejecuta el cálculo exponencial de la constante “e” elevada a la potencia del valor de IN y deposita el resultado en OUT. La operación Exponencial natural se puede utilizar en combinación con la operación Logaritmo natural para elevar cualquier número real a la potencia de otro número real, incluyendo exponentes fraccionarios. Por consiguiente, X elevado a la potencia de Y se puede calcular de la siguiente forma:
 $EXP(Y * LN(X))$.

Ejemplos:

5 elevado al cubo = $5^3 = EXP(3 * LN(5)) = 125$

Raíz cúbica de 125 = $125^{(1/3)} = EXP(1/3 * LN(125)) = 5$

Raíz cuadrada de

5 elevado al cubo = $5^{(3/2)} = EXP(3/2 * LN(5)) = 11,18034 \dots$

Condiciones de error que ponen ENO a 0:

SM1.1 (desbordamiento), 0006 (direccionamiento indirecto)

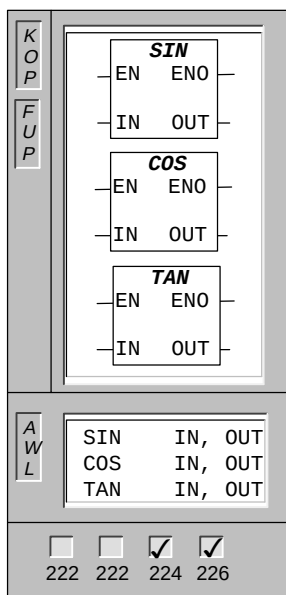
Estas operaciones afectan a las siguientes marcas especiales:

SM1.0 (cero); SM1.1 (desbordamiento); SM1.2 (negativo);

SM4.3 (división por cero)

Entradas/salidas	Operandos	Tipos de datos
IN	VD, ID, QD, MD, SMD, SD, LD, AC, constante, *VD, *AC, *LD	REAL
OUT	VD, ID, QD, MD, SMD, SD, LD, AC, *VD, *AC, *LD	REAL

Seno, Coseno y Tangente



Las operaciones **Seno, Coseno y Tangente** evalúan la función trigonométrica del valor del ángulo IN y depositan el resultado en OUT. El valor del ángulo de entrada se indica en radianes. Para convertir un ángulo de grados a radianes, utilice la operación Multiplicar reales (MUL_R (*R)) para multiplicar el ángulo en grados por $1,745329E-2$ (aproximadamente por $\pi / 180$).

Condiciones de error que ponen ENO a 0:

SM1.1 (desbordamiento), 0006 (direccionamiento indirecto)

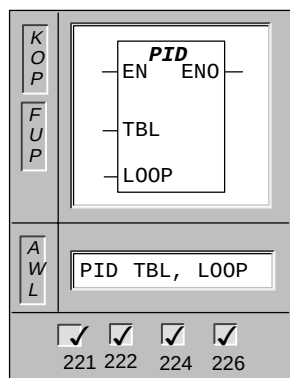
Estas operaciones afectan a las siguientes marcas especiales:

SM1.0 (cero); SM1.1 (desbordamiento); SM1.2 (negativo);

SM1.3 (división por cero)

Entradas/salidas	Operandos	Tipos de datos
IN	VD, ID, QD, MD, SMD, SD, LD, AC, constante, *VD, *AC, *LD	REAL
OUT	VD, ID, QD, MD, SMD, SD, LD, AC, *VD, *AC, *LD	REAL

Regulación PID



La operación **Regulación PID** ejecuta el cálculo de un lazo de regulación PID en el LOOP referenciado en base a las informaciones de entrada y configuración definidas en Table (TBL).

Condiciones de error que ponen ENO a 0:
SM1.1 (desbordamiento), SM4.3 (tiempo de ejecución),
0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales:
SM1.1 (desbordamiento)

Entradas/salidas	Operandos	Tipos de datos
TBL	VB	BYTE
LOOP	Constante(0 a 7)	BYTE

La operación PID (lazo de regulación con acción proporcional, integral, derivada) sirve para ejecutar el cálculo PID. Para habilitar el cálculo PID, el primer nivel de la pila lógica (TOS) deberá estar a ON (circulación de corriente). Esta operación tiene dos operandos: una dirección TABLE que constituye la dirección inicial de la tabla del lazo y un número LOOP que es una constante comprendida entre 0 y 7. Un programa sólo admite ocho operaciones PID. Si se utilizan dos o más operaciones PID con el mismo número de lazo (aunque tengan diferentes direcciones de tabla), los dos cálculos PID se interferirán mutuamente siendo impredecible la salida resultante.

La tabla del lazo almacena nueve parámetros que sirven para controlar y supervisar la operación del mismo. Incluye el valor actual y previo de la variable del proceso (valor real), la consigna, la salida o magnitud manipulada, la ganancia, el tiempo de muestreo, el tiempo de acción integral, el tiempo de acción derivada y la suma integral (bias).

Para poder realizar el cálculo PID con el intervalo de muestreo deseado, la operación PID deberá ejecutarse bien dentro de una rutina de interrupción temporizada o desde el programa principal, a intervalos controlados por un temporizador. El tiempo de muestreo debe definirse en calidad de entrada para la operación PID a través de la tabla del lazo.

Utilizar el Asistente PID en STEP 7-Micro/WIN 32

STEP 7-Micro/WIN 32 incorpora el Asistente PID que ayuda a definir un algoritmo PID para un proceso de control de bucle cerrado. Elija el comando de menú **Herramientas ► Asistente de operaciones** y seleccione "PID" en la ventana del Asistente.

Algoritmo PID

En estado estacionario, un regulador PID varía el valor de su salida para llevar a cero el error de regulación (e). El error es la diferencia entre el valor de consigna (SP) (el punto de trabajo deseado) y la variable del proceso (PV) (el punto de trabajo real). El principio de una regulación PID se basa en la ecuación que se indica a continuación y que expresa la salida $M(t)$ como una función de un término proporcional, uno integral y uno diferencial:

$M(t)$	=	$K_C * e$	+	$K_C \int_0^t e \, dt + M_{\text{inicial}}$	+	$K_C * de/dt$
Salida	=	término proporcional	+	término integral	+	término diferencial

donde:

$M(t)$ es la salida del lazo en función del tiempo
 K_C es la ganancia del lazo
 e es el error de regulación (diferencia entre consigna y variable de proceso)
 M_{inicial} es el valor inicial de la salida del lazo

Para poder implementar esta función de regulación en un sistema digital, la función continua deberá cuantificarse mediante muestreos periódicos del valor del error, calculándose seguidamente el valor de la salida. La ecuación que constituye la base de la solución en un sistema digital es:

M_n	=	$K_C * e_n$	+	$K_I * \sum_1^n + M_{\text{inicial}}$	+	$K_D * (e_n - e_{n-1})$
Salida	=	término proporcional	+	término integral	+	término diferencial

donde:

M_n es el valor de salida del lazo calculado en el muestreo n-ésimo
 K_C es la ganancia del lazo
 e_n es el valor del error de regulación en el muestreo n-ésimo
 e_{n-1} es el valor previo del error de regulación (en el muestreo (n-1)-ésimo)
 K_I es la constante proporcional del término integral
 M_{inicial} es el valor inicial de la salida del lazo
 K_D es la constante proporcional del término diferencial

Para esta ecuación, el término integral se muestra en función de todos los términos del error, desde el primer muestreo hasta el muestreo actual. El término diferencial es una función del muestreo actual y del muestreo previo; mientras que el término proporcional sólo es función del muestreo actual. En un sistema digital no es práctico almacenar todos los muestreos del término del error, además de no ser necesario.

Como un sistema digital debe calcular el valor de salida cada vez que se muestre el error, comenzando por el primer muestreo, basta con almacenar el valor previo del error y el valor previo del término integral. Debido a la naturaleza repetitiva de la solución basada en un sistema digital es posible simplificar la ecuación a resolver en cada muestreo. La ecuación simplificada es:

M_n	=	K_C * e_n	+	K_I * e_n + MX	+	K_D * (e_n-e_{n-1})
Salida	=	término proporcional	+	término integral	+	término diferencial

donde:

M _n	es el valor de salida del lazo calculado en el muestreo n-ésimo
K _C	es la ganancia del lazo
e _n	es el valor del error de regulación en el muestreo n-ésimo
e _{n-1}	es el valor previo del error de regulación (en el muestreo (n-1)-ésimo)
K _I	es la constante proporcional del término integral
MX	es el valor previo del término integral (en el muestreo (n-1)-ésimo)
K _D	es la constante proporcional del término diferencial

Para calcular el valor de salida del lazo, la CPU utiliza una forma modificada de la ecuación simplificada anterior. Esta ecuación modificada equivale a:

M_n	=	MP_n	+	MI_n	+	MD_n
Salida	=	término proporcional	+	término integral	+	término diferencial

donde:

M _n	es el valor de salida del lazo calculado en el muestreo n-ésimo
MP _n	es el valor del término proporcional de salida del lazo en el muestreo n-ésimo
MI _n	es el valor del término integral de salida del lazo en el muestreo n-ésimo
MD _n	es el valor del término diferencial de salida del lazo en el muestreo n-ésimo

Término proporcional

El término proporcional MP es el producto de la ganancia (K_C), la cual controla la sensibilidad del cálculo de la salida, y del error (e), que es la diferencia entre el valor de consigna (SP) y el valor real o de la variable del proceso (PV) para un instante de muestreo determinado. La ecuación que representa el término proporcional según la resuelve la CPU es la siguiente:

$$MP_n = K_C * (SP_n - PV_n)$$

donde:

MP _n	es el valor del término proporcional de salida del lazo en el muestreo n-ésimo
K _C	es la ganancia del lazo
SP _n	es el valor de la consigna en el muestreo n-ésimo
PV _n	es el valor de la variable del proceso en el muestreo n-ésimo

Término integral

El término integral MI es proporcional a la suma del error a lo largo del tiempo. La ecuación que representa el término integral tal y como la resuelve la CPU es:

$$MI_n = K_C * T_S / T_I * (SP_n - PV_n) + MX$$

donde:

MI_n	es el valor del término integral de salida del lazo en el muestreo n-ésimo
K_C	es la ganancia del lazo
T_S	es el tiempo de muestreo del lazo
T_I	es el período de integración del lazo (también llamado tiempo de acción integral)
SP_n	es el valor de la consigna en el muestreo n-ésimo
PV_n	es el valor de la variable del proceso en el muestreo n-ésimo
MX	es el valor previo del término integral (en el muestreo (n-1)-ésimo) (también llamado suma integral o "bias")

La suma integral o bias (MX) es la suma acumulada de todos los valores previos del término integral. Después de cada cálculo de MI_n se actualiza la suma integral con el valor de MI_n que puede ajustarse o limitarse (para más detalles, v. la sección "Variables y márgenes" en la página 9-95). Por regla general, el valor inicial de la suma integral se ajusta al valor de salida ($M_{inicial}$) justo antes de calcular la primera salida del lazo. El término integral incluye también varias constantes tales como la ganancia (K_C), el tiempo de muestreo (T_S), que define el intervalo con que se recalcula periódicamente el valor de salida del lazo PID, y el tiempo de acción integral (T_I), que es un tiempo que se utiliza para controlar la influencia del término integral en el cálculo de la salida.

Término diferencial

El término diferencial MD es proporcional a la tasa de cambio del error. La ecuación del término diferencial es la siguiente:

$$MD_n = K_C * T_D / T_S * ((SP_n - PV_n) - (SP_{n-1} - PV_{n-1}))$$

Para evitar cambios o saltos bruscos de la salida debidos a cambios de la acción derivada o de la consigna se ha modificado esta ecuación bajo la hipótesis de que la consigna es constante ($SP_n = SP_{n-1}$). En consecuencia, se calcula el cambio en la variable del proceso en lugar del cambio en el error, como puede verse a continuación:

$$MD_n = K_C * T_D / T_S * (SP_n - PV_n - SP_n + PV_{n-1})$$

o simplificando:

$$MD_n = K_C * T_D / T_S * (PV_{n-1} - PV_n)$$

donde:

MD_n	es el valor del término diferencial de la salida del lazo en el muestreo n-ésimo
K_C	es la ganancia del lazo
T_S	es el tiempo de muestreo del lazo
T_D	es el período de diferenciación de lazo (también llamado tiempo de acción derivada)
SP_n	es el valor de la consigna en el muestreo n-ésimo
SP_{n-1}	es el valor de la consigna en el muestreo (n-1)-ésimo
PV_n	es el valor de la variable del proceso en el muestreo n-ésimo
PV_{n-1}	es el valor de la variable del proceso en el muestreo (n-1)-ésimo

En lugar del error es necesario guardar la variable del proceso para usarla en el próximo cálculo del término diferencial. En el instante del primer muestreo, el valor de PV_{n-1} se inicializa a un valor igual a PV_n .

Elegir el tipo de regulación

En muchos sistemas de regulación basta emplear una o dos acciones de regulación. Así, por ejemplo, puede requerirse únicamente regulación proporcional o regulación proporcional e integral. El tipo de regulación se selecciona ajustando correspondientemente los valores de los parámetros constantes.

Así, si no se desea acción integral (sin "I" en el cálculo PID), entonces el tiempo de acción integral deberá ajustarse a infinito. Incluso sin acción integral el valor del término integral puede no ser cero debido a que la suma integral MX puede tener un valor inicial.

Si no se desea acción derivada (sin "D" en el cálculo PID), entonces el tiempo de acción derivada deberá ajustarse a 0.0.

Si no se desea acción proporcional (sin "P" en el cálculo PID) y se desea regulación I o ID, entonces la ganancia deberá ajustarse a 0.0. Como la ganancia interviene en las ecuaciones para calcular los términos integral y diferencial, si se ajusta a 0.0 resulta un valor de 1.0, que es el utilizado para calcular los términos integral y diferencial.

Convertir y normalizar las entradas del lazo

El lazo tiene dos variables o magnitudes de entrada: la consigna y la variable del proceso. La consigna es generalmente un valor fijo como el ajuste de velocidad en el computador de abordaje de su automóvil. La variable del proceso es una magnitud relacionada con la salida del lazo y que mide por ello el efecto que tiene la misma sobre el sistema regulado. En el ejemplo del computador de abordaje, la variable del proceso sería la entrada al tacómetro que es una señal proporcional a la velocidad de giro de las ruedas.

Tanto la consigna como la variable del proceso son valores físicos que pueden tener diferente magnitud, margen y unidades de ingeniería. Para que la operación PID pueda utilizar esos valores físicos, éstos deberán convertirse a representaciones normalizadas en coma flotante.

El primer paso es convertir el valor físico de un valor entero de 16 bits a un valor en coma flotante o real. La siguiente secuencia de instrucciones muestra la forma de convertir un valor entero en un número real.

```
XORD  AC0, AC0           // Borrar el acumulador.
MOVW  AIW0, AC0          // Guardar en el acumulador el valor analógico.
LDW>= AC0, 0             // Si el valor analógico es positivo,
JMP    0                 // entonces convertir a número real.
NOT    0                 // Si no,
ORD    16#FFFF0000, AC0  // el signo amplía el valor en AC0.
LBL    0
DTR    AC0, AC0          // Convertir entero de 32 bits a un número real.
```

El próximo paso consiste en convertir el número real representativo del valor físico en un valor normalizado entre 0.0 y 1.0. La ecuación siguiente se utiliza para normalizar tanto la consigna como el valor de la variable del proceso.

$$R_{\text{Norm}} = ((R_{\text{No norm}} / \text{Alcance}) + \text{Offset})$$

donde:

R_{Norm} es la representación como número real normalizado del valor físico

$R_{\text{No norm}}$ es la representación como número real no normalizado del valor físico

Offset vale 0.0 para valores unipolares
vale 0.5 para valores bipolares

Alcance es la diferencia entre el máximo valor posible menos el mínimo valor posible
= 32.000 para valores unipolares (típico)
= 64.000 para valores bipolares (típico)

La siguiente secuencia de instrucciones muestra la forma de normalizar el valor bipolar contenido en AC0 (cuyo alcance vale 64.000), continuando la secuencia previa:

```
/R      64000.0, AC0      // Normaliza el valor en el acumulador
+R      0.5, AC0          // Desplaza el valor al margen entre 0.0 y 1.0
MOVR    AC0, VD100        // Almacena el valor normalizado en la tabla del lazo
```

Convertir la salida del lazo en un valor entero escalado

La salida del lazo constituye la variable manipulada; en el caso del automóvil, la posición de la mariposa en el carburador. La salida del lazo es un valor real normalizado comprendido entre 0.0 y 1.0. Antes de que la salida del lazo pueda utilizarse para excitar una salida analógica, deberá convertirse a un valor escalado de 16 bits. Esta operación constituye el proceso inverso de convertir PV y SP en un valor normalizado. El primer paso es convertir la salida del lazo en un valor real escalado usando la fórmula siguiente:

$$R_{\text{Scal}} = (M_n - \text{Offset}) * \text{Alcance}$$

donde:

R_{Scal}	es el valor real escalado de la salida del lazo
M_n	es el valor real normalizado de la salida del lazo
Offset	vale 0,0 para valores unipolares vale 0,5 para valores bipolares
Alcance	es la diferencia entre el máximo valor posible menos el mínimo valor posible = 32.000 para valores unipolares (típico) = 64.000 para valores bipolares (típico)

La siguiente secuencia de instrucciones muestra la forma de escalar la salida del lazo:

```
MOVR  VD108, AC0      // Mover la salida del lazo al acumulador.
-R     0.5, AC0        // Incluir esta operación sólo si el valor es
                        // bipolar.
*R     64000.0, AC0    // Escalar el valor en el acumulador.
```

Seguidamente es necesario convertir en un entero de 16 bits el valor real escalado representativo de la salida del lazo. La siguiente secuencia muestra la forma de realizar esta conversión:

```
ROUND AC0 AC0          // Convertir entero de 32 bits a un número real.
MOVW  AC0, AQW0        // Escribir el entero de 16 bits en la salida
                        // analógica.
```

Lazos con acción positiva o negativa

El lazo tiene acción positiva si la ganancia es positiva y acción negativa si la ganancia es negativa. (En regulación I o ID, donde la ganancia vale 0.0, si se especifica un valor positivo para el tiempo de acción integral y derivada resulta un lazo de acción positiva y de acción negativa al especificarse valores negativos).

Variables y márgenes

La variable del proceso y la consigna son magnitudes de entrada para el cálculo PID. Por ello, la operación PID lee los campos definidos para estas variables en la tabla del lazo, pero no los modifica.

El valor de salida se genera al realizar el cálculo PID; como consecuencia, el campo en la tabla del lazo que contiene el valor de salida se actualiza cada vez que se termina un cálculo PID. El valor de salida está limitado entre 0.0 y 1.0. El usuario puede utilizar el campo de valor de salida en calidad de campo de entrada para especificar un valor de salida inicial cuando se conmute de control manual a automático (consulte también la sección "Modos").

Si se utiliza regulación integral, la suma integral es actualizada por el cálculo PID y el valor actualizado se utiliza como entrada para el siguiente cálculo PID. Si el valor de salida calculado se sale de margen (salida inferior a 0.0 o superior a 1.0), la suma integral se ajusta de acuerdo con las fórmulas siguientes:

$$MX = 1.0 - (MP_n + MD_n) \quad \text{si la salida calculada, } M_n > 1.0$$

o bien,

$$MX = - (MP_n + MD_n) \quad \text{si la salida calculada, } M_n < 0.0$$

donde:

MX	es el valor de la suma integral ajustada
MP _n	es el valor del término proporcional de salida del lazo en el muestreo n-ésimo
MD _n	es el valor del término diferencial de la salida del lazo en el muestreo n-ésimo
M _n	es el valor de la salida del lazo en el muestreo n-ésimo

Si la suma integral se calcula de la forma descrita, se mejora la respuesta del sistema cuando la salida calculada retorna al margen adecuado. Es decir, la suma integral calculada se limita entre 0.0 y 1.0 y luego se escribe en el campo reservado para ella en la tabla del lazo cada vez que se finaliza un cálculo PID. El valor almacenado en la tabla del lazo se utiliza para el próximo cálculo PID.

A fin de evitar problemas con valores de la suma integral en determinadas situaciones de aplicación, el usuario puede modificar, antes de ejecutar la operación PID, el valor de la suma integral en la tabla del lazo. Cualquier modificación manual de la suma integral deberá realizarse con mucho cuidado. En cualquier caso, el valor de la suma integral escrito en la tabla del lazo deberá ser un número real comprendido entre 0.0 y 1.0.

En la tabla del lazo se mantiene un valor de comparación de la variable del proceso para su uso en la parte de acción derivada del cálculo PID. El usuario no deberá modificar este valor.

Modos

Los lazos PID del S7-200 no incorporan control de modo de operación. El cálculo PID sólo se ejecuta si circula corriente hacia el cuadro PID. Por ello resulta el modo "automático" o "auto" cuando se ejecuta cíclicamente el cálculo PID. Resulta el modo "manual" cuando no se ejecuta el cálculo PID.

La operación PID tiene un bit de historial de circulación de corriente similar a una operación de contador. La operación utiliza este bit para detectar una transición de 0 a 1 de la circulación de la corriente. Cuando se detecta la transición, la operación ejecutará una serie de acciones a fin de asegurar un cambio sin choques entre control manual y automático. Para evitar choques en la transición al modo automático, el valor de la salida ajustado por control manual deberá entregarse en calidad de entrada a la operación PID (escrita en la entrada para M_n en la tabla del lazo) antes de conmutar a modo automático. La operación PID ejecuta las siguientes acciones con los valores de la tabla del lazo a fin de asegurar un cambio sin choques entre control manual y automático cuando se detecta una transición de la circulación de corriente de 0 a 1:

- Ajusta consigna (SP_n) = variable de proceso (PV_n)
- Ajusta variable del proceso antigua (PV_{n-1}) = variable del proceso (PV_n)
- Ajusta suma integral (MX) = valor de salida (M_n)

El estado por defecto de los bits de historial PID es "activado"; este estado se establece en el arranque de la CPU o cada vez que hay una transición de modo STOP a RUN en el sistema de automatización. Si circula corriente hacia el cuadro PID la primera vez que se ejecuta tras entrar en el modo RUN, entonces no se detecta ninguna transición de la circulación de corriente y, por consecuencia, no se ejecutan las acciones destinadas a evitar choques en el cambio de modo.

Alarmas y operaciones especiales

La operación PID es simple, pero ofrece grandes prestaciones para ejecutar cálculos PID. Si se precisan funciones de postprocesamiento tales como funciones de alarma o cálculos especiales en base a las variables de lazo, ésto deberá implementarse utilizando las instrucciones básicas admitidas por la CPU en cuestión.

Condiciones de error

A la hora de compilar, la CPU generará un error de compilación (error de margen) y la compilación fallará si los operandos correspondientes a la dirección inicial o al número de lazo PID en la tabla del lazo están fuera de margen.

La operación PID no comprueba si todos los valores de entrada en la tabla del lazo respetan los límites de margen. Es decir, el usuario debe asegurarse de que la variable del proceso y la consigna (al igual que la suma integral y la variable del proceso previa, si se utilizan como entradas) sean números reales comprendidos entre 0.0 y 1.0.

Si se detecta algún error al ejecutar las operaciones aritméticas del cálculo PID se activa la marca SM1.1 (desbordamiento o valor no válido) y se finaliza la ejecución de la operación PID. (La actualización de los valores de salida en la tabla del lazo puede ser incompleta por lo que deberán descartarse estos valores y corregir el valor de entrada que ha provocado el error matemático antes de volver a ejecutar la operación de regulación PID.)

Tabla del lazo

La tabla de lazo tiene 36 bytes de longitud y el formato que muestra la tabla 9-19:

Tabla 9-19 Formato de la tabla del lazo

Offset	Campo	Formato	Tipo	Descripción
0	Variable del proceso (PV_n)	Palabra doble – real	IN	Contiene la variable del proceso que debe estar escalada entre 0.0 y 1.0.
4	Consigna (SP_n)	Palabra doble – real	IN	Contiene la consigna que debe estar escalada entre 0.0 y 1.0.
8	Salida (M_n)	Palabra doble – real	IN/OUT	Contiene la salida calculada, escalada entre 0.0 y 1.0.
12	Ganancia (K_C)	Palabra doble – real	IN	Contiene la ganancia, que es una constante proporcional. Puede ser un número positivo o negativo.
16	Tiempo de muestreo (T_S)	Palabra doble – real	IN	Contiene el tiempo de muestreo en segundos. Tiene que ser un número positivo.
20	Tiempo de acción integral (T_I)	Palabra doble – real	IN	Contiene el tiempo de acción integral en minutos. Tiene que ser un número positivo.
24	Tiempo de acción derivada (T_D)	Palabra doble – real	IN	Contiene el tiempo de acción derivada en minutos. Tiene que ser un número positivo.
28	Suma integral (MX)	Palabra doble – real	IN/OUT	Contiene el valor de la suma integral entre 0.0 y 1.0.
32	Variable del proceso previa (PV_{n-1})	Palabra doble – real	IN/OUT	Contiene el último valor real, el cual se almacenó al ejecutar por última vez la instrucción PID.

Ejemplo de programa PID

En este ejemplo se utiliza un depósito para mantener una presión de agua constante. Para ello se toma continuamente agua del depósito en una cantidad variable. Una bomba de velocidad variable se utiliza para añadir agua al depósito con un caudal apto para mantener una presión adecuada del agua y evitar así que se vacíe.

La consigna de este sistema es el nivel de agua; en este caso, un valor equivalente al 75% de llenado. La variable del proceso la suministra un sensor flotador que señala el nivel de llenado del depósito; equivale a 0 % cuando está vacío y a 100 % cuando está completamente lleno. La salida es una señal que permite controlar la velocidad de la bomba, del 0 al 100 % de su velocidad máxima.

La consigna está predeterminada y se introduce directamente en la tabla del lazo. El sensor flotador suministra la variable del proceso que es un valor analógico unipolar. La salida del lazo se escribe en una salida analógica unipolar que se utiliza para controlar la velocidad de la bomba. El alcance tanto de la entrada como de la salida analógica es de 32.000.

En este ejemplo sólo se utiliza acción proporcional e integral. La ganancia del lazo y las constantes de tiempo se han determinado durante cálculos de ingeniería y se ajustan para obtener una regulación óptima. Los valores calculados de las constantes de tiempo se indican a continuación:

K_C es 0,25

T_S es 0,1 segundos

T_I es 30 minutos

La velocidad de la bomba se controlará de forma manual hasta que el depósito esté lleno al 75 %, seguidamente se abre la válvula para sacar agua del mismo. Simultáneamente se conmuta la bomba de modo manual a automático. La entrada digital se utiliza para conmutar de manual a automático. Esta entrada se describe seguidamente:

I0.0 es control manual/automático; 0 = manual, 1 = automático

En modo manual, el operador ajusta la velocidad de la bomba en VD108 mediante un valor real de 0.0 a 1.0.

La figura 9-28 muestra el programa de control (regulación) para esta aplicación.

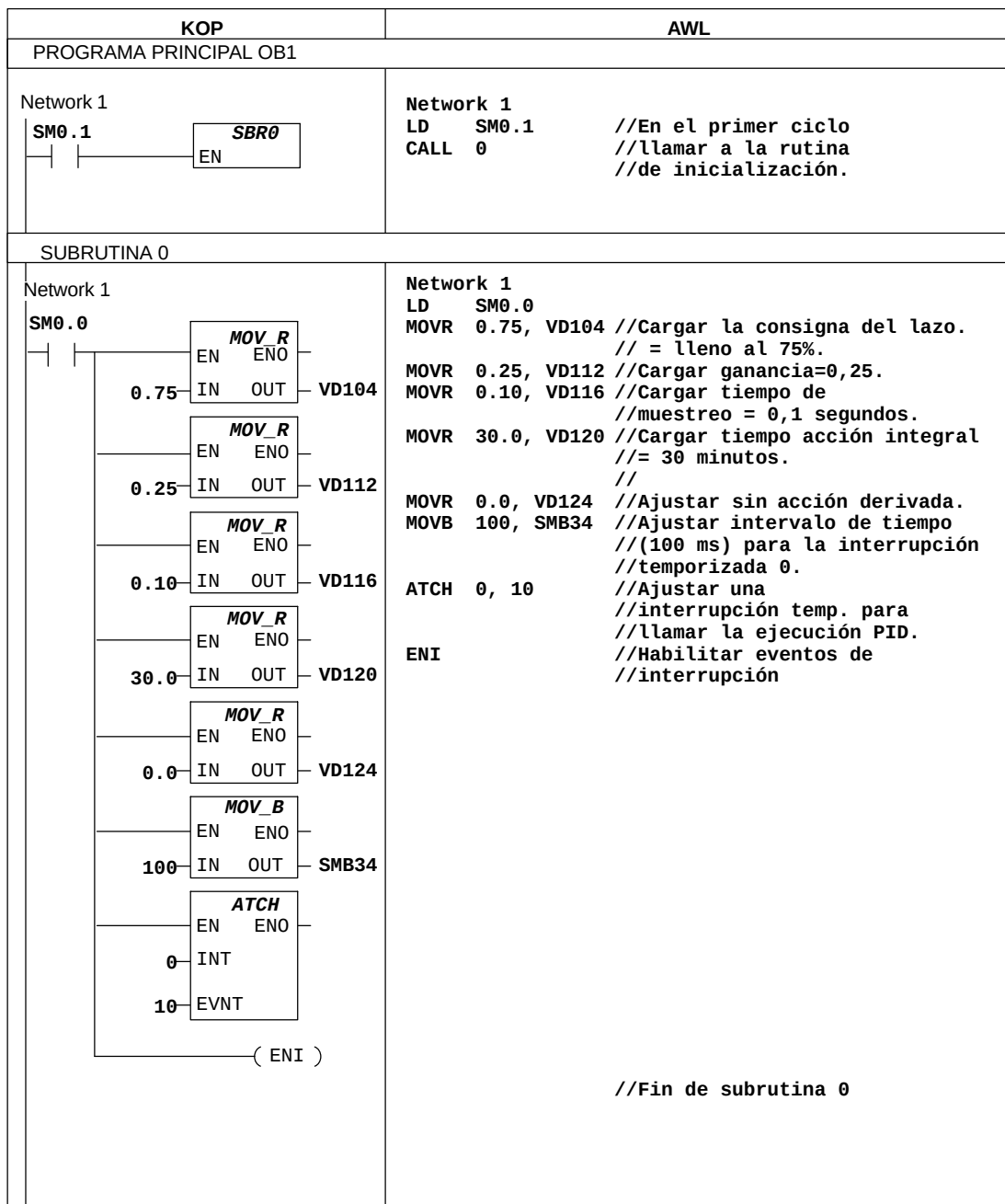


Figura 9-28 Ejemplo de una regulación PID en KOP, AWL y FUP (SIMATIC)

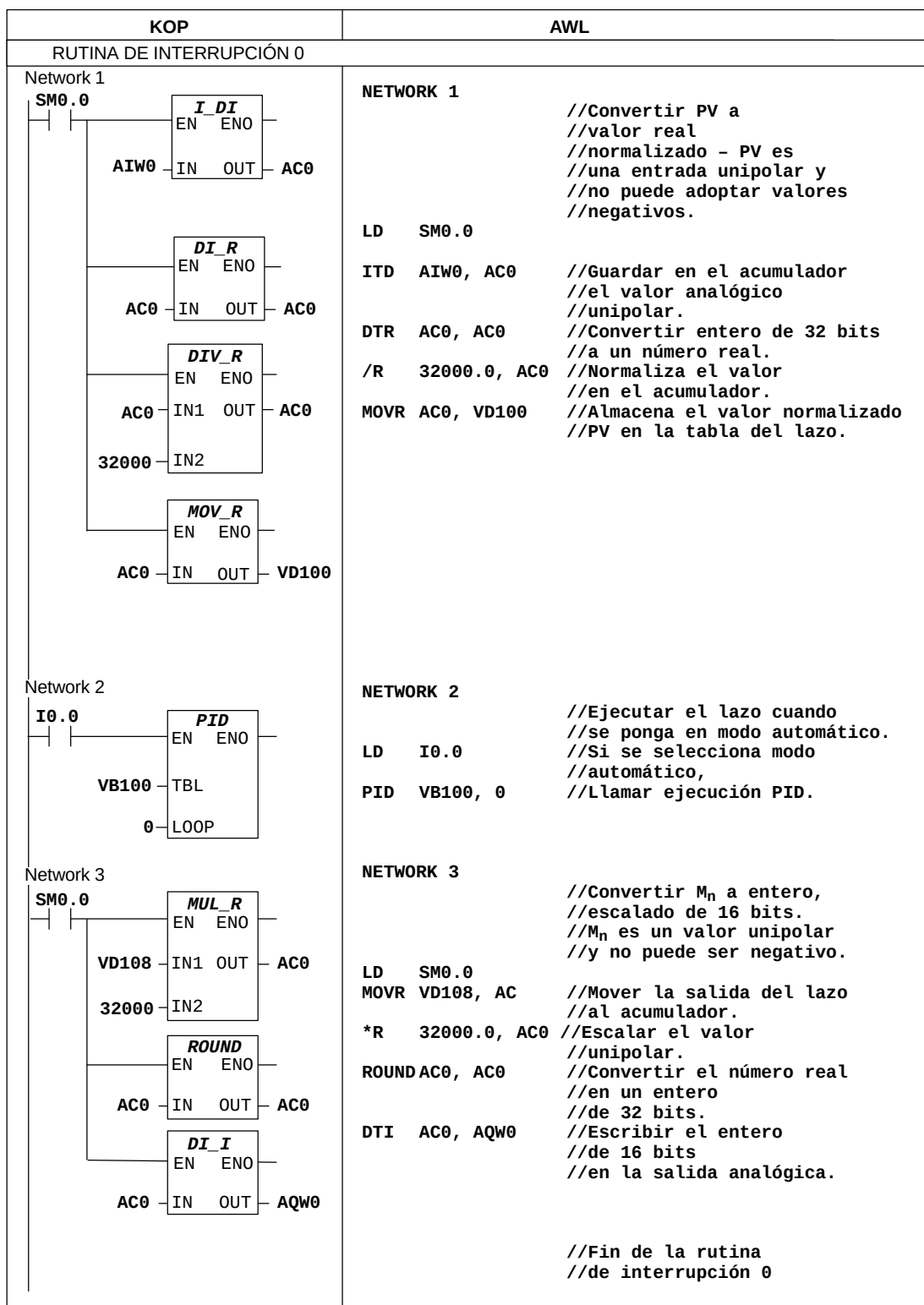


Figura 9-28 Ejemplo de una regulación PID en KOP, AWL y FUP (SIMATIC), (continuación)

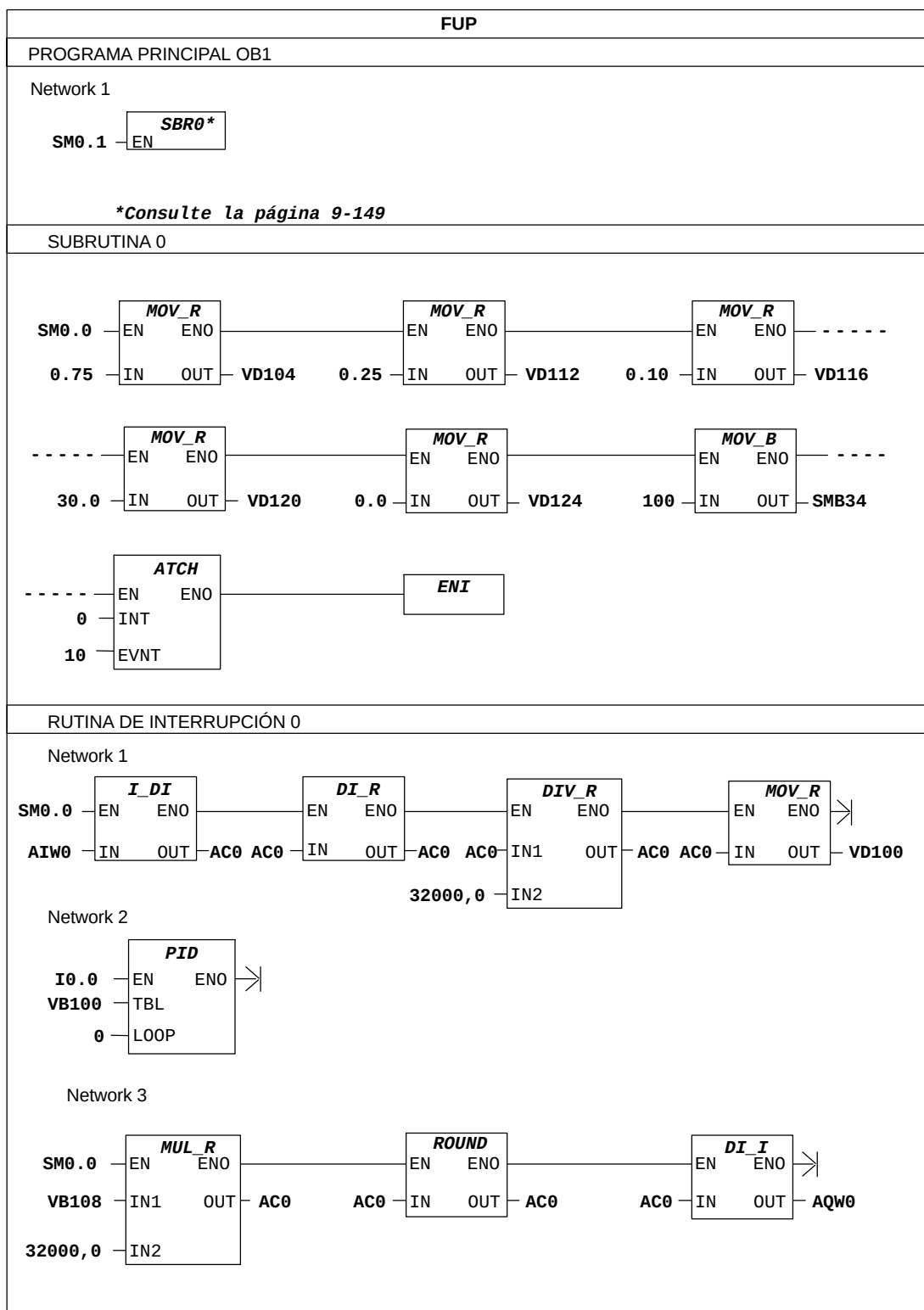
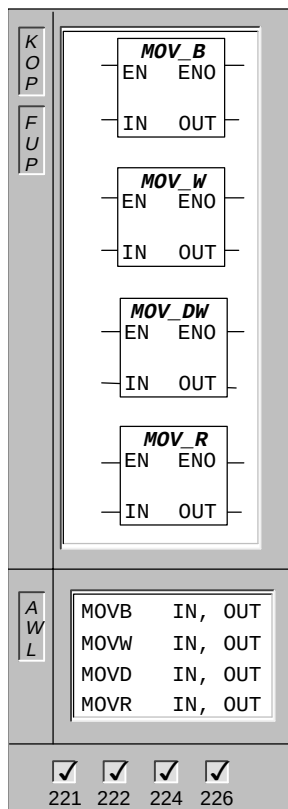


Figura 9-28 Ejemplo de una regulación PID en KOP, AWL y FUP (SIMATIC), (continuación)

9.9 Operaciones de transferencia (SIMATIC)

Transferir byte, Transferir palabra, Transferir palabra doble y Transferir real



La operación **Transferir byte** transfiere el byte de entrada (IN) al byte de salida (OUT). El byte de entrada permanece inalterado.

La operación **Transferir palabra** transfiere la palabra de entrada (IN) a la palabra de salida (OUT). La palabra de entrada permanece inalterada.

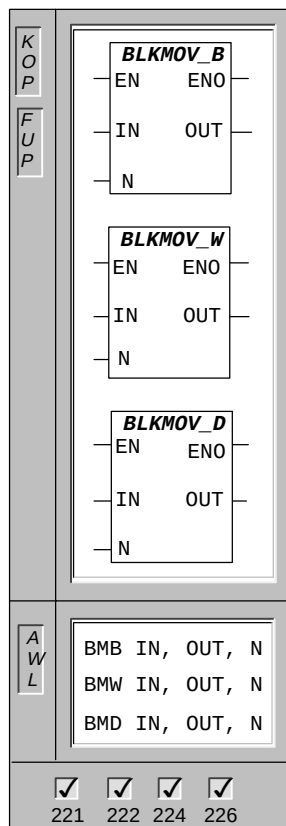
La operación **Transferir palabra doble** transfiere la palabra doble de entrada (IN) a la palabra doble de salida (OUT). La palabra doble de entrada permanece inalterada.

La operación **Transferir real** transfiere un número real de 32 bits de la palabra doble de entrada (IN) a la palabra doble de salida (OUT). La palabra doble de entrada permanece inalterada.

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Transferir...	Entradas/salidas	Operandos	Tipos de datos
Byte	IN	VB, IB, QB, MB, SB, SMB, LB, AC, constante, *VD, *AC, *LD	BYTE
	OUT	VB, IB, QB, MB, SB, SMB, LB, AC, *VD, *AC, *LD	BYTE
Palabra	IN	VW, IW, QW, MW, SW, SMW, LW, T, C, AIW, constante, AC *VD, *AC, *LD	WORD, INT
	OUT	VW, T, C, IW, QW, SW, MW, SMW, LW, AC, AQW, *VD, *AC, *LD	WORD, INT
Palabra doble	IN	VD, ID, QD, MD, SD, SMD, LD, HC, &VB, &IB, &QB, &MB, &SB, &T, &C, AC, constante, *VD, *AC, *LD	DWORD, DINT
	OUT	VD, ID, QD, MD, SD, SMD, LD, AC, *VD, *AC, *LD	DWORD, DINT
Real	IN	VD, ID, QD, MD, SD, SMD, LD, AC, constante, *VD, *AC, *LD	REAL
	OUT	VD, ID, QD, MD, SD, SMD, LD, AC, *VD, *AC, *LD	REAL

Transferir bytes en bloque, Transferir palabras en bloque, Transferir palabras dobles en bloque



La operación **Transferir bytes en bloque** transfiere un número determinado de bytes (N) de la dirección de entrada IN a la dirección de salida OUT. N puede estar comprendido entre 1 y 255.

La operación **Transferir palabras en bloque** transfiere un número determinado de palabras (N) de la dirección de entrada IN a la dirección de salida OUT. N puede estar comprendido entre 1 y 255.

La operación **Transferir palabras dobles en bloque** transfiere un número determinado de palabras dobles (N) de la dirección de entrada IN a la dirección de salida OUT. N puede estar comprendido entre 1 y 255.

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto), 0091 (operando fuera de área)

Transferir ... en bloque	Entradas/salidas	Operandos	Tipos de datos
Byte	IN, OUT	VB, IB, QB, MB, SB, SMB, LB, *VD, *AC, *LD	BYTE
	N	VB, IB, QB, MB, SB, SMB, LB, AC, constante, *VD, *AC, *LD	BYTE
Palabra	IN	VW, IW, QW, MW, SW, SMW, LW, T, C, AIW, *VD, *AC, *LD	WORD
	N	VB, IB, QB, MB, SB, SMB, LB, AC, constante, *VD, *AC, *LD	BYTE
	OUT	VW, IW, QW, MW, SW, SMW, LW, T, C, AQW, *VD, *LD, *AC	WORD
Palabras dobles	IN, OUT	VD, ID, QD, MD, SD, SMD, LD, *VD, *AC, *LD	DWORD
	N	VB, IB, QB, MB, SB, SMB, LB, AC, constante, *VD, *AC, *LD	BYTE

Ejemplo de una operación de transferencia de bloques

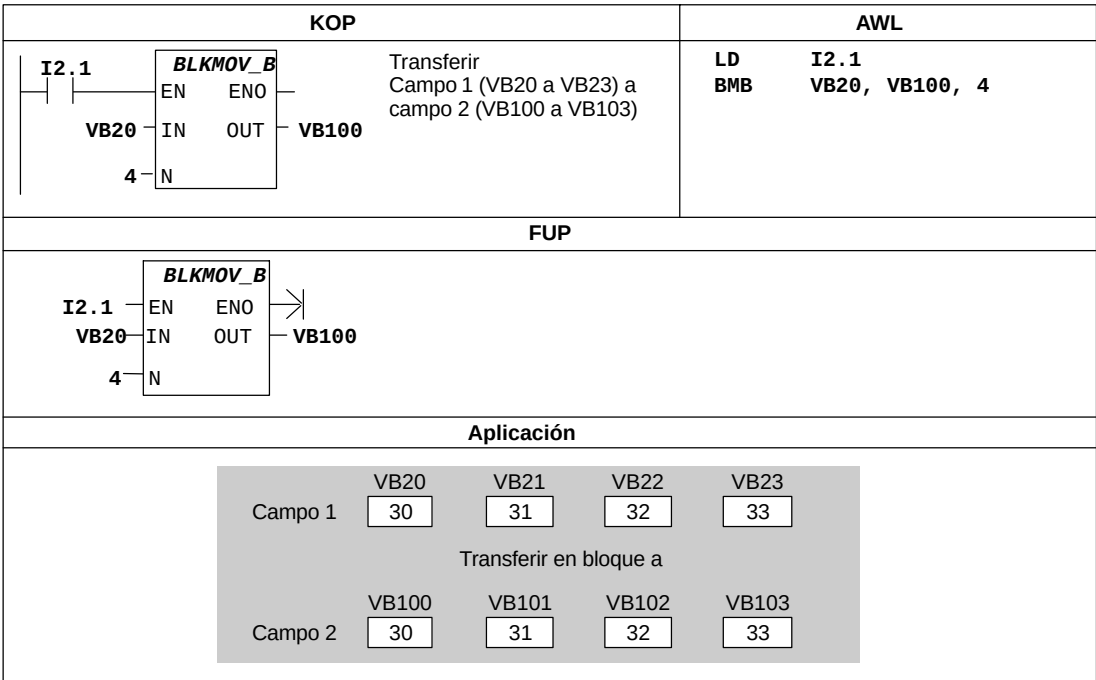
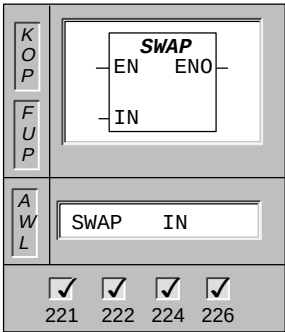


Figura 9-29 Ejemplo de operaciones de transferencia en bloque en KOP, AWL y FUP (SIMATIC)

Invertir bytes de una palabra



La operación **Invertir bytes de una palabra** intercambia el byte más significativo y el byte menos significativo de una palabra (IN).

Condiciones de error que ponen ENO a 0:
SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Entradas/salidas	Operandos	Tipos de datos
IN	VW, IW, QW, MW, SW, SMW, LW, T, C, AC, *VD, *AC, *LD	WORD

Ejemplos de operaciones de transferir e invertir

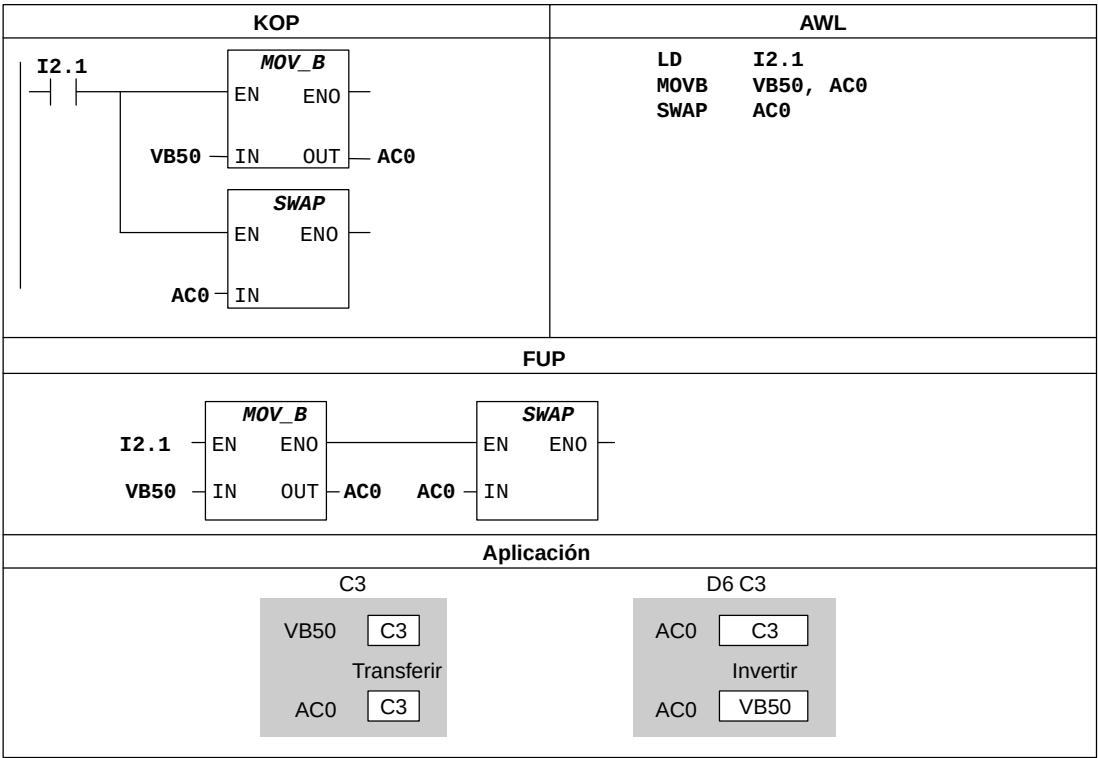
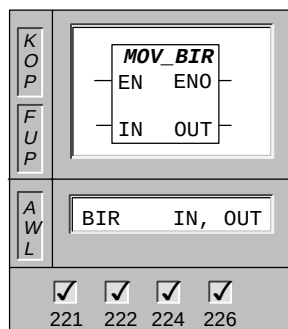


Figura 9-30 Ejemplo de operaciones de transferir e invertir en KOP, AWL y FUP (SIMATIC)

Lectura directa y transferencia de bytes

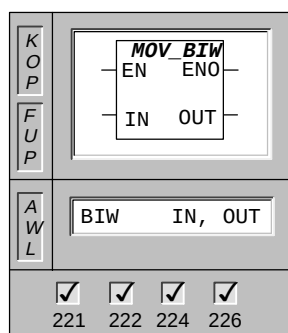


La operación **Lectura directa y transferencia de bytes** lee la entrada física IN y escribe el resultado en OUT.

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Entradas/salidas	Operandos	Tipos de datos
IN	IB	BYTE
OUT	VB, IB, QB, MB, SB, SMB, LB, AC, *VD, *LD, *AC	BYTE

Escritura directa y transferencia de bytes



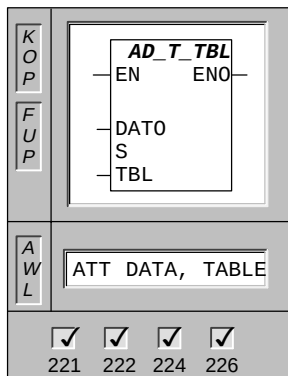
La operación **Escritura directa y transferencia de bytes** lee de la dirección IN y escribe en la salida física OUT.

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Entradas/salidas	Operandos	Tipos de datos
IN	VB, IB, QB, MB, SB, SMB, LB, AC, constante, *VD, *LD, *AC	BYTE
OUT	QB	BYTE

9.10 Operaciones de tabla (SIMATIC)

Registrar valor en tabla



La operación **Registrar valor en tabla** registra valores de palabra (DATA) en la tabla (TBL).

El primer valor de la tabla indica la longitud máxima de la misma (TL). El segundo valor (EC) indica el número de registros que contiene la tabla (v. figura 9-31). Los nuevos datos se añaden al final de la tabla, debajo del último registro. Cada vez que se añade un registro a la tabla, se incrementa el número efectivo de registros. Una tabla puede tener como máximo 100 registros.

Condiciones de error que ponen ENO a 0: SM1.4 (desbordamiento de tabla), SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto), 0091 (operando fuera de área)

Estas operaciones afectan a las siguientes marcas especiales: SM1.4 se activa si se intenta introducir demasiados registros en la tabla.

Entradas/salidas	Operandos	Tipos de datos
DATOS	VW, IW, QW, MW, SW, SMW, LW, T, C, AIW, AC, constante, *VD, *AC, *LD	INT
TBL	VW, IW, QW, MW, SW, SMW, LW, T, C, *VD, *AC, *LD	WORD

Ejemplo de la operación Registrar valor en tabla

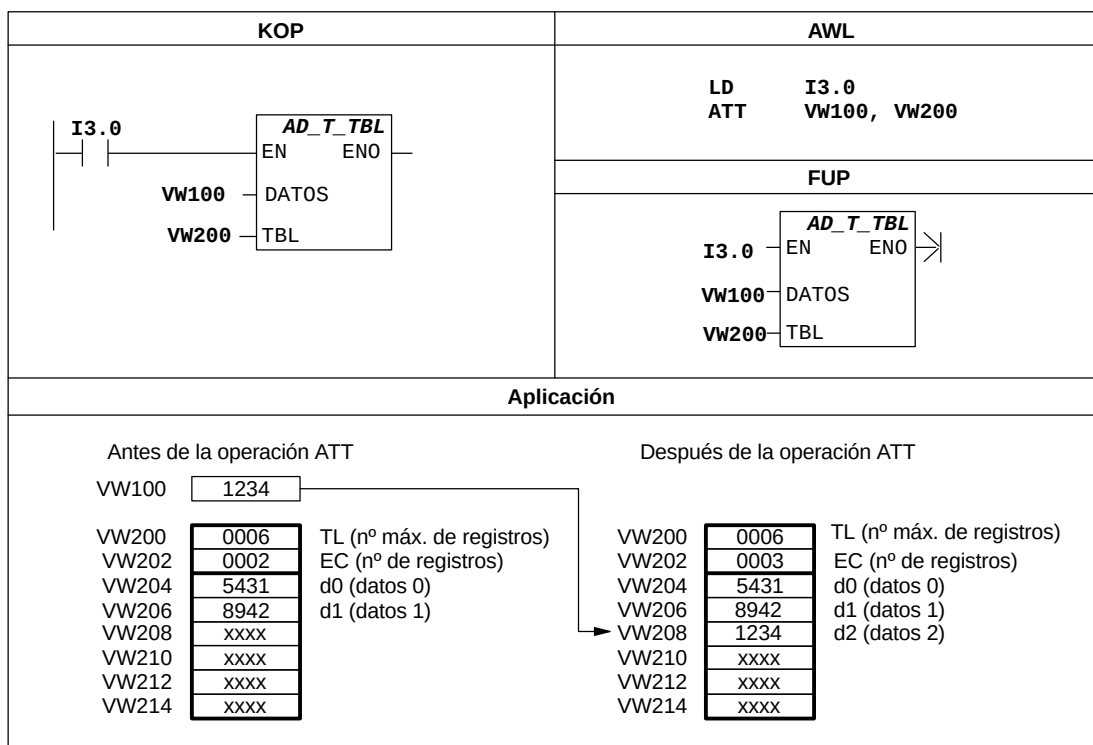
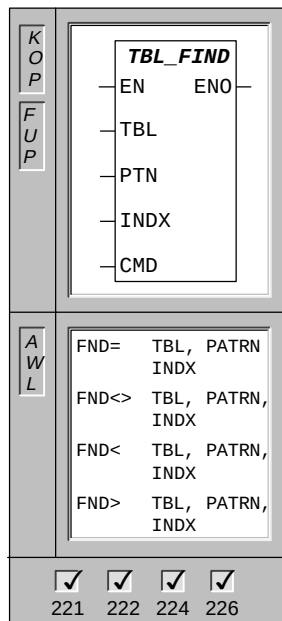


Figura 9-31 Ejemplo de la operación Registrar valor en tabla en KOP, AWL y FUP (SIMATIC)

Buscar valor en tabla



La operación **Buscar valor en tabla** rastrea la tabla (TBL), comenzando con el registro indicado por INDX, y busca el valor (PTN) que corresponda a los criterios de búsqueda definidos por CMD. El parámetro de comando (CMD) indica un valor numérico comprendido entre 1 y 4 que corresponde a la relación =, <>, <, y >, respectivamente.

Si se cumple un criterio, INDX señalará el registro en cuestión. Para buscar el siguiente registro se habrá de incrementar INDX antes de volver a llamar nuevamente a la operación Buscar valor en tabla. Si no se encuentra ningún registro que corresponda al criterio, el valor INDX será igual al número de registros que contiene la tabla.

Una tabla puede tener como máximo 100 registros. Los registros de la tabla (el área donde se desea buscar) están numerados de 0 hasta el valor máximo (99).

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto), 0091 (operando fuera de área)

Entradas/salidas	Operandos	Tipos de datos
SRC	VW, IW, QW, MW, SMW, LW, T, C, *VD, *AC, *LD	WORD
PTN	VW, IW, QW, MW, SW, SMW, AIW, LW, T, C, AC, constante, *VD, *AC, *LD	INT
INDX	VW, IW, QW, MW, SW, SMW, LW, T, C, AC, *VD, *AC, *LD	WORD
CMD	Constante	BYTE

Nota

Si las operaciones de búsqueda se utilizan en tablas creadas con las operaciones ATT, LIFO y FIFO, el valor de contaje corresponderá al número de registros. Contrariamente a las operaciones ATT, LIFO y FIFO, donde una palabra indica el número máximo de registros, las operaciones de búsqueda no requieren esta palabra. Por consiguiente, la dirección del operando SRC de una operación de búsqueda supera en una palabra (dos bytes) al operando TBL correspondiente a la operación ATT, LIFO o FIFO, como muestra la figura 9-32.

Formato de tabla para ATT, LIFO y FIFO			Formato de tabla para TBL_FIND		
VW200	0006	TL (nº máx. de registros)	VW202	0006	EC (nº de registros)
VW202	0006	EC (nº de registros)	VW204	xxxx	d0 (datos 0)
VW204	xxxx	d0 (datos 0)	VW206	xxxx	d1 (datos 1)
VW206	xxxx	d1 (datos 1)	VW208	xxxx	d2 (datos 2)
VW208	xxxx	d2 (datos 2)	VW210	xxxx	d3 (datos 3)
VW210	xxxx	d3 (datos 3)	VW212	xxxx	d4 (datos 4)
VW212	xxxx	d4 (datos 4)	VW214	xxxx	d5 (datos 5)
VW214	xxxx	d5 (datos 5)			

Figura 9-32 Diferencia de los formatos de tabla entre las operaciones de búsqueda y las operaciones ATT, LIFO, FIFO

Ejemplo de la operación Buscar valor en tabla

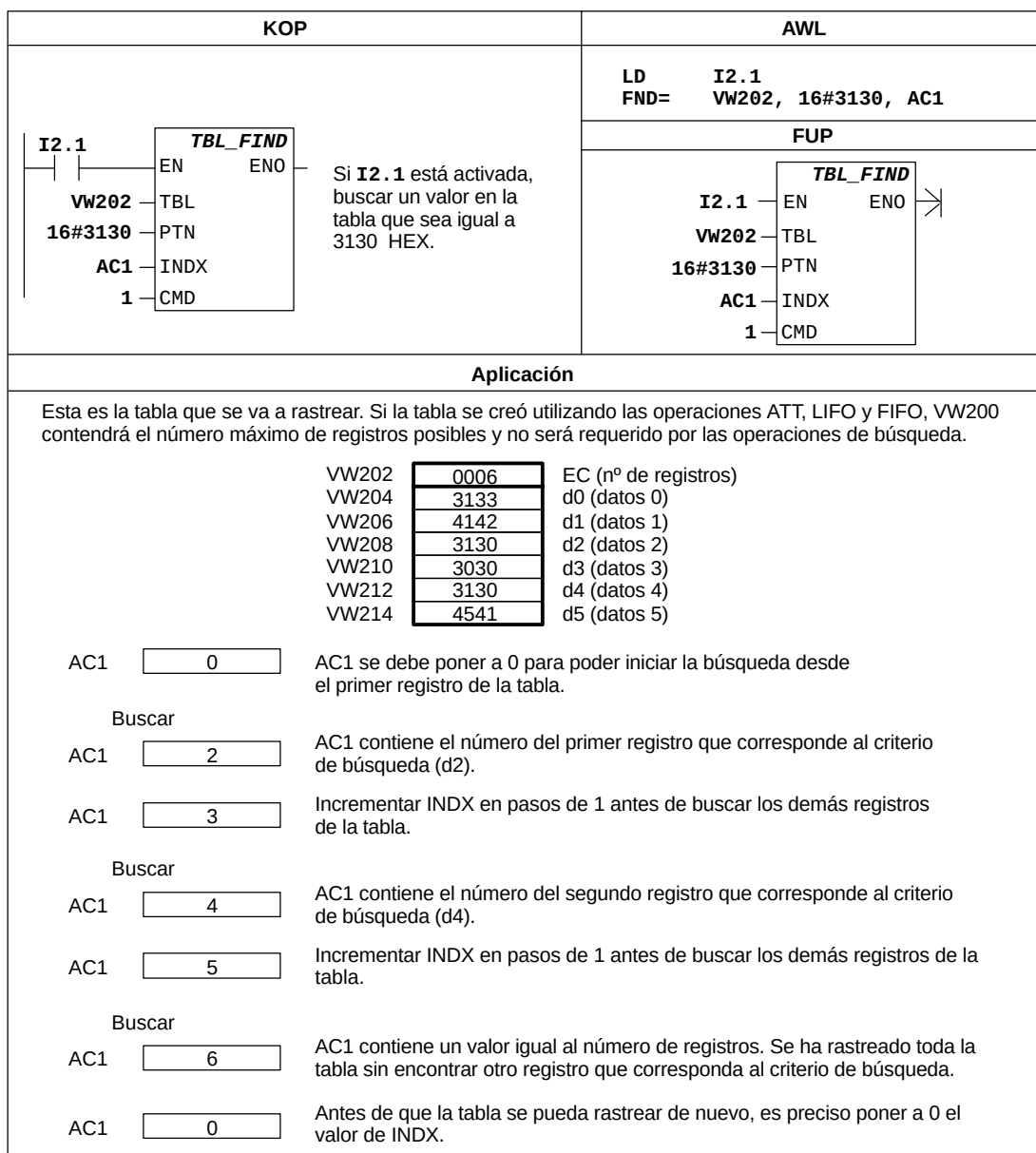
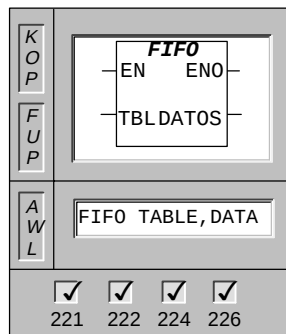


Figura 9-33 Ejemplo de una operación Buscar valor en tabla en KOP, AWL y FUP (SIMATIC)

Borrar primer registro de la tabla



La operación **Borrar primer registro de la tabla** borra el primer registro de la tabla (TBL) y transfiere el valor a la dirección DATA. Todos los demás registros se desplazan una posición hacia arriba. El número de registros (EC) de la tabla decrementa cada vez que se ejecuta esta operación.

Condiciones de error que ponen ENO a 0: SM1.5 (tabla vacía), SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto), 0091 (operando fuera de área)

Estas operaciones afectan a las siguientes marcas especiales: SM1.5 se activa si se intenta borrar un registro de una tabla vacía.

Entradas/salidas	Operandos	Tipos de datos
TABLE	VW, IW, QW, MW, SW, SMW, LW, T, C, *VD, *AC, *LD	INT
DATOS	VW, IW, QW, MW, SW, SMW, LW, AC, AQW, T, C, *VD, *AC, *LD	WORD

Ejemplo de la operación Borrar primer registro de la tabla

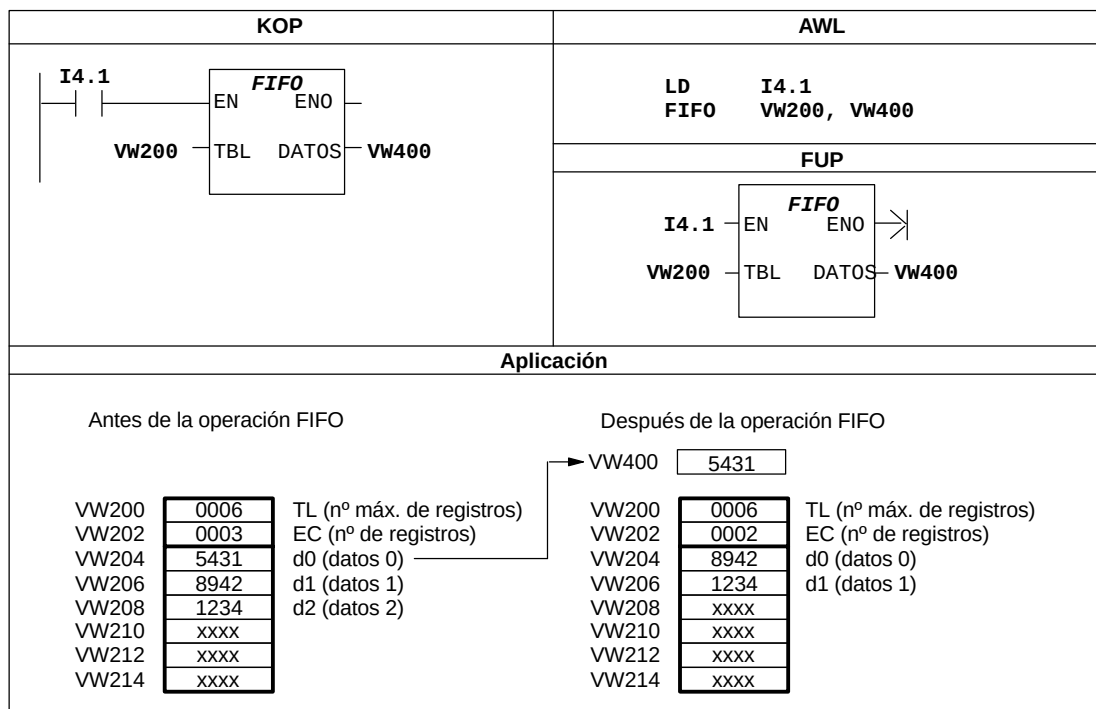
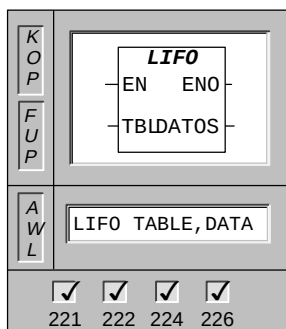


Figura 9-34 Ejemplo de la operación Borrar primer registro de la tabla en KOP, AWL y FUP (SIMATIC)

Borrar último registro de la tabla



La operación **Borrar último registro de la tabla** borra el último registro de la tabla (TBL) y transfiere el valor a la dirección indicada por DATA. El número de registros (EC) de la tabla decrementa cada vez que se ejecuta esta operación.

Condiciones de error que ponen ENO a 0: SM1.5 (tabla vacía), SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto), 0091 (operando fuera de área)

Estas operaciones afectan a las siguientes marcas especiales: SM1.5 se activa si se intenta borrar un registro de una tabla vacía.

Entradas/salidas	Operandos	Tipos de datos
TABLE	VW, IW, QW, MW, SW, SMW, LW, T, C, *VD, *AC, *LD	INT
DATOS	VW, IW, QW, MW, SW, SMW, LW, AQW, T, C, AC, *VD, *AC, *LD	WORD

Ejemplo de la operación Borrar último registro de la tabla

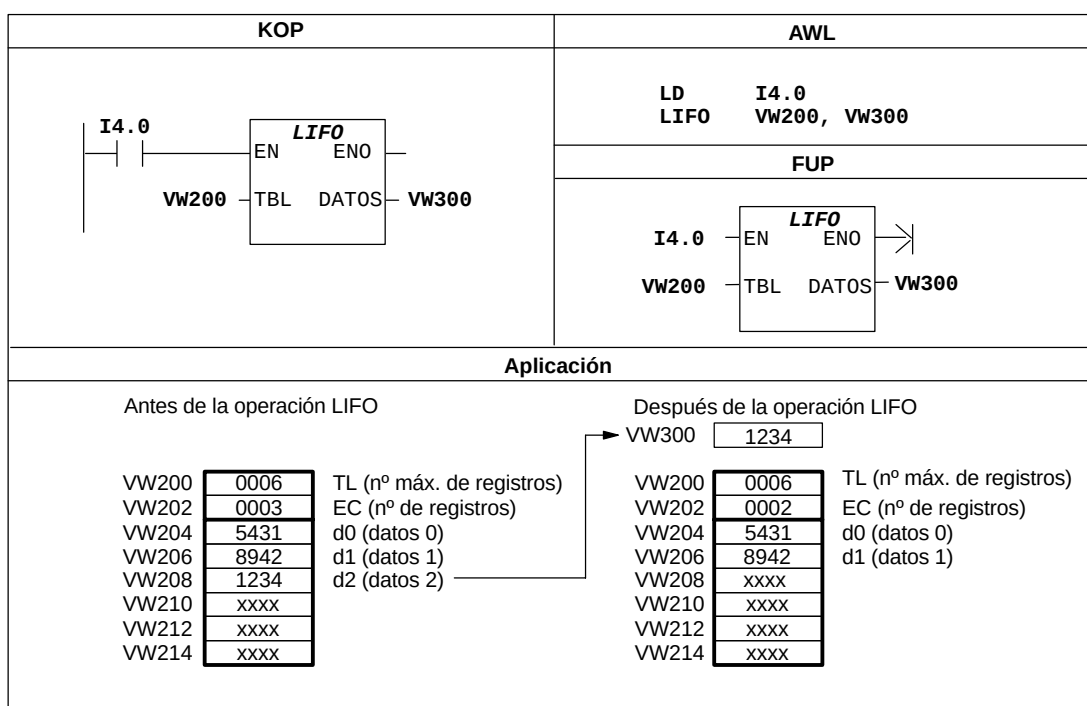
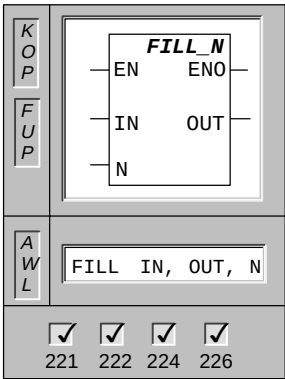


Figura 9-35 Ejemplo de la operación Borrar último registro de la tabla en KOP, AWL y FUP (SIMATIC)

Inicializar memoria



La operación **Inicializar memoria** rellena el número de palabras N de la dirección IN en la dirección OUT. N puede estar comprendido entre 1 y 255.

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto), 0091 (operando fuera de área)

Entradas/salidas	Operandos	Tipos de datos
IN	VW, IW, QW, MW, SW, SMW, LW, AIW, T, C, AC, constante, *VD, *AC, *LD	WORD
N	VB, IB, QB, MB, SB, SMB, LB, AC, constante, *VD, *AC, *LD	BYTE
OUT	VW, IW, QW, MW, SW, SMW, LW, T, C, AQW, *VD, *AC, *LD	WORD

Ejemplo de la operación Inicializar memoria

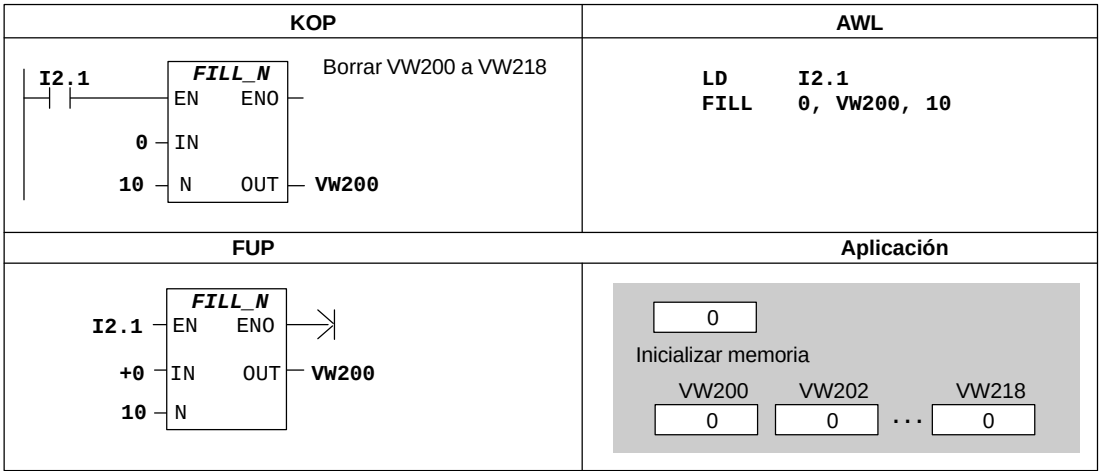
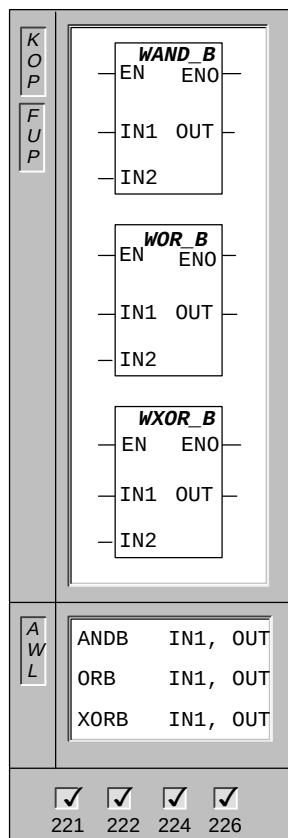


Figura 9-36 Ejemplo de la operación Inicializar memoria en KOP, AWL y FUP (SIMATIC)

9.11 Operaciones lógicas (SIMATIC)

Combinación Y con bytes, Combinación O con bytes y Combinación O-exclusiva con bytes



La operación **Combinación Y con bytes** combina los bits correspondientes de los dos bytes de entrada mediante Y, y carga el resultado (OUT) en un byte.

La operación **Combinación O con bytes** combina los bits correspondientes de los dos bytes de entrada mediante O, y carga el resultado (OUT) en un byte.

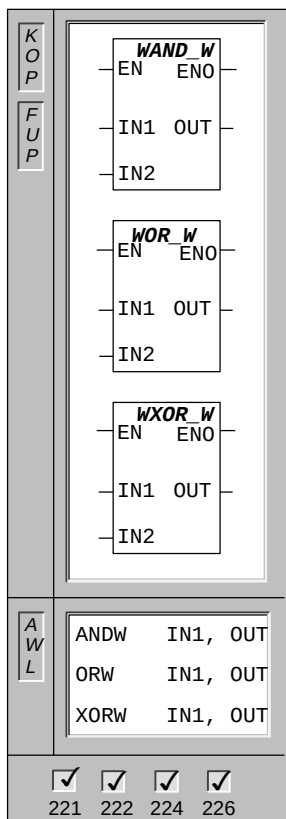
La operación **Combinación O-exclusiva con bytes** combina los bits correspondientes de los dos bytes de entrada mediante O-exclusiva, y carga el resultado (OUT) en un byte.

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales: SM1.0 (cero)

Entradas/salidas	Operandos	Tipos de datos
IN1, IN2	VB, IB, QB, MB, SB, SMB, LB, AC, constante, *VD, *AC, *LD	BYTE
OUT	VB, IB, QB, MB, SB, SMB, LB, AC, *VD, *AC, *LD	BYTE

Combinación Y con palabras, Combinación O con palabras y Combinación O-exclusiva con palabras



La operación **Combinación Y con palabras** combina los bits correspondientes de las dos palabras de entrada mediante Y, y carga el resultado (OUT) en una palabra.

La operación **Combinación O con palabras** combina los bits correspondientes de las dos palabras de entrada mediante O, y carga el resultado (OUT) en una palabra.

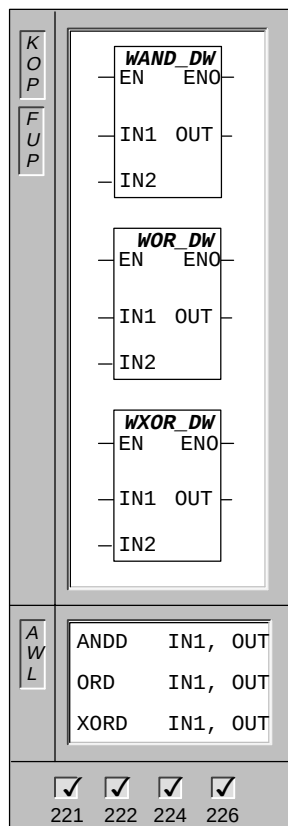
La operación **Combinación O-exclusiva con palabras** combina los bits correspondientes de las dos palabras de entrada mediante O-exclusiva, y carga el resultado (OUT) en una palabra.

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales: SM1.0 (cero)

Entradas/salidas	Operandos	Tipos de datos
IN1, IN2	VW, IW, QW, MW, SW, SMW, LW, T, C, AIW, AC, constante, *VD, *AC, *LD	WORD
OUT	VW, IW, QW, MW, SW, SMW, LW, T, C, AC, *VD, *AC, *LD	WORD

Combinación Y con palabras dobles, Combinación O con palabras dobles y Combinación O-exclusiva con palabras dobles



La operación **Combinación Y con palabras dobles** combina los bits correspondientes de las dos palabras dobles de entrada mediante Y, y carga el resultado (OUT) en una palabra doble.

La operación **Combinación O con palabras dobles** combina los bits correspondientes de las dos palabras dobles de entrada mediante O, y carga el resultado (OUT) en una palabra doble.

La operación **Combinación O-exclusiva con palabras dobles** combina los bits correspondientes de las dos palabras dobles de entrada mediante O-exclusiva, y carga el resultado (OUT) en una palabra doble.

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales: SM1.0 (cero)

Entradas/salidas	Operandos	Tipos de datos
IN1, IN2	VD, ID, QD, MD, SD, SMD, AC, LD, HC, constante, *VD, *AC, SD, *LD	DWORD
OUT	VD, ID, QD, MD, SMD, LD, AC, *VD, *AC, SD, *LD	DWORD

Ejemplos de las operaciones de combinación con Y, O y O-exclusiva

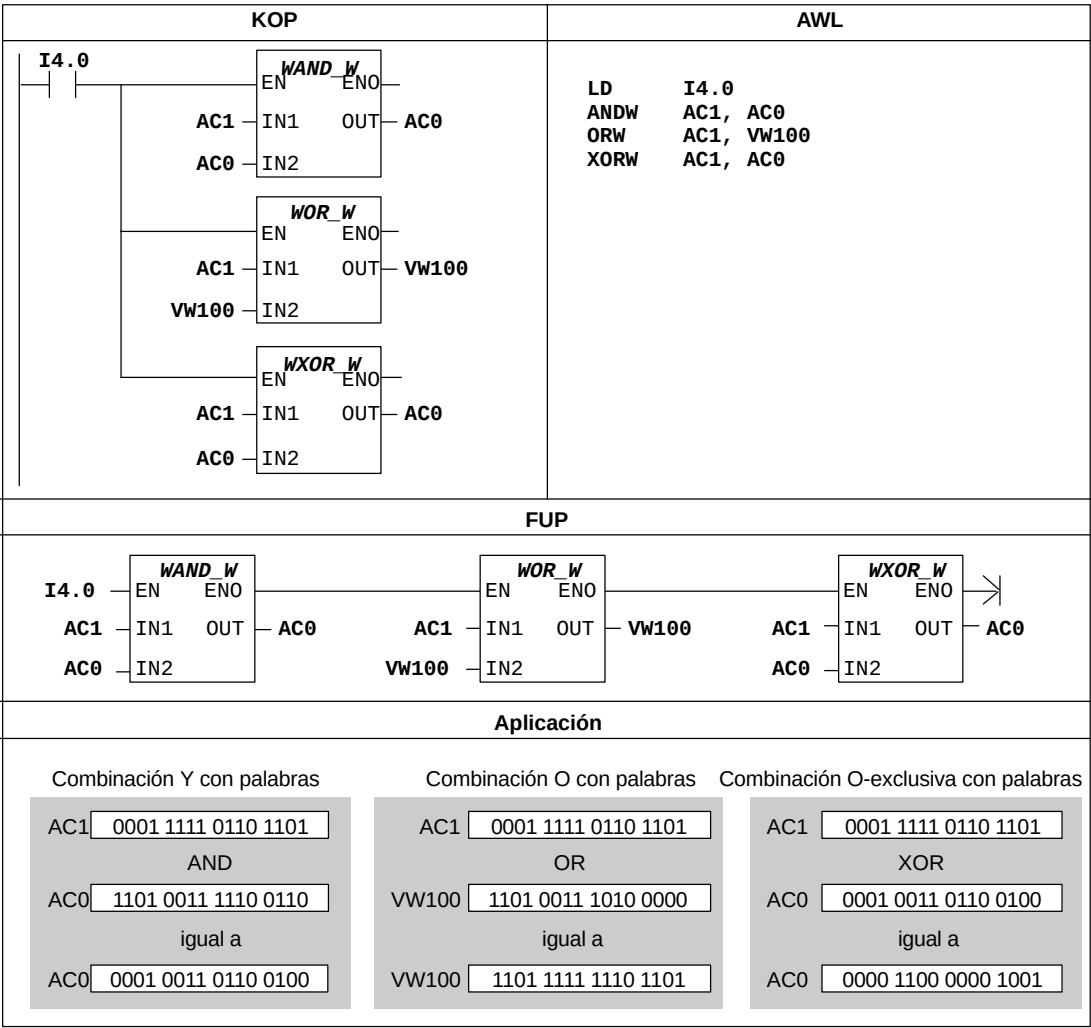
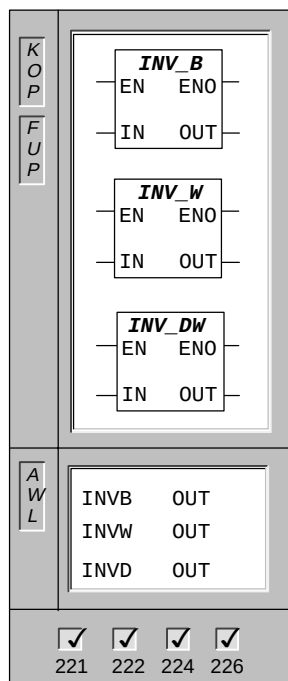


Figura 9-37 Ejemplos de las operaciones de combinación con Y, O y O-exclusiva en KOP, AWL y FUP (SIMATIC)

Invertir byte, Invertir palabra, Invertir palabra doble



La operación **Invertir byte** forma el complemento a 1 del valor del byte de entrada IN y carga el resultado en el valor de byte OUT.

La operación **Invertir palabra** forma el complemento a 1 del valor de la palabra de entrada IN y carga el resultado en el valor de palabra OUT.

La operación **Invertir palabra doble** forma el complemento a 1 del valor de la palabra doble de entrada IN y carga el resultado en el valor de palabra doble OUT.

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales: SM1.0 (cero)

Invertir...	Entradas/salidas	Operandos	Tipos de datos
Byte	IN	VB, IB, QB, MB, SB, SMB, LB, AC, constante, *VD, *AC, *LD	BYTE
	OUT	VB, IB, QB, MB, SB, SMB, LB, AC, *VD, *AC, *LD	BYTE
Palabra	IN	VW, IW, QW, MW, SW, SMW, T, C, AIW, LW, AC, constante, *VD, *AC, *LD	WORD
	OUT	VW, IW, QW, MW, SW, SMW, T, C, LW, AC, *VD, *AC, *LD	WORD
Palabra doble	IN	VD, ID, QD, MD, SD, SMD, LD, HC, AC, constante, *VD, *AC, *LD	DWORD
	OUT	VD, ID, QD, MD, SD, SMD, LD, AC, *VD, *AC, *LD	DWORD

Ejemplo de la operación Invertir

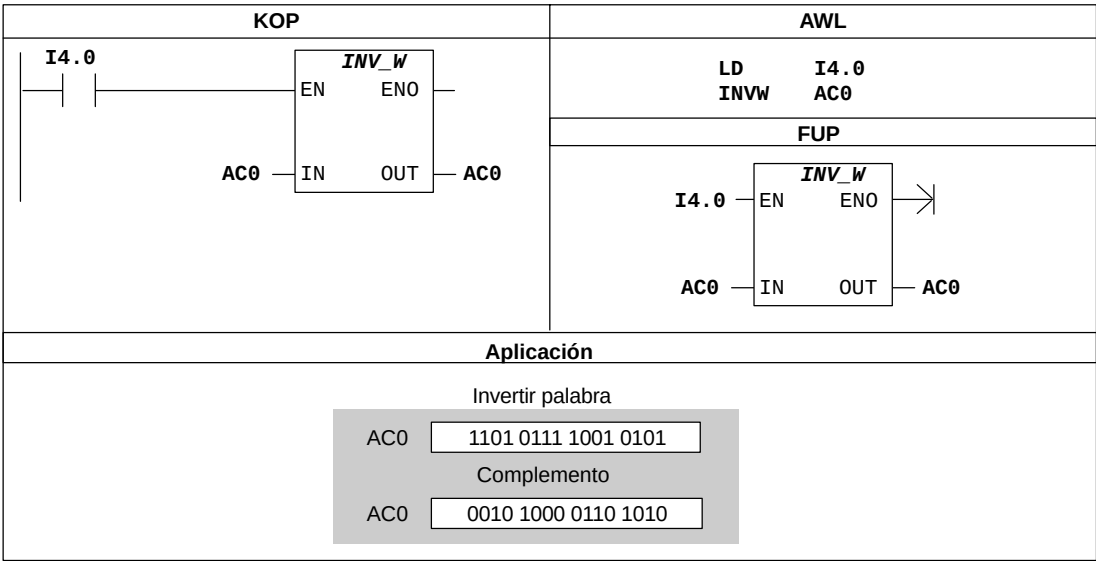
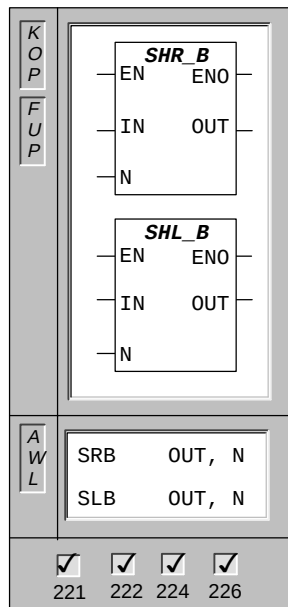


Figura 9-38 Ejemplo de la operación Invertir en KOP, AWL y FUP (SIMATIC)

9.12 Operaciones de desplazamiento y rotación (SIMATIC)

Desplazar byte a la derecha, Desplazar byte a la izquierda



Las operaciones **Desplazar byte a la derecha** y **Desplazar byte a la izquierda** desplazan el valor del byte de entrada (IN) a la derecha y a la izquierda respectivamente, tantas posiciones como indique el valor de desplazamiento (N), y cargan el resultado en el byte de salida (OUT).

Las **operaciones de desplazamiento** se rellenan con ceros cada vez que se desplaza un bit. Si el valor de desplazamiento (N) es mayor o igual a 8, el valor se desplazará como máximo 8 veces.

Si el valor de desplazamiento es mayor que 0, la marca de desbordamiento (SM1.1) adoptará el valor del último bit desplazado hacia afuera. La marca cero (SM1.0) se activará si el resultado de la operación de desplazamiento es cero.

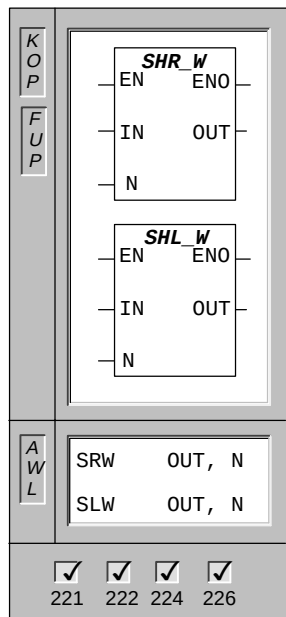
Las operaciones de desplazamiento de bytes no llevan signo.

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales: SM1.0 (cero); SM1.1 (desbordamiento)

Entradas/salidas	Operandos	Tipos de datos
IN	VB, IB, QB, MB, SB, SMB, LB, AC, constante, *VD, *AC, *LD	BYTE
OUT	VB, IB, QB, MB, SB, SMB, LB, AC, *VD, *AC, *LD	BYTE
N	VB, IB, QB, MB, SB, SMB, LB, AC, constante, *VD, *AC, *LD	BYTE

Desplazar palabra a la derecha, Desplazar palabra a la izquierda



Las operaciones **Desplazar palabra a la derecha** y **Desplazar palabra a la izquierda** desplazan el valor de la palabra de entrada (IN) a la derecha y a la izquierda respectivamente, tantas posiciones como indique el valor de desplazamiento (N) y cargan el resultado en la palabra de salida (OUT).

Las **operaciones de desplazamiento** se rellenan con ceros cada vez que se desplaza un bit. Si el valor de desplazamiento (N) es mayor o igual a 16, el valor se desplazará como máximo 16 veces. Si el valor de desplazamiento es mayor que 0, la marca de desbordamiento (SM1.1) adoptará el valor del último bit desplazado hacia afuera. La marca cero (SM1.0) se activará si el resultado de la operación de desplazamiento es cero.

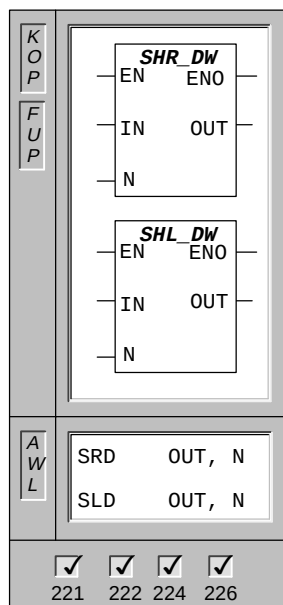
El bit de signo se desplaza cuando se utilizan tipos de datos con signo.

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales: SM1.0 (cero); SM1.1 (desbordamiento)

Entradas/salidas	Operandos	Tipos de datos
IN	VW, IW, QW, MW, SW, SMW, LW, T, C, AIW, AC, constante, *VD, *AC, *LD	WORD
N	VB, IB, QB, MB, SB, SMB, LB, AC, constante, *VD, *AC, *LD	BYTE
OUT	VW, IW, QW, MW, SW, SMW, LW, T, C, AC, *VD, *AC, *LD	WORD

Desplazar palabra doble a la derecha, Desplazar palabra doble a la izquierda



Las operaciones **Desplazar palabra doble a la derecha** y **Desplazar palabra doble a la izquierda** desplazan el valor de la palabra doble de entrada (IN) a la derecha y a la izquierda respectivamente, tantas posiciones como indique el valor de desplazamiento (N) y cargan el resultado en la palabra doble de salida (OUT).

Las **operaciones de desplazamiento** se rellenan con ceros cada vez que se desplaza un bit. Si el valor de desplazamiento (N) es mayor o igual a 32, el valor se desplazará como máximo 32 veces. Si el valor de desplazamiento es mayor que 0, la marca de desbordamiento (SM1.1) adoptará el valor del último bit desplazado hacia afuera. La marca cero (SM1.0) se activará si el resultado de la operación de desplazamiento es cero.

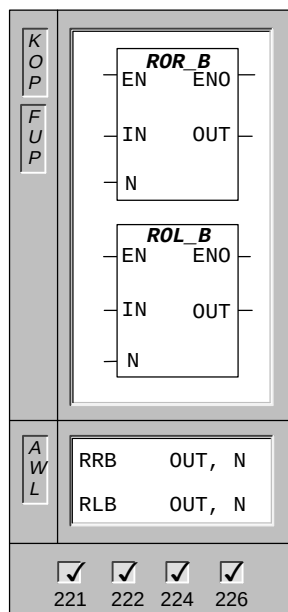
El bit de signo se desplaza cuando se utilizan tipos de datos con signo.

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales: SM1.0 (cero); SM1.1 (desbordamiento)

Entradas/salidas	Operandos	Tipos de datos
IN	VD, ID, QD, MD, SD, SMD, LD, AC, HC, constante, *VD, *AC, *LD	DWORD
N	VB, IB, QB, MB, SB, SMB, LB, AC, constante, *VD, *AC, *LD	BYTE
OUT	VD, ID, QD, MD, SD, SMD, LD, AC, *VD, *AC, *LD	DWORD

Rotar byte a la derecha, Rotar byte a la izquierda



Las operaciones **Rotar byte a la derecha** y **Rotar byte a la izquierda** rotan el valor del byte de entrada (IN) a la derecha y a la izquierda respectivamente, tantas posiciones como indique el valor de desplazamiento (N) y cargan el resultado en el byte de salida (OUT). La rotación es circular.

Si el valor de desplazamiento (N) es mayor o igual a 8, antes de la operación de rotación se ejecutará una operación módulo 8 en el valor de desplazamiento (N). De ello resulta un valor de rotación de 0 a 7. Si el valor de desplazamiento es igual a 0, no se rotará el valor. Si se ejecuta la rotación, el valor del último bit rotado se copiará en la marca de desbordamiento (SM1.1).

Si el valor de desplazamiento no es un entero múltiplo de 8, el último bit rotado se copiará en la marca de desbordamiento (SM1.1). La marca cero (SM1.0) se activará si el valor a rotar es igual a cero.

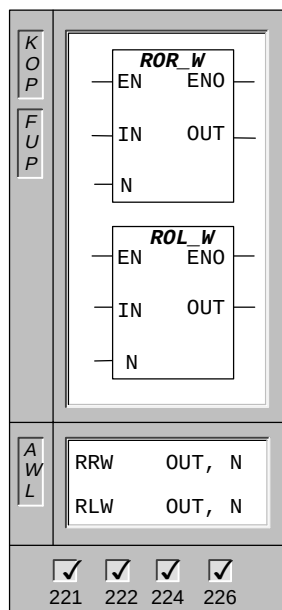
Las operaciones de rotación de bytes no llevan signo.

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales: SM1.0 (cero); SM1.1 (desbordamiento)

Entradas/salidas	Operandos	Tipos de datos
IN	VB, IB, QB, MB, SMB, SB, LB, AC, *VD, *AC, *LD	BYTE
N	VB, IB, QB, MB, SMB, SB, LB, AC, constante, *VD, *AC, *LD	BYTE
OUT	VB, IB, QB, MB, SMB, SB, LB, AC, *VD, *AC, *LD	BYTE

Rotar palabra a la derecha, Rotar palabra a la izquierda



Las operaciones **Rotar palabra a la derecha** y **Rotar palabra a la izquierda** rotan el valor de la palabra de entrada (IN) a la derecha y a la izquierda respectivamente, tantas posiciones como indique el valor de desplazamiento (N) y cargan el resultado en la palabra de salida (OUT). La rotación es circular.

Si el valor de desplazamiento (N) es mayor o igual a 16, antes de la operación de rotación se ejecutará una operación módulo 16 en el valor de desplazamiento (N). De ello resulta un valor de rotación de 0 a 15. Si el valor de desplazamiento es igual a 0, no se rotará el valor. Si se ejecuta la rotación, el valor del último bit rotado se copiará en la marca de desbordamiento (SM1.1).

Si el valor de desplazamiento no es un entero múltiplo de 16, el último bit rotado se copiará en la marca de desbordamiento (SM1.1). La marca cero (SM1.0) se activará si el valor a rotar es igual a cero.

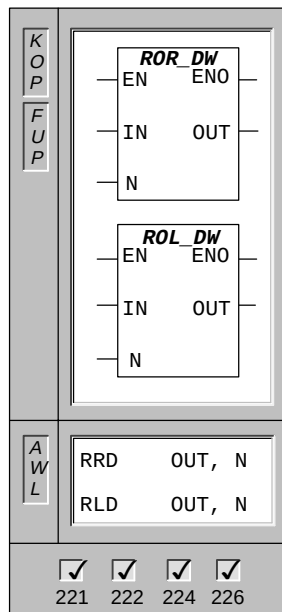
El bit de signo se desplaza cuando se utilizan tipos de datos con signo.

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales: SM1.0 (cero); SM1.1 (desbordamiento)

Entradas/salidas	Operandos	Tipos de datos
IN	VW, T, C, IW, MW, SW, SMW, AC, QW, LW, AIW, constante *VD, *AC, *LD	WORD
N	VB, IB, QB, MB, SMB, LB, AC, constante, *VD, *AC, SB, *LD	BYTE
OUT	VW, T, C, IW, QW, MW, SW, SMW, LW, AC, *VD, *AC, *LD	WORD

Rotar palabra doble a la derecha, Rotar palabra doble a la izquierda



Las operaciones **Rotar palabra doble a la derecha** y **Rotar palabra doble a la izquierda** rotan el valor de la palabra doble de entrada (IN) a la derecha y a la izquierda respectivamente, tantas posiciones como indique el valor de desplazamiento (N) y cargan el resultado en la palabra doble de salida (OUT). La rotación es circular.

Si el valor de desplazamiento (N) es mayor o igual a 32, antes de la operación de rotación se ejecutará una operación módulo 32 en el valor de desplazamiento (N). De ello resulta un valor de rotación de 0 a 31. Si el valor de desplazamiento es igual a 0, no se rotará el valor. Si se ejecuta la rotación, el valor del último bit rotado se copiará en la marca de desbordamiento (SM1.1).

Si el valor de desplazamiento no es un entero múltiplo de 32, el último bit rotado se copiará en la marca de desbordamiento (SM1.1). La marca cero (SM1.0) se activará si el valor a rotar es igual a cero.

El bit de signo se desplaza cuando se utilizan tipos de datos con signo.

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales: SM1.0 (cero); SM1.1 (desbordamiento)

Entradas/salidas	Operandos	Tipos de datos
IN	VD, ID, QD, MD, SMD, LD, AC, HC, constante, *VD, *AC, SD, *LD	DWORD
N	VB, IB, QB, MB, SMB, LB, AC, constante, *VD, *AC, SB, *LD	BYTE
OUT	VD, ID, QD, MD, SMD, LD, AC, *VD, *AC, SD, *LD	DWORD

Ejemplos de operaciones de rotación y desplazamiento

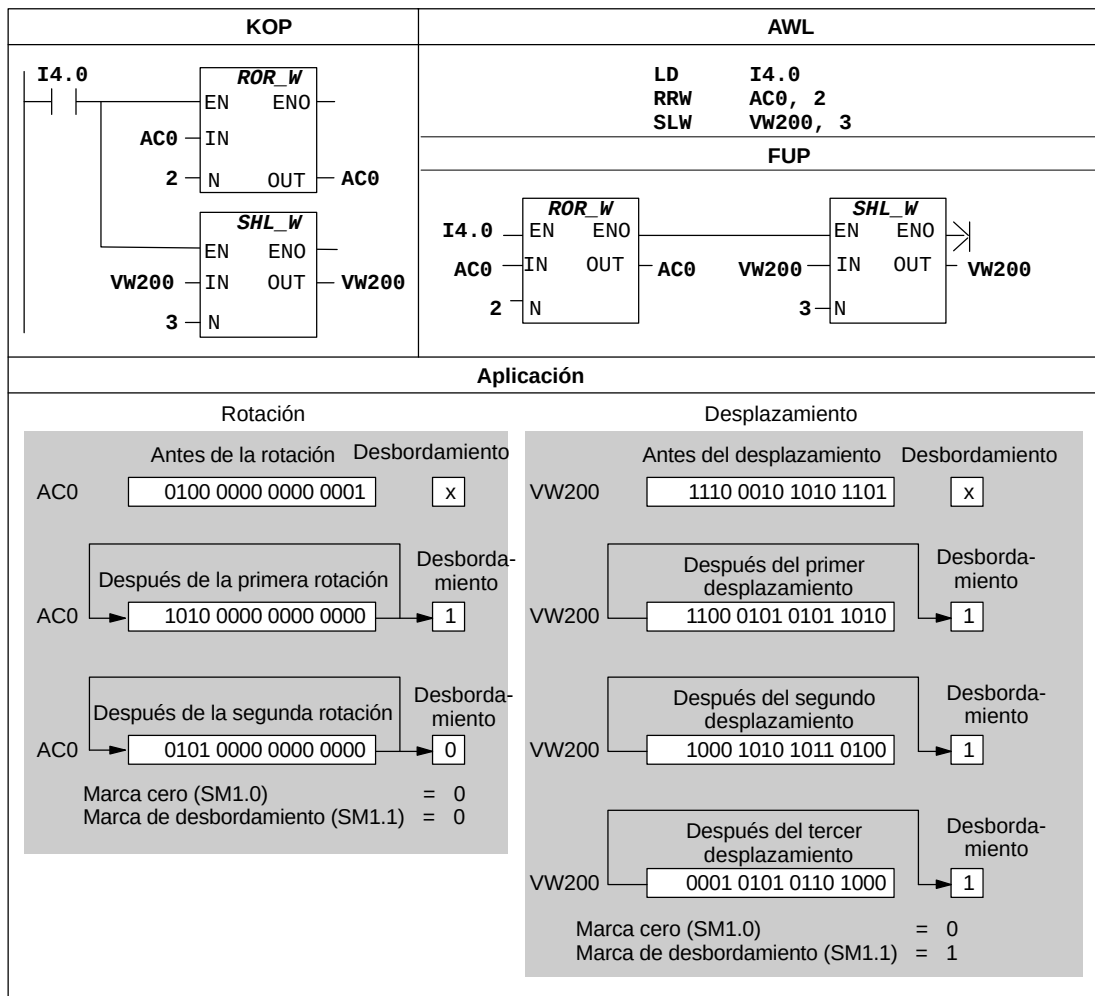
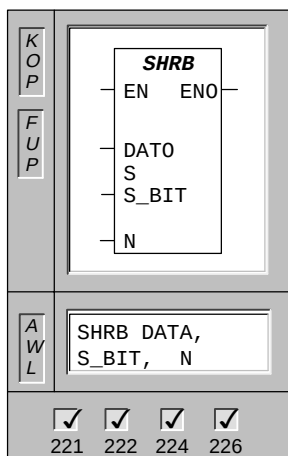


Figura 9-39 Ejemplo de operaciones de desplazamiento y rotación en KOP, AWL y FUP (SIMATIC)

Registro de desplazamiento



La operación **Registro de desplazamiento** (SHRB) desplaza el valor de DATA al registro de desplazamiento. S_BIT indica el bit menos significativo de este registro. N indica la longitud del registro y el sentido de desplazamiento (valor positivo = N, valor negativo = -N).

Los bits desplazados por la operación Registro de desplazamiento se depositarán en la marca de desbordamiento (SM1.1).

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto), 0091 (operando fuera de área), 0092 (error en campo de conteo)

Estas operaciones afectan a las siguientes marcas especiales: SM1.1 (desbordamiento)

Entradas/salidas	Operandos	Tipos de datos
DATA, S_BIT	I, Q, M, SM, T, C, V, S, L	BOOL
N	VB, IB, QB, MB, SMB, LB, AC, constante, *VD, *AC, SB, *LD	BYTE

Descripción de la operación Registro de desplazamiento

La operación Registro de desplazamiento permite secuenciar y controlar fácilmente el flujo de productos o de datos. Esta operación se debe utilizar para desplazar todo el registro un bit en cada ciclo. El registro de desplazamiento está definido por el bit menos significativo (S_BIT) y por el número de bits indicados por la longitud (N). La figura 9-41 muestra un ejemplo de la operación Registro de desplazamiento.

La dirección del bit más significativo del registro de desplazamiento (MSB.b) se calcula con la siguiente ecuación:

$$\text{MSB.b} = [(\text{byte de S\_BIT}) + ([N] - 1 + (\text{bit de S\_BIT})) / 8] \cdot [\text{resto de la división por 8}]$$

Se debe restar 1 bit, porque S_BIT es uno de los bits del registro de desplazamiento.

Por ejemplo, si S_BIT es V33.4 y N es 14, el bit MSB.b será V35.1 ó:

$$\begin{aligned} \text{MSB.b} &= \text{V33} + ([14] - 1 + 4) / 8 \\ &= \text{V33} + 17 / 8 \\ &= \text{V33} + 2 \text{ con el resto de } 1 \\ &= \text{V35.1} \end{aligned}$$

Si el valor de desplazamiento es negativo, es decir, si la longitud (N) indicada es negativa, los datos de entrada se desplazarán desde el bit menos significativo (S_BIT) al bit más significativo del registro de desplazamiento.

Si el valor de desplazamiento es positivo, es decir, si la longitud (N) indicada es positiva, los datos de entrada (DATA) se desplazarán desde el bit más significativo al bit menos significativo (indicado por S_BIT) del registro de desplazamiento.

Los datos desplazados se depositan en la marca de desbordamiento (SM1.1). El registro de desplazamiento puede tener una longitud máxima de 64 bits (positiva o negativa). La figura 9-40 muestra el desplazamiento de bits de un valor N positivo y de un valor N negativo.

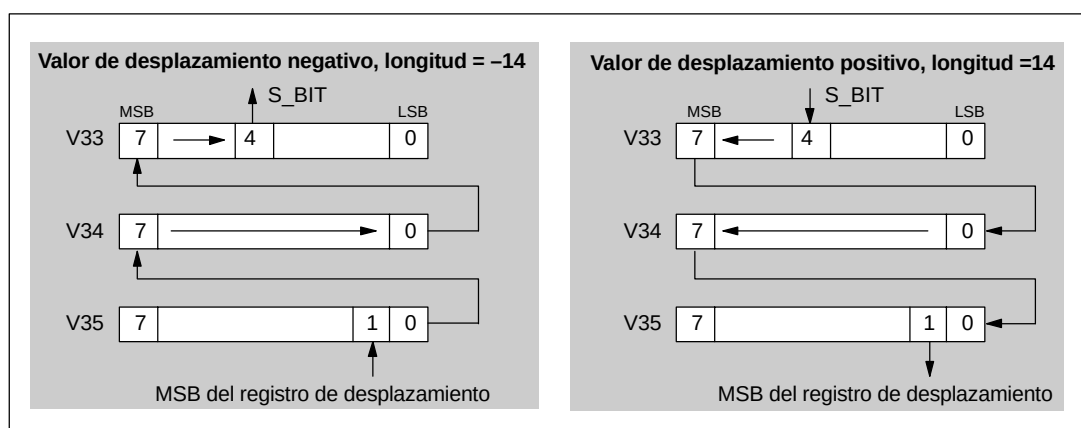


Figura 9-40 Entrada y salida de valores positivos y negativos en el registro de desplazamiento

Ejemplo de la operación Registro de desplazamiento

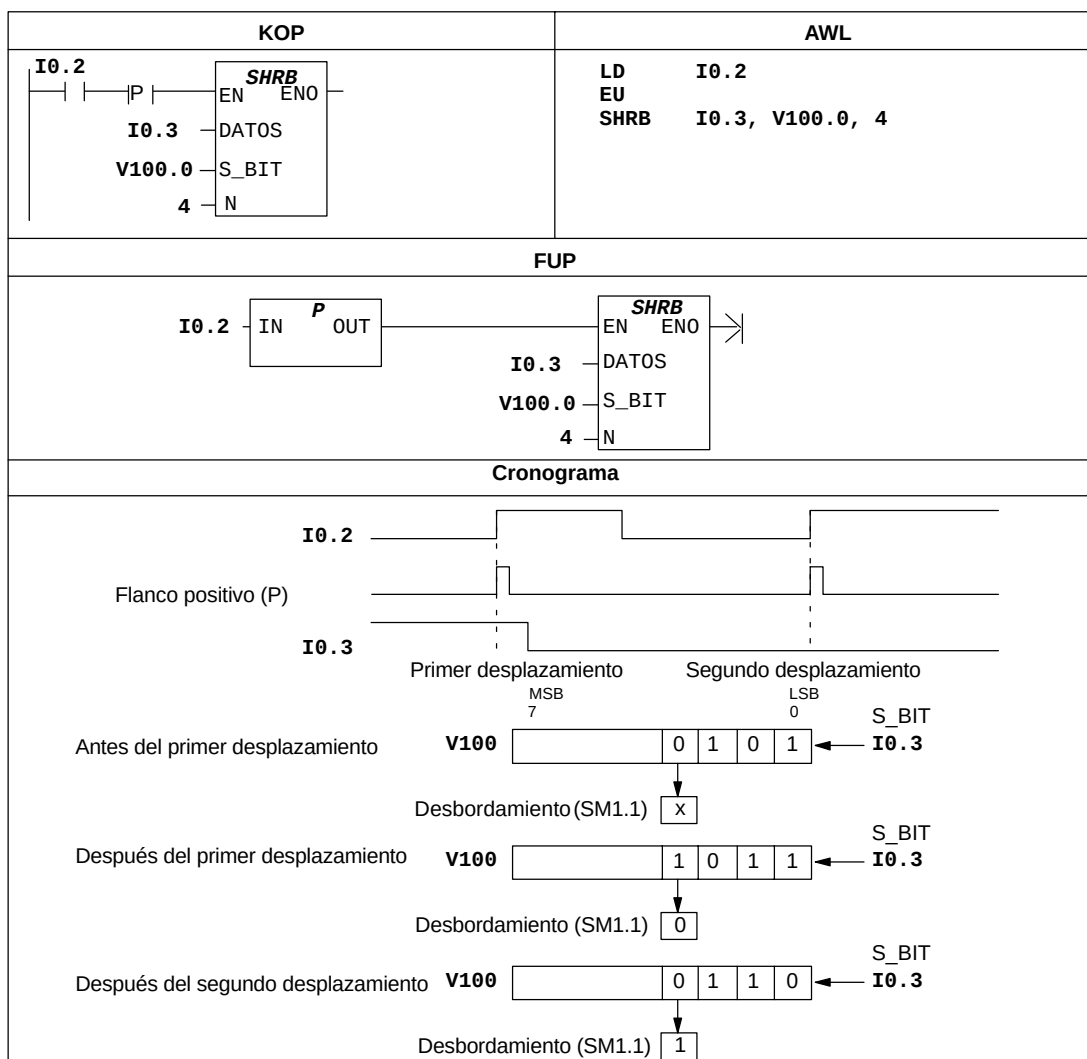
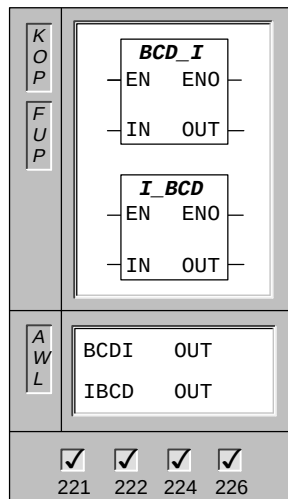


Figura 9-41 Ejemplo de la operación Registro de desplazamiento en KOP, AWL y FUP (SIMATIC)

9.13 Operaciones de conversión (SIMATIC)

Convertir de BCD a entero, Convertir de entero a BCD



La operación **Convertir de BCD a entero** convierte el valor BCD de entrada (IN) en un valor de entero y carga el resultado en la variable indicada por OUT. El margen válido de IN está comprendido entre 0 y 9999 BCD.

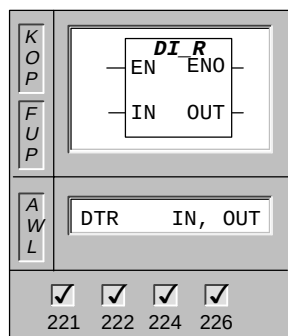
La operación **Convertir de entero a BCD** convierte el valor entero de entrada (IN) en un valor BCD y carga el resultado en la variable indicada por OUT. El margen válido de IN está comprendido entre 0 y 9999 entero.

Condiciones de error que ponen ENO a 0: SM1.6 (error BCD), SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales: SM1.6 (BCD no válido)

Entradas/salidas	Operandos	Tipos de datos
IN	VW, T, C, IW, QW, MW, SMW, LW, AC, AIW, constante, *VD, *AC, SW, *LD	WORD
OUT	VW, T, C, IW, QW, MW, SMW, LW, AC, *VD, *AC, SW, *LD	WORD

Convertir de entero doble a real

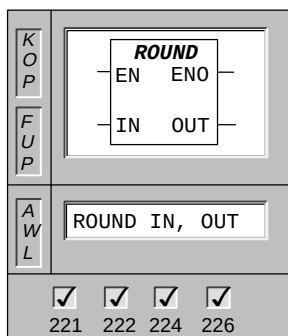


La operación **Convertir de entero doble a real** convierte un entero de 32 bits con signo (IN) en un número real de 32 bits y deposita el resultado en la variable indicada por OUT.

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Entradas/salidas	Operandos	Tipos de datos
IN	VD, ID, QD, MD, SMD, AC, LD, HC, constante, *VD, *AC, SD, *LD	DINT
OUT	VD, ID, QD, MD, SMD, LD, AC, *VD, *AC, SD, *LD	REAL

Redondear a entero doble



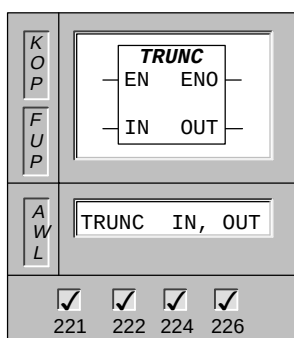
La operación **Redondear a entero doble** convierte el valor real (IN) en un valor de entero doble y deposita el resultado en la variable indicada por OUT. Si la fracción es 0,5 o superior, el número se redondeará al próximo entero superior.

Condiciones de error que ponen ENO a 0:
SM1.1 (desbordamiento), SM4.3 (tiempo de ejecución),
0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales:
SM1.1 (desbordamiento)

Entradas/salidas	Operandos	Tipos de datos
IN	VD, ID, QD, MD, SMD, AC, LD, constante, *VD, *AC, SD, *LD	REAL
OUT	VD, ID, QD, MD, SMD, LD, AC, *VD, *AC, SD, *LD	DINT

Truncar



La operación **Truncar** convierte un número real de 32 bits (IN) en un entero de 32 bits con signo y carga el resultado en la variable indicada por OUT. Sólo se convierte la parte entera del número real. La fracción se pierde.

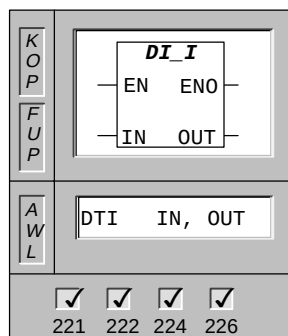
Si el valor a convertir no es un número real válido o si es demasiado grande para ser representado en la salida, la marca de desbordamiento se activará y la salida no se verá afectada.

Condiciones de error que ponen ENO a 0:
SM1.1 (desbordamiento), SM4.3 (tiempo de ejecución),
0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales:
SM1.1 (desbordamiento)

Entradas/salidas	Operandos	Tipos de datos
IN	VD, ID, QD, MD, SMD, LD, AC, constante, *VD, *AC, SD, *LD	REAL
OUT	VD, ID, QD, MD, SMD, LD, AC, *VD, *AC, SD, *LD	DINT

Convertir de entero doble a entero



La operación **Convertir de entero doble a entero** convierte el valor de entero doble (IN) en un valor de entero y deposita el resultado en la variable indicada por OUT.

Si el valor a convertir es demasiado grande para ser representado en la salida, la marca de desbordamiento se activará y la salida no se verá afectada.

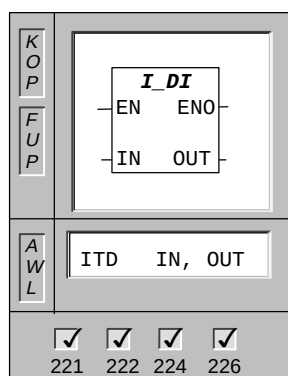
Condiciones de error que ponen ENO a 0:

SM1.1 (desbordamiento), SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales: SM1.1 (desbordamiento)

Entradas/salidas	Operandos	Tipos de datos
IN	VD, ID, QD, MD, SMD, AC, LD, HC, constante, *VD, *AC, SD, *LD	DINT
OUT	VW, IW, QW, MW, SW, SMW, LW, T, C, AC, *VD, *LD, *AC	INT

Convertir de entero a entero doble



La operación **Convertir de entero a entero doble** convierte el valor de entero (IN) en un valor de entero doble y deposita el resultado en la variable indicada por OUT. El signo se amplía.

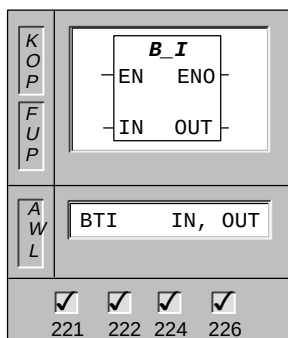
Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Entradas/salidas	Operandos	Tipos de datos
IN	VW, IW, QW, MW, SW, SMW, LW, T, C, AIW, AC, constante, *AC, *VD, *LD	INT
OUT	VD, ID, QD, MD, SD, SMD, LD, AC, *VD, *LD, *AC	DINT

Convertir de entero a real

Para convertir un entero a un número real, utilice la operación Convertir de entero a entero doble (v. pág. 9-130) y luego la operación Convertir de entero doble a real (v. fig. 9-42).

Convertir de byte a entero

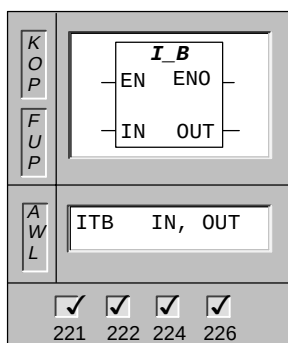


La operación **Convertir de byte a entero** convierte el valor de byte (IN) en un valor de entero y deposita el resultado en la variable indicada por OUT. El byte no tiene signo. Por tanto, no hay ampliación de signo.

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Entradas/salidas	Operandos	Tipos de datos
IN	VB, IB, QB, MB, SB, SMB, LB, AC, constante, *AC, *VD, *LD	BYTE
OUT	VW, IW, QW, MW, SW, SMW, LW, T, C, AC, *VD, *LD, *AC	INT

Convertir de entero a byte



La operación **Convertir de entero a byte** convierte el valor de entero (IN) en un valor de byte y deposita el resultado en la variable indicada por OUT.

Se convierten los valores comprendidos entre 0 y 255. Todos los demás valores producen un desbordamiento y la salida no se ve afectada.

Condiciones de error que ponen ENO a 0:
SM1.1 (desbordamiento), SM4.3 (tiempo de ejecución),
0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales:
SM1.1 (desbordamiento)

Entradas/salidas	Operandos	Tipos de datos
IN	VW, IW, QW, MW, SW, SMW, LW, T, C, AIW, AC, constante, *VD, *LD, *AC	INT
OUT	VB, IB, QB, MB, SB, SMB, LB, AC, *VD, *AC, *LD	BYTE

Ejemplos de conversión

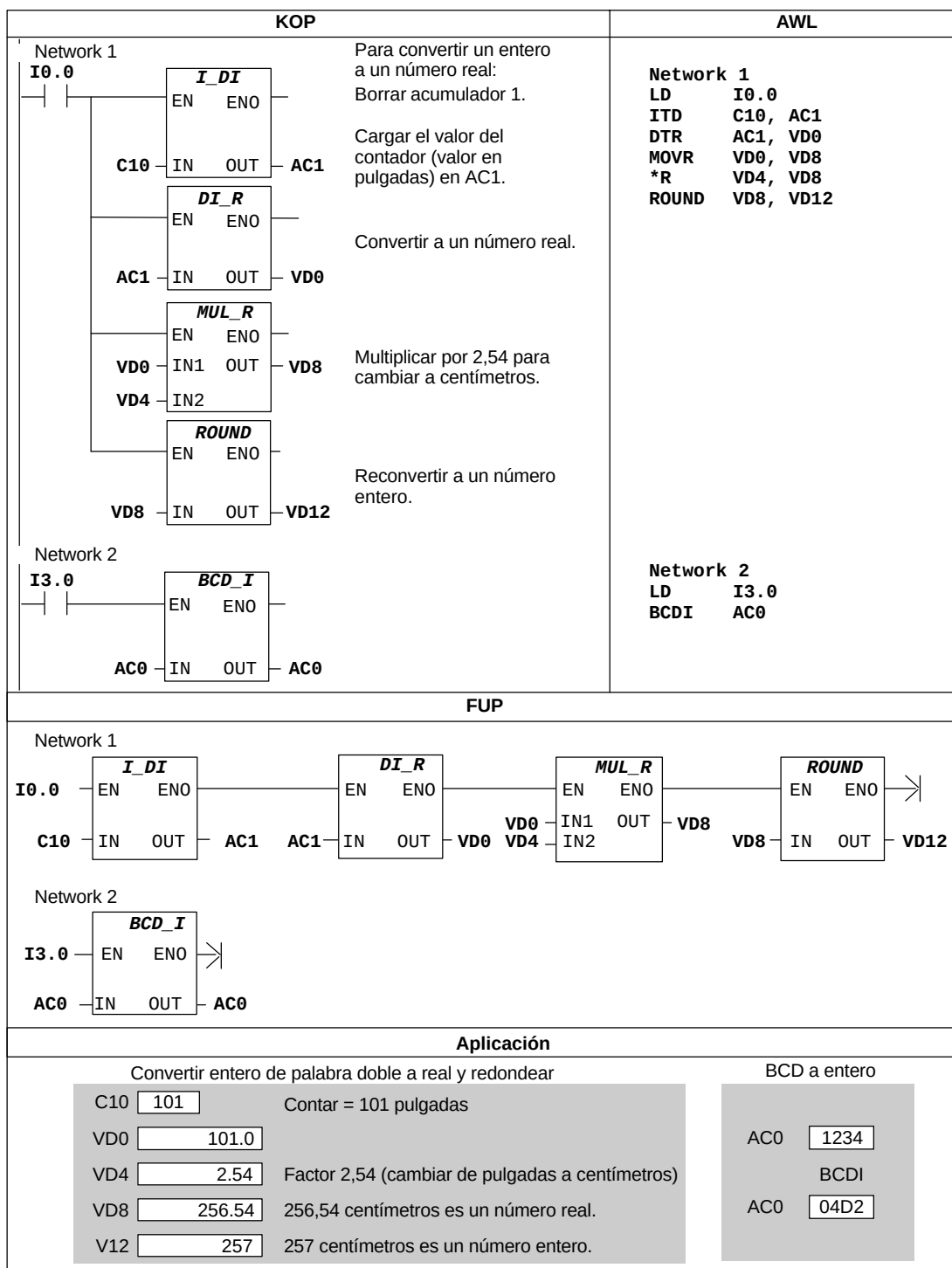
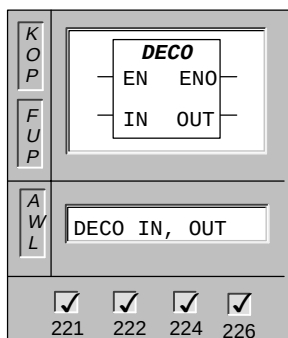


Figura 9-42 Ejemplo de operaciones de conversión en KOP, AWL y FUP (SIMATIC)

Decodificar

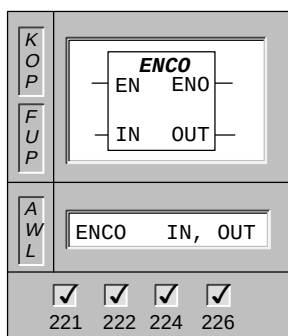


La operación **Decodificar** activa el bit de la palabra de salida (OUT). Este bit corresponde al número de bit representado por el medio byte menos significativo (4 bits) del byte de entrada (IN). Todos los demás bits de la palabra de salida se ponen a 0.

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Entradas/salidas	Operandos	Tipos de datos
IN	VB, IB, QB, MB, SMB, LB, SB, AC, constante, *VD, *AC, *LD	BYTE
OUT	VW, IW, QW, MW, SMW, LW, SW, AQW, T, C, AC, *VD, *AC, *LD	WORD

Codificar



La operación **Codificar** escribe el número del bit menos significativo de la palabra de entrada (IN) en el medio byte menos significativo (4 bits) del byte de salida (OUT).

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Entradas/salidas	Operandos	Tipos de datos
IN	VW, T, C, IW, QW, MW, SMW, AC, LW, AIW, constante, *VD, *AC, SW, *LD	WORD
OUT	VB, IB, QB, MB, SMB, LB, AC, *VD, *AC, SB, *LD	BYTE

Ejemplos de las operaciones Decodificar y Codificar

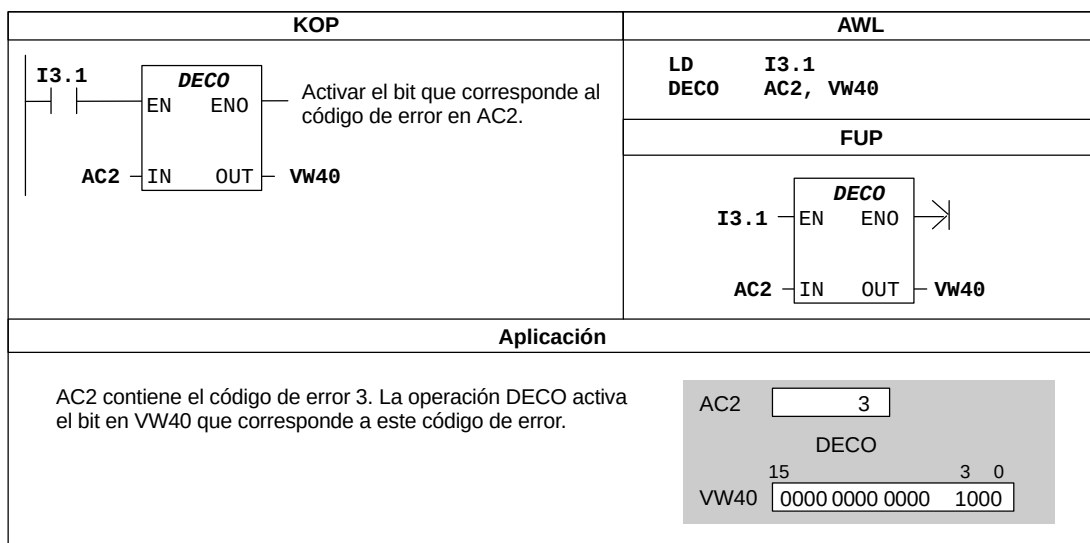


Figura 9-43 Ejemplo de activación de un bit de error utilizando la operación Decodificar en KOP, AWL y FUP (SIMATIC)

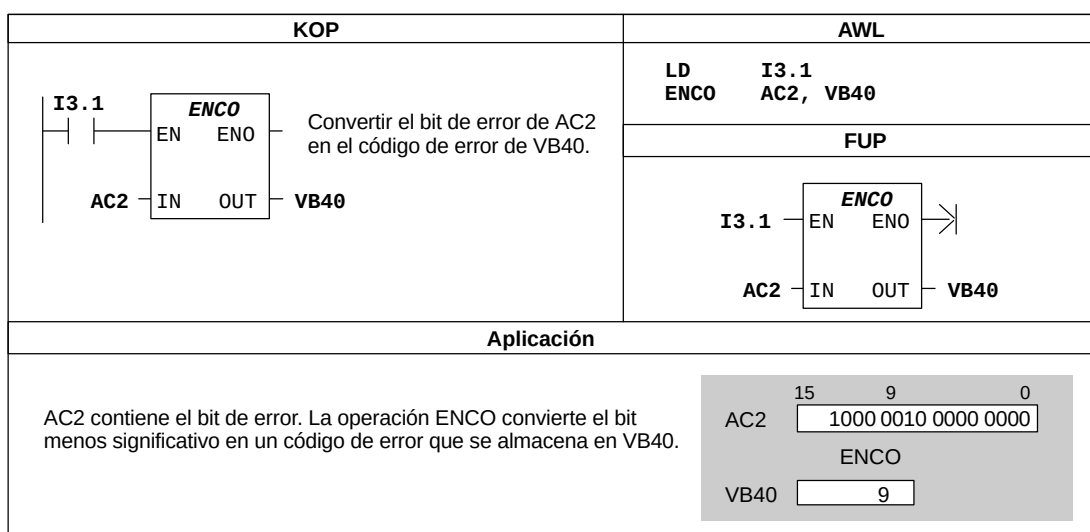
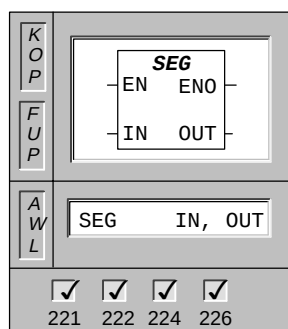


Figura 9-44 Convertir el bit de error en un código de error con la operación Codificar en KOP, AWL y FUP

Segmento



La operación **Segmento** utiliza el carácter indicado por IN para generar una configuración binaria (OUT) que ilumina los segmentos de un indicador de siete segmentos. Los segmentos iluminados representan el carácter depositado en el dígito menos significativo del byte de entrada (IN).

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

La figura 9-45 muestra la codificación del indicador de siete segmentos utilizado por la operación Segmento.

Entradas/salidas	Operandos	Tipos de datos
IN	VB, IB, QB, MB, SMB, LB, AC, constante, *VD, *AC, SB, *LD	BYTE
OUT	VB, IB, QB, MB, SMB, LB, AC, *VD, *AC, SB, *LD	BYTE

(IN) LSD	Indicador Segmentos	(OUT) - g f e d c b a		(IN) LSD	Indicador Segmentos	(OUT) - g f e d c b a
0		0 0 1 1 1 1 1 1		8		0 1 1 1 1 1 1 1
1		0 0 0 0 0 1 1 0		9		0 1 1 0 0 1 1 1
2		0 1 0 1 1 0 1 1		A		0 1 1 1 0 1 1 1
3		0 1 0 0 1 1 1 1		B		0 1 1 1 1 1 0 0
4		0 1 1 0 0 1 1 0		C		0 0 1 1 1 0 0 1
5		0 1 1 0 1 1 0 1		D		0 1 0 1 1 1 1 0
6		0 1 1 1 1 1 0 1		E		0 1 1 1 1 0 0 1
7		0 0 0 0 0 1 1 1		F		0 1 1 1 0 0 0 1

Figura 9-45 Codificación del indicador de siete segmentos

Ejemplo de la operación Segmento

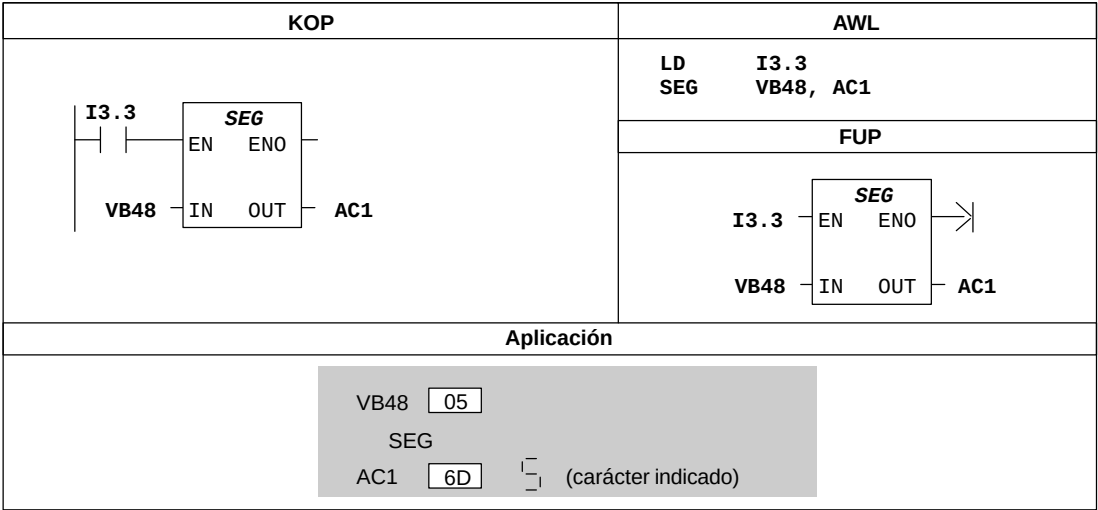
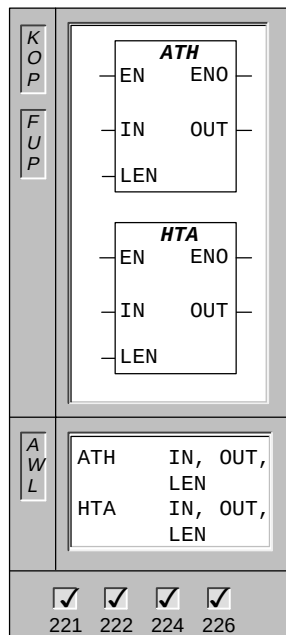


Figura 9-46 Ejemplo de la operación Segmento en KOP, AWL y FUP (SIMATIC)

Convertir de ASCII a hexadecimal, Convertir de hexadecimal a ASCII



La operación **Convertir de ASCII a hexadecimal** convierte la cadena ASCII de longitud (LEN), a partir del carácter IN, en dígitos hexadecimales, comenzando en OUT. La cadena ASCII puede tener una longitud máxima de 255 caracteres.

La operación **Convertir de hexadecimal a ASCII** convierte los dígitos hexadecimales a partir del byte de entrada (IN) en una cadena ASCII, comenzando en OUT. El número de dígitos hexadecimales a convertir viene indicado por la longitud (LEN). Es posible convertir 255 dígitos hexadecimales como máximo.

Los caracteres ASCII admisibles son los valores hexadecimales 30 a 39 y 41 a 46.

Convertir de ASCII a hexadecimal: Condiciones de error que ponen ENO a 0: SM1.7 (ASCII no válido), SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto), 0091 (operando fuera de área)

Convertir de hexadecimal a ASCII: Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto), 0091 (operando fuera de área)

Estas operaciones afectan a las siguientes marcas especiales: SM1.7 (ASCII no válido)

Entradas/salidas	Operandos	Tipos de datos
IN, OUT	VB, IB, QB, MB, SMB, LB, *VD, *AC, SB, *LD	BYTE
LEN	VB, IB, QB, MB, SMB, LB, AC, constante, *VD, *AC, SB, *LD	BYTE

Ejemplo de la operación Convertir de ASCII a hexadecimal

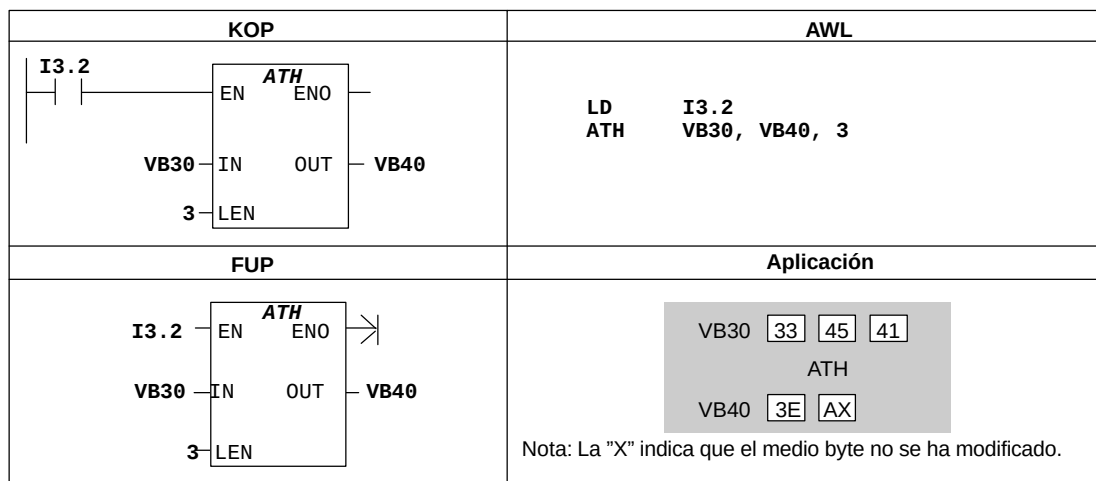
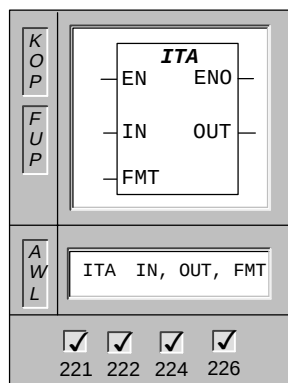


Figura 9-47 Ejemplo de la operación Convertir de ASCII a hexadecimal en KOP, AWL y FUP (SIMATIC)

Convertir de entero a ASCII



La operación **Convertir de entero a ASCII** convierte un entero (IN) en una cadena ASCII. El formato (FMT) indica la precisión de la conversión a la derecha del decimal, así como si el punto decimal debe aparecer en forma de coma o de punto. La conversión resultante se deposita en 8 bytes consecutivos comenzando en OUT. La cadena ASCII comprende siempre 8 caracteres.

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto), sin salida (formato no válido)

Entradas/salidas	Operandos	Tipos de datos
IN	VW, IW, QW, MW, SW, SMW, LW, AIW, T, C, AC, constante, *VD, *AC, *LD	INT
FMT	VB, IB, QB, MB, SMB, LB, AC, constante, *VD, *AC, SB, *LD	BYTE
OUT	VB, IB, QB, MB, SMB, LB, *VD, *AC, SB, *LD	BYTE

El operando de formato (FMT) para la operación ITA (Convertir de entero a ASCII) se define en la figura 9-48. El tamaño del búfer de salida es siempre de 8 bytes. El campo nnn indica el número de dígitos a la derecha del punto decimal en el búfer de salida. El margen válido para el campo nnn está comprendido entre 0 y 5. Si se indican 0 dígitos a la derecha del punto decimal, el valor se visualizará sin punto decimal. Si el valor nnn es mayor que 5, el búfer de salida se llenará con espacios ASCII. El bit c indica si se utiliza una coma (c = 1) o un punto decimal (c = 0) como separador entre el número entero y la fracción. Los 4 bits superiores deben ser cero.

El búfer de salida se formatea conforme a las siguientes reglas:

1. Los valores positivos se escriben sin signo en el búfer de salida.
2. Los valores negativos se escriben precedidos de un signo menos (–) en el búfer de salida.
3. Los ceros a la izquierda del punto decimal (con excepción del dígito adyacente a dicho punto) se suprimen.
4. Los valores se justifican a la derecha en el búfer de salida.

La figura 9-48 muestra ejemplos de valores que se formatean utilizando un punto decimal (c= 0) con tres dígitos a la derecha del punto decimal (nnn = 011).

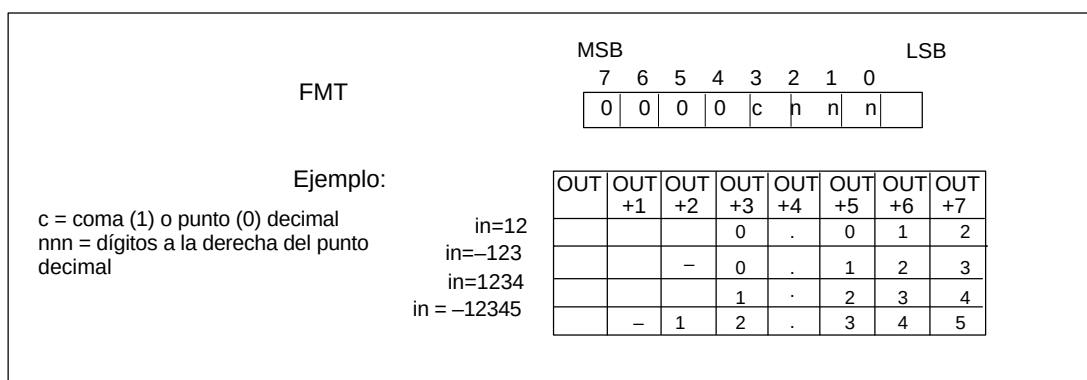
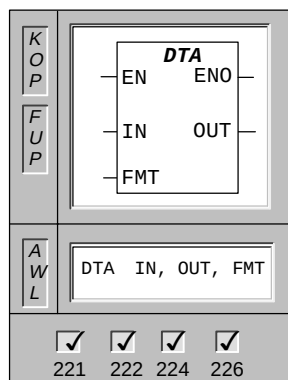


Figura 9-48 Operando FMT para la operación Convertir de entero a ASCII (ITA)

Convertir de entero doble a ASCII



La operación **Convertir de entero doble a ASCII** convierte una palabra doble (IN) en una cadena ASCII. El formato (FMT) indica la precisión de conversión a la derecha del decimal. La conversión resultante se deposita en 12 bytes consecutivos comenzando en OUT.

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto), sin salida (formato no válido)

Entradas/salidas	Operandos	Tipos de datos
IN	VD, ID, QD, MD, SD, SMD, LD, HC, constante, AC, *VD, *AC, *LD	DINT
FMT	VB, IB, QB, MB, SMB, LB, AC, constante, *VD, *AC, SB, *LD	BYTE
OUT	VB, IB, QB, MB, SMB, LB, *VD, *AC, SB, *LD	BYTE

El operando de formato (FMT) para la operación DTA se define en la figura 9-49. El tamaño del búfer de salida es siempre de 12 bytes. El campo nnn indica el número de dígitos a la derecha del punto decimal en el búfer de salida. El margen válido para el campo nnn está comprendido entre 0 y 5. Si se indican 0 dígitos a la derecha del punto decimal, el valor se visualizará sin punto decimal. Si el valor nnn es mayor que 5, el búfer de salida se llenará con espacios ASCII. El bit c indica si se utiliza una coma (c = 1) o un punto decimal (c = 0) como separador entre el número entero y la fracción. Los 4 bits superiores deben ser cero. El búfer de salida se formatea conforme a las siguientes reglas:

1. Los valores positivos se escriben sin signo en el búfer de salida.
2. Los valores negativos se escriben precedidos de un signo menos (–) en el búfer de salida.
3. Los ceros a la izquierda del punto decimal (con excepción del dígito adyacente a dicho punto) se suprimen.
4. Los valores se justifican a la derecha en el búfer de salida.

La figura 9-49 muestra ejemplos de valores que se formatean utilizando un punto decimal (c = 0) con cuatro dígitos a la derecha del punto decimal (nnn = 100).

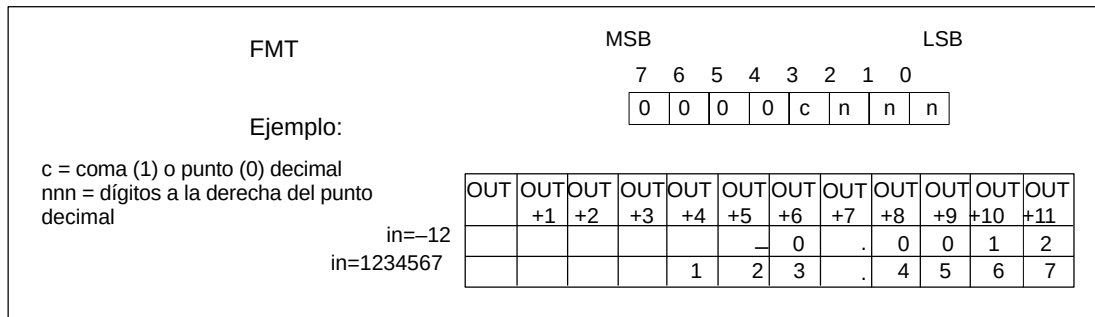
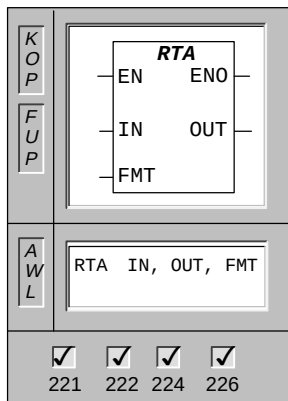


Figura 9-49 Operando FMT para la operación Convertir de entero doble a ASCII

Convertir de real a ASCII



La operación **Convertir de real a ASCII** convierte el valor en coma flotante (IN) en una cadena ASCII. El formato (FMT) indica la precisión de la conversión a la derecha del decimal, así como si el punto decimal debe aparecer en forma de coma o de punto, y también el tamaño del búfer de salida. La conversión resultante se deposita en un búfer de salida que comienza en OUT. La longitud de la cadena ASCII resultante corresponde al tamaño del búfer de salida, pudiendo indicarse en un margen comprendido entre 3 y 15.

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto), sin salida (formato no válido o búfer demasiado pequeño)

Entradas/salidas	Operandos	Tipos de datos
IN	VD, ID, QD, MD, SD, SMD, LD, AC, *VD, *AC, *LD	REAL
FMT	VB, IB, QB, MB, SMB, LB, AC, constante, *VD, *AC, SB, *LD	BYTE
OUT	VB, IB, QB, MB, SMB, LB, *VD, *AC, SB, *LD	BYTE

El operando de formato (FMT) para la operación RTA se define en la figura 9-50. El campo ssss indica el tamaño del búfer de salida. No es válido un tamaño de 0, 1 ó 2 bytes. El campo nnn indica el número de dígitos a la derecha del punto decimal en el búfer de salida. El margen válido para el campo nnn está comprendido entre 0 y 5. Si se indican 0 dígitos a la derecha del punto decimal, el valor se visualizará sin punto decimal. El búfer de salida se rellena con espacios ASCII si los valores nnn son mayores que 5 o si el búfer es demasiado pequeño para almacenar el valor convertido. El bit c indica si se utiliza una coma (c = 1) o un punto decimal (c = 0) como separador entre el número entero y la fracción. El búfer de salida se formatea conforme a las siguientes reglas:

1. Los valores positivos se escriben sin signo en el búfer de salida.
2. Los valores negativos se escriben precedidos de un signo menos (–) en el búfer de salida.
3. Los ceros a la izquierda del punto decimal (con excepción del dígito adyacente a dicho punto) se suprimen.
4. Los valores a la derecha del punto decimal se redondean para que correspondan al número de dígitos indicado.
5. El búfer de salida deberá ser por lo menos tres bytes más grande que el número de dígitos a la derecha del punto decimal.
6. Los valores se justifican a la derecha en el búfer de salida.

La figura 9-50 muestra ejemplos de valores que se formatean utilizando un punto decimal (c = 0) con un dígito a la derecha del punto decimal (nnn = 001) y un tamaño de búfer de seis bytes (ssss = 0110).

MSB								LSB
7	6	5	4	3	2	1	0	
s	s	s	s	c	n	n	n	

ssss = tamaño del búfer de salida	in = 1234.5	Out	Out +1	Out +2	Out +3	Out +4	Out +5
c = coma (1) o punto (0) decimal	in = –0.0004	1	2	3	4	.	5
nnn = dígitos a la derecha del punto decimal	in = –3.67526				0	.	0
	in = 1.95			–	3	.	7
					2	.	0

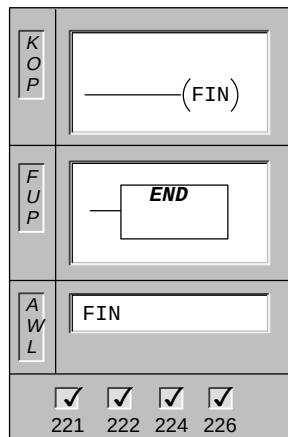
Figura 9-50 Operando FMT para la operación Convertir de real a ASCII

Nota

El formato en coma flotante utilizado por la CPU S7-200 soporta 7 dígitos significativos como máximo. Si se intenta visualizar más de estos 7 dígitos significativos se producirá un error de redondeo.

9.14 Operaciones de control del programa (SIMATIC)

END condicional



La operación condicional **Finalizar programa principal** finaliza el programa en función de la combinación lógica precedente.

Operandos: Ninguno

Tipos de datos: Ninguno

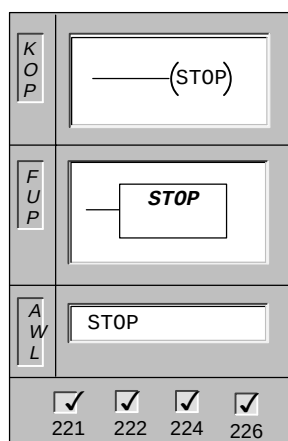
Nota

La operación END condicional se puede utilizar en el programa principal, pero no en subrutinas ni en rutinas de interrupción.

Nota

STEP 7-Micro/WIN 32 añade automáticamente un fin absoluto al programa principal de usuario.

STOP

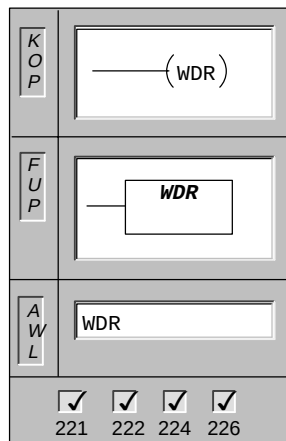


La operación **STOP** finaliza inmediatamente la ejecución del programa haciendo que la CPU cambie de RUN a STOP.

Operandos: Ninguno

Si la operación STOP se ejecuta en una rutina de interrupción, ésta se finalizará inmediatamente ignorando las interrupciones pendientes. Las demás acciones en el ciclo actual se completan, incluyendo la ejecución del programa principal. El cambio de RUN a STOP se produce al final del ciclo actual.

Borrar temporizador de vigilancia



La operación **Borrar temporizador de vigilancia** permite que la CPU redispere el temporizador de vigilancia. Así se prolonga el tiempo de ciclo sin que se indique un error de vigilancia.

Operandos: Ninguno

Utilizar la operación WDR para inicializar el temporizador de vigilancia

Esta operación se debe utilizar con mucha cautela. En caso de utilizar bucles para que no finalice el ciclo o para prolongarlo excesivamente, es posible que no se ejecuten los procesos siguientes hasta completar el ciclo:

- Comunicación (excepto modo Freeport)
- Actualización de las entradas y salidas (excepto control directo de las E/S)
- Actualización de los valores forzados
- Actualización de las marcas especiales (no se actualizan las marcas SM0 y SM5 a SM29)
- Tareas de diagnóstico en el tiempo de ejecución
- Los temporizadores con resolución de 10 ms y 100 ms no contarán correctamente los ciclos que excedan los 25 segundos.
- Operación STOP si se utiliza en una rutina de interrupción

Nota

Si se prevé que el tiempo de ciclo durará más de 300 ms o que la actividad de interrupción aumentará de modo que el ciclo principal quede interrumpido más de 300 ms, es preciso utilizar la operación WDR para redispere el temporizador de vigilancia.

Cambiando el selector a la posición STOP, la CPU pasará a modo STOP en 1,4 segundos.

Ejemplos de las operaciones STOP, WDR y END

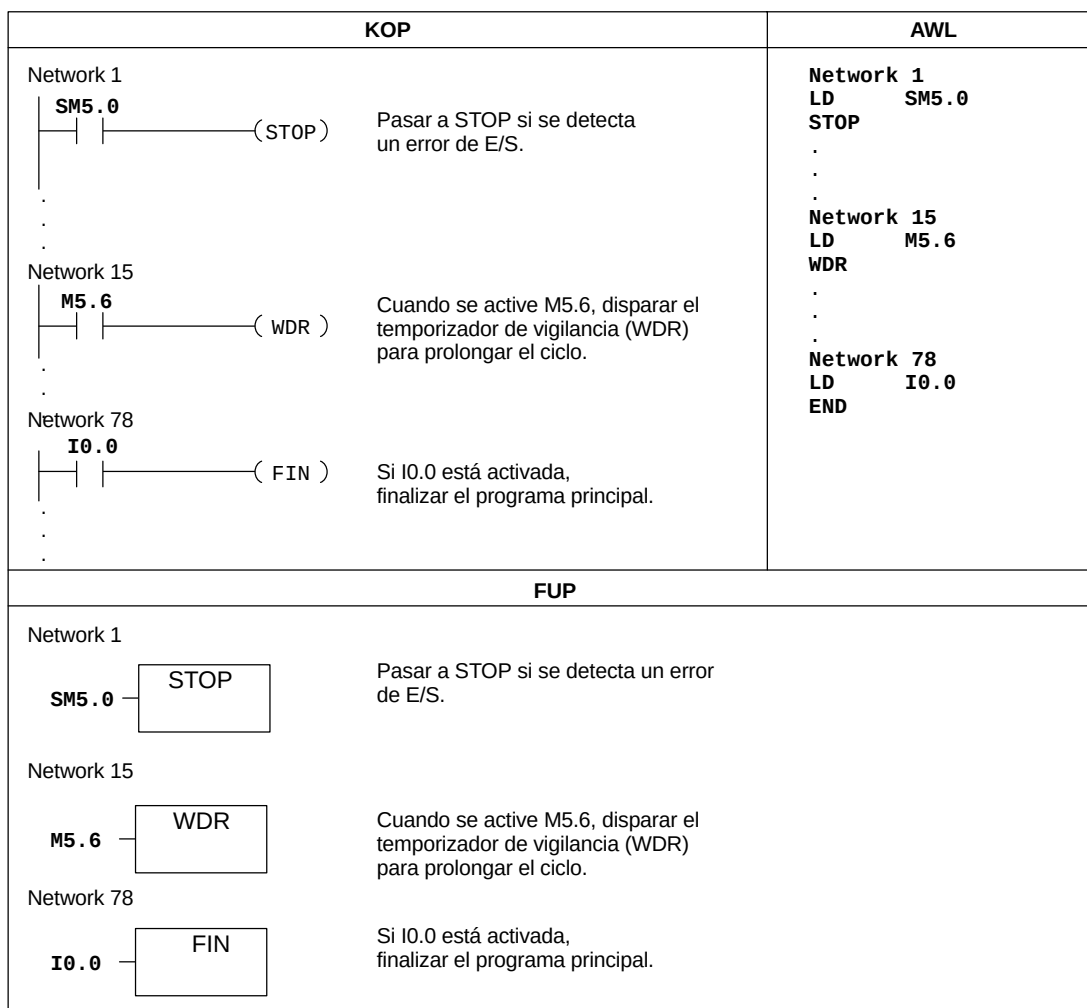
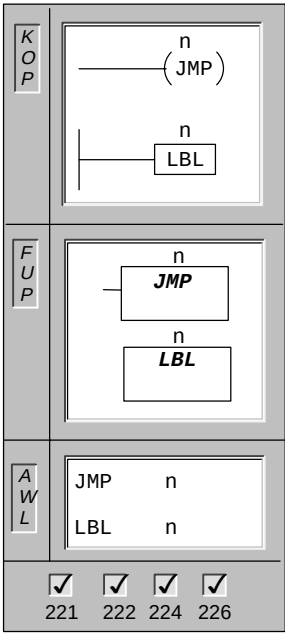


Figura 9-51 Ejemplos de las operaciones STOP, WDR y END en KOP, AWL y FUP (SIMATIC)

Saltar a meta, Definir meta



La operación **Saltar a meta** deriva la ejecución del programa a la meta indicada (n). Al saltar, el primer valor de la pila es siempre un "1" lógico.

La operación **Definir meta** indica la dirección de la meta de salto (n).

Operandos: n: Constante (0 a 255)

Tipos de datos: WORD

Tanto la operación de salto como la correspondiente meta deben encontrarse en el programa principal, en una subrutina o en una rutina de interrupción. Desde el programa principal no se puede saltar a una meta que se encuentre en una subrutina o en una rutina de interrupción. Tampoco es posible saltar desde una subrutina o una rutina de interrupción a una meta que se encuentre fuera de ella.

Ejemplo de la operación Saltar a meta

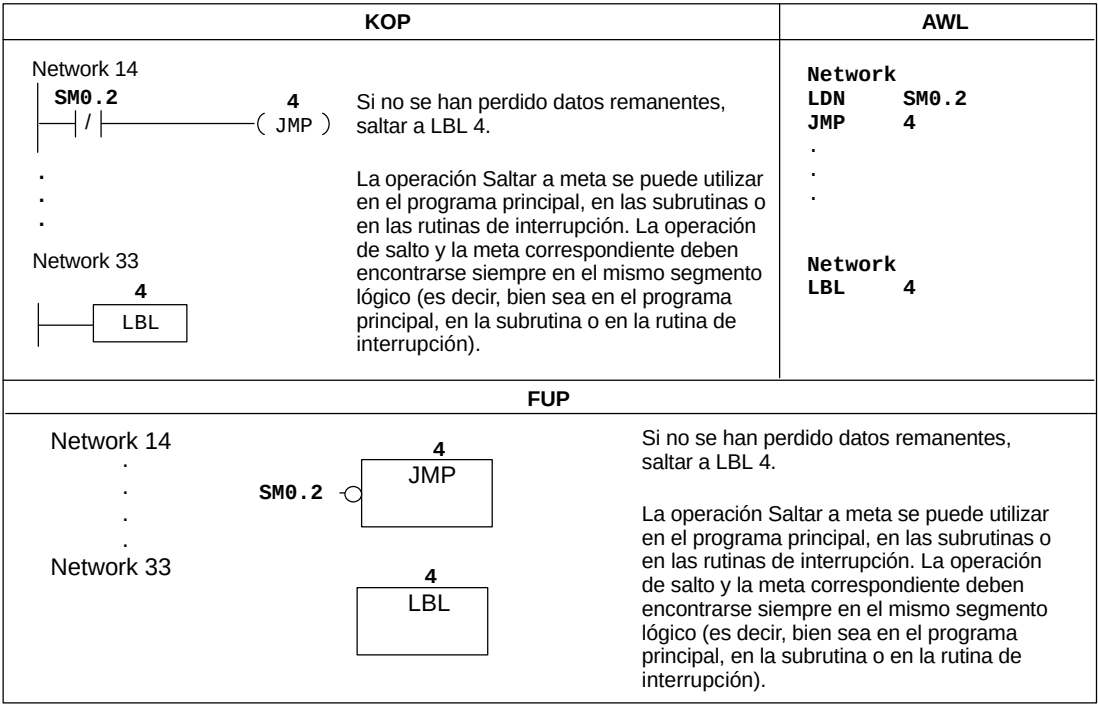
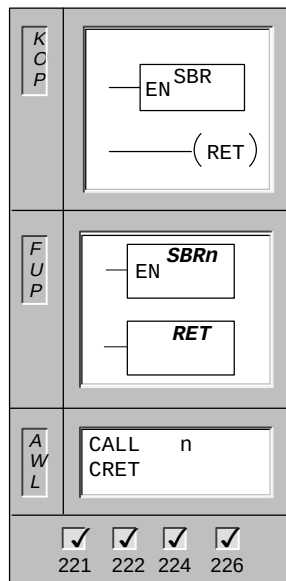


Figura 9-52 Ejemplo de las operaciones Saltar a meta y Definir meta en KOP, AWL y FUP (SIMATIC)

Llamar subrutina, Retorno de subrutina



La operación **Llamar subrutina** transfiere el control a la subrutina (n). Esta operación se puede utilizar con o sin parámetros. Para añadir una subrutina, elija los comandos de menú **Edición > Insertar > Subrutina**.

La operación **Retorno condicional de subrutina** se utiliza para finalizar una subrutina en función de la combinación lógica precedente.

Operandos: Ninguno

Tipos de datos: Ninguno

Una vez ejecutada la subrutina, el control vuelve a la operación que sigue a la llamada de la subrutina (CALL).

La figura 9-55 muestra ejemplos de las operaciones Llamar subrutina y Retorno de subrutina.

Condiciones de error que ponen ENO a 0 para la llamada de subrutina con parámetros:

SM4.3 (tiempo de ejecución), 0008 (excedida la profundidad máxima de anidamiento)

Nota

STEP 7-Micro/WIN 32 añade automáticamente un retorno desde cada subrutina.

En el programa principal, se pueden anidar (situar una llamada a subrutina en otra) hasta ocho subrutinas. Las subrutinas no se pueden anidar en una rutina de interrupción. Una llamada a subrutina no se puede disponer en ninguna otra subrutina a la que se llame desde una rutina de interrupción. Si bien la recursión (la subrutina se llama a sí misma) está permitida, hay que utilizarla con cautela.

Cuando se llama a una subrutina, se almacena toda la pila lógica, poniéndose a "1" el nivel superior de la pila. Sus demás niveles se ponen a "0" y la ejecución se transfiere a la subrutina que se ha llamado. Cuando ésta se termina de ejecutar, se restablece la pila con los valores almacenados al llamar a la subrutina y se retorna a la rutina que ha efectuado la llamada.

Los acumuladores son comunes a las subrutinas y a la rutina de llamada. Los acumuladores no se almacenan ni se restablecen si se utilizan con subrutinas.

Llamar a una subrutina con parámetros

Las subrutinas pueden contener parámetros que hayan sido transferidos. Los parámetros se definen en la tabla de variables locales de la subrutina (v. figura 9-53). Estos parámetros deben tener un nombre simbólico (de 8 caracteres como máximo), un tipo de variable y un tipo de datos. Se pueden transferir 16 parámetros a o desde una subrutina.

El campo "Tipo de variable" en la tabla de variables locales define si la variable se transfiere a la subrutina (IN), a y desde la subrutina (IN_OUT) o desde la subrutina (OUT). A continuación se indican las características de los tipos de parámetros:

- **IN:** los parámetros se transfieren a la subrutina. Si el parámetro es una dirección directa (p.ej. VB10), el valor de la dirección indicada se transferirá a la subrutina. Si el parámetro es una dirección indirecta (p.ej. *AC1), el valor de la dirección a la que señala el puntero se transferirá a la subrutina. Si el parámetro es una constante de datos (16#1234) o una dirección (VB100), la constante o el valor de dirección se transferirán a la subrutina.
- **IN_OUT:** el valor de la dirección del parámetro indicado se transferirá a la subrutina y el valor resultante de la subrutina se retornará luego a la misma dirección. Para los parámetros de entrada/salida no se pueden utilizar ni constantes (p.ej. 16#1234) ni direcciones (p.ej. &VB100).
- **OUT:** El valor resultante de la subrutina se retorna a la dirección del parámetro indicado. Para los parámetros de salida no se pueden utilizar ni constantes (p.ej. 16#1234) ni direcciones (p.ej. &VB100).
- **TEMP:**
Cualquier memoria local que no se utilice para la transferencia de parámetros se puede emplear para el almacenamiento temporal dentro de la subrutina.

Para añadir un registro de parámetro, en el campo "Tipo de variable" sitúe el cursor en el tipo (IN, IN_OUT<OUT) que desea añadir. Pulse el botón derecho del ratón para visualizar un menú que ofrece diversas opciones. Seleccione la opción "Insertar" y luego la opción "Fila inferior". Debajo del registro actual aparecerá un nuevo registro de parámetro.

	Nombre	Tipo de var.	Tipo de datos	Comentario
	EN	IN	BOOL	
L0.0	IN1	IN	BOOL	
LB1	IN2	IN	BYTE	
LB2.0	IN3	IN	BOOL	
LD3	IN4	IN	DWORD	
LW7	IN/OUT1	IN_OUT	WORD	
LD9	OUT1	OUT	DWORD	
		TEMP		

Figura 9-53 Tabla de variables locales de STEP 7-Micro/WIN 32

El campo "Tipo de datos" de la tabla de variables locales define el tamaño y el formato del parámetro. Los tipos de parámetros son:

- **Circulación de corriente:** La circulación de corriente booleana sólo se permite en las entradas binarias (booleanas). Esta declaración le indica a STEP 7-Micro/WIN 32 que este tipo de parámetro de entrada es el resultado de la circulación de la corriente conforme a una combinación de operaciones lógicas con bits. Las entradas booleanas de circulación de corriente deben aparecer en la tabla de variables locales antes de cualquier otro tipo de entrada. Aquí se permite utilizar sólo parámetros de entrada. La entrada de habilitación (EN) y las entradas IN1 en la figura 9-54 usan la lógica booleana.
- **BOOL:** Este tipo de datos se utiliza para entradas y salidas binarias sencillas. IN2 en la figura 9-54 es una entrada booleana.
- **BYTE, WORD, DWORD:** Estos tipos de datos identifican parámetros de entrada o de salida sin signo compuestos por 1, 2 ó 4 bytes, respectivamente.
- **INT, DINT:** Estos tipos de datos identifican parámetros de entrada o de salida con signo compuestos por 2 ó 4 bytes, respectivamente.
- **REAL:** Este tipo de datos identifica un valor en coma flotante IEEE de precisión simple (4 bytes).

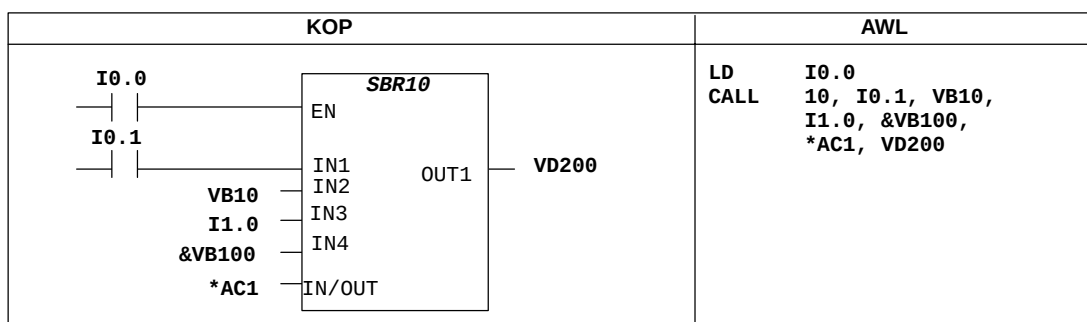


Figura 9-54 Llamada a subrutina en KOP y AWL

Los parámetros de dirección tales como IN4 en la figura 9-54 (&VB100) se transfieren a una subrutina como valor DWORD (palabra doble sin signo). El tipo de parámetro de una constante se debe indicar en la rutina de llamada mediante un descriptor delante del valor de la constante. Por ejemplo, para transferir como parámetro una constante de palabra doble sin signo cuyo valor sea 12.345, el parámetro de esta constante se deberá indicar de la siguiente forma: DW#12345. Si se omite el descriptor de la constante, se podría deducir que la constante es de un tipo diferente.

En el caso de los parámetros de entrada o de salida no se realiza una conversión automática de datos. Por ejemplo, si en la tabla de variables locales se indica que un parámetro es del tipo de datos REAL y en la rutina de llamada se indica una palabra doble (DWORD) para este parámetro, el valor en la subrutina será una palabra doble.

Los valores que se transfieren a una subrutina se depositan en la memoria local de la misma. La columna del extremo izquierdo de la tabla de variables locales (v. figura 9-53) muestra la dirección local de cada parámetro que se ha transferido. Cuando se llama a la subrutina, los valores de los parámetros de entrada se copian a la memoria local de la subrutina. Cuando se finaliza la ejecución de la subrutina, los valores de los parámetros de salida se copian de la memoria local de la subrutina a las direcciones indicadas de los parámetros de salida.

El tamaño y el tipo de los elementos de datos está codificado en los parámetros. Los valores de los parámetros se asignan de la siguiente forma a la memoria local de la subrutina:

- Los valores de parámetros se asignan a la memoria local en el orden indicado por la operación Llamar subrutina, comenzando estos parámetros en L.0.
- Uno a ocho valores binarios de parámetros consecutivos se asignan a un sólo byte comenzando en Lx.0 hasta Lx.7.
- Los valores de byte, palabra y palabra doble se asignan a la memoria local en bytes (LBx, LWx ó LDx).

En la operación Llamar subrutina con parámetros, éstos se deben organizar de la siguiente forma: primero los parámetros de entrada, luego los de entrada/salida y, por último, los de salida.

En AWL, el formato de la operación Llamar subrutina (CALL) es el siguiente:

CALL número de subrutina, parámetro 1, parámetro 2, ... , parámetro

Condiciones de error que ponen ENO a 0 para la llamada de una subrutina con parámetros: SM4.3 (tiempo de ejecución), 0008 (excedida la profundidad máxima de anidamiento)

Ejemplo de las operaciones Llamar subrutina y Retorno de subrutina

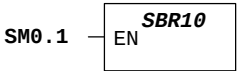
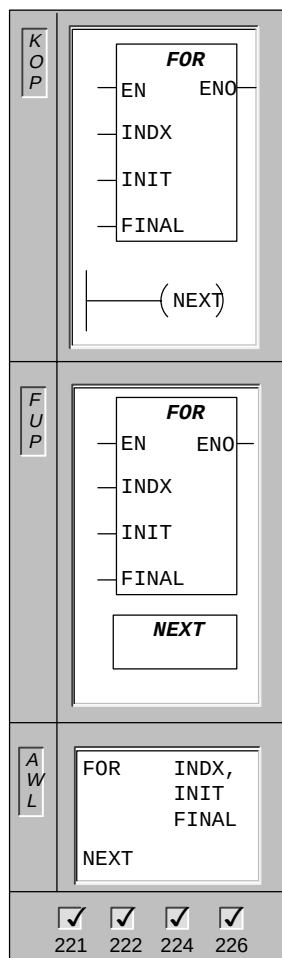
KOP		AWL
PROGRAMA PRINCIPAL		
Network 1  En el primer ciclo: Llamar SBR10 para inicializar.		Network 1 LD SM0.1 CALL 10
SUBROUTINA 10		
Comienzo de la subrutina 10. Network 6  Se puede utilizar un retorno condicional (RET) desde la subrutina 10.		Network 6 LD M14.3 CRET
FUP		
PROGRAMA PRINCIPAL		
		
SUBROUTINA 10		
		

Figura 9-55 Ejemplos de operaciones de subrutinas en KOP, AWL y FUP (SIMATIC)

FOR, NEXT



La operación **FOR** ejecuta las operaciones que se encuentren entre FOR y NEXT. Se deben indicar el valor del índice o el conteo actual del bucle (INDX), el valor inicial (INIT) y el valor final (FINAL).

La operación **NEXT** marca el final del bucle FOR y pone a "1" el primer valor de la pila.

Ejemplo: si el valor de INIT es 1 y si el de FINAL es 10, las operaciones que se encuentren entre FOR y NEXT se ejecutarán 10 veces, incrementando el valor de conteo INDX en 1, 2, 3, ...10.

Si el valor inicial es mayor que el valor final, no se ejecuta el bucle. Después de ejecutarse las operaciones que se encuentran entre FOR y NEXT, se incrementa el valor de INDX y el resultado se compara con el valor final. Si INDX es mayor que el valor final, finaliza el bucle.

para: Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Entradas/salidas	Operandos	Tipos de datos
INDX	VW, IW, QW, MW, SW, SMW, LW, T, C, AC, *VD, *AC, *LD	INT
INIT	VW, IW, QW, MW, SW, SMW, T, C, AC, LW, AIW, constante, *VD, *AC, *LD	INT
FINAL	VW, IW, QW, MW, SW, SMW, LW, T, C, AC, AIW, constante, *VD, *AC, *LD	INT

Reglas para utilizar el bucle FOR/NEXT:

- Al habilitar el bucle FOR/NEXT, éste se ejecuta hasta finalizar las iteraciones, a menos que Ud. cambie el valor final dentro del bucle. Los valores se pueden cambiar mientras se ejecute FOR/NEXT.
- Si se vuelve a habilitar el bucle, éste copia el valor inicial (INIT) en el valor actual de conteo del bucle (IDX). La operación FOR/NEXT se desactiva automáticamente la próxima vez que se habilite.

Las operaciones FOR/NEXT repiten un bucle del programa un número determinado de veces. Toda operación FOR exige una operación NEXT. Los bucles FOR/NEXT pueden anidarse (insertar un bucle FOR/NEXT dentro de otro) hasta una profundidad de ocho niveles.

Ejemplo de las operaciones FOR/NEXT

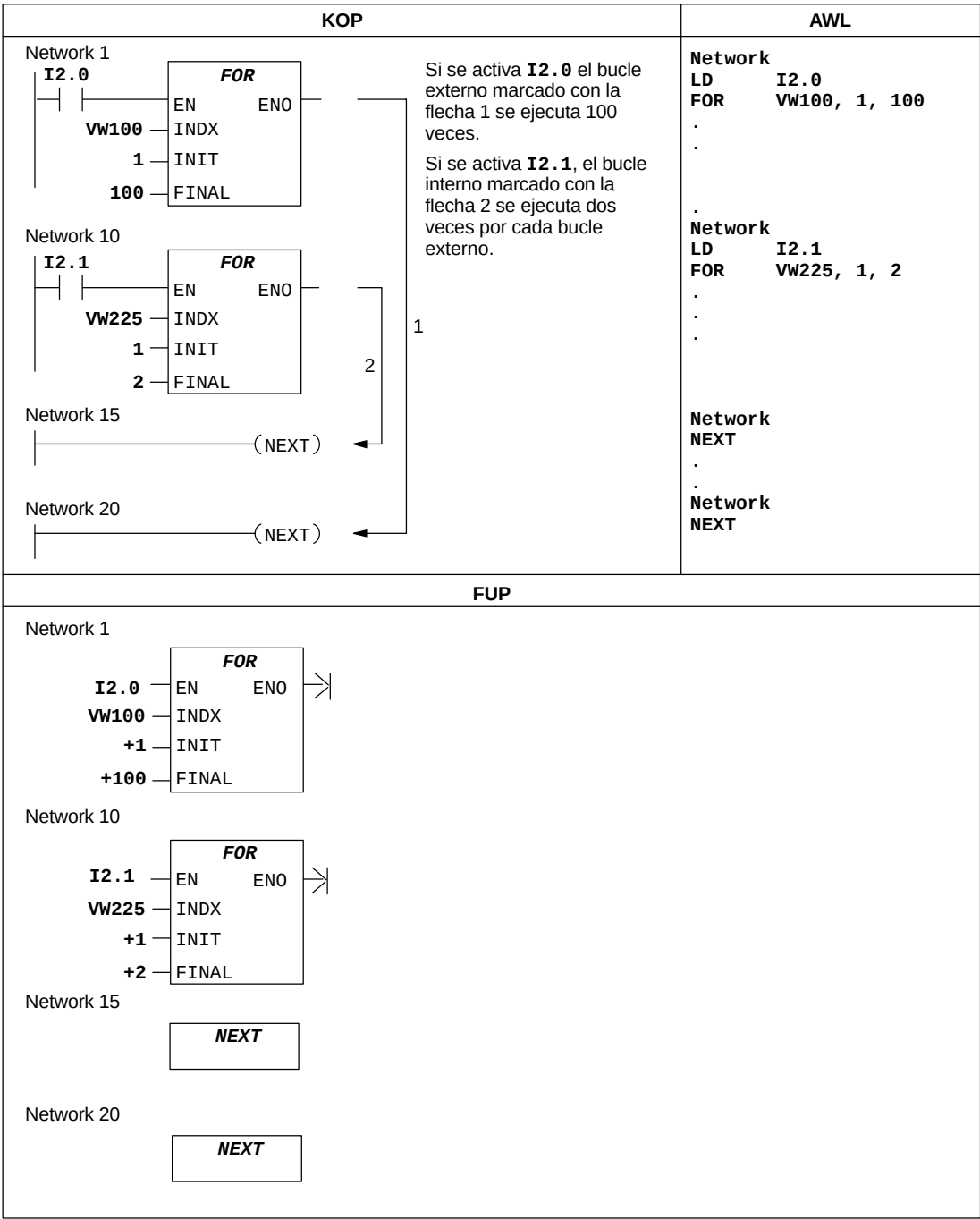
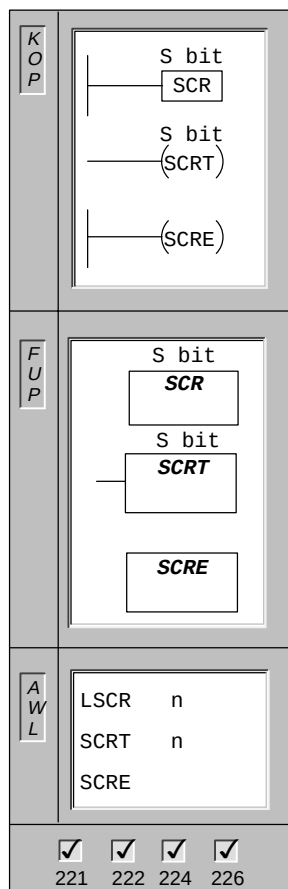


Figura 9-56 Ejemplo de las operaciones FOR/NEXT en KOP, AWL y FUP (SIMATIC)

Relé de control secuencial



La operación **Cargar relé de control secuencial** indica el comienzo de un segmento SCR. Si $n = 1$, se habilita la circulación de la corriente hacia el segmento SCR. La operación LSCR se debe finalizar con una operación SCRE.

La operación **Transición del relé de control secuencial** identifica el bit SCR que se debe habilitar (el siguiente bit S a activar). Cuando la corriente fluye hasta la bobina o hasta el cuadro FUP, el bit S direccionado se activa y el bit S de la operación LSCR (que habilitó este segmento SCR) se desactiva.

La operación **Fin del relé de control secuencial** indica el fin de un segmento SCR.

Entradas/salidas	Operandos	Tipos de datos
n	S	BOOL

Descripción de las operaciones del relé de control secuencial

En KOP, FUP y AWL, los relés de control secuencial (SCRs) se utilizan para estructurar instalaciones o etapas en segmentos equivalentes del programa. Los SCR permiten segmentar lógicamente el programa de usuario.

La operación LSCR carga el valor del bit S que indica la operación en la pila del relé de control secuencial (pila SCR), así como en la pila lógica. El segmento SCR se activa o se desactiva en función del resultado de la pila SCR. El valor superior de la pila se carga en el bit S indicado, pudiéndose conectar directamente los cuadros y las bobinas a la barra de alimentación izquierda sin necesidad de interconectar un contacto. La figura 9-57 muestra la pila SCR y la pila lógica, así como los efectos de la operación LSCR.

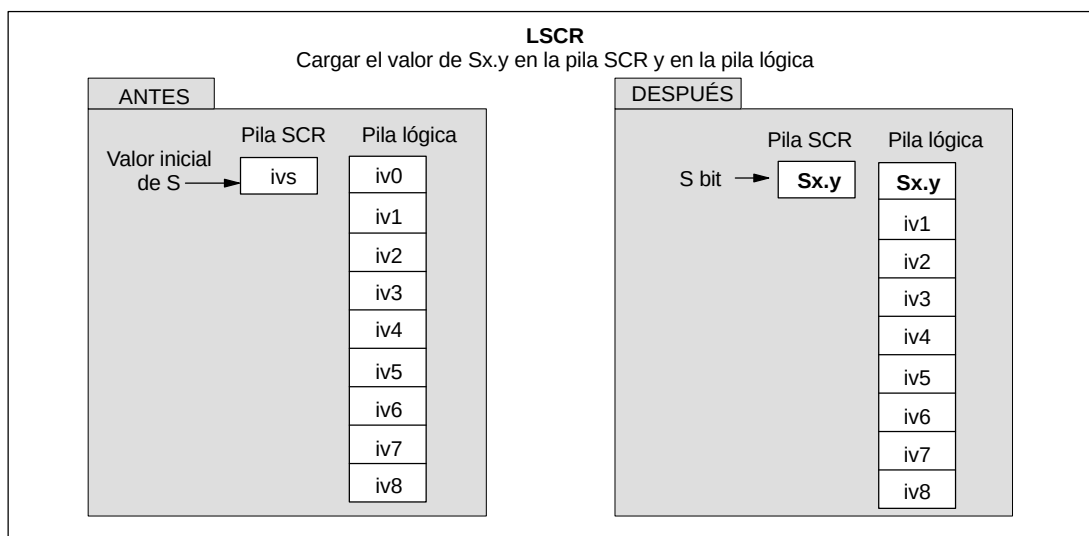


Figura 9-57 Efectos de la operación LSCR en la pila lógica

Observaciones en relación con las operaciones del relé de control secuencial:

- Todas las operaciones que se encuentren entre la operación LSCR y la operación SCRE constituyen el segmento SCR, dependiendo su ejecución del valor de la pila SCR. La lógica que se encuentra entre la operación SCRE y la siguiente operación LSCR no depende del valor de la pila SCR.
- La operación SCRT activa un bit S que habilita el siguiente relé de control secuencial. Asimismo, desactiva el bit S que se cargó para habilitar esta parte del segmento SCR.

Uso restringido

Al utilizar los relés de control secuencial es preciso tener en cuenta los siguientes puntos:

- Un mismo bit S no se puede utilizar en más de una rutina. Por ejemplo, si S0.1 se utiliza en el programa principal, no se podrá utilizar además en la subrutina.
- En un segmento SCR no se pueden usar las operaciones Saltar a meta (JMP) ni Definir meta (LBL). Por tanto, no sirven para saltar dentro ni fuera del segmento SCR, ni tampoco en el mismo. No obstante, las operaciones de salto y de meta se pueden emplear para saltar segmentos SCR.
- En un segmento SCR no se pueden utilizar las operaciones FOR, NEXT ni END.

Ejemplo de una operación SCR

La figura 9-58 muestra cómo funciona un relé de control secuencial.

- En el ejemplo, S0.1 se activa con la marca especial SM0.1 (marca del primer ciclo). S0.1 será entonces la etapa 1 activa en el primer ciclo.
- Una vez transcurrido un retardo de 2 segundos, T37 provoca una transición a la etapa 2. Esta transición desactiva el segmento SCR (S0.1) de la primera etapa y activa el segmento SCR (S0.2) de la segunda etapa.

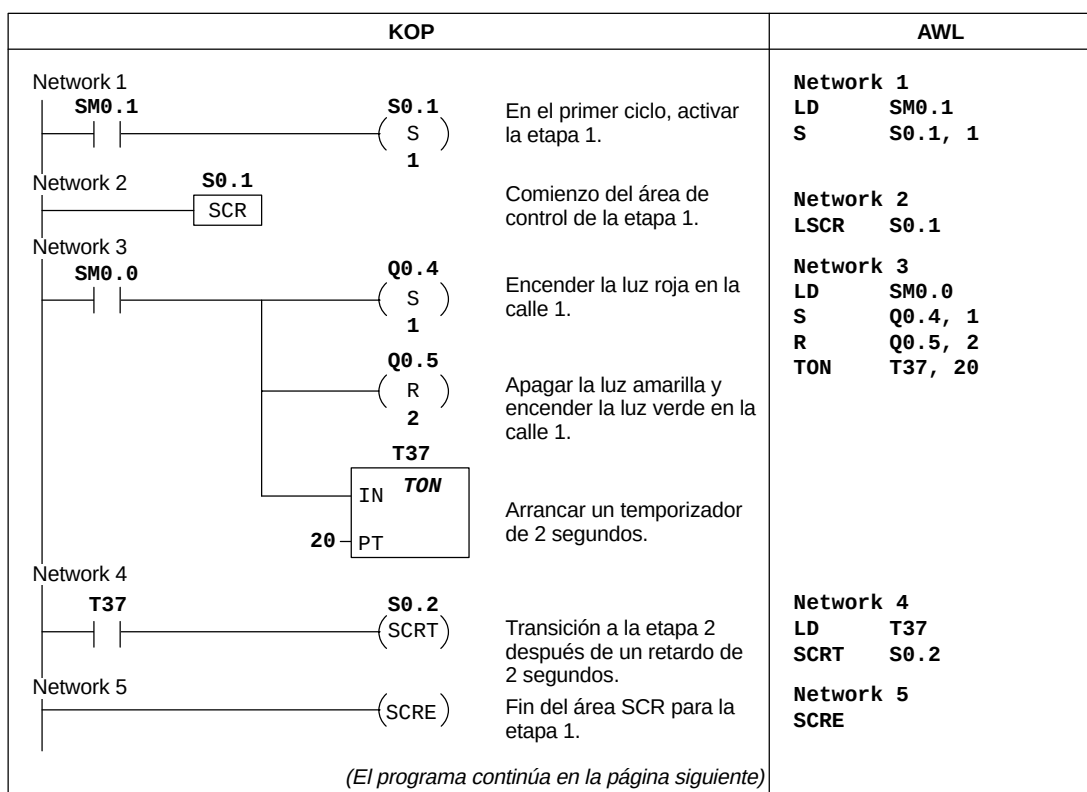


Figura 9-58 Ejemplo de una operación relé de control secuencial (SCRs) en KOP, AWL y FUP (SIMATIC)

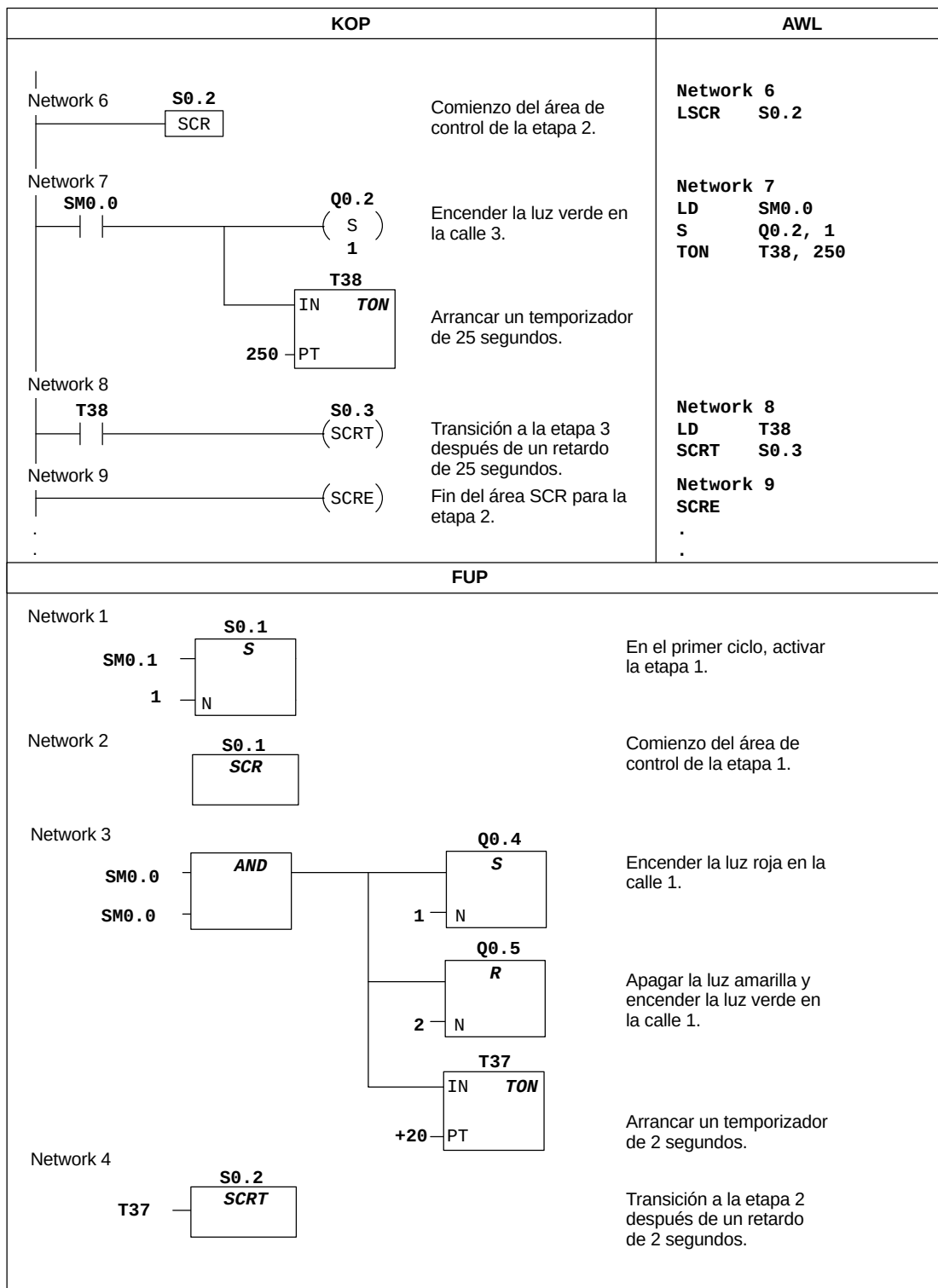


Figura 9-58 Ejemplo de una operación relé de control secuencial (SCRs) en KOP, AWL y FUP (SIMATIC) (continuación)

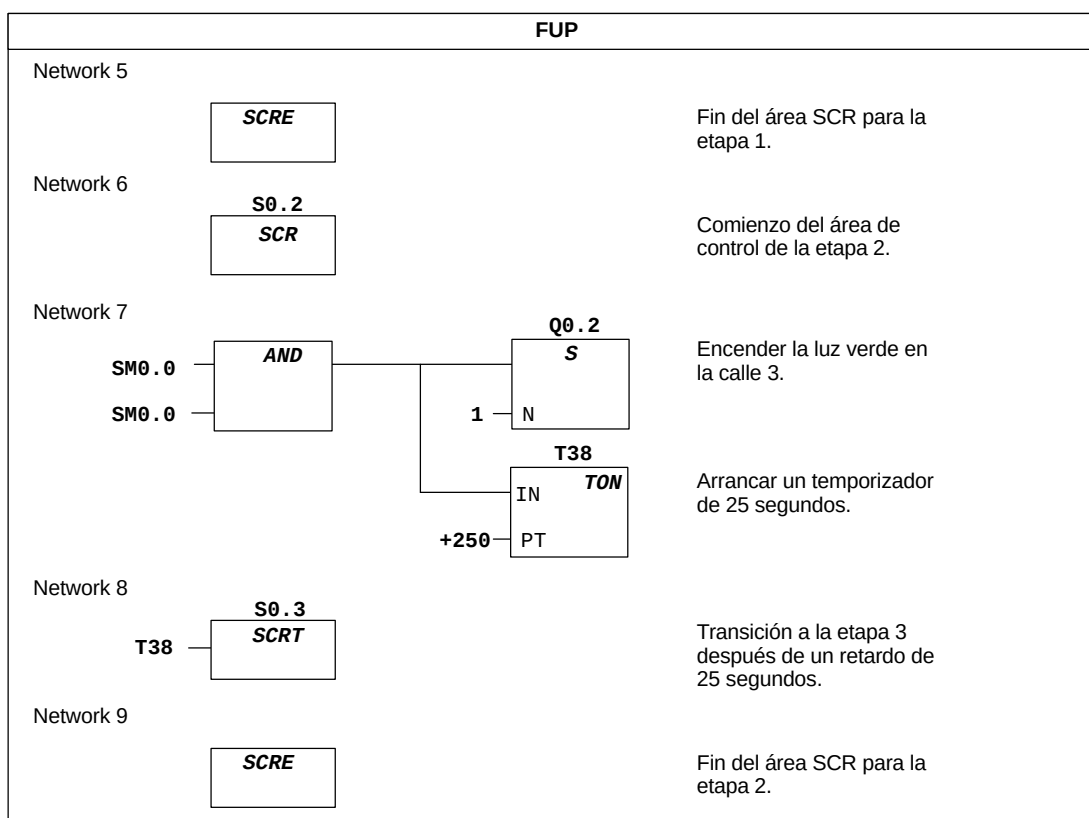


Figura 9-58 Ejemplo de una operación relé de control secuencial (SCRs) en KOP, AWL y FUP (SIMATIC) (continuación)

Dividir cadenas secuenciales

En numerosas aplicaciones es necesario dividir una cadena secuencial en dos o más cadenas. Si una cadena secuencial se divide en varias cadenas, es preciso activar simultáneamente todas las nuevas cadenas secuenciales, como muestra la figura 9-59.

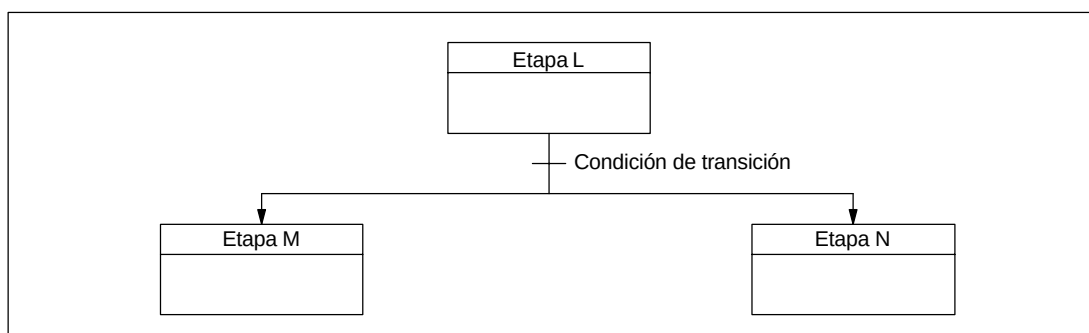


Figura 9-59 División de una cadena secuencial

La división de cadenas secuenciales se puede implementar en un programa SCR, activando varias operaciones SCRT con una misma condición de transición como muestra la figura 9-60.

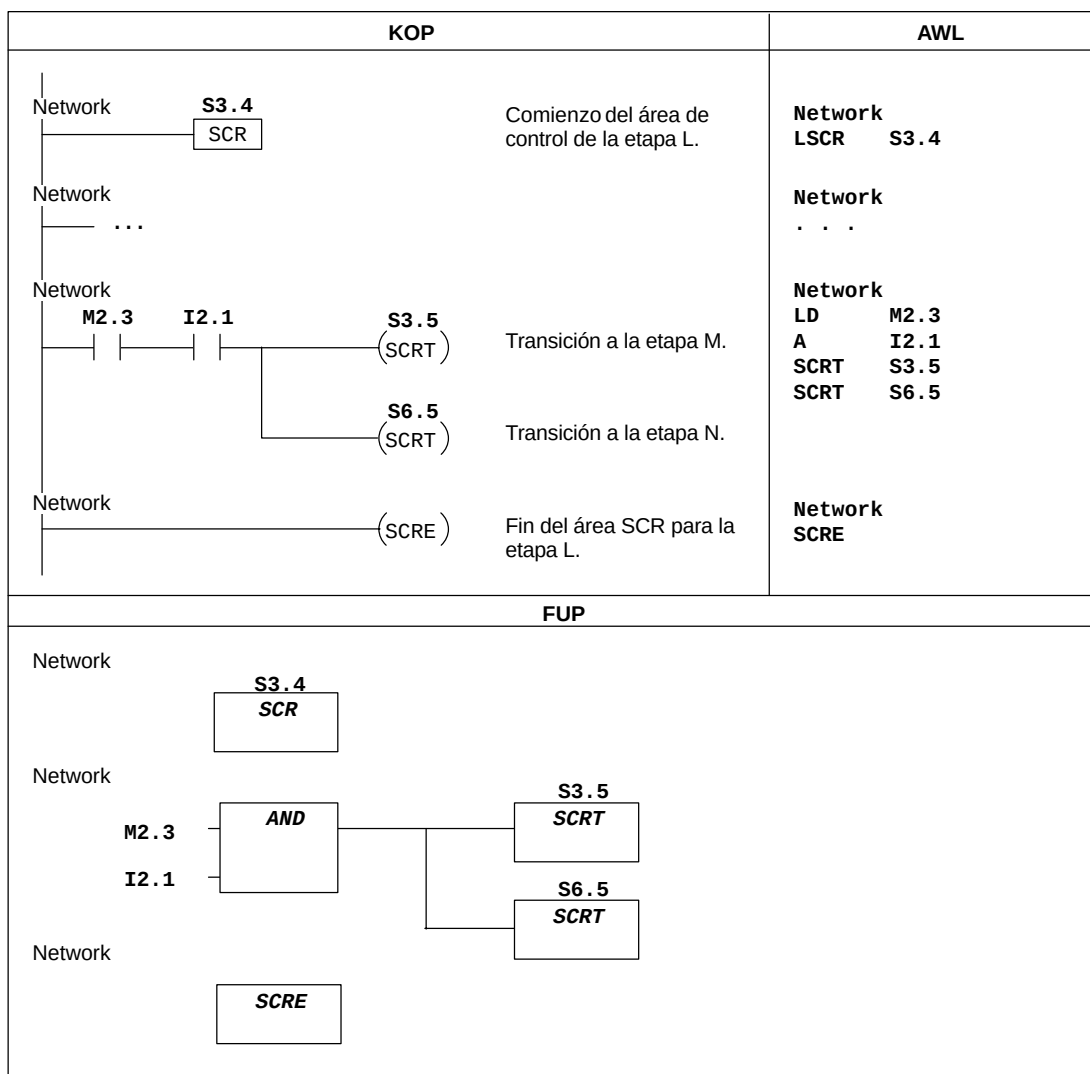


Figura 9-60 Ejemplo de la división de una cadena secuencial en KOP, AWL y FUP

Convergir cadenas secuenciales

Algo similar ocurre cuando dos o varias cadenas secuenciales deban convergir en una sola. Todas las cadenas secuenciales se deben terminar antes de poder ejecutar la siguiente etapa. La figura 9-61 muestra la convergencia de dos cadenas secuenciales.

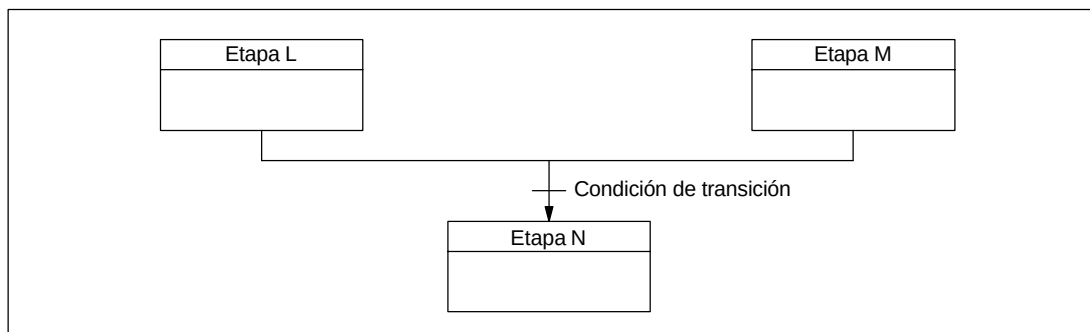


Figura 9-61 Convergenencia de cadenas secuenciales

La convergencia de cadenas secuenciales se puede implementar en un programa SCR creando una transición de la etapa L a la etapa L', y de la etapa M a la etapa M'. Si los bits SCR que representan L' y M' son verdaderos, se podrá habilitar la etapa N como muestra la figura 9-62.

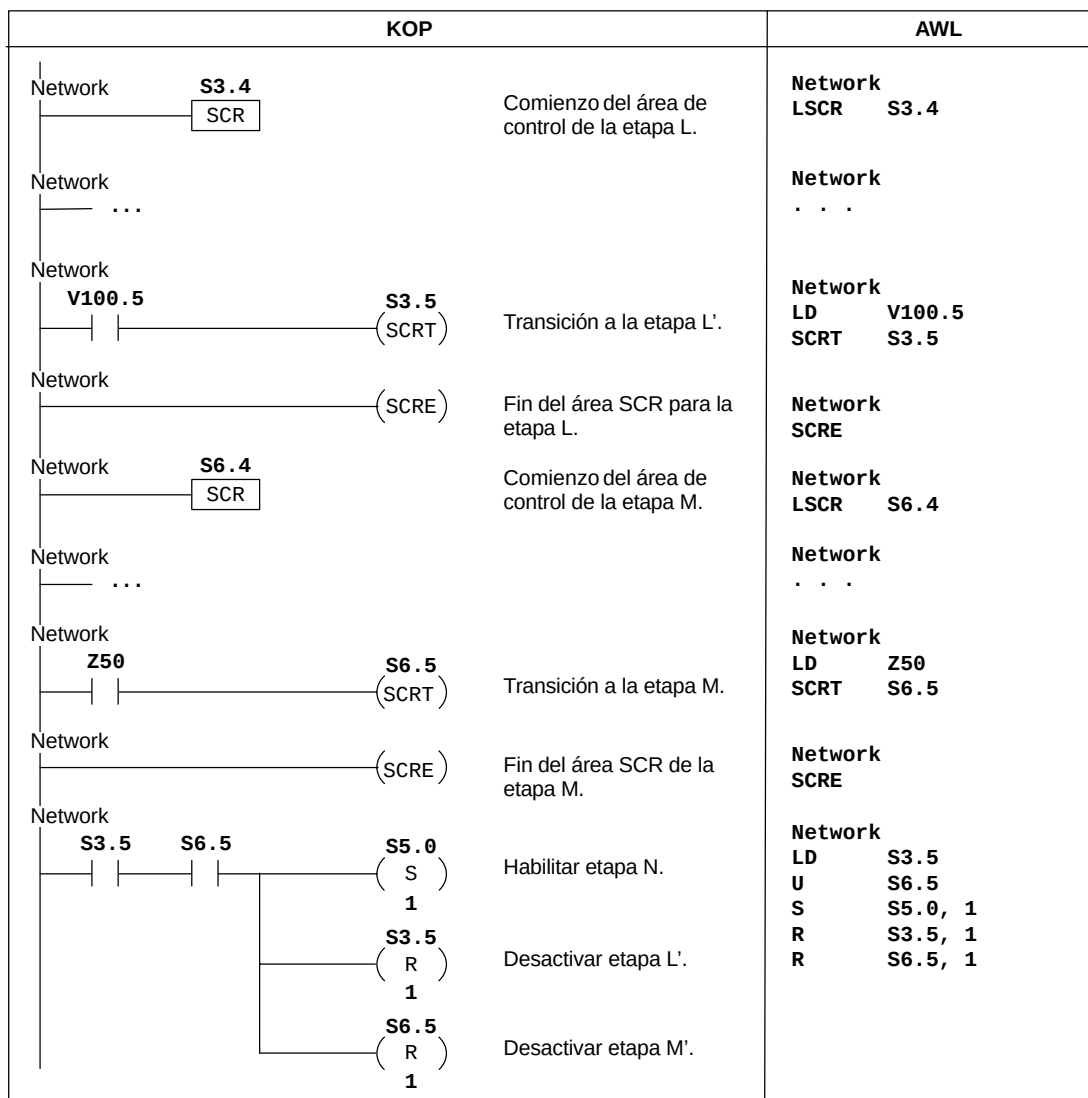


Figura 9-62 Ejemplo de convergencia de cadenas secuenciales en KOP, AWL y FUP

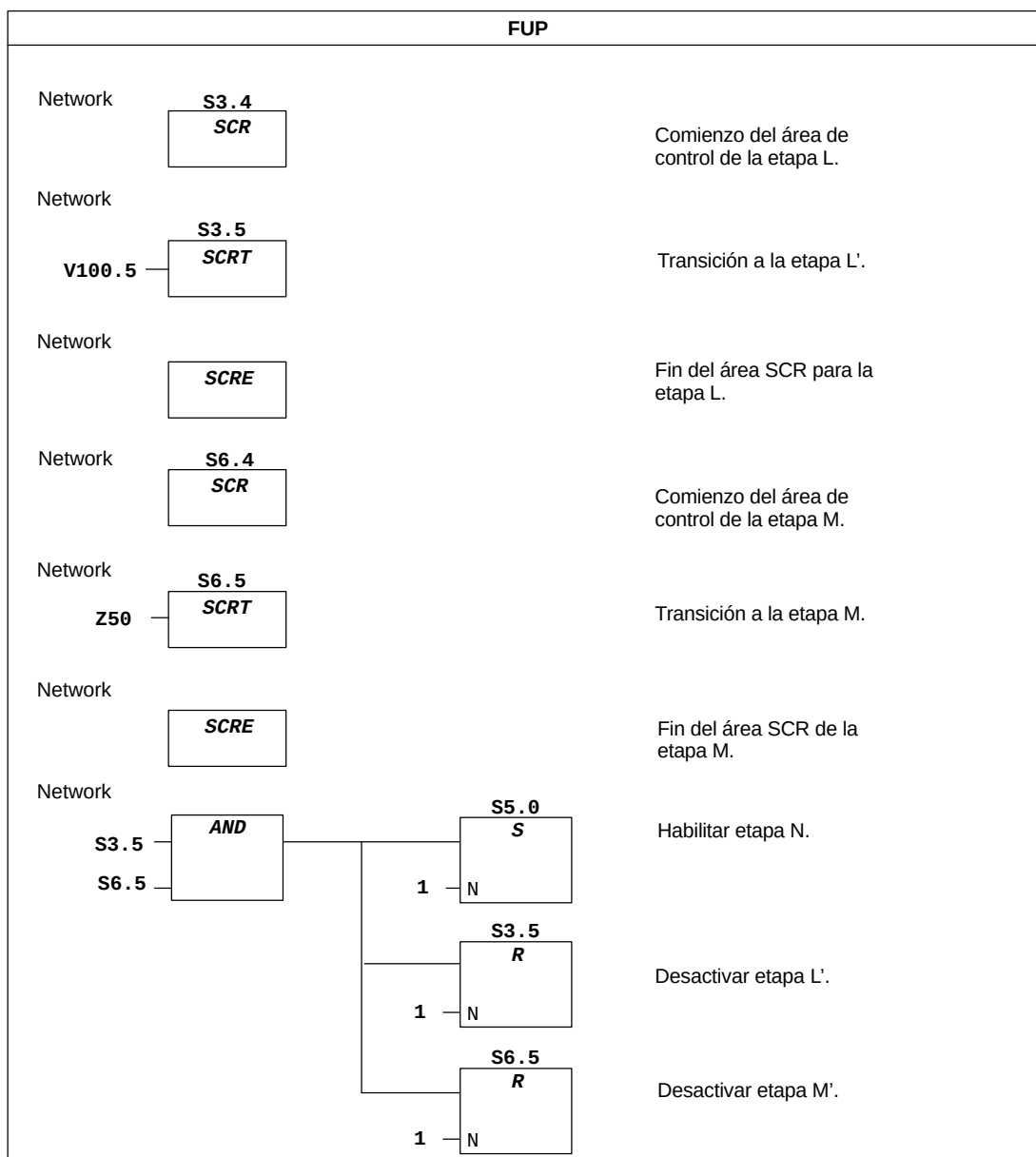


Figura 9-62 Ejemplo de convergencia de cadenas secuenciales en KOP, AWL y FUP, (continuación)

En otras situaciones, una cadena secuencial se puede dirigir a una de varias cadenas secuenciales posibles, dependiendo de la primera condición de transición que sea verdadera. La figura 9-63 muestra esta situación.

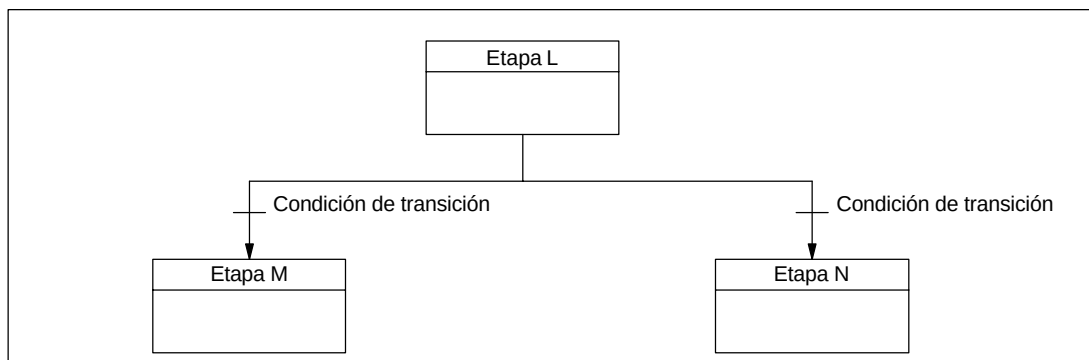


Figura 9-63 Dirigir una cadena secuencial a otra, dependiendo de la condición de transición

La figura 9-64 muestra el correspondiente programa SCR.

KOP			AWL
Network	S3.4 SCR	Comienzo del área de control de la etapa L.	Network LSCR S3.4
Network	...		Network ...
Network	M2.3 ———— S3.5 (SCRT)	Transición a la etapa M'.	Network LD M2.3 SCRT S3.5
Network	I3.3 ———— S6.5 (SCRT)	Transición a la etapa N.	Network LD I3.3 SCRT S6.5
Network	————— (SCRE)	Fin del área SCR para la etapa L.	Network SCRE

Figura 9-64 Ejemplo de transiciones condicionales en KOP, AWL y FUP

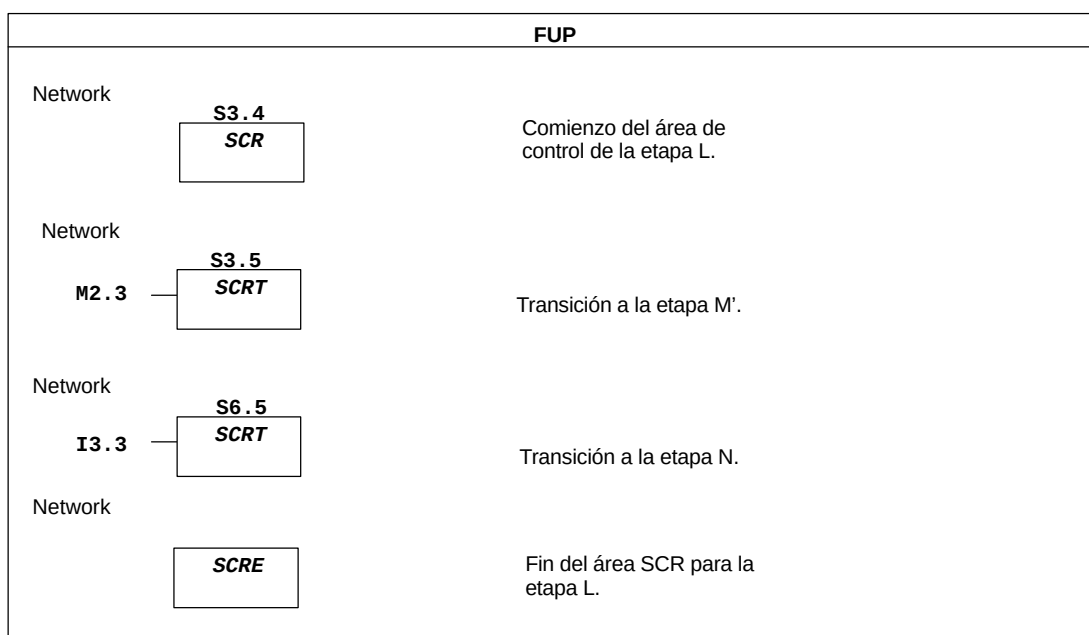
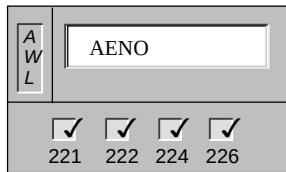


Figura 9-64 Ejemplo de transiciones condicionales en KOP, AWL y FUP (SIMATIC), (continuación)

ENO

ENO es una salida booleana para los cuadros KOP y FUP. Si la corriente fluye en un cuadro por la entrada EN y el cuadro se ejecuta sin error, la salida ENO conduce la corriente al siguiente elemento. ENO se puede utilizar como bit de habilitación para indicar que una operación se ha ejecutado correctamente.

El bit ENO se utiliza en el primer nivel de la pila para influir en la circulación de la corriente al ejecutar las operaciones posteriores.

Las operaciones AWL no tienen una entrada de habilitación (EN). El primer nivel de la pila debe ser un 1 lógico para que la operación se pueda ejecutar.

En AWL no existe la salida de habilitación (ENO), pero las instrucciones AWL correspondientes a las operaciones KOP y FUP con salidas ENO activan un bit ENO especial. A este bit se puede acceder con la operación **Y-ENO** (AENO). AENO sirve para generar el mismo efecto que el bit ENO de un cuadro. La operación AENO sólo está disponible en AWL.

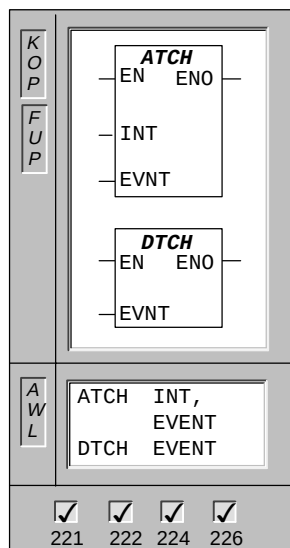
AENO combina el bit ENO y el nivel superior de la pila mediante Y. El resultado de la operación de combinación mediante Y es el nuevo valor en el nivel superior de la pila.

Operandos: Ninguno

Tipos de datos: Ninguno

9.15 Operaciones de interrupción y comunicación (SIMATIC)

Asociar interrupción, Desasociar interrupción



La operación **Asociar interrupción** asocia el número de una rutina de interrupción (INT) a un evento de interrupción (EVNT), habilitando así éste último.

La operación **Desasociar interrupción** desasocia un evento de interrupción (EVNT) de todas las rutinas de interrupción, deshabilitando así el evento.

Asociar interrupción: Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0002 (conflicto de asignación de entradas a un contador rápido).

Entradas/salidas	Operandos	Tipos de datos
INT	Constante	BYTE
EVNT	Constante (CPU 221/222: 0 – 12, 19 – 23, 27 – 33; CPU 224: 0 – 23; 27 – 33; CPU 226: 0 – 33)	BYTE

Descripción de las operaciones Asociar interrupción y Desasociar interrupción

Antes de poder llamar a una rutina de interrupción es preciso establecer un enlace entre el evento de interrupción y la parte del programa que se desee ejecutar cuando se presente el evento. La operación Asociar interrupción (ATCH) sirve para asignar el evento de interrupción (indicado por el número de evento) a una parte del programa (indicada por el número de la rutina de interrupción). También es posible asociar varios eventos de interrupción a una única rutina de interrupción. Por el contrario, no se puede asociar un sólo evento a distintas rutinas. Cuando se produce un evento estando habilitadas las interrupciones, se ejecuta únicamente la última rutina de interrupción asociada a dicho evento.

Cuando se asocia un evento a una rutina de interrupción, se habilita automáticamente el evento. Si se inhiben todos los eventos de interrupción, entonces cada vez que se presente la interrupción, se pondrá en cola de espera hasta que las interrupciones se habiliten de nuevo, utilizando para ello la operación Habilitar todos los eventos de interrupción.

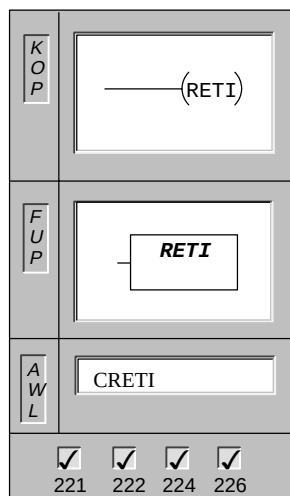
También es posible inhibir ciertos eventos de interrupción, eliminando la asociación entre el evento y la correspondiente rutina mediante la operación DTCH (Desasociar interrupción). Esta operación retorna la interrupción a un estado inactivo o ignorado.

La tabla 9-20 muestra los diferentes tipos de eventos de interrupción.

Tabla 9-20 Eventos de interrupción

Nº de evento	Descripción de la interrupción	CPU 221	CPU 222	CPU 224	CPU 226
0	Flanco positivo, I0.0	Sí	Sí	Sí	Sí
1	Flanco negativo, I0.0	Sí	Sí	Sí	Sí
2	Flanco positivo, I0.1	Sí	Sí	Sí	Sí
3	Flanco negativo, I0.1	Sí	Sí	Sí	Sí
4	Flanco positivo, I0.2	Sí	Sí	Sí	Sí
5	Flanco negativo, I0.2	Sí	Sí	Sí	Sí
6	Flanco positivo, I0.3	Sí	Sí	Sí	Sí
7	Flanco negativo, I0.3	Sí	Sí	Sí	Sí
8	Puerto 0: Recibir carácter	Sí	Sí	Sí	Sí
9	Puerto 0: Transmisión finalizada	Sí	Sí	Sí	Sí
10	Interrupción temporizada 0, SMB34	Sí	Sí	Sí	Sí
11	Interrupción temporizada 1, SMB35	Sí	Sí	Sí	Sí
12	HSC0 CV=PV (valor actual = valor predeterminado)	Sí	Sí	Sí	Sí
13	HSC1 CV=PV (valor actual = valor predeterminado)			Sí	Sí
14	HSC1 cambio de sentido			Sí	Sí
15	HSC1 puesto a 0 externamente			Sí	Sí
16	HSC2 CV=PV (valor actual = valor predeterminado)			Sí	Sí
17	HSC2 cambio de sentido			Sí	Sí
18	HSC2 puesto a 0 externamente			Sí	Sí
19	Interrupción Valor de conteo de impulsos PLS0	Sí	Sí	Sí	Sí
20	Interrupción Valor de conteo de impulsos PLS1	Sí	Sí	Sí	Sí
21	Interrupción temporizador T32 CT=PT	Sí	Sí	Sí	Sí
22	Interrupción temporizador T96 CT=PT	Sí	Sí	Sí	Sí
23	Puerto 0: Recepción de mensajes finalizada	Sí	Sí	Sí	Sí
24	Puerto 1: Recepción de mensajes finalizada				Sí
25	Puerto 1: Recibir carácter				Sí
26	Puerto 1: Transmisión finalizada				Sí
27	HSC0 cambio de sentido	Sí	Sí	Sí	Sí
28	HSC0, puesto a 0 externamente	Sí	Sí	Sí	Sí
29	HSC4 CV=PV (valor actual = valor predeterminado)	Sí	Sí	Sí	Sí
30	HSC4 cambio de sentido	Sí	Sí	Sí	Sí
31	HSC4, puesto a 0 externamente	Sí	Sí	Sí	Sí
32	HSC3 CV=PV (valor actual = valor predeterminado)	Sí	Sí	Sí	Sí
33	HSC5 CV=PV (valor actual = valor predeterminado)	Sí	Sí	Sí	Sí

Retorno condicional desde rutina de interrupción



La operación **Retorno condicional desde rutina de interrupción** finaliza una rutina en función de la combinación lógica precedente. Para añadir una interrupción, elija los comandos de menú **Edición > Insertar > Interrupción**.

Operandos: Ninguno

Tipos de datos: Ninguno

En la pantalla de STEP 7-Micro/WIN 32, los retornos desde rutinas de interrupción se visualizan en fichas por separado.

Rutinas de interrupción

La rutina de interrupción se ejecuta como respuesta a un evento interno o externo asociado. Tras haberse ejecutado la última operación de la rutina de interrupción, el control retorna al programa principal. Para salir de la rutina se puede ejecutar una operación Retorno condicional desde rutina de interrupción (CRETI).

Reglas para utilizar interrupciones

El procesamiento de interrupciones permite reaccionar rápidamente ante determinados eventos internos o externos. Las rutinas de interrupción se deben estructurar de forma que - una vez ejecutadas determinadas tareas - devuelvan el control al programa principal. A tal efecto es conveniente crear rutinas de interrupción cortas con indicaciones precisas, de manera que se puedan ejecutar rápidamente sin interrumpir otros procesos durante períodos demasiado largos. Si no se observan estas medidas, es posible que se produzcan estados imprevistos que pueden afectar a la instalación controlada por el programa principal. Al utilizar interrupciones, conviene atenerse al lema de "cuanto más breve, mejor".

Restricciones

No utilice las operaciones DISI, ENI, HDEF, LSCR y END en las rutinas de interrupción.

Soporte del sistema durante las interrupciones

Como las interrupciones pueden afectar a la lógica de contactos, bobinas y acumuladores, el sistema almacena la pila lógica, los acumuladores y las marcas especiales (SM) que indican el estado de los acumuladores y las operaciones, volviéndolos a cargar posteriormente. De este modo se previenen perturbaciones en el programa principal debidas a derivaciones a rutinas de interrupción o desde ellas.

Llamar a subrutinas desde rutinas de interrupción

Desde una rutina de interrupción se puede llamar a un nivel de anidamiento de subrutinas. Los acumuladores y la pila lógica son compartidos por la rutina de interrupción y por la subrutina invocada.

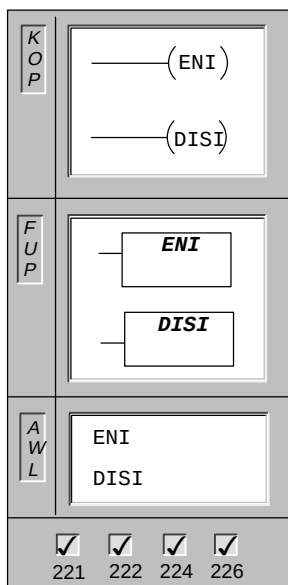
Datos compartidos por el programa principal y las rutinas de interrupción

El programa principal y una o varias rutinas de interrupción pueden compartir datos. Por ejemplo, una parte del programa principal puede suministrar datos a una rutina de interrupción o viceversa. En el caso de que el programa esté compartiendo datos, habrá que considerar también el hecho de que las rutinas de interrupción se ejecutan de forma asíncrona al programa principal. Por lo tanto, se pueden presentar en cualquier momento durante la ejecución de éste último. Los problemas de coherencia de los datos compartidos pueden ser ocasionados por las acciones de las rutinas de interrupción, al interrumpir éstas la ejecución de las operaciones del programa principal.

Hay diversas técnicas de programación que se pueden utilizar para garantizar que el programa principal y las rutinas de interrupción compartan los datos correctamente. Estas técnicas restringen la forma de acceder a las direcciones compartidas en la memoria o evitan que se interrumpan las secuencias de operaciones que utilicen direcciones compartidas.

- En un programa AWL que comparta sólo una variable: Si los datos compartidos son una sola variable en formato de byte, palabra o palabra doble, y el programa se ha escrito en AWL, los resultados intermedios de operaciones con datos compartidos sólo se podrán almacenar en direcciones o en acumuladores que no se compartan.
- En un programa KOP que comparta sólo una variable: Si los datos compartidos son una sola variable en formato de byte, palabra o palabra doble, y el programa se ha escrito en KOP, es preciso acceder a las direcciones compartidas utilizando las operaciones de transferencia (MOVB, MOVW, MOVDW, MOVR). En tanto que numerosas operaciones KOP comprenden secuencias de instrucciones AWL que se pueden interrumpir, estas operaciones de transferencia equivalen a una sola operación AWL, cuya ejecución no se ve afectada por los eventos de interrupción.
- En un programa AWL o KOP que comparta varias variables: Si los datos compartidos son varios bytes, palabras o palabras dobles contiguas, la ejecución de la rutina de interrupción se puede controlar con las operaciones Habilitar todos los eventos de interrupción (ENI) e Inhibir todos los eventos de interrupción (DISI). Las interrupciones se deben inhibir en aquel punto del programa principal donde figuran las operaciones que acceden a las direcciones compartidas. Una vez ejecutadas todas las operaciones que utilicen las direcciones compartidas, se deberán habilitar de nuevo las interrupciones. Mientras esté inhibida la interrupción no se podrá ejecutar la rutina correspondiente. Por lo tanto, no será posible acceder entonces a las direcciones compartidas. Sin embargo, esta técnica de programación puede causar que se ignoren los eventos de interrupción.

Habilitar todos los eventos de interrupción, Inhibir todos los eventos de interrupción



La operación **Habilitar todos los eventos de interrupción** habilita la ejecución de todos los eventos asociados.

La operación **Inhibir todos los eventos de interrupción** inhibe la ejecución de todos los eventos asociados.

Operandos: Ninguno

Tipos de datos: Ninguno

Cuando la CPU pasa a modo RUN, las interrupciones se inhiben. Estando en modo RUN, se pueden habilitar todos los eventos de interrupción con la operación global ENI. La operación DISI permite poner las interrupciones en cola de espera, pero no llamar a ninguna rutina de interrupción.

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0004 (intento de ejecución de operaciones ENI, DISI ó HDEF en una rutina de interrupción).

Interrupciones de comunicación

El puerto serie de comunicación del sistema de automatización se puede controlar mediante un programa KOP o AWL. La comunicación a través de este puerto se denomina modo Freeport (comunicación programable por el usuario). En modo Freeport, el programa define la velocidad de transferencia, los bits por carácter, la paridad y el protocolo. Las interrupciones de transmisión y recepción permiten controlar la comunicación mediante el programa. Para obtener más información al respecto, consulte la descripción de las operaciones Transmitir mensaje y Recibir mensaje.

Interrupciones E/S

Las interrupciones E/S abarcan interrupciones en flancos positivos y negativos, interrupciones de los contadores rápidos, así como interrupciones de la salida de impulsos. La CPU puede generar una interrupción en flancos positivos y/o negativos en una entrada. En la tabla 9-21 figuran las entradas disponibles para las interrupciones. Los eventos Flanco positivo y Flanco negativo se pueden capturar para cada una de dichas entradas. Estos eventos también sirven para indicar una condición que requiera atención inmediata en cuanto se produzca el evento.

Tabla 9-21 Interrupciones soportadas en los flancos positivos y/o negativos

Interrupciones E/S	CPU S7-200
Entradas y salidas	I0.0 a I0.3

Las interrupciones de los contadores rápidos permiten responder rápidamente a condiciones tales como: a) el valor actual alcanza el valor predeterminado, b) el sentido de conteo cambia de forma inversa al sentido de giro del árbol de accionamiento y c) el contador se pone a 0 externamente. Todos estos eventos de los contadores rápidos permiten reaccionar ante eventos que no se puedan controlar durante el tiempo de ciclo del sistema de automatización.

Las interrupciones de salida de impulsos dan un aviso inmediato cuando finaliza la salida del número indicado de impulsos. Por lo general, las salidas de impulsos se utilizan para controlar motores paso a paso.

Todas estas interrupciones se habilitan asociando una rutina de interrupción al evento E/S en cuestión.

Interrupciones temporizadas

Las interrupciones temporizadas incluyen también las de los temporizadores T32/T96. La CPU puede soportar interrupciones temporizadas. Las interrupciones temporizadas se utilizan para indicar tareas que deban ejecutarse cíclicamente. El tiempo de ciclo se incrementa en intervalos de 1 ms, abarcando desde 1 ms hasta 255 ms. El tiempo de ciclo de la interrupción temporizada 0 se debe escribir en SMB34, y el de la interrupción temporizada 1, en SMB35.

Cada vez que termina la temporización, el evento de interrupción temporizado transfiere el control a la rutina de interrupción correspondiente. Típicamente, las interrupciones temporizadas se utilizan para controlar el muestreo de las entradas analógicas en intervalos regulares o para ejecutar un bucle PID.

Asociando un evento de interrupción temporizado a una rutina de interrupción, se habilita el evento e inmediatamente se empieza a temporizar. Durante ese proceso, el sistema captura el valor del tiempo de ciclo, de forma que los cambios siguientes no lo pueden alterar. Para poder modificar el tiempo de ciclo se deberá cambiar el valor del mismo y reasociar luego la rutina de interrupción al evento de la interrupción temporizada. Al reasociar la rutina de interrupción, la función borra los tiempos acumulados de la asociación anterior, con lo cual se vuelve a temporizar a partir del nuevo valor.

Una vez habilitada, la interrupción funciona de forma continua ejecutando la rutina asociada cada vez que transcurre el intervalo de tiempo indicado. La interrupción temporizada se inhibe saliendo del modo RUN o desasociándola de la rutina correspondiente (mediante la operación DTCH). Si se ejecuta la operación Inhibir todos los eventos de interrupción, se siguen generando interrupciones temporizadas, pero se ponen en cola de espera (hasta que se habiliten nuevamente o hasta llenarse la cola). La figura 9-66 muestra un ejemplo de utilización de una interrupción temporizada.

Las interrupciones de los temporizadores T32 y T96 permiten reaccionar de forma temporizada una vez transcurrido un determinado intervalo de tiempo. Estas interrupciones se soportan únicamente en T32 y T96, siendo éstos temporizadores de retardo a la conexión (TON) con resolución de 1 ms. Por lo demás, T32 y T96 disponen de las funciones habituales. Una vez habilitada la interrupción, la rutina asociada se ejecutará cuando el valor actual del temporizador activo sea igual a su valor de preselección al actualizar la CPU el temporizador de 1 ms. Estas interrupciones se habilitan asociando una rutina de interrupción a los eventos de interrupción T32/T96.

Prioridades de las interrupciones y colas de espera

La prioridad de las interrupciones es la siguiente:

- Interrupciones de comunicación (prioridad más alta)
- Interrupciones E/S
- Interrupciones temporizadas (prioridad más baja)

La CPU procesa las interrupciones según su prioridad y después en el orden que aparecen. Sólo se ejecuta una rutina de interrupción en cada caso. Cuando se comienza a ejecutar una rutina de interrupción, se procesa hasta el final. No puede ser interrumpida por otra, ni siquiera por una rutina de mayor prioridad. Las interrupciones que aparezcan mientras se esté ejecutando otra interrupción se ponen en cola de espera para ser procesadas posteriormente.

La tabla 9-22 muestra las tres colas de espera y el número máximo de interrupciones que pueden acoger.

Tabla 9-22 Colas de espera y número máximo de interrupciones que pueden acoger

Cola de espera	CPU 221	CPU 222	CPU 224	CPU 226
Interrupciones de comunicación	4	4	4	8
Interrupciones E/S	16	16	16	16
Interrupciones temporizadas	8	8	8	8

Pueden presentarse más interrupciones de las que puede acoger la cola de espera. Por esta razón, el sistema dispone de marcas de desbordamiento que indican qué eventos de interrupción no se han podido acoger en la cola de espera. La tabla 9-23 muestra estas marcas de desbordamiento. Estas sólo se pueden utilizar en una rutina de interrupción, porque se desactivan tras vaciarse la cola de espera y reanudarse la ejecución del programa principal.

Tabla 9-23 Definiciones de las marcas especiales para el desbordamiento de las colas de espera

Descripción (0 = sin desbordamiento, 1 = desbordamiento)	Marca especial
Desbordamiento de la cola de espera de interrupciones de comunicación	SM4.0
Desbordamiento de la cola de espera de interrupciones E/S	SM4.1
Desbordamiento de la cola de espera de interrupciones temporizadas	SM4.2

La tabla 9-24 muestra las interrupciones, sus prioridades y los números de los eventos asociados.

Tabla 9-24 Prioridades de los eventos de interrupción

Nº de evento	Descripción de la interrupción	Prioridad	Prioridad de grupo
8	Puerto 0: Recibir carácter	Comunicación (más alta)	0
9	Puerto 0: Transmisión finalizada		0
23	Puerto 0: Recepción de mensajes finalizada		0
24	Puerto 1: Recepción de mensajes finalizada		1
25	Puerto 1: Recibir carácter		1
26	Puerto 1: Transmisión finalizada		1
19	PTO 0 interrupción completa	Digital (media)	0
20	PTO 1 interrupción completa		1
0	Flanco positivo, I0.0		2
2	Flanco positivo, I0.1		3
4	Flanco positivo, I0.2		4
6	Flanco positivo, I0.3		5
1	Flanco negativo, I0.0		6
3	Flanco negativo, I0.1		7
5	Flanco negativo, I0.2		8
7	Flanco negativo, I0.3		9
12	HSC0 CV=PV (valor actual = valor predeterminado)		10
27	HSC0 cambio de sentido		11
28	HSC0, puesto a 0 externamente		12
13	HSC1 CV=PV (valor actual = valor predeterminado)		13
14	HSC1, cambio de sentido		14
15	HSC1 puesto a 0 externamente		15
16	HSC2 CV=PV		16
17	HSC2 cambio de sentido		17
18	HSC2 puesto a 0 externamente		18
32	HSC3 CV=PV (valor actual = valor predeterminado)		19
29	HSC4 CV=PV (valor actual = valor predeterminado)		20
30	HSC4 cambio de sentido		21
31	HSC4, puesto a 0 externamente		22
33	HSC5 CV=PV (valor actual = valor predeterminado)		23
10	Interrupción temporizada 0	Temporizada (más baja)	0
11	Interrupción temporizada 1		1
21	Interrupción temporizador T32 CT=PT		2
22	Interrupción temporizador T96 CT=PT		3

Ejemplo de interrupciones

La figura 9-65 muestra un ejemplo de operaciones con rutinas de interrupción.

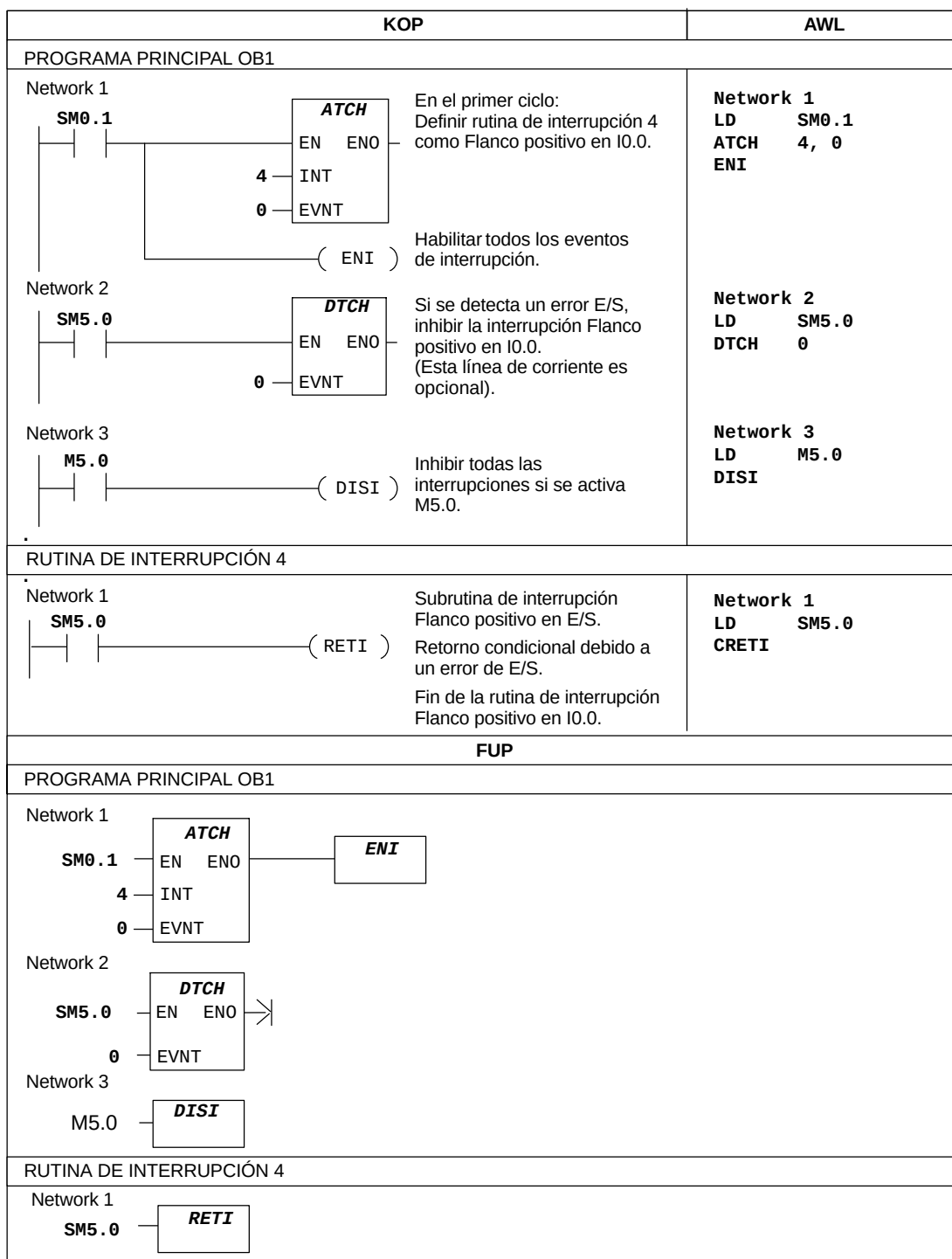


Figura 9-65 Ejemplo de operaciones de interrupción en KOP, AWL y FUP (SIMATIC)

La figura 9-66 muestra cómo leer el valor de una entrada analógica mediante una interrupción temporizada.

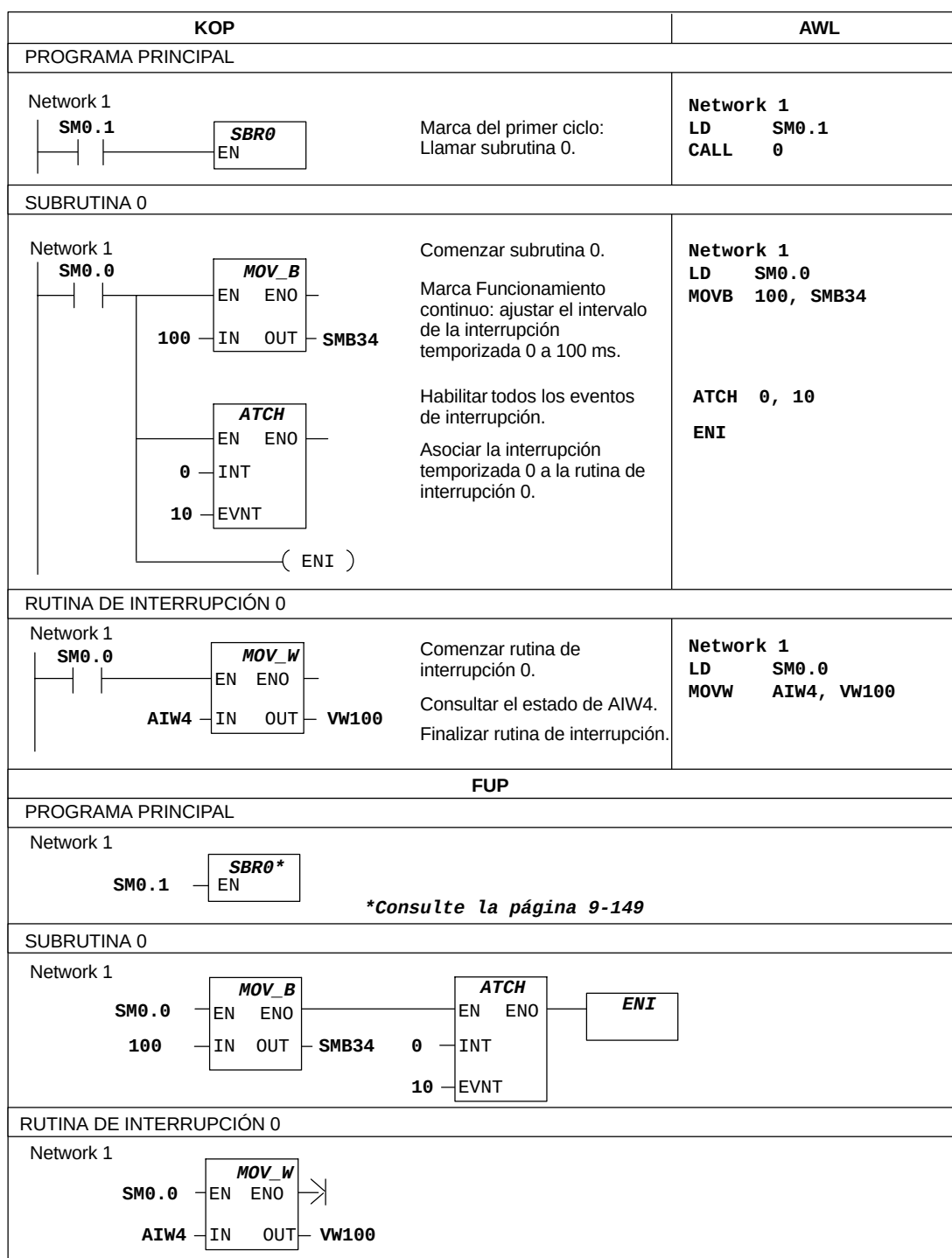
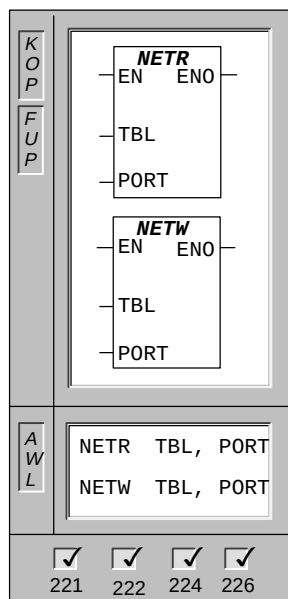


Figura 9-66 Ejemplo de lectura de una entrada analógica mediante una interrupción temporizada

Leer de la red, Escribir en la red



La operación **Leer de la red** inicia una comunicación para registrar datos de una estación remota a través del puerto indicado (PORT), según se define en la tabla (TBL).

La operación **Escribir en la red** inicia una comunicación para escribir datos en una estación remota a través del puerto indicado (PORT), según se define en la tabla (TBL).

Con la operación NETR se pueden leer hasta 16 bytes de información de una estación remota, en tanto que con la operación NETW se pueden escribir hasta 16 bytes de información en una estación remota. El programa puede contener un número cualquiera de operaciones NETR/NETW, pero sólo ocho de ellas (en total) pueden estar activadas simultáneamente. Por ejemplo, pueden estar activadas cuatro operaciones NETR y cuatro NETW, o bien, dos operaciones NETR y seis NETW en un sistema de automatización S7-200.

La figura 9-67 muestra la tabla a la que hace referencia el parámetro TBL en las operaciones NETR y NETW.

NETR: Condiciones de error que ponen ENO a 0:
SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

NETW: Condiciones de error que ponen ENO a 0:
SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Entradas/salidas	Operandos	Tipos de datos
TBL	VB, MB, *VD, *AC, *LD	BYTE
PORT	Constante	BYTE

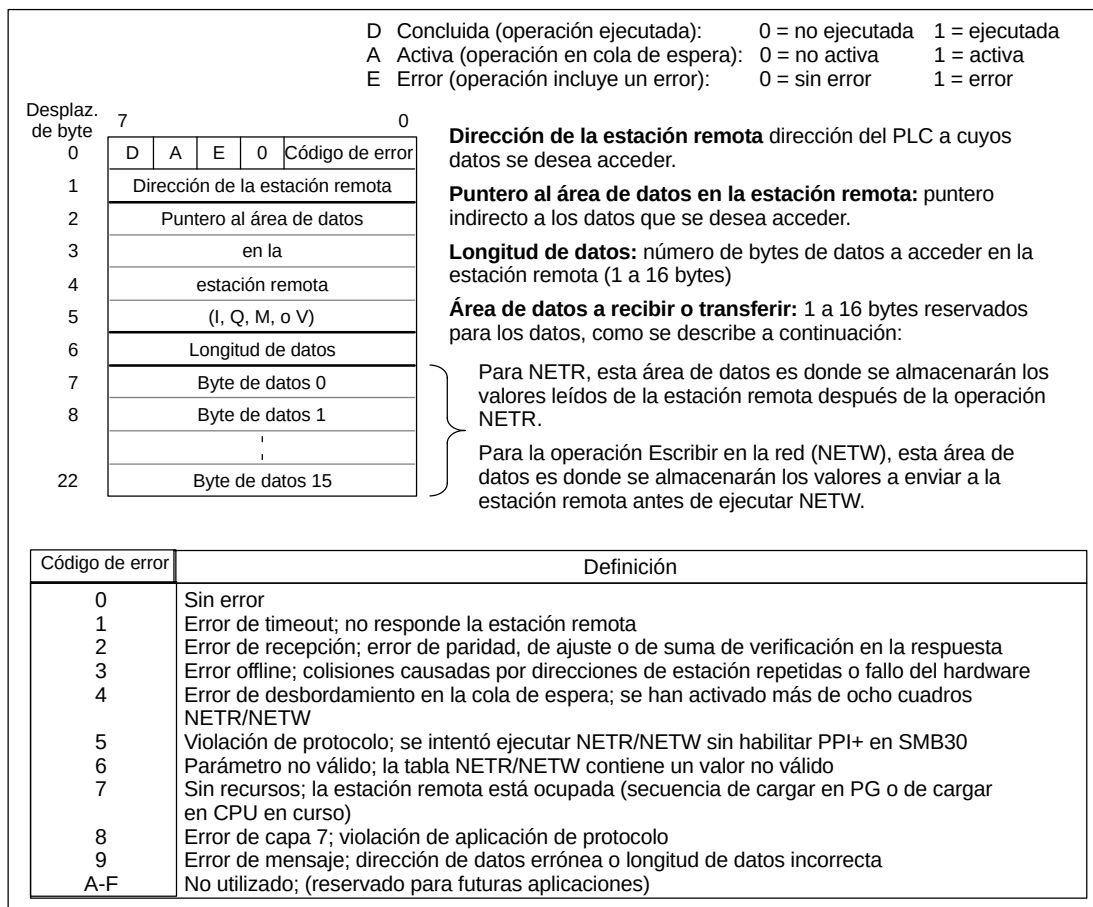


Figura 9-67 Definición de TABLE para NETR y NETW

Ejemplo de las operaciones Leer de la red y Escribir en la red

La figura 9-68 muestra un ejemplo para ilustrar la utilidad de las operaciones NETR (Leer de la red) y NETW (Escribir en la red). Se trata de un ejemplo de una línea de producción donde se están llenando paquetes de mantequilla que se envían a una de las cuatro máquinas empaquetadoras. La empaquetadora embala ocho paquetes de mantequilla en cada caja. Una máquina distribuidora controla el flujo de los paquetes de mantequilla hacia cada una de las empaquetadoras. Se utilizan cuatro CPUs 221 para controlar las empaquetadoras y una CPU 222 equipada con un visualizador de textos TD 200 para controlar a la distribuidora. La figura 9-68 muestra la configuración de la red.

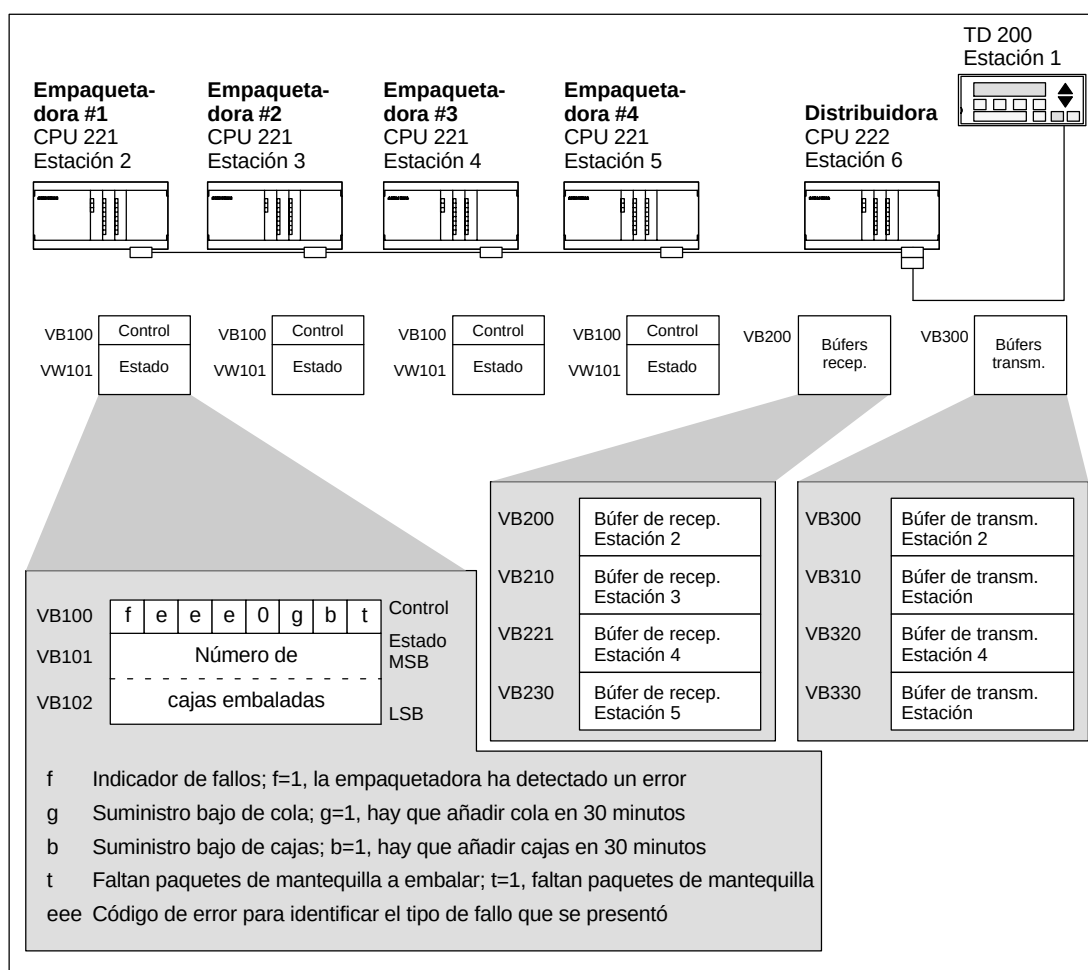


Figura 9-68 Ejemplo de las operaciones NETR y NETW

La figura 9-69 muestra detalladamente los búfers de recepción y transmisión para acceder a los datos de la estación 2 (en VB200 y VB300, respectivamente).

La CPU 224 utiliza la operación NETR para leer continuamente el control y el estado de cada una de las empaquetadoras. Cada vez que una empaquetadora ha embalado 100 cajas, la máquina distribuidora lo registra y envía un mensaje para borrar la palabra de estado utilizando una operación NETW.

La figura 9-70 muestra el programa para leer el byte de control, el número de cajas embaladas y para poner a 0 el número de cajas embalado por una sola empaquetadora (empaquetadora nº 1).

Búfer de recepción de la distribuidora para leer la empaquetadora nº 1						Búfer de transmisión de la distribuidora para borrar el conteo de la empaquetadora nº 1					
	7				0		7				0
VB200	D	A	E	0	Código de error	VB300	D	A	E	0	Código de error
VB201	Dirección de la estación remota					VB301	Dirección de la estación remota				
VB202	Puntero al					VB302	Puntero al				
VB203	área de datos					VB303	área de datos				
VB204	en la					VB304	en la				
VB205	estación remota = (&VB100)					VB305	estación remota = (&VB101)				
VB206	Longitud de datos = 3 bytes					VB306	Longitud de datos = 2 bytes				
VB207	Control					VB307	0				
VB208	Estado (MSB)					VB308	0				
VB209	Estado (LSB)										

Figura 9-69 Ejemplo de TABLE para NETR y NETW

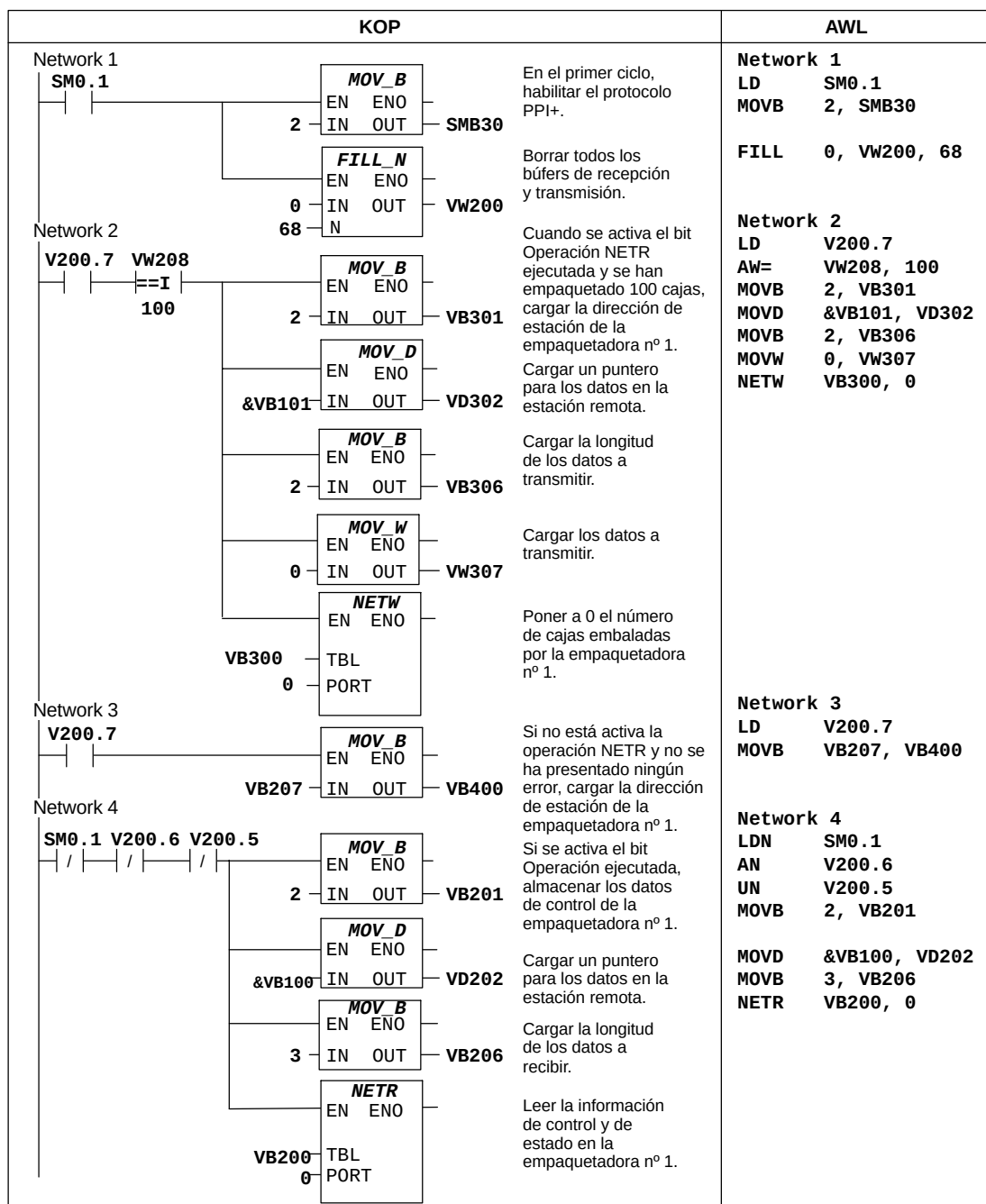


Figura 9-70 Ejemplo de las operaciones NETR y NETW en KOP y AWL (SIMATIC)

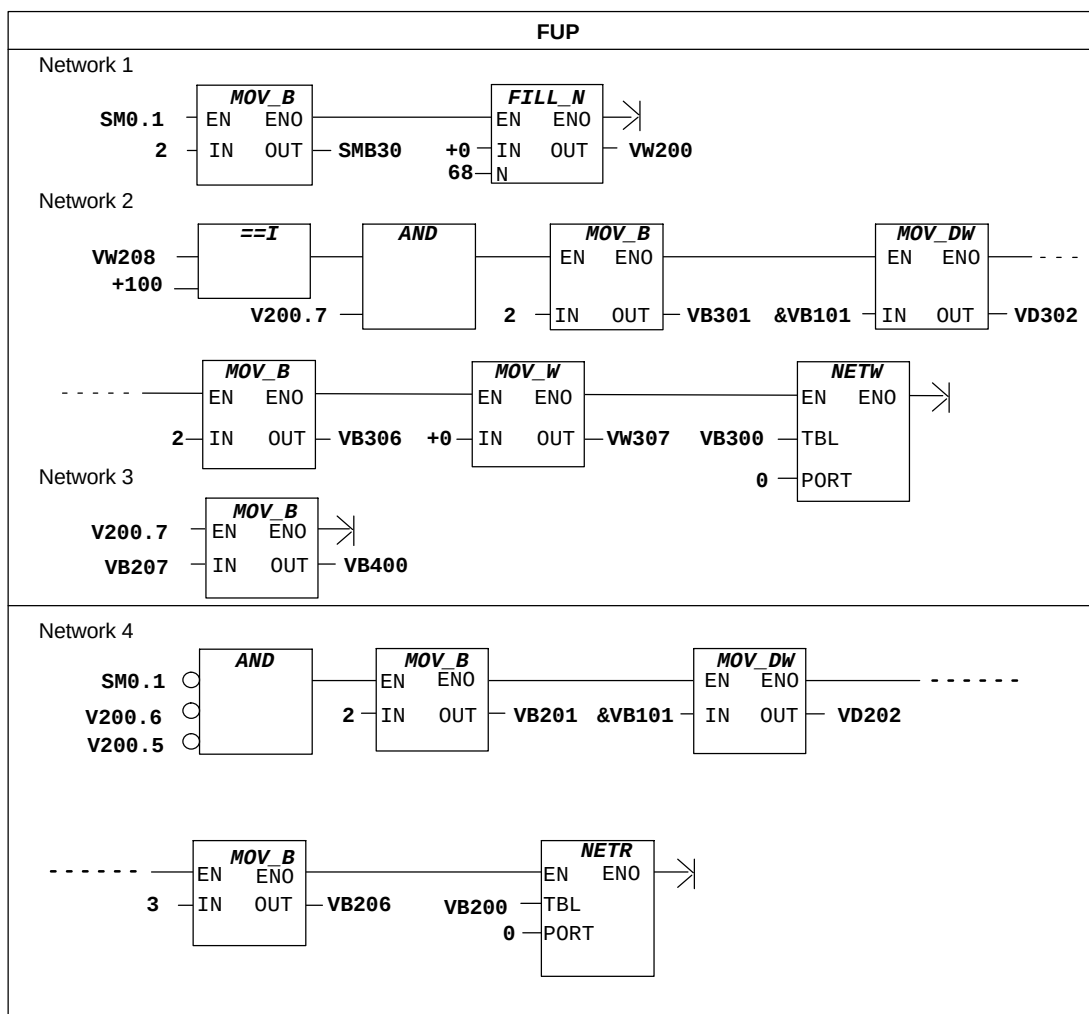
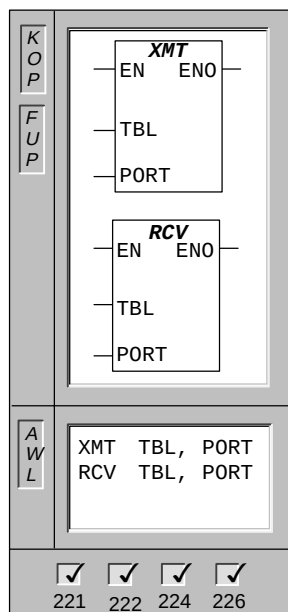


Figura 9-71 Ejemplo de las operaciones NETR y NETW en FUP (SIMATIC)

Transmitir mensaje, Recibir mensaje



La operación **Transmitir mensaje** activa la transmisión del búfer de datos (TBL). La primera entrada del búfer indica cuántos bytes se han de transmitir. PORT indica el puerto de programación por donde se va a transmitir.

La operación XMT se utiliza en modo Freeport para transmitir datos por el (los) puerto(s) de comunicación.

El formato del búfer XMT buffer es el siguiente:

La operación **Recibir mensaje** inicia o finaliza la función Recibir mensaje. Para el cuadro Recibir mensaje es preciso indicar una condición inicial y final. Los mensajes que se hayan recibido a través del puerto indicado (PORT) se almacenan en el búfer de datos (TBL). La primera entrada del búfer indica el número de bytes que se han recibido.

Transmitir mensaje: Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto), 0009 (XMT/RCV simultáneos en el puerto 0), 000B (XMT/RCV simultáneos en el puerto 1)

La CPU no está en modo Freeport.

Recibir mensaje: Condiciones de error que ponen ENO a 0: SM86.6 y SM186.6 (error de parámetro RCV), SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto), 0009 (XMT/RCV simultáneos en el puerto 0), 000B (XMT/RCV simultáneos en el puerto 1), la CPU no está en modo Freeport.

Entradas/salidas	Operandos	Tipos de datos
TABLE	VB, IB, QB, MB, SB, SMB, *VD, *AC, *LD	BYTE
PORT	Constante (0 para CPU 221, CPU 222, CPU 224; 0 ó 1 para CPU 226)	BYTE

Modo Freeport

El programa de usuario puede controlar el puerto serie de la CPU. La comunicación a través de este puerto se denomina modo Freeport (comunicación programable por el usuario). Eligiendo el modo Freeport, el programa KOP controla el puerto de comunicación utilizando interrupciones de recepción y de transmisión, así como las operaciones Transmitir mensaje (XMT) y Recibir mensaje (RCV). En modo Freeport, el programa KOP controla todo el protocolo de comunicación. Las marcas especiales SMB30 (para el puerto 0) y SMB130 (para el puerto 1, si la CPU dispone de dos puertos) se utilizan para elegir la velocidad de transferencia y la paridad.

Cuando la CPU pasa a modo STOP se inhibe el modo Freeport y se restablece la comunicación (p.ej. acceso a través de la unidad de programación).

En el caso más simple se puede enviar un mensaje a la impresora o a la pantalla con sólo utilizar la operación Transmitir mensaje (XMT). Otros ejemplos incluyen la conexión a un lector de código de barras, una báscula o una soldadora. En todo caso, el programa deberá soportar el protocolo con el que la CPU se comunica en modo Freeport.

Para poder utilizar el modo Freeport, es preciso que la CPU esté en modo RUN. El modo Freeport se habilita ajustando el valor 01 en el campo de selección del protocolo de SMB30 (puerto 0) o de SMB130 (puerto 1). Estando en modo Freeport, la CPU no se puede comunicar con la unidad de programación.

Nota

La conmutación a modo Freeport se puede controlar con la marca especial SM0.7 que indica la posición actual del selector de modos de operación. Si SM0.7 = 0, el selector está en posición TERM; si SM0.7 = 1, el selector está en posición RUN. Si el modo Freeport se habilita sólo cuando el selector esté en RUN, la unidad de programación se podrá utilizar para vigilar o controlar el funcionamiento de la CPU, cambiando el selector a una posición diferente.

Inicializar el modo Freeport

SMB30 y SMB130 se utilizan para inicializar el modo Freeport en los puertos de comunicación 0 y 1, respectivamente, permitiendo elegir la velocidad de transferencia, la paridad y el número de bits por carácter. La tabla 9-25 muestra los bytes de control del modo Freeport.

Tabla 9-25 Bytes de marcas especiales SMB30 y SMB130

Puerto 0	Puerto 1	Descripción
Formato de SMB30	Formato de SMB130	MSB 7 p p d b b b m m LSB 0 Byte de control del modo Freeport
SM30.6 y SM30.7	SM130.6 y SM130.7	pp: Selección de paridad 00 = sin paridad 01 = paridad par 10 = sin paridad 11 = paridad impar
SM30.5	SM130.5	d: Bits por carácter 0 = 8 bits por carácter 1 = 7 bits por carácter
SM30.2 a SM30.4	SM130.2 a SM130.4	bbb: Velocidad de transferencia 000 = 38.400 bit/s 001 = 19.200 bit/s 010 = 9.600 bit/s 011 = 4.800 bit/s 100 = 2.400 bit/s 101 = 1.200 bit/s 110 = 600 bit/s 111 = 300 bit/s
SM30.0 y SM30.1	SM130.0 y SM130.1	mm: Selección de protocolo 00 = Protocolo de interface punto a punto (PPI/modo esclavo) 01 = Protocolo Freeport 10 = PPI/modo maestro 11 = Reservado (estándar: PPI/modo esclavo)
Nota: Se genera un bit de parada para todas las configuraciones.		

Utilizar la operación XMT para transmitir datos

Con la operación XMT se puede enviar un búfer de uno o más caracteres (hasta un máximo de 255). Una vez transmitido el último carácter del búfer, se genera una interrupción (evento de interrupción 9 para el puerto 0 y evento de interrupción 26 para el puerto 1), si una rutina de interrupción se ha asociado al evento Transmisión finalizada. También es posible transmitir datos sin utilizar interrupciones (p.ej. enviar un mensaje a una impresora), vigilando SM4.5 ó SM4.6 hasta que finalice la transmisión.

La operación XMT sirve para generar una condición BREAK, poniendo el número de caracteres a cero y ejecutando luego la operación XMT. Así se genera una condición BREAK en la línea de temporizadores de 16 bits a la velocidad de transferencia actual. La transmisión de una condición BREAK se gestiona de la misma forma que la de cualquier otro mensaje. Una interrupción de transmisión se genera cuando se termina de transmitir la condición BREAK, indicando SM4.5 ó SM4.6 el estado actual de la transmisión.

La figura 9-72 muestra el formato del búfer XMT.

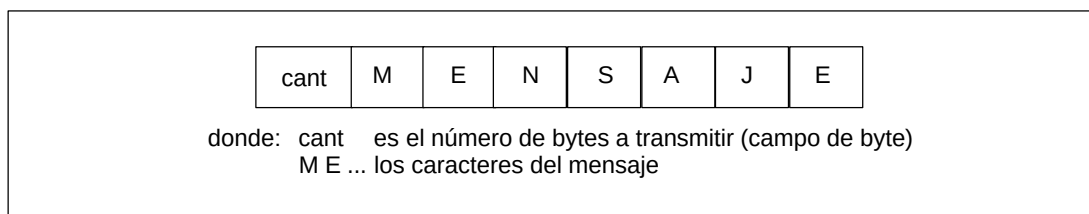


Figura 9-72 Formato del búfer XMT

Utilizar la operación RCV para recibir datos

Con la operación RCV se puede recibir un búfer de uno o más caracteres (hasta un máximo de 255). Una vez recibido el último carácter del búfer, se genera una interrupción (evento de interrupción 23 para el puerto 0 y evento de interrupción 24 para el puerto 1), si una rutina de interrupción se ha asociado al evento Recepción de mensajes finalizada.

También es posible recibir mensajes sin utilizar interrupciones, vigilando para ello la marca especial SMB86. SMB86 (o SMB186) no serán iguales a cero cuando el cuadro RCV esté inactivo o cuando se haya finalizado. En cambio, será igual a cero cuando se estén recibiendo datos.

La operación RCV permite seleccionar las condiciones para el comienzo y el final de un mensaje. En la tabla 9-26 (SM86 a SM94 para el puerto 0, y SM186 a SM194 para el puerto 1) se describen las condiciones de comienzo y de final de mensajes. La figura muestra el formato del búfer RCV. La figura 9-73 muestra el formato del búfer XMT.

Nota

La recepción de mensajes se finalizará automáticamente si se produce un desbordamiento o un error de paridad. Para la operación Recibir mensaje es preciso definir una condición inicial (x ó z) y una condición final (y, t ó el número máximo de caracteres).

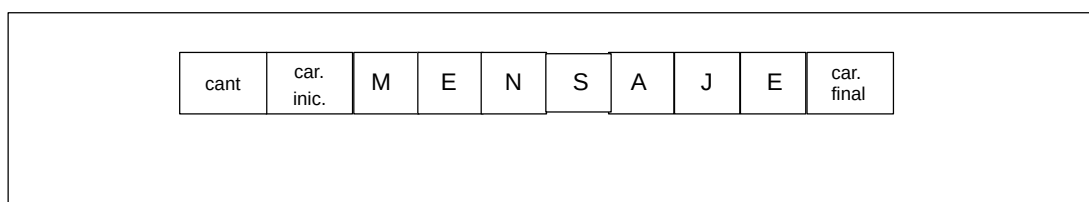


Figura 9-73 Formato del búfer RCV

Tabla 9-26 Marcas especiales SMB86 a SMB94 y SMB186 a SMB194

Puerto 0	Puerto 1	Descripción
SMB86	SMB186	MSB 7 LSB 0 n r e 0 0 t c p Byte de estado de recepción de mensajes n: 1 = El usuario ha inhibido la recepción de mensajes r: 1 = Se ha finalizado la recepción de mensajes: error en parámetros de entrada o falta condición inicial o final e: 1 = Carácter final recibido. t: 1 = Se ha finalizado la recepción de mensajes: ha transcurrido la temporización. c: 1 = Se ha finalizado la recepción de mensajes: se ha excedido el número máximo de caracteres. p: 1 = Se ha finalizado la recepción de mensajes debido a un error de paridad

Tabla 9-26 Marcas especiales SMB86 a SMB94 y SMB186 a SMB194 (continuación)

Puerto 0	Puerto 1	Descripción								
SMB87	SMB187	MSB 7 LSB 0 <table><tr><td>en</td><td>sc</td><td>ec</td><td>il</td><td>c/m</td><td>tmr</td><td>bk</td><td>0</td></tr></table> Byte de control de recepción de mensajes en: 0 = Inhibida la función de recibir mensajes. 1 = Habilitada la función de recibir mensajes. El bit para habilitar/inhibir la recepción de mensajes se comprueba cada vez que se ejecuta la operación RCV. sc: 0 = Ignorar SMB88 o SMB188. 1 = Utilizar el valor de SMB88 o de SMB188 para detectar el comienzo del mensaje. ec: 0 = Ignorar SMB89 o SMB189. 1 = Utilizar el valor de SMB89 o de SMB189 para detectar el final del mensaje. il: 0 = Ignorar SMB90 o SMB190. 1 = Utilizar el valor de SMW90 para detectar una condición de inactividad. c/m: 0 = Utilizar el temporizador como temporizador entre caracteres. 1= Utilizar el temporizador como temporizador de mensajes. tmr: 0 = Ignorar SMB92 o SMB192. 1 = Finalizar la recepción si se excede el período de tiempo indicado en SMW92 o SMW192. bk: 0 = Ignorar condiciones BREAK; 1 = Utilizar condición BREAK como comienzo de la detección de mensajes. Las marcas del byte de control de interrupción de mensajes se utilizan para definir los criterios con los que se identifica el mensaje. Se definen los criterios tanto de comienzo como de final del mensaje. Para determinar el comienzo de un mensaje, uno de los dos juegos de criterios de comienzo de mensaje combinados lógicamente mediante Y deberán ser verdaderos y deberán ocurrir en secuencia (inactividad seguida de un comienzo de carácter o condición BREAK seguida de un comienzo de carácter). Para determinar el fin de un mensaje, los criterios habilitados se combinan mediante O. A continuación se indican las ecuaciones de comienzo y de final: Comienzo del mensaje = $il * sc + bk * sc$ Final del mensaje = $ec + tmr + \text{contaje máximo de caracteres alcanzado}$ Programar los criterios de comienzo de mensaje para: 1. Detección de inactividad: il=1, sc=0, bk=0, SMW90>0 2. Detección de carácter inicial: il=0, sc=1, bk=0, SMW90 no es relevante 3. Detección BREAK: il=0, sc=1, bk=1, SMW90 no es relevante 4. Cualquier respuesta a una petición: il=1, sc=0, bk=0, SMW90=0 (El temporizador de mensajes sirve para terminar la recepción si no hay respuesta). 5. BREAK y carácter inicial: il=0, sc=1, bk=1, SMW90 no es relevante 6. Inactividad y carácter inicial: il=1, sc=1, bk=0, SMW90 >0 7. Inactividad y carácter inicial (no válido): il=1, sc=1, bk=0, SMW90=0 Nota: La recepción se finalizará automáticamente si ocurre un error de desbordamiento o de paridad (si se han habilitado).	en	sc	ec	il	c/m	tmr	bk	0
en	sc	ec	il	c/m	tmr	bk	0			
SMB88	SMB188	Carácter de comienzo del mensaje.								

Tabla 9-26 Marcas especiales SMB86 a SMB94 y SMB186 a SMB194 (continuación)

Puerto 0	Puerto 1	Descripción
SMB89	SMB189	Carácter de fin del mensaje.
SMB90 SMB91	SMB190 SMB191	Tiempo de línea de inactividad en milisegundos. El primer carácter recibido una vez transcurrido el tiempo de línea de inactividad es el comienzo del nuevo mensaje. SM90 (o SM190) es el byte más significativo y SM91 (o SM191) es el byte menos significativo.
SMB92 SMB93	SMB192 SMB193	Vigilancia de tiempo del temporizador entre caracteres/mensajes en milisegundos. Si se excede el tiempo, se finaliza la recepción de mensajes. SM92 (o SM192) es el byte más significativo y SM93 (o SM193) es el byte menos significativo.
SMB94	SMB194	Número máximo de caracteres a recibir (1 a 255 bytes). Nota: Este margen debe ajustarse al tamaño máximo esperado para el búfer, incluso si no se utiliza la finalización de mensajes por el conteo de caracteres.

Recibir datos mediante interrupciones de caracteres

Para disponer de una mayor flexibilidad en los protocolos soportados, los datos se pueden recibir también de forma controlada por interrupciones de caracteres. Cada carácter recibido genera una interrupción. El carácter recibido se deposita en SMB2 y el estado de la paridad (si se ha habilitado) se deposita en SM3.0. Ello sucede inmediatamente antes de ejecutarse la rutina de interrupción asociada al evento Recibir carácter.

- SMB2 es el búfer de recepción de caracteres en modo Freeport. Cada carácter recibido en modo Freeport se deposita en esta dirección para que el programa de usuario pueda acceder rápidamente a los valores.
- SMB3 se utiliza para el modo Freeport y contiene un bit de error de paridad que se activa si se detecta un error de ese tipo en un carácter recibido. Todos los demás bits del byte se reservan. Utilice este bit para rechazar el mensaje o para generar un acuse negativo del mensaje.

Nota

SMB2 y SMB3 son compartidos por los puertos 0 y 1. Si debido a la recepción de un carácter por el puerto 0 se ejecuta la rutina de interrupción asociada a ese evento (evento de interrupción 8), SMB2 contendrá el carácter recibido por el puerto 0, en tanto que SMB3 contendrá la paridad de dicho carácter. Si debido a la recepción de un carácter por el puerto 1 se ejecuta la rutina de interrupción asociada a ese evento (evento de interrupción 25), SMB2 contendrá el carácter recibido por el puerto 1, en tanto que SMB3 contendrá la paridad de dicho carácter.

Ejemplo de las operaciones Recibir mensaje y Transmitir mensaje

Este programa de ejemplo (fig. 9-74) muestra la utilización de las operaciones Recibir mensaje y Transmitir mensaje. El programa recibirá una cadena de caracteres hasta que se reciba un carácter que indique un cambio de línea. El mensaje se retornará entonces al emisor.

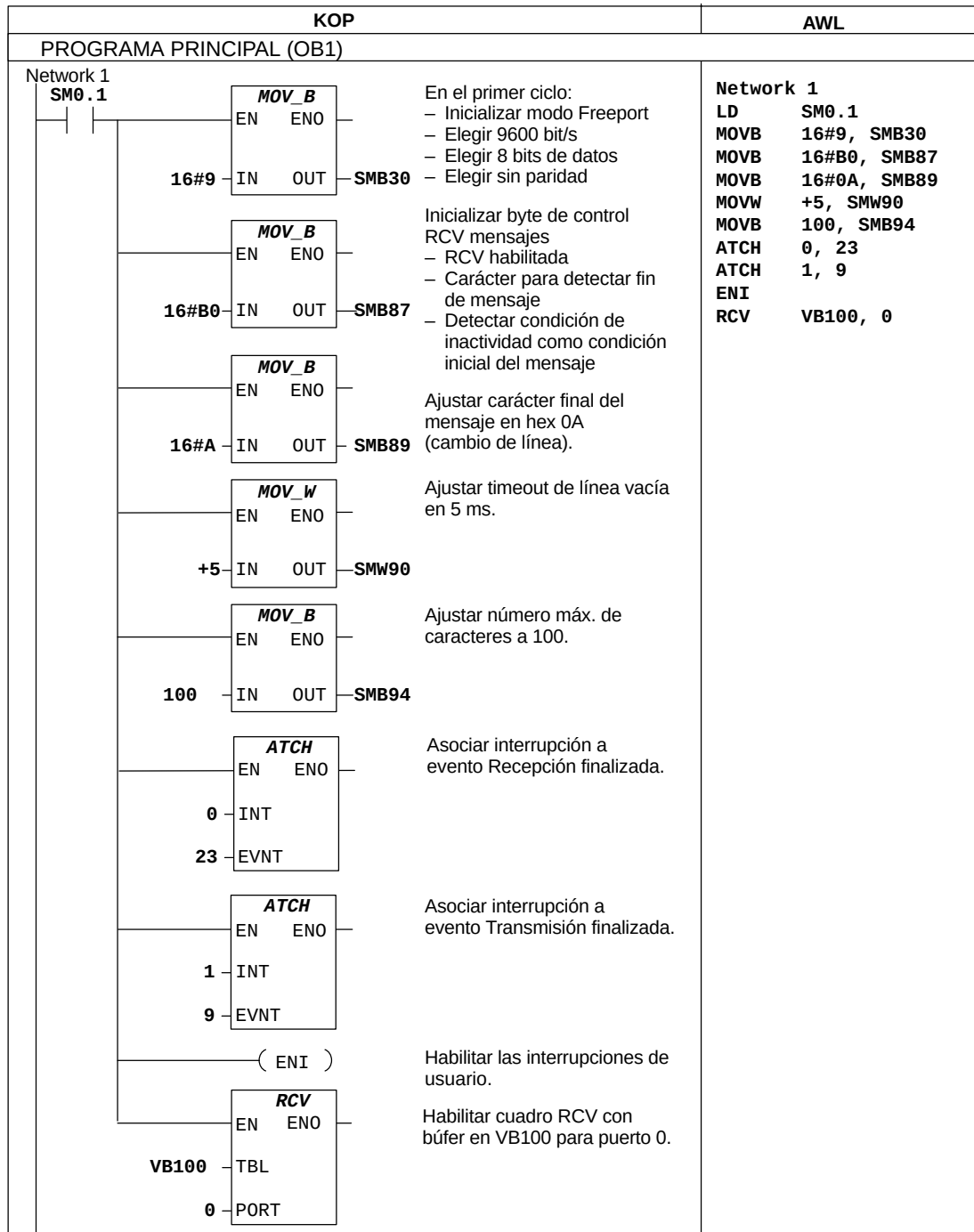


Figura 9-74 Ejemplo de la operación Transmitir mensaje en KOP, AWL y FUP (SIMATIC)

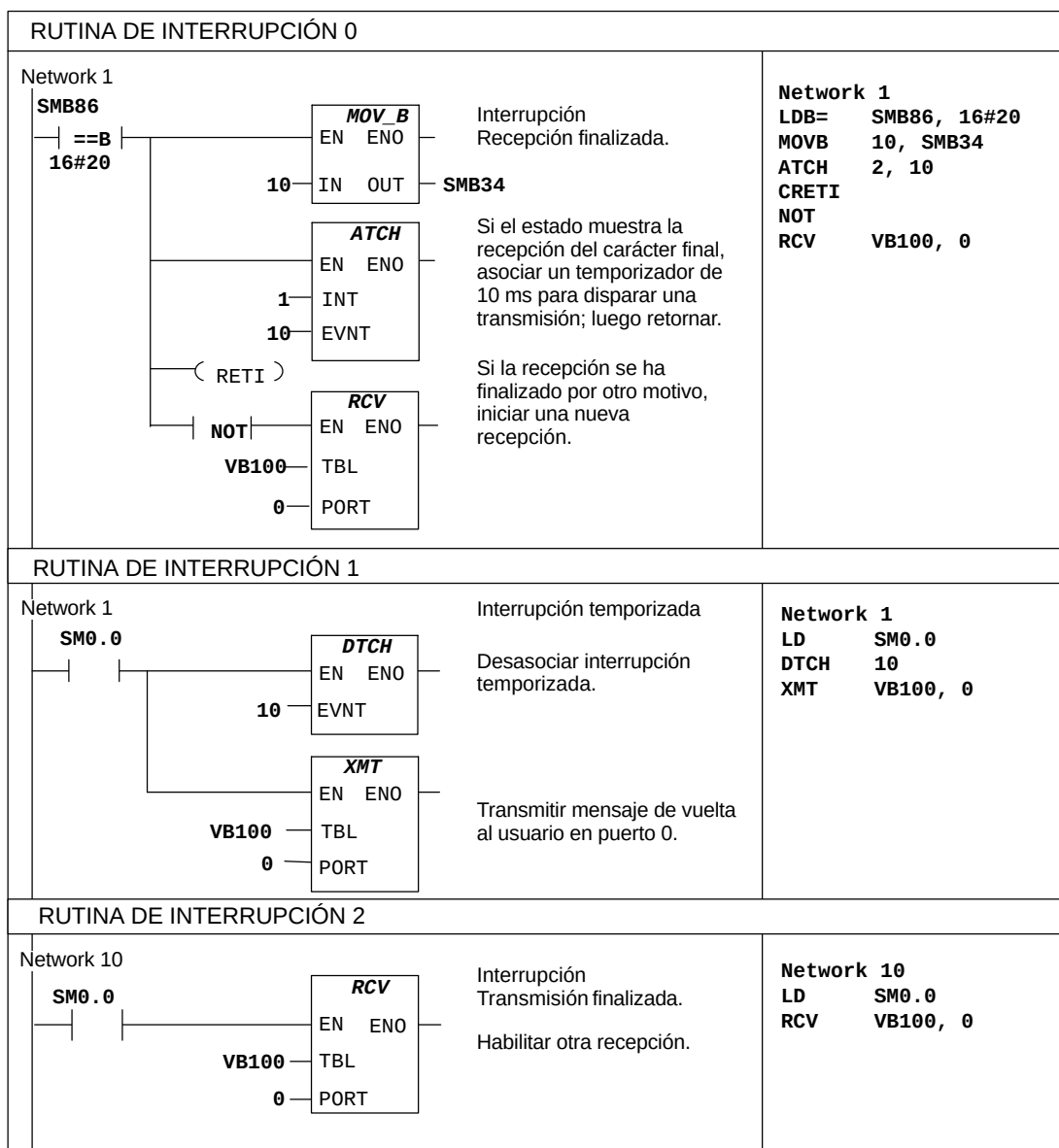


Figura 9-23 Ejemplo de la operación Transmitir mensaje en KOP, AWL y FUP (SIMATIC) (continuación)

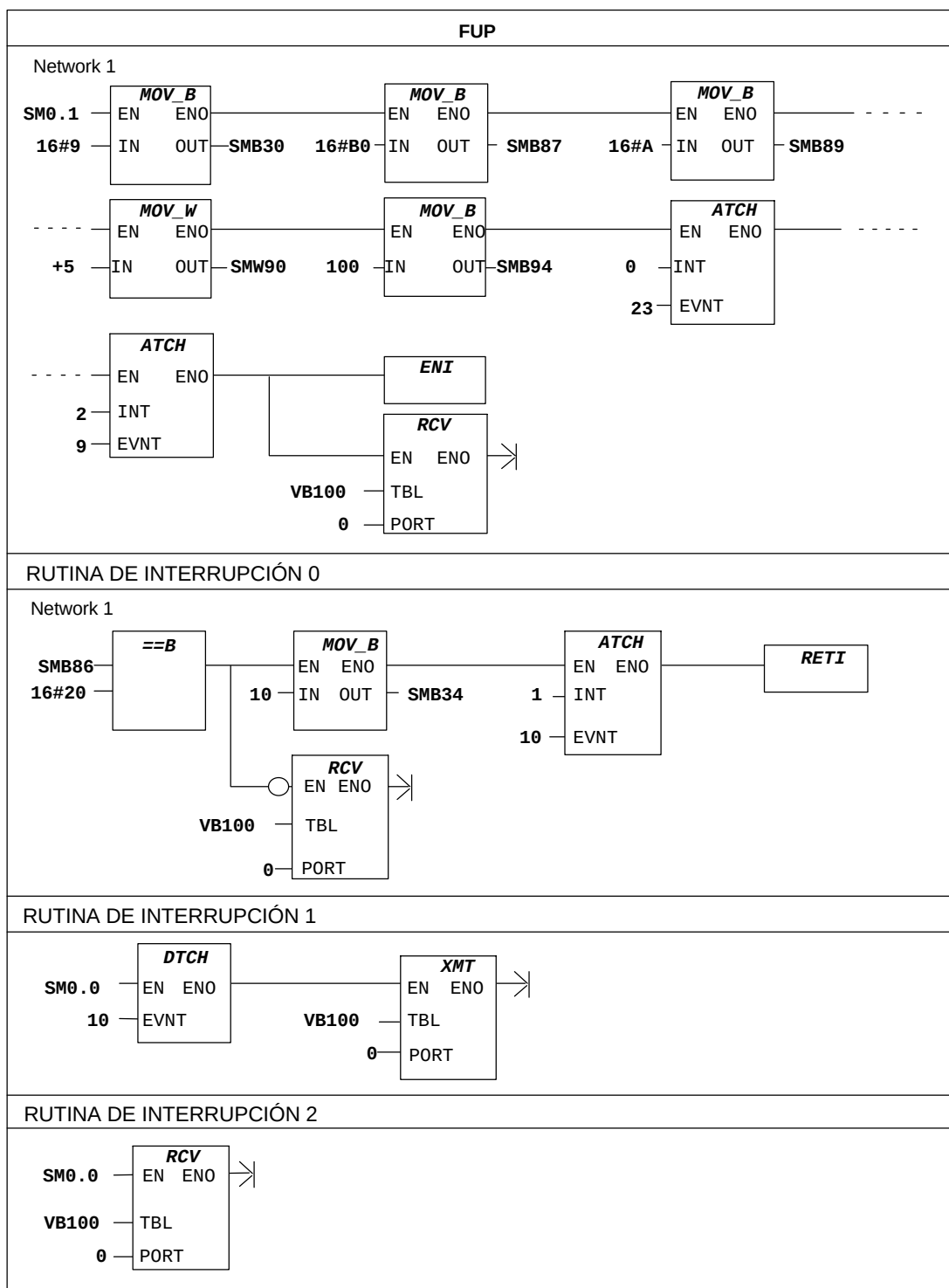
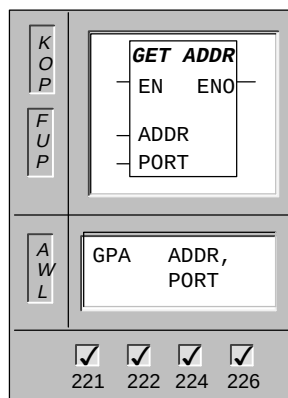


Figura 9-23 Ejemplo de la operación Transmitir mensaje en KOP, AWL y FUP (SIMATIC)
(continuación)

Leer dirección de puerto

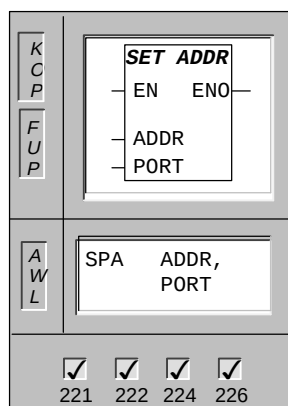


La operación **Leer dirección de puerto** lee la dirección de estación del puerto de la CPU indicado en PORT y deposita el valor en la dirección especificada en ADDR.

Leer dirección de puerto: Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Entradas/salidas	Operandos	Tipos de datos
ADDR	VB, IB, QB, MB, SB, SMB, LB, AC, *VD, *AC, *LD	BYTE
PORT	Constante	BYTE

Ajustar dirección de puerto



La operación **Ajustar dirección de puerto** ajusta la dirección de estación del puerto (PORT) al valor indicado en ADDR.

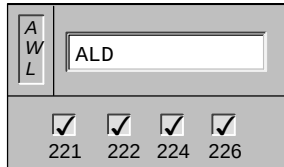
Ajustar dirección de puerto: Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

La nueva dirección no se almacena de forma permanente. Tras desconectar la CPU y conectarla de nuevo, el puerto en cuestión recuperará su dirección antigua (la que se había cargado junto con el bloque de sistema).

Entradas/salidas	Operandos	Tipos de datos
ADDR	VB, IB, QB, MB, SB, SMB, LB, AC, constante, *VD, *AC, *LD	BYTE
PORT	Constante	BYTE

9.16 Operaciones lógicas de pilas (SIMATIC)

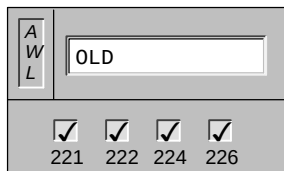
Combinar primer y segundo valor mediante Y



La operación **Combinar primer y segundo valor mediante Y** (ALD) combina los valores del primer y segundo nivel de la pila mediante una operación lógica Y. El resultado se carga en el nivel superior de la pila. Una vez ejecutada la operación ALD, la profundidad de la pila tiene un nivel menos.

Operandos: Ninguno

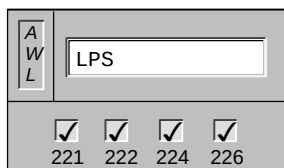
Combinar primer y segundo valor mediante O



La operación **Combinar primer y segundo valor mediante O** (OLD) combina los valores del primer y segundo nivel de la pila mediante una operación lógica O. El resultado se carga en el nivel superior de la pila. Una vez ejecutada la operación OLD, la profundidad de la pila tiene un nivel menos.

Operandos: Ninguno

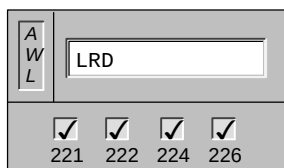
Duplicar primer valor



La operación **Duplicar primer valor** duplica el primer valor de la pila y lo desplaza dentro de la misma. El último valor de la pila se expulsa y se pierde.

Operandos: Ninguno

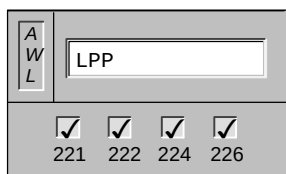
Copiar segundo valor



La operación **Copiar segundo valor** copia el segundo valor de la pila en el nivel superior de la misma. En la pila no se carga ni se expulsa ningún valor. No obstante, el valor que se encontraba en el nivel superior se sobrescribe con el nuevo valor.

Operandos: Ninguno

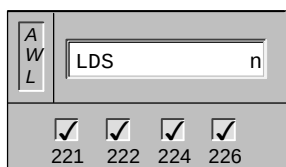
Sacar primer valor



La operación **Sacar primer valor** desplaza el primer valor fuera de la pila. El segundo valor se convierte entonces en el primer nivel de la pila.

Operandos: Ninguno

Cargar pila



La operación **Cargar pila** duplica el bit n de la pila y lo deposita en el nivel superior de la misma. El último valor de la pila se expulsa y se pierde.

Operandos: n (1 a 8)

Operaciones lógicas de pilas

La figura 9-75 muestra cómo funcionan las operaciones Combinar primer y segundo valor mediante Y y mediante O.

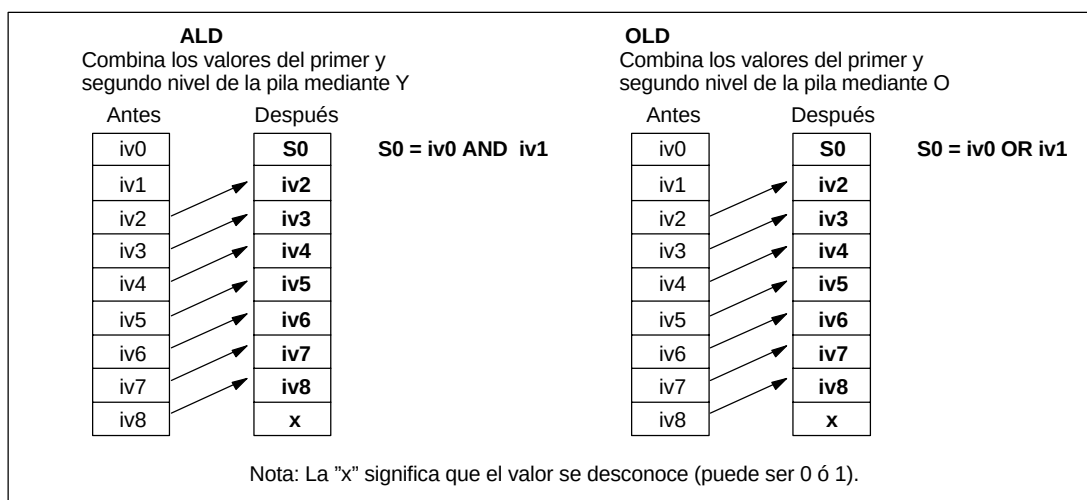


Figura 9-75 Operaciones Combinar primer y segundo valor mediante Y y mediante O

La figura 9-76 muestra cómo funcionan las operaciones Duplicar primer valor, Copiar segundo valor y Sacar primer valor.

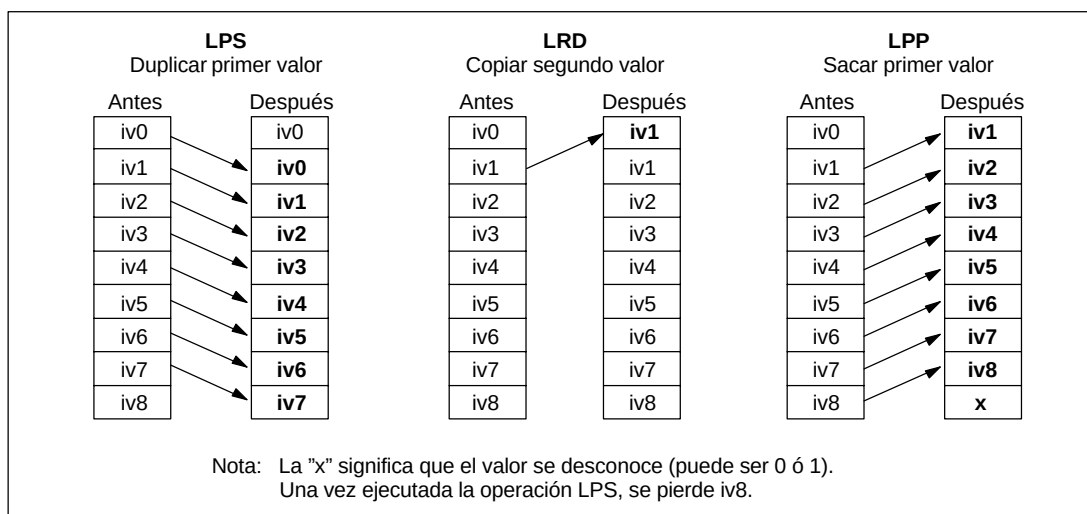


Figura 9-76 Operaciones Duplicar primer valor, Copiar segundo valor y Sacar primer valor

La figura muestra 9-77 cómo funciona la operación Cargar pila.

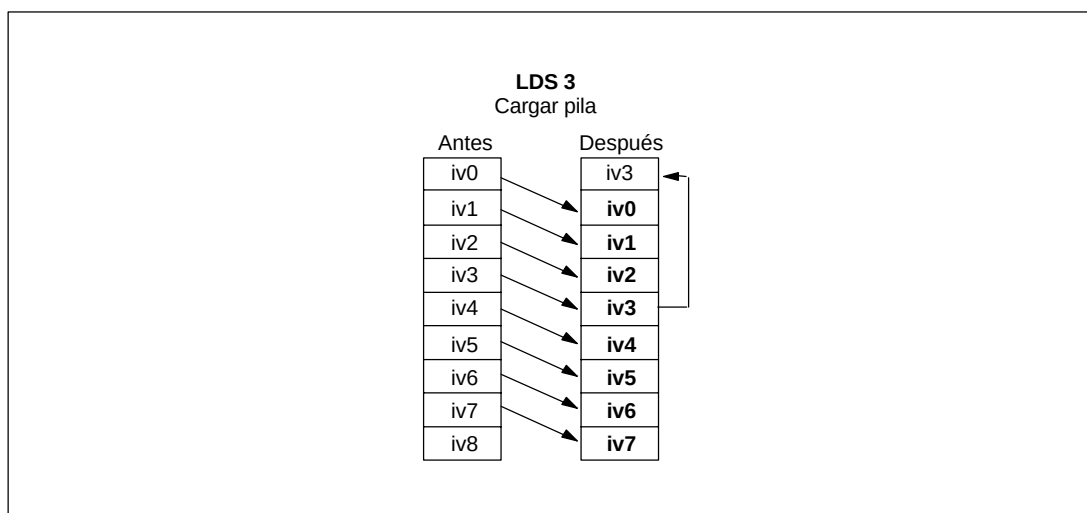


Figura 9-77 Operación Cargar pila

Ejemplo de una operación lógica de pila

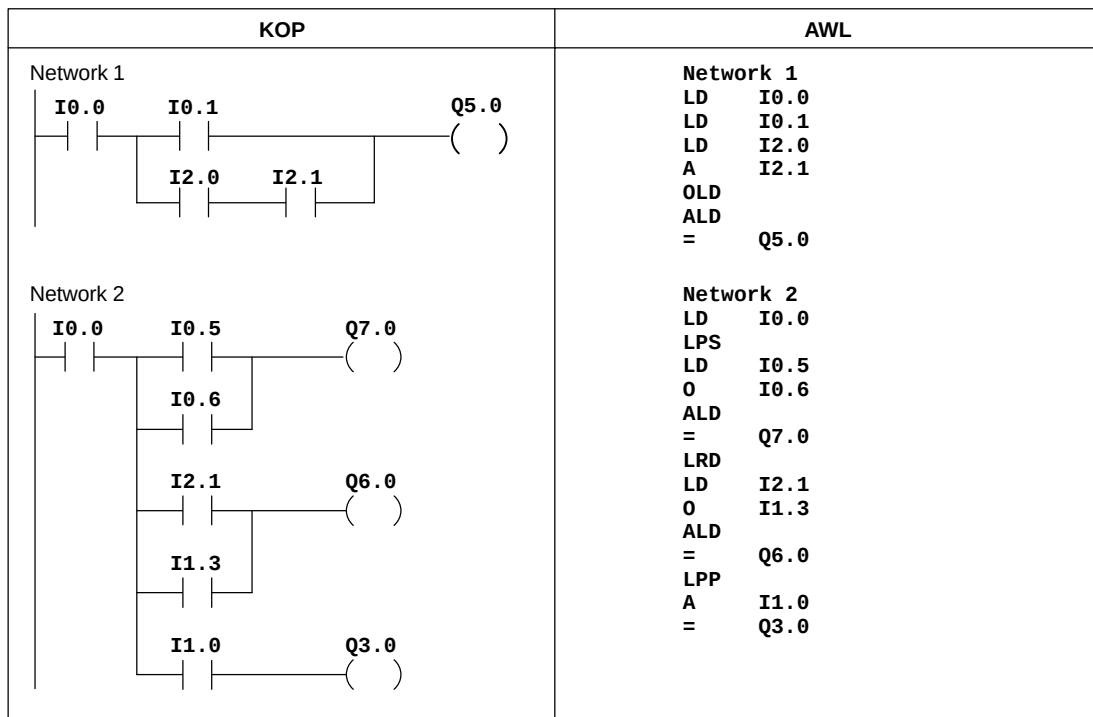


Figura 9-78 Ejemplo de una operación lógica de pila en KOP y AWL (SIMATIC)

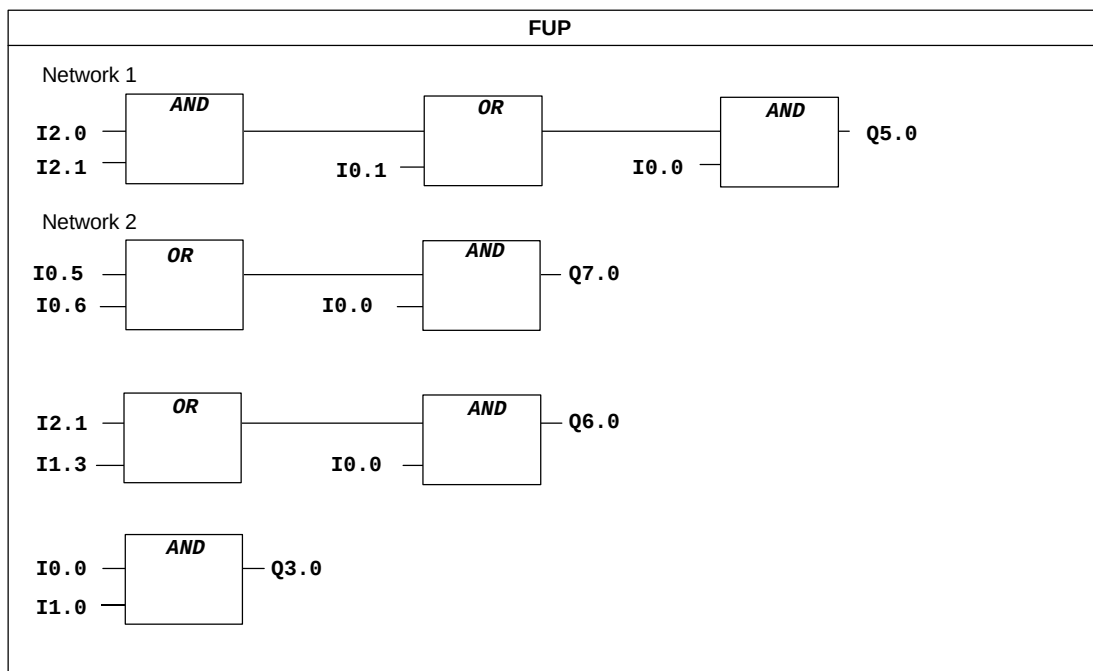


Figura 9-79 Ejemplo de una operación lógica de pila en FUP (SIMATIC)

Operaciones IEC 1131-3

10

El presente capítulo describe las operaciones IEC 1131-3 estándar. Hay algunas operaciones SIMATIC que se pueden utilizar en programas IEC. Éstas se denominan operaciones IEC no normalizadas, indicándose al comienzo de cada apartado.

Índice del capítulo

Apartado	Descripción	Página
10.1	Operaciones lógicas con bits (IEC)	10-2
10.2	Operaciones de comparación (IEC)	10-8
10.3	Operaciones de temporización (IEC)	10-11
10.4	Operaciones con contadores (IEC)	10-15
10.5	Operaciones aritméticas (IEC)	10-18
10.6	Operaciones con funciones numéricas (IEC)	10-21
10.7	Operaciones de transferencia (IEC)	10-23
10.8	Operaciones lógicas (IEC)	10-25
10.9	Operaciones de desplazamiento y rotación (IEC)	10-27
10.10	Operaciones de conversión (IEC)	10-30

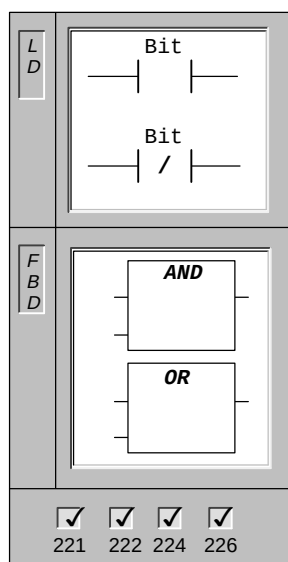
10.1 Operaciones lógicas con bits (IEC)

En la tabla 10-1 se indican las páginas donde se describen las operaciones lógicas con bits IEC no normalizadas a las que hace referencia el presente apartado.

Tabla 10-1 Operaciones lógicas con bits IEC no normalizadas

Descripción	Página
Contactos estándar	9-2
Contactos directos	9-3
Contacto NOT	9-4
Detectar flanco positivo y negativo	9-4
Asignar	9-6
Asignar directamente	9-6
Poner a 1 y Poner a 0 (bits N)	9-7

Contactos estándar (IEC 1131-3 no normalizados)



El **Contacto normalmente abierto** se cierra (ON) si el bit es igual a 1.

El **Contacto normalmente cerrado** se cierra (ON) si el bit es igual a 0.

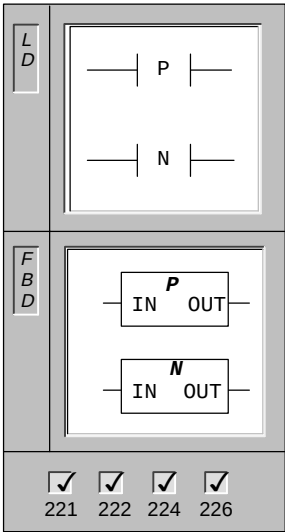
Estas operaciones leen el valor direccionado de la memoria o de la imagen del proceso si el tipo de memoria es I ó Q.

En LD, las operaciones Contacto normalmente abierto y Contacto normalmente cerrado se representan mediante contactos.

En FBD, los contactos normalmente abiertos se representan mediante cuadros AND/OR. Estas operaciones se pueden utilizar para manipular señales booleanas de la misma forma que los contactos LD. Los contactos normalmente cerrados también se representan mediante cuadros. Una operación Contacto normalmente cerrado se construye situando el símbolo de negación en la raíz de la señal de entrada. La cantidad de entradas de los cuadros AND y OR se puede incrementar a siete como máximo.

Entradas/salidas	Operandos	Tipos de datos
Bit (LD)	I, Q, M, SM, T, C, V, S, L	BOOL
Entrada (LD)	Circulación de corriente	BOOL
Entradas (FBD)	I, Q, M, SM, T, C, V, S, L, circulación de corriente	BOOL
Salidas (LD, FBD)	I, Q, M, SM, T, C, V, S, L, circulación de corriente	BOOL

Transición positiva, Transición negativa



El contacto detector de **Transición positiva** permite que la corriente circule durante un ciclo cada vez que se produce un cambio de 0 a 1 (de "off" a "on").

El contacto detector de **Transición negativa** permite que la corriente circule durante un ciclo cada vez que se produce un cambio de 1 a 0 (de "on" a "off").

En LD, las operaciones Transición positiva y Transición negativa se representan mediante contactos.

En FBD, dichas operaciones se representan mediante los cuadros P y N.

Entradas/salidas	Operandos	Tipos de datos
IN (LD)	Circulación de corriente	BOOL
IN (FBD)	I, Q, M, SM, T, C, V, S, L, circulación de corriente	BOOL
OUT (LD)	Circulación de corriente	BOOL
OUT (FBD)	I, Q, M, SM, T, C, V, S, L, circulación de corriente	BOOL

Ejemplos de operaciones con contactos

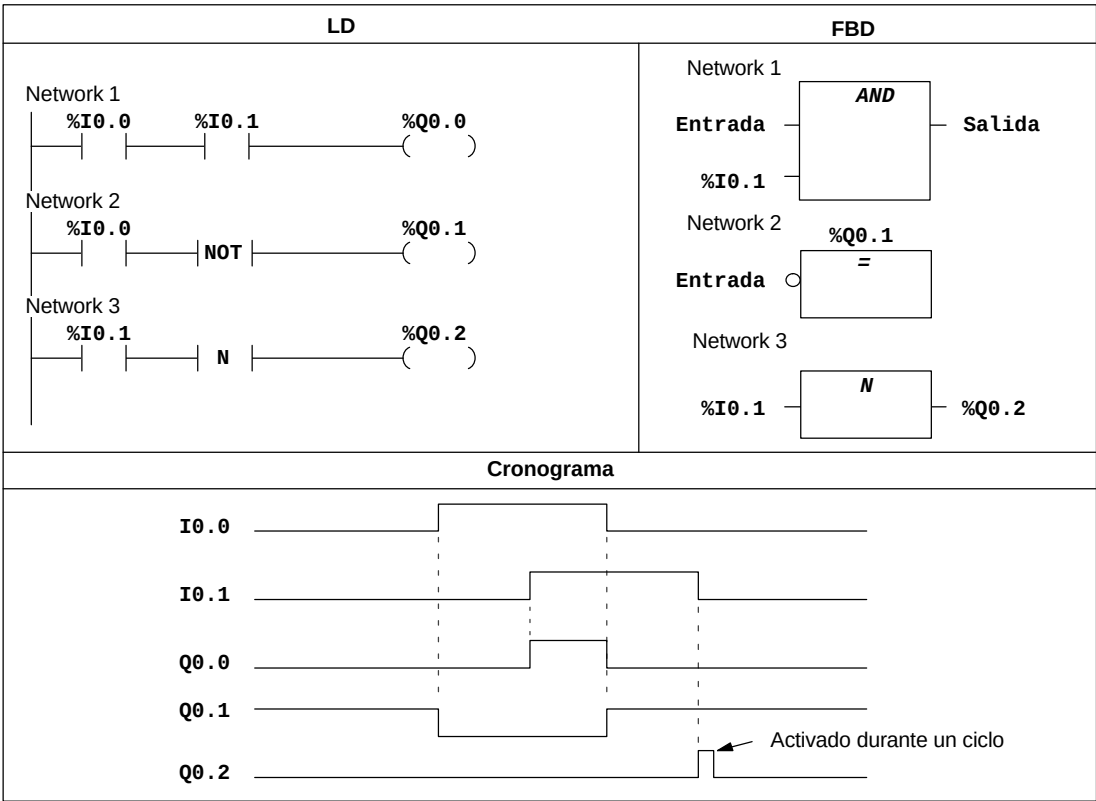
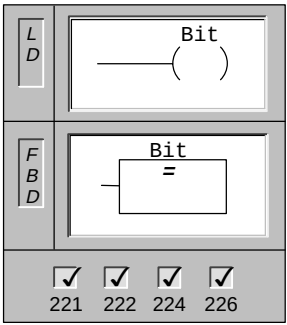


Figura 10-1 Ejemplos de operaciones lógicas con contactos en LD y FBD

Bobina



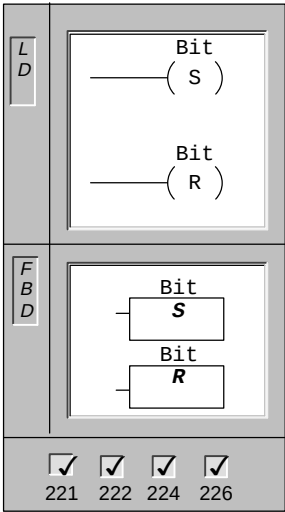
Al ejecutar la **Bobina** se activa la salida.

En LD, la operación Bobina se representa mediante una bobina.

En FBD, dicha operación se representa mediante el cuadro =.

Entradas/salidas	Operandos	Tipos de datos
Bit (LD/FBD)	I, Q, M, SM, T, C, V, S, L	BOOL
Entrada (LD)	Circulación de corriente	BOOL
Entrada (FBD)	I, Q, M, SM, T, C, V, S, L, circulación de corriente	BOOL

SET, RESET



Cuando se ejecutan las operaciones **SET** y **RESET**, el valor indicado por el bit o por el parámetro OUT se activa o se desactiva, respectivamente.

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto), 0091 (operando fuera de área)

Entradas/salidas	Operandos	Tipos de datos
Bit (LD, FBD)	I, Q, M, SM, T, C, V, S, L	BOOL
Entrada (LD)	Circulación de corriente	BOOL
Entrada (FBD)	I, Q, M, SM, T, C, V, S, L, circulación de corriente	BOOL

Ejemplos de operaciones con salidas

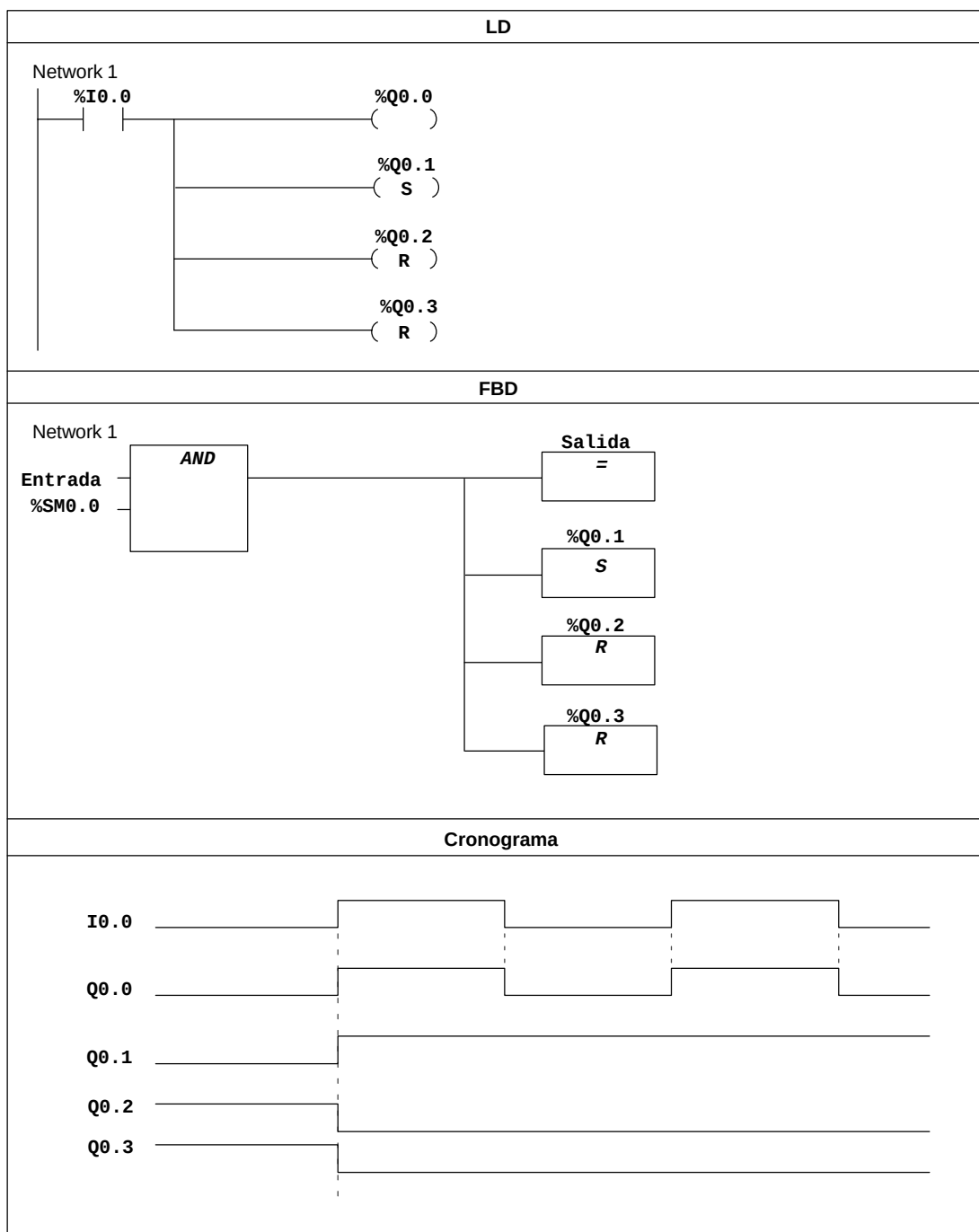
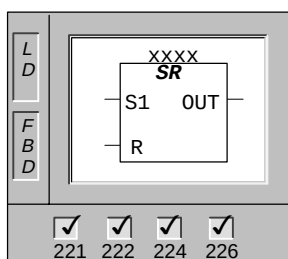


Figura 10-2 Ejemplos de operaciones con salidas en LD y FBD

Bloque funcional biestable (posicionar dominante)

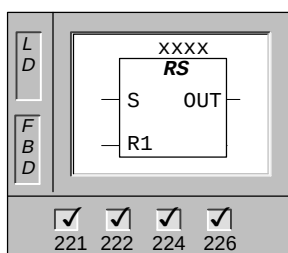


El **Bloque funcional biestable (posicionar dominante)** es un flip-flop en el que domina la señal "posicionar". Si tanto la señal "posicionar" (S1) como la señal "rearmar" (R) son verdaderas, la salida (OUT) será verdadera.

El parámetro xxxx del bloque funcional especifica el parámetro booleano que está activado ("posicionado") o desactivado ("rearmado"). La salida opcional refleja el estado de señal del parámetro xxxx.

Entradas/salidas	Operandos	Tipos de datos
S1, R (LD)	Circulación de corriente	BOOL
S1, R (FBD)	I, Q, M, SM, T, C,V, S, circulación de corriente	BOOL
OUT (LD)	Circulación de corriente	BOOL
OUT (FBD)	I, Q, M, SM, T, C, V, S, L, circulación de corriente	BOOL
xxxx	I, Q, M, V, S	BOOL

Bloque funcional biestable (rearmar dominante)



El **Bloque funcional biestable (rearmar dominante)** es un flip-flop en el que domina la señal "rearmar". Si tanto la señal "posicionar" (S) como la señal "rearmar" (R1) son verdaderas, la salida (OUT) será falsa.

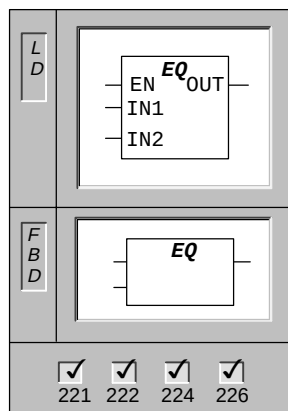
El parámetro xxxx del bloque funcional especifica el parámetro booleano que está activado ("posicionado") o desactivado ("rearmado"). La salida opcional refleja el estado de señal del parámetro xxxx.

Entradas/salidas	Operandos	Tipos de datos
S, R1 (LD)	Circulación de corriente	BOOL
S, R1 (FBD)	I, Q, M, SM, T, C, V, S, L, circulación de corriente	BOOL
OUT (LD)	Circulación de corriente	BOOL
OUT (FBD)	I, Q, M, SM, T, C, V, S, L, circulación de corriente	BOOL
xxxx	I, Q, M, V, S	BOOL

10.2 Operaciones de comparación (IEC)

No existen operaciones de comparación IEC no normalizadas.

Igualdad (EQ)

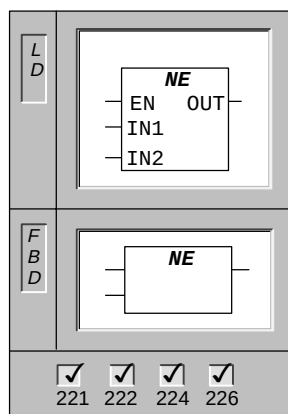


La función **Igualdad** (EQ) compara IN1 e IN2 con el resultado booleano depositado en OUT. Los tipos de datos de entrada y salida pueden variar pero deben ser del mismo tipo.

Las comparaciones de bytes no llevan signo. Las comparaciones de enteros, de enteros dobles y de reales llevan signo. Las operaciones de tiempo son enteros con signo.

Entradas/salidas	Operandos	Tipos de datos
Entradas (LD y FBD)	IB, QB, MB, SB, SMB, VB, LB, IW, QW, MW, SW, SMW, VW, LW, T, C, AIW, ID, QD, MD, SD, SMD, VD, LD, HC, AC, constante, *VD, *AC, *LD	BYTE, INT, DINT, REAL
OUT (sólo LD)	Circulación de corriente	BOOL
OUT (sólo FBD)	I, Q, M, SM, T, C, V, S, L, circulación de corriente	BOOL

Desigualdad (NE)

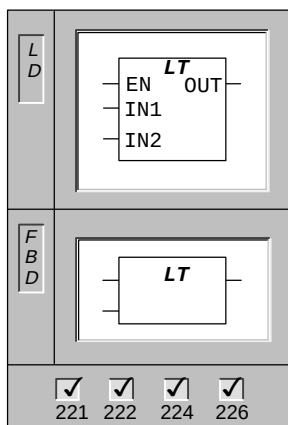


La función **Desigualdad** (NE) compara IN1 e IN2 con el resultado booleano depositado en OUT. Los tipos de datos de entrada y salida pueden variar pero deben ser del mismo tipo.

Las comparaciones de bytes no llevan signo. Las comparaciones de enteros, de enteros dobles y de reales llevan signo. Las operaciones de tiempo son enteros con signo.

Entradas/salidas	Operandos	Tipos de datos
Entradas (LD y FBD)	IB, QB, MB, SB, SMB, VB, LB, IW, QW, MW, SW, SMW, VW, LW, T, C, AIW, ID, QD, MD, SD, SMD, VD, LD, HC, AC, constante, *VD, *AC, *LD	BYTE, INT, DINT, REAL
OUT (sólo LD)	Circulación de corriente	BOOL
OUT (sólo FBD)	I, Q, M, SM, T, C, V, S, L, circulación de corriente	BOOL

Secuencia ascendente (LT)

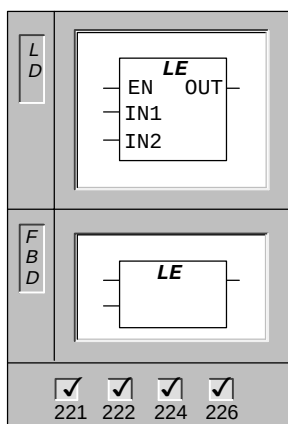


La función **Secuencia ascendente (LT)** compara $IN1 < IN2$ con el resultado booleano depositado en OUT. Los tipos de datos de entrada y salida pueden variar pero deben ser del mismo tipo.

Las comparaciones de bytes no llevan signo. Las comparaciones de enteros, de enteros dobles y de reales llevan signo. Las operaciones de tiempo son enteros con signo.

Entradas/salidas	Operandos	Tipos de datos
Entradas (LD y FBD)	IB, QB, MB, SB, SMB, VB, LB, IW, QW, MW, SW, SMW, VW, LW, T, C, AIW, ID, QD, MD, SD, SMD, VD, LD, HC, AC, constante, *VD, *AC, *LD	BYTE, INT, DINT, REAL
OUT (sólo LD)	Circulación de corriente	BOOL
OUT (sólo FBD)	I, Q, M, SM, V, S, L, circulación de corriente	BOOL

Secuencia monótona (LE)

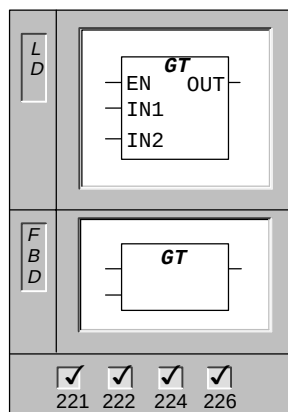


La función **Secuencia monótona (LE)** compara $IN1 \leq IN2$ con el resultado booleano depositado en OUT. Los tipos de datos de entrada y salida pueden variar pero deben ser del mismo tipo.

Las comparaciones de bytes no llevan signo. Las comparaciones de enteros, de enteros dobles y de reales llevan signo. Las operaciones de tiempo son enteros con signo.

Entradas/salidas	Operandos	Tipos de datos
Entradas (LD y FBD)	IB, QB, MB, SB, SMB, VB, LB, IW, QW, MW, SW, SMW, VW, LW, T, C, AIW, ID, QD, MD, SD, SMD, VD, LD, HC, AC, constante, *VD, *AC, *LD	BYTE, INT, DINT, REAL
OUT (sólo LD)	Circulación de corriente	BOOL
OUT (sólo FBD)	I, Q, M, SM, V, S, L, circulación de corriente	BOOL

Secuencia decreciente (GT)

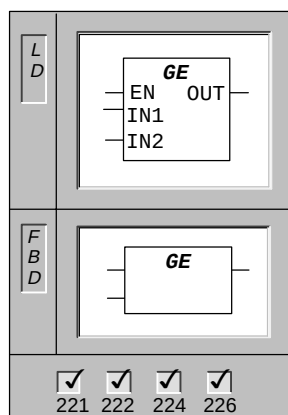


La función **Secuencia decreciente (GT)** compara $IN1 > IN2$ con el resultado booleano depositado en OUT. Los tipos de datos de entrada y salida pueden variar pero deben ser del mismo tipo.

Las comparaciones de bytes no llevan signo. Las comparaciones de enteros, de enteros dobles y de reales llevan signo. Las operaciones de tiempo son enteros con signo.

Entradas/salidas	Operandos	Tipos de datos
Entradas (LD y FBD)	IB, QB, MB, SB, SMB, VB, LB, IW, QW, MW, SW, SMW, VW, LW, T, C, AIW, ID, QD, MD, SD, SMD, VD, LD, HC, AC, constante, *VD, *AC, *LD	BYTE, INT, DINT, REAL
OUT (sólo LD)	Circulación de corriente	BOOL
OUT (sólo FBD)	I, Q, M, SM, V, S, L, circulación de corriente	BOOL

Secuencia monótona (GE)



La función **Secuencia monótona (GE)** compara $IN1 \geq IN2$ con el resultado booleano depositado en OUT. Los tipos de datos de entrada y salida pueden variar pero deben ser del mismo tipo.

Las comparaciones de bytes no llevan signo. Las comparaciones de enteros, de enteros dobles y de reales llevan signo.

Entradas/salidas	Operandos	Tipos de datos
Entradas (LD y FBD)	IB, QB, MB, SB, SMB, VB, LB, IW, QW, MW, SW, SMW, VW, LW, T, C, AIW, ID, QD, MD, SD, SMD, VD, LD, HC, AC, constante, *VD, *AC, *LD	BYTE, INT, DINT, REAL
OUT (sólo LD)	Circulación de corriente	BOOL
OUT (sólo FBD)	I, Q, M, SM, V, S, L, circulación de corriente	BOOL

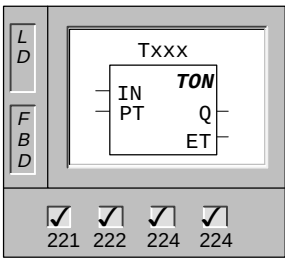
10.3 Operaciones de temporización (IEC)

En la tabla 10-2 se indican las páginas donde se describen las operaciones de temporización (IEC) no normalizadas a las que hace referencia el presente apartado.

Tabla 10-2 Operaciones de temporización (IEC) no normalizadas

Descripción	Página
Temporizador de retardo a la conexión memorizado	9-15

Temporizador con retardo al conectar

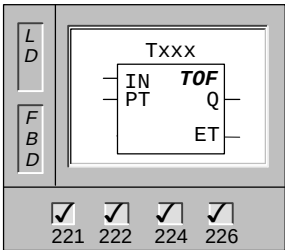


El bloque funcional **Temporizador con retardo al conectar** temporiza hasta el valor prefijado cuando la entrada de habilitación (IN) cambia a "verdadero". Si el tiempo transcurrido (ET) es mayor que o igual al tiempo prefijado (PT), se activará el bit de salida del temporizador (Q).

El bit de salida se desactiva cuando la entrada de habilitación cambie a "falso". Cuando se alcanza el tiempo prefijado (PT), la temporización se detiene y el temporizador se inhibe.

Entradas/salidas	Operandos	Tipos de datos
IN (LD)	Circulación de corriente	BOOL
IN (FBD)	I, Q, M, SM, T, C, V, S, L, circulación de corriente	BOOL
PT (LD y FBD)	VW, IW, QW, MW, SMW, LW, SW, AIW, AC, constante, *VD, *AC, *LD	INT
Q (LD y FBD)	I, Q, M, SM, V, S, L	BOOL
ET (LD y FBD)	VW, IW, QW, MW, SMW, LW, SW, AQW, AC, *VD, *AC, *LD	INT
Txxx	Constante (v. tabla 10-3, página 10-12)	TON

Temporizador con retardo al desconectar



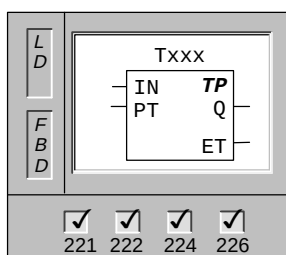
El bloque funcional **Temporizador con retardo al desconectar** se utiliza para retardar el estado "falso" de una salida durante un período determinado tras haber adoptado la entrada el estado "falso". Temporiza hasta el valor predefinido cuando la entrada de habilitación (IN) cambio a "falso". Si el tiempo transcurrido (ET) es mayor que o igual al tiempo prefijado (PT), se activará el bit de salida del temporizador (Q).

Una vez alcanzado el valor prefijado, el bit de salida del temporizador cambia a "falso" y el tiempo transcurrido se mantiene hasta que la entrada de habilitación (IN) cambia a "verdadero". Si la entrada de habilitación (IN) cambia a "falso" durante un período inferior al tiempo prefijado (PT), el bit de salida seguirá siendo "verdadero".

Para obtener más información sobre los números y las resoluciones de los temporizadores, consulte la tabla 10-3.

Entradas/salidas	Operandos	Tipos de datos
IN (LD)	Circulación de corriente	BOOL
IN (FBD)	I, Q, M, SM, T, C, V, S, L, circulación de corriente	BOOL
PT (LD y FBD)	VW, IW, QW, MW, SMW, LW, SW, AIW, AC, constante, *VD, *AC, *LD	INT
Q (LD y FBD)	I, Q, M, SM, V, S, L	BOOL
ET (LD y FBD)	VW, IW, QW, MW, SMW, LW, SW, AQW, AC, *VD, *AC, *LD	INT
Txxx	Constante (v. tabla 10-3, página 10-12)	TOF

Temporizador por impulsos



El bloque funcional **Temporizador por impulsos** se utiliza para generar impulsos de una duración determinada. Cuando el estado de señal de la entrada de habilitación (IN) cambia a "verdadero", se activa el bit de salida (Q). Éste último sigue siendo "verdadero" durante el impulso especificado en el tiempo prefijado (PT). Cuando el tiempo transcurrido (ET) alcanza el valor del tiempo prefijado (PT), el estado de señal del bit de salida (Q) cambia a "falso". El tiempo transcurrido (ET) se conserva hasta que la entrada de habilitación cambie a "falso". Una vez que el bit de salida (Q) cambie a "verdadero", permanecerá en ese estado hasta que haya transcurrido el tiempo prefijado (PT).

Para obtener más información sobre los números y las resoluciones de los temporizadores, consulte la tabla 10-3.

Entradas/salidas	Operandos	Tipos de datos
IN (LD)	Circulación de corriente	BOOL
IN (FBD)	I, Q, M, SM, T, C, V, S, L, circulación de corriente	BOOL
PT (LD y FBD)	VW, IW, QW, MW, SMW, LW, SW, AIW, AC, constante, *VD, *AC, *LD	INT
Q (LD y FBD)	I, Q, M, SM, S, V, L	BOOL
ET (LD y FBD)	VW, IW, QW, MW, SW, SMW, LW, AQW, AC, *VD, *AC, *LD	INT
Txxx	Constante (v. tabla 10-3)	TP

Operaciones de temporización IEC 1131-3

Se dispone de temporizadores TON, TOF y TP con tres resoluciones. La resolución viene determinada por el número de temporizador (v. tabla 10-3). El valor actual resulta del valor de conteo multiplicado por la base de tiempo. Por ejemplo, el valor de conteo 50 en un temporizador de 10 ms equivale a 500 ms.

Tabla 10-3 Temporizadores y sus resoluciones

Tipo de temporizador	Resolución en milisegundos (ms)	Valor máximo en segundos (s)	Nº de temporizador
TON, TOF, TP	1 ms	32.767 s	T32, T96
	10 ms	327.67 s	T33 a T36, T97 a T100
	100 ms	3276.7 s	T37 a T63, T101 a T255

Nota

No se pueden compartir números iguales para los temporizadores TOF, TP y TON. Por ejemplo, no puede haber tanto un TON T32 como un TOF T32.

Ejemplo de un temporizador de retardo al conectar

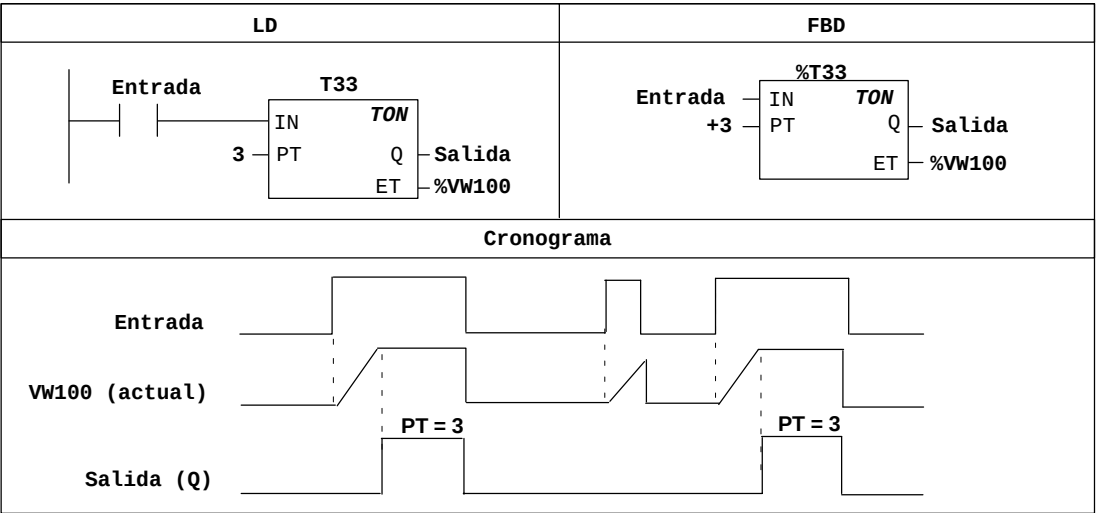


Figura 10-3 Ejemplo de un temporizador de retardo al conectar en LD y FBD

Ejemplo de un temporizador de retardo al desconectar

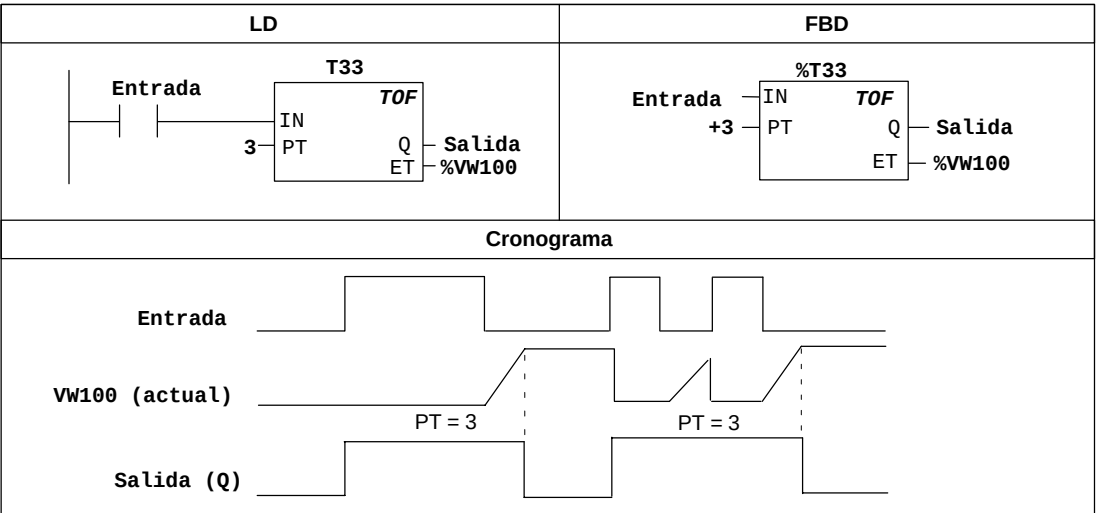


Figura 10-4 Ejemplo de un temporizador de retardo al desconectar en LD y FBD

Ejemplo de un temporizador por impulsos

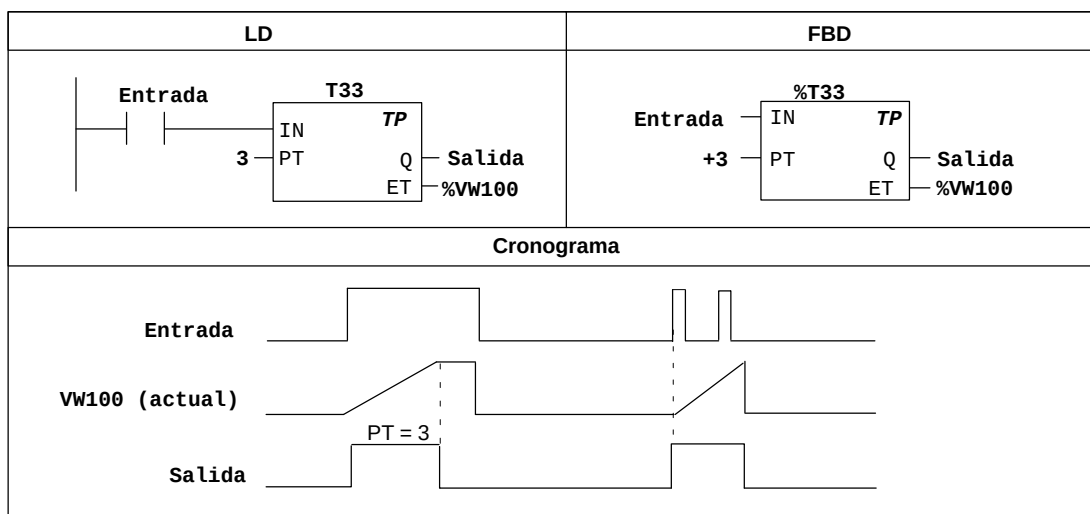


Figura 10-5 Ejemplo de un temporizador por impulsos en LD y FBD

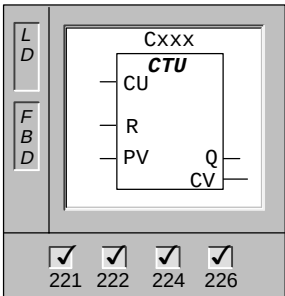
10.4 Operaciones con contadores (IEC)

En la tabla 10-4 se indican las páginas donde se describen las operaciones con contadores (IEC) no normalizadas a las que hace referencia el presente apartado.

Tabla 10-4 Operaciones con contadores (IEC) no normalizadas

Descripción	Página
Activar contador rápido	9-27
Definir modo para contador rápido	9-27
Salida de impulsos	9-49

Contador ascendente



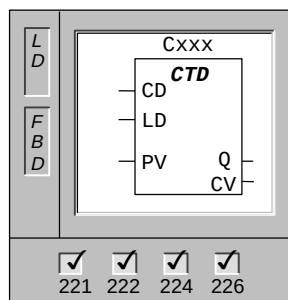
El bloque funcional **Contador ascendente** cuenta adelante desde el valor actual hasta el valor prefijado al producirse un flanco positivo en la entrada de contaje adelante (CU). Si el valor actual (CV) es mayor o igual al valor prefijado (PV), se activa el bit de contaje (Q). El contador se inicializa al activarse la entrada de desactivación (R). El contador ascendente detiene el conteo al alcanzar el valor prefijado (PV).

Nota

Puesto que cada contador dispone sólo de un valor actual, no se podrá asignar un mismo número a varios contadores. (Los contadores ascendentes, descendentes y ascendentes-descendentes acceden a un mismo valor actual).

Entradas/salidas	Operandos	Tipos de datos
CU, R (sólo LD)	Circulación de corriente	BOOL
CU, R (sólo FBD)	I, Q, M, SM, V, S, L, T, C, circulación de corriente	BOOL
PV (LD y FBD)	VW, IW, QW, MW, SMW, LW, SW, AIW, AC, constante, *VD, *AC, *LD	INT
Q (LD y FBD)	I, Q, M, SM, V, S, L	BOOL
CV (LD y FBD)	VW, IW, QW, MW, SW, LW, AC, *VD, *AC, *LD	INT
Cxxx (LD & FBD)	Constante	CTU

Contador descendente



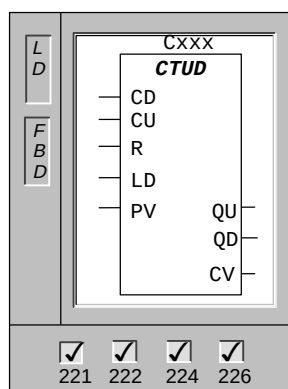
El bloque funcional **Contador descendente** cuenta hacia atrás desde el valor prefijado (PV) al producirse un flanco positivo en la entrada de contaje atrás (CD). Si el valor actual (CV) es igual a cero, se activa el bit de salida del contador (Q). El contador se inicializa y carga el valor actual (CV) en el valor prefijado (PV) cuando se habilita la entrada de carga (LD). El contador descendente detiene el contaje cuando alcanza el valor cero.

Nota

Puesto que cada contador dispone sólo de un valor actual, no se podrá asignar un mismo número a varios contadores. (Los contadores ascendentes, descendentes y ascendentes-descendentes acceden a un mismo valor actual).

Entradas/salidas	Operandos	Tipos de datos
CD, LD (LD)	Circulación de corriente	BOOL
CD, LD (FBD)	I, Q, M, SM, V, S, L, T, C, circulación de corriente	BOOL
PV (LD, FBD)	VW, IW, QW, MW, SMW, LW, SW, AIW, AC, constante, *VD, *AC, *LD	INT
Q (LD y FBD)	I, Q, M, SM, V, S, L	BOOL
CV (LD y FBD)	VW, IW, QW, MW, SW, LW, AC, *VD, *AC, *LD	INT
Cxxx	Constante	CTD

Contador ascendente-descendente



El bloque funcional **Contador ascendente-descendente** cuenta adelante o atrás desde el valor prefijado al producirse un flanco positivo en la entrada de contaje adelante (CU) o de contaje atrás (CD), respectivamente. La salida (QU) se activa cuando el valor actual (CV) es igual al valor prefijado. La salida (QD) se activa cuando el valor actual (CV) es igual a cero. El contador carga el valor actual (CV) en el valor prefijado (PV) cuando se habilita la entrada de carga (LD). De forma similar, el contador se inicializa y carga el valor actual (CV) con cero cuando se habilita la desactivación (R). El contador detiene el contaje cuando alcanza el valor prefijado, o bien cero.

Nota

Puesto que cada contador dispone sólo de un valor actual, no se podrá asignar un mismo número a varios contadores. (Los contadores ascendentes, descendentes y ascendentes-descendentes acceden a un mismo valor actual).

Entradas/salidas	Operandos	Tipos de datos
CU, CD, R, LD (sólo LD)	Circulación de corriente	BOOL
CU, CD, R, LD (sólo FBD)	I, Q, M, SM, V, S, L, T, C, circulación de corriente	BOOL
PV (LD y FBD)	VW, IW, QW, MW, SMW, LW, SW, AIW, AC, constante, *VD, *AC, *LD	INT
QU (LD y FBD)	I, Q, M, SM, V, S, L	BOOL
QD (LD y FBD)	I, Q, M, SM, V, S, L	BOOL
CV (LD y FBD)	VW, IW, QW, MW, SW, LW, AC, *VD, *AC, *LD	INT
Cxxx	Constante	CTUD

Ejemplo de una operación de contaje

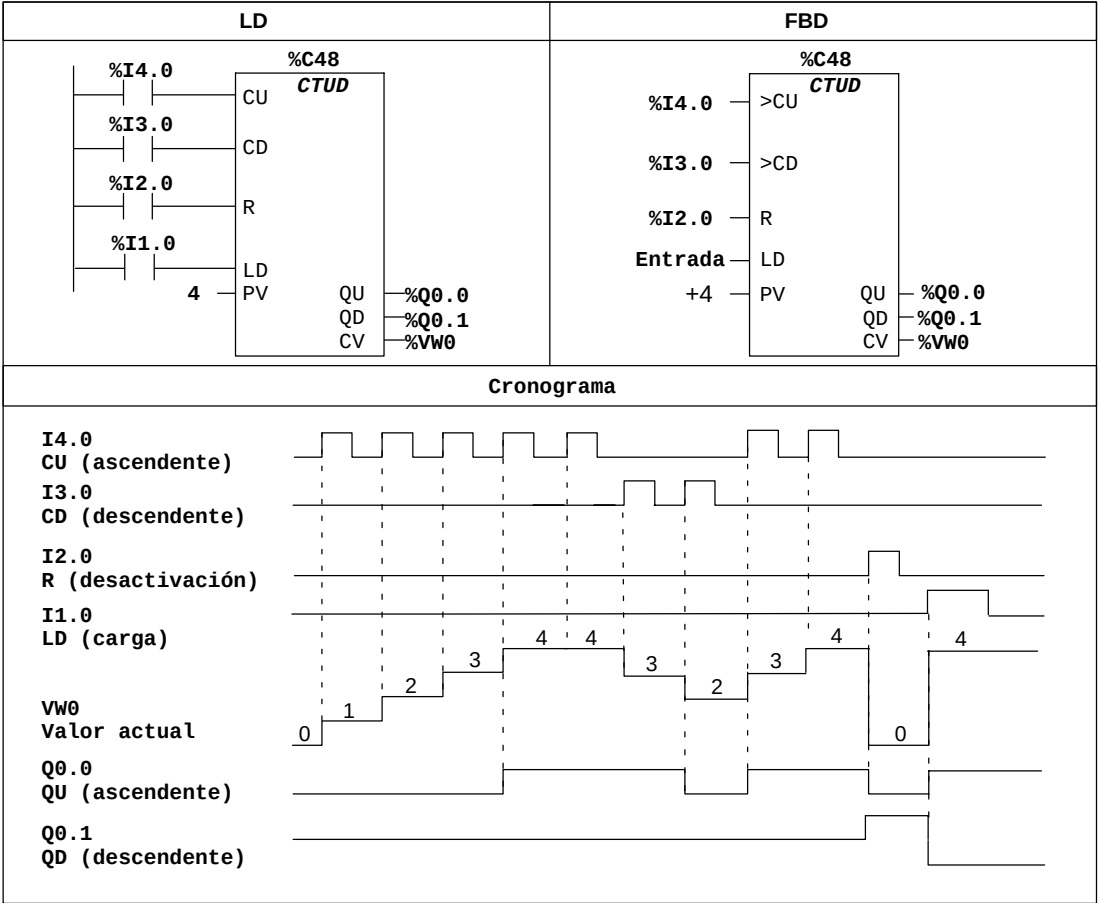
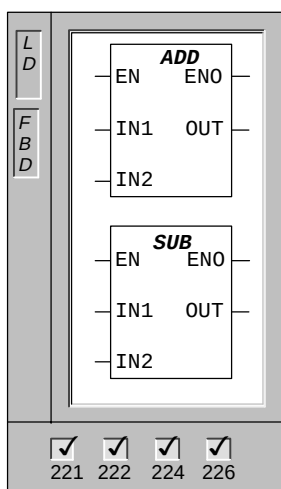


Figura 10-6 Ejemplo de una operación de contaje en LD y FBD

10.5 Operaciones aritméticas (IEC)

Sumar y restar



Las funciones **Sumar** y **Restar** suman o restan IN1 e IN2 y depositan el resultado en OUT. Los tipos de datos de entrada y salida pueden variar pero deben ser del mismo tipo. Por ejemplo, dos variables de 16 bits se pueden sumar o restar, pero el resultado se debe depositar en una variable de 16 bits. El resultado de una suma o de una resta de dos variables de 32 bits se debe depositar en una variable de 32 bits.

En LD: $IN1 + IN2 = OUT$
 $IN1 - IN2 = OUT$

Condiciones de error que ponen ENO a 0:
 SM1.1 (desbordamiento), SM4.3 (tiempo de ejecución),
 0006 (direccionamiento indirecto)

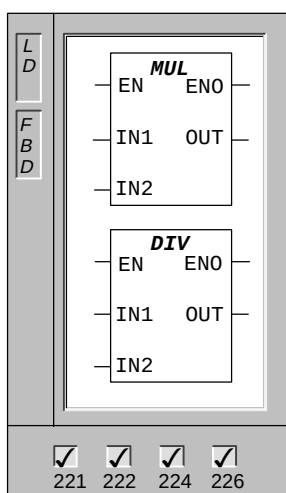
Estas funciones afectan a las siguientes marcas especiales:
 SM1.0 (cero); SM1.1 (desbordamiento); SM1.2 (negativo)

Entradas/salidas	Operandos	Tipos de datos
IN1, IN2	VW, IW, QW, MW, SW, SMW, LW, AIW, T, C, VD, ID, QD, MD, SMD, SD, LD, HC, AC, constante, *VD, *AC, *LD	INT, DINT, REAL
OUT	VW, IW, QW, MW, SW, SMW, LW, T, C, VD, ID, QD, MD, SMD, SD, LD, AC, *VD, *AC, *LD	INT, DINT, REAL

Nota

Los números reales (o números en coma flotante) se representan en el formato descrito en la norma ANSI/IEEE 754-1985 (precisión sencilla). Para obtener más información al respecto, consulte dicha norma.

Multiplicar y dividir



La función **Multiplicar** (MUL) multiplica IN1 por IN2 y deposita el resultado en la variable indicada por OUT.

La función **Dividir** (DIV) divide IN1 por IN2 y deposita el resultado en la variable indicada por OUT.

Los tipos de datos de entrada y salida pueden variar pero deben ser del mismo tipo. Por ejemplo, el producto de dos variables de 16 bits se debe depositar en una variable de 16 bits. El producto de dos variables de 32 bits se debe depositar en una variable de 32 bits.

En LD: $IN1 * IN2 = OUT$
 $IN1 / IN2 = OUT$

Condiciones de error que ponen ENO a 0:
 SM1.1 (desbordamiento), SM1.3 (división por cero),
 SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Estas funciones afectan a las siguientes marcas especiales: SM1.0 (cero); SM1.1 (desbordamiento); SM1.2 (negativo); SM1.3 (división por cero)

Si SM1.1 (marca de desbordamiento) está activada, los demás bits de estado aritméticos se borrarán y el operando de salida no se alterará. En el caso de operaciones con enteros, si SM1.3 se activa durante una operación de división, permanecerán inalterados los demás bits aritméticos de estado, así como los operandos de entrada originales. En otro caso, todos los bits aritméticos de estado asistidos contendrán el estado válido al finalizar la operación aritmética.

Entradas/salidas	Operandos	Tipos de datos
IN1, IN2	VW, IW, QW, MW, SW, SMW, LW, AIW, T, C, VD, ID, QD, MD, SMD, SD, LD, HC, AC, constante, *VD, *AC, *LD	INT, DINT, REAL
OUT	VW, IW, QW, MW, SW, SMW, T, C, LW, VD, ID, QD, MD, SMD, SD, LD, AC, *VD, *AC, *LD	INT, DINT, REAL

Nota

Los números reales (o números en coma flotante) se representan en el formato descrito en la norma ANSI/IEEE 754-1985 (precisión sencilla). Para obtener más información al respecto, consulte dicha norma.

Ejemplos de operaciones aritméticas

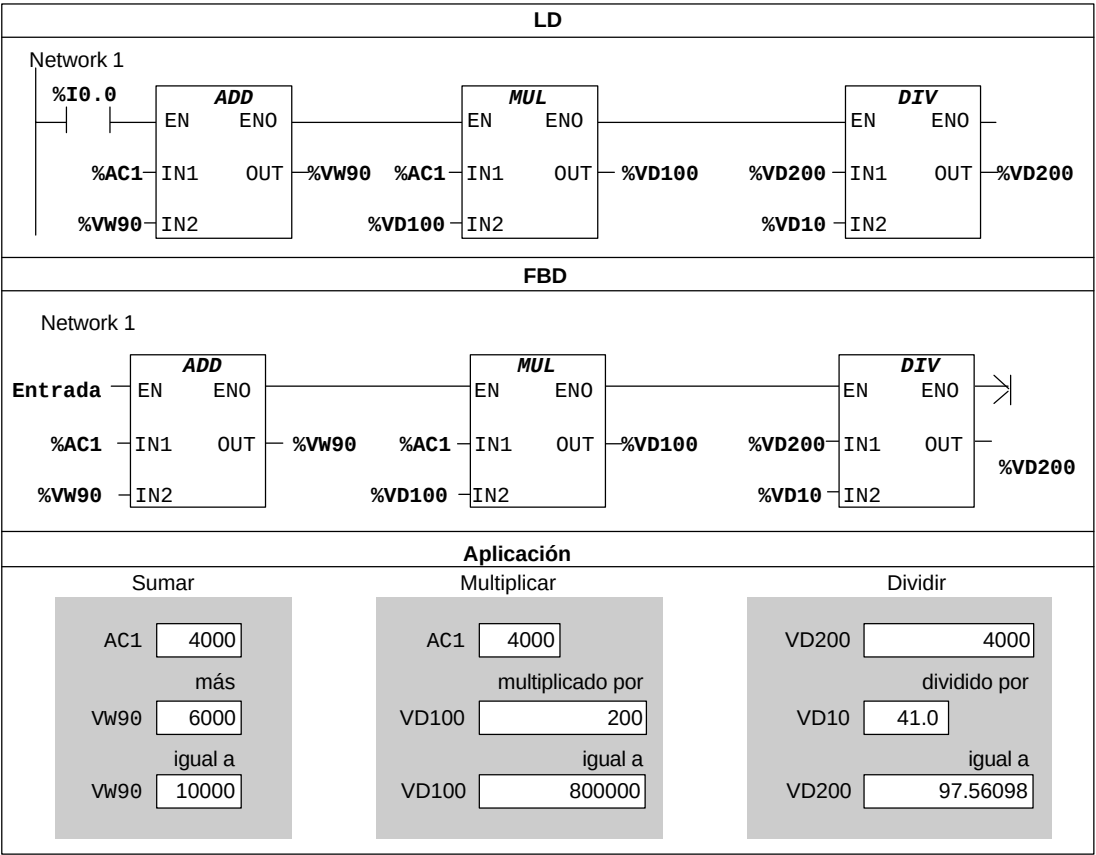
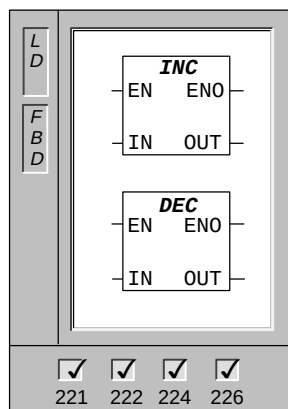


Figura 10-7 Ejemplos de operaciones aritméticas en LD y FBD

Incrementar, Decrementar



Las funciones **Incrementar** y **Decrementar** suman/restan 1 a IN y depositan el resultado en OUT.

Las funciones Incrementar byte y Decrementar byte no llevan signo.

Condiciones de error que ponen ENO a 0:

SM1.1 (desbordamiento), SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Estas funciones afectan a las siguientes marcas especiales: SM1.0 (cero); SM1.1 (desbordamiento), SM1.2 (negativo)

Entradas/salidas	Operandos	Tipos de datos
IN	VB, IB, QB, MB, SB, SMB, LB, VW, IW, QW, MW, SW, SMW, LW, T, C, AIW, VD, ID, QD, MD, SD, SMD, LD, HC, AC, constante, *VD, *AC, *LD	BYTE, INT, DINT
OUT	VB, IB, QB, MB, SB, SMB, LB, VW, IW, QW, MW, SW, SMW, T, C, LW, VD, ID, QD, MD, SMD, SD, LD, AC, *VD, *AC, *LD	BYTE, INT, DINT

Ejemplo de las operaciones Incrementar y Decrementar

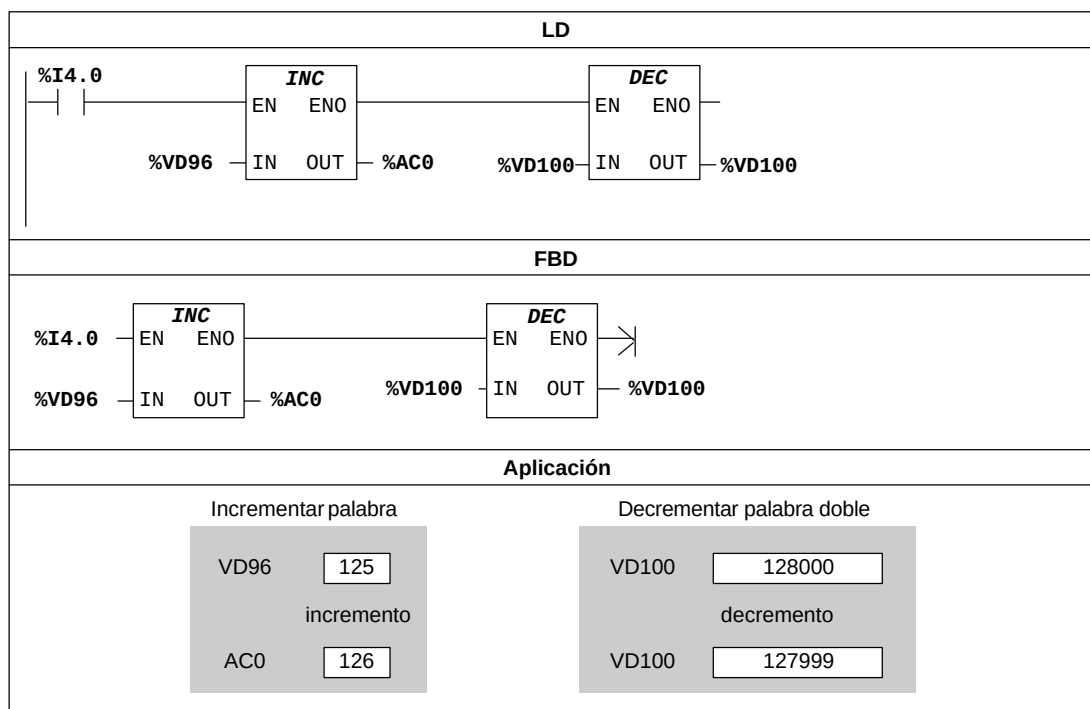


Figura 10-8 Ejemplos de las funciones Incrementar y Decrementar en LD y FBD

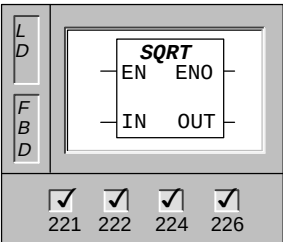
10.6 Operaciones con funciones numéricas (IEC)

En la tabla 10-5 se indican las páginas donde se describen las operaciones con funciones numéricas IEC no normalizadas a las que hace referencia el presente apartado.

Tabla 10-5 Operaciones con funciones numéricas IEC no normalizadas

Descripción	Página
Operación PID	9-87

Raíz cuadrada



La operación **Raíz cuadrada** saca la raíz cuadrada de un valor especificado por IN y deposita el resultado en OUT.

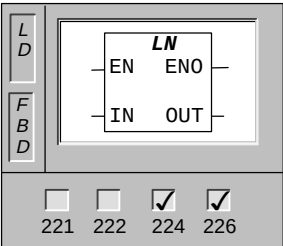
Condiciones de error que ponen ENO a 0:
SM1.1 (desbordamiento), SM4.3 (tiempo de ejecución),
0006 (direccionamiento indirecto)

Esta función afecta a las siguientes marcas especiales:
SM1.0 (cero); SM1.1 (desbordamiento); SM1.2 (negativo)

Si SM1.1 (marca de desbordamiento) está activada, los demás bits de estado aritméticos se borrarán y el operando de salida no se alterará.

Entradas/salidas	Operandos	Tipos de datos
IN	VD, ID, QD, MD, SMD, SD, LD, AC, constante, *VD, *AC, *LD	REAL
OUT	VD, ID, QD, MD, SMD, SD, LD, AC, *VD, *AC, *LD	REAL

Logaritmo natural



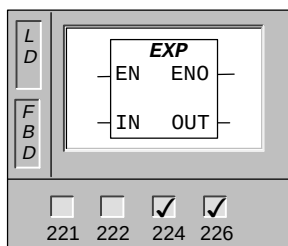
La operación **Logaritmo natural** calcula el logaritmo natural del valor de IN y deposita el resultado en OUT.

Condiciones de error que ponen ENO a 0:
SM1.1 (desbordamiento), 0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales:
SM1.0 (cero); SM1.1 (desbordamiento); SM1.2 (negativo)

Entradas/salidas	Operandos	Tipos de datos
IN	VD, ID, QD, MD, SMD, SD, LD, AC, constante, *VD, *AC, *LD	REAL
OUT	VD, ID, QD, MD, SMD, SD, LD, AC, *VD, *AC, *LD	REAL

Exponencial natural



La operación **Exponencial natural** ejecuta el cálculo exponencial de la constante e elevada a la potencia del valor de IN y deposita el resultado en OUT.

Condiciones de error que ponen ENO a 0:

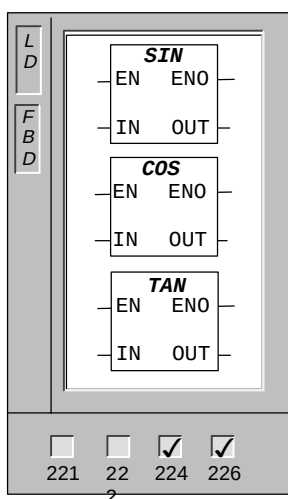
SM1.1 (desbordamiento), 0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales:

SM1.0 (cero); SM1.1 (desbordamiento); SM1.2 (negativo)

Entradas/salidas	Operandos	Tipos de datos
IN	VD, ID, QD, MD, SMD, SD, LD, AC, constante, *VD, *AC, *LD	REAL
OUT	VD, ID, QD, MD, SMD, SD, LD, AC, *VD, *AC, *LD	REAL

Seno, Coseno, Tangente



La operación **Seno** calcula el seno del valor del ángulo IN y deposita el resultado en OUT. El ángulo de entrada se indica en radianes. Si el ángulo se indica en grados, es preciso multiplicarlo por $\pi / 180$ para convertirlo a radianes.

La operación **Coseno** calcula el coseno del valor del ángulo IN y deposita el resultado en OUT. El ángulo de entrada se indica en radianes. Si el ángulo se indica en grados, es preciso multiplicarlo por $\pi / 180$ para convertirlo a radianes.

La operación **Tangente** calcula la tangente del valor del ángulo IN y deposita el resultado en OUT. El ángulo de entrada se indica en radianes. Si el ángulo se indica en grados, es preciso multiplicarlo por $\pi / 180$ para convertirlo a radianes.

Condiciones de error que ponen ENO a 0:

SM1.1 (desbordamiento), 0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales:

SM1.0 (cero); SM1.1 (desbordamiento); SM1.2 (negativo)

Entradas/salidas	Operandos	Tipos de datos
IN	VD, ID, QD, MD, SMD, SD, LD, AC, constante, *VD, *AC, *LD	REAL
OUT	VD, ID, QD, MD, SMD, SD, LD, AC, *VD, *AC, *LD	REAL

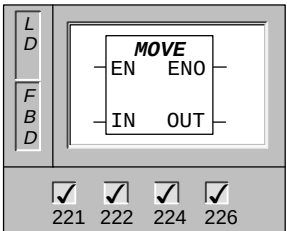
10.7 Operaciones de transferencia (IEC)

En la tabla 10-6 se indican las páginas donde se describen las operaciones de transferencia (IEC) no normalizadas a las que hace referencia el presente apartado.

Tabla 10-6 Operaciones de transferencia (IEC) no normalizadas

Descripción	Página
Operaciones de invertir	9-105
Lectura directa y transferencia de bytes	9-106
Escritura directa y transferencia de bytes	9-106

Transferir (MOVE)



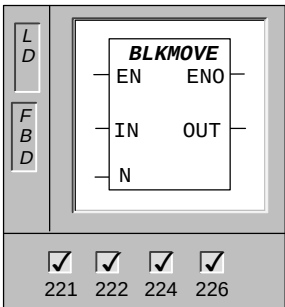
La función **Transferir y asignar valores** transfiere el valor IN a la dirección OUT. Esta función ejecuta una operación de asignación. El parámetro de entrada no se modifica durante la ejecución.

Los tipos de datos de entrada y salida pueden variar pero deben ser del mismo tipo.

Condiciones de error que ponen ENO a 0:
SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Entradas/salidas	Operandos	Tipos de datos
IN	VB, IB, QB, MB, SB, SMB, LB, VW, IW, QW, MW, SM, SMW, LW, T, C, AIW, VD, ID, QD, MD, SMD, SD, LD, HC, &VB, &IB, &QB, &MB, &SB, AC, constante, *VD, *AC, *LD	BYTE, WORD, INT, DWORD, DINT, REAL
OUT	VB, IB, QB, MB, SB, SMB, LB, VW, IW, QW, MW, SW, SMW, LW, T, C, AQW, VD, ID, QD, MD, SMD, SD, LD, AC, *VD, *AC, *LD	BYTE, WORD, INT, DWORD, DINT, REAL

Transferir en bloque



La operación **Transferir en bloque** transfiere un número determinado de palabras (N) indicado por la dirección IN a la dirección OUT. N puede estar comprendido entre 1 y 255.

Los tipos de datos de entrada y salida pueden variar pero deben ser del mismo tipo.

La función Transferir en bloque es una función IEC no normalizada.

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto), 0091 (operando fuera de área)

Entradas/salidas	Operandos	Tipos de datos
IN	VB, IB, QB, MB, SB, SMB, LB, VW, IW, QW, MW, SM, SMW, LW, T, C, AIW, VD, ID, QD, MD, SMD, SD, LD, *VD, *AC, *LD	BYTE, WORD, DWORD, INT, DINT
OUT	VB, IB, QB, MB, SB, SMB, LB, VW, IW, QW, MW, SW, SMW, LW, T, C, AQW, VD, ID, QD, MD, SMD, SD, LD, AC, *VD, *AC, *LD	BYTE, WORD, DWORD, INT, DINT
N	VB, IB, QB, MB, SB, SMB, LB, AC, constante, *VD, *AC, *LD	Byte

Ejemplos de operaciones de transferir

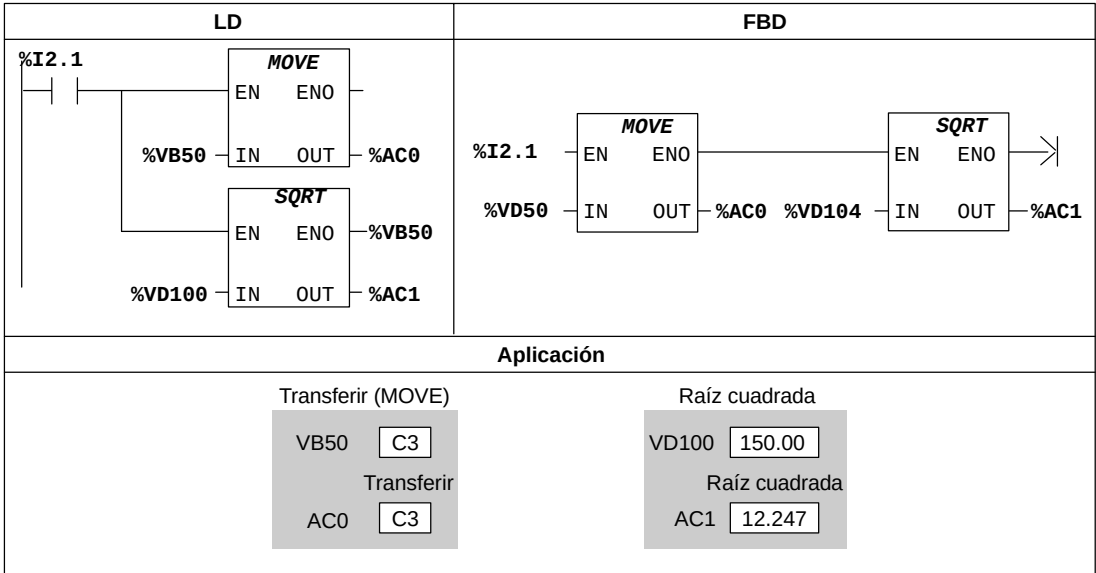
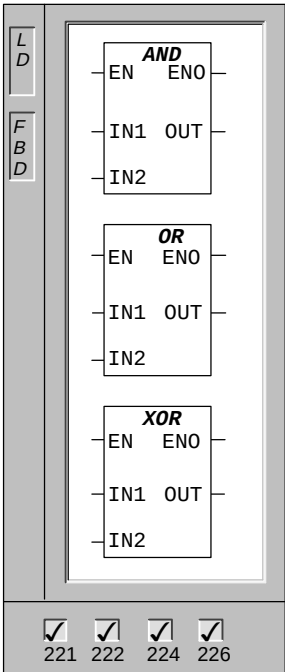


Figura 10-9 Ejemplo de una operación de transferencia en LD y FBD

10.8 Operaciones lógicas (IEC)

No existen operaciones lógicas IEC no normalizadas.

AND, OR, XOR



La función **AND** combina mediante Y los bits correspondientes de IN1 e IN2 y carga el resultado en OUT.

La función **OR** combina mediante O los bits correspondientes de IN1 e IN2 y carga el resultado en OUT.

La función **XOR** combina mediante O-exclusiva los bits correspondientes de IN1 e IN2 y carga el resultado en OUT.

Los tipos de datos de entrada y salida pueden variar pero deben ser del mismo tipo.

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales: SM1.0 (cero)

Entradas/salidas	Operandos	Tipos de datos
IN1, IN2	VB, IB, QB, MB, SB, SMB, LB, VW, IW, QW, MW, SW, SMW, AIW, T, C, LW, VD, ID, QD, MD, SD, SMD, LD, HC, AC, constante, *VD, *AC, *LD	BYTE, WORD DWORD
OUT	VB, IB, QB, MB, SB, SMB, LB, VW, IW, QW, MW, SW, SMW, T, C, LW, VD, ID, QD, MD, SD, SMD, LD, AC, *VD, *AC, *LD	BYTE, WORD DWORD

Ejemplo de las operaciones AND, OR y XOR

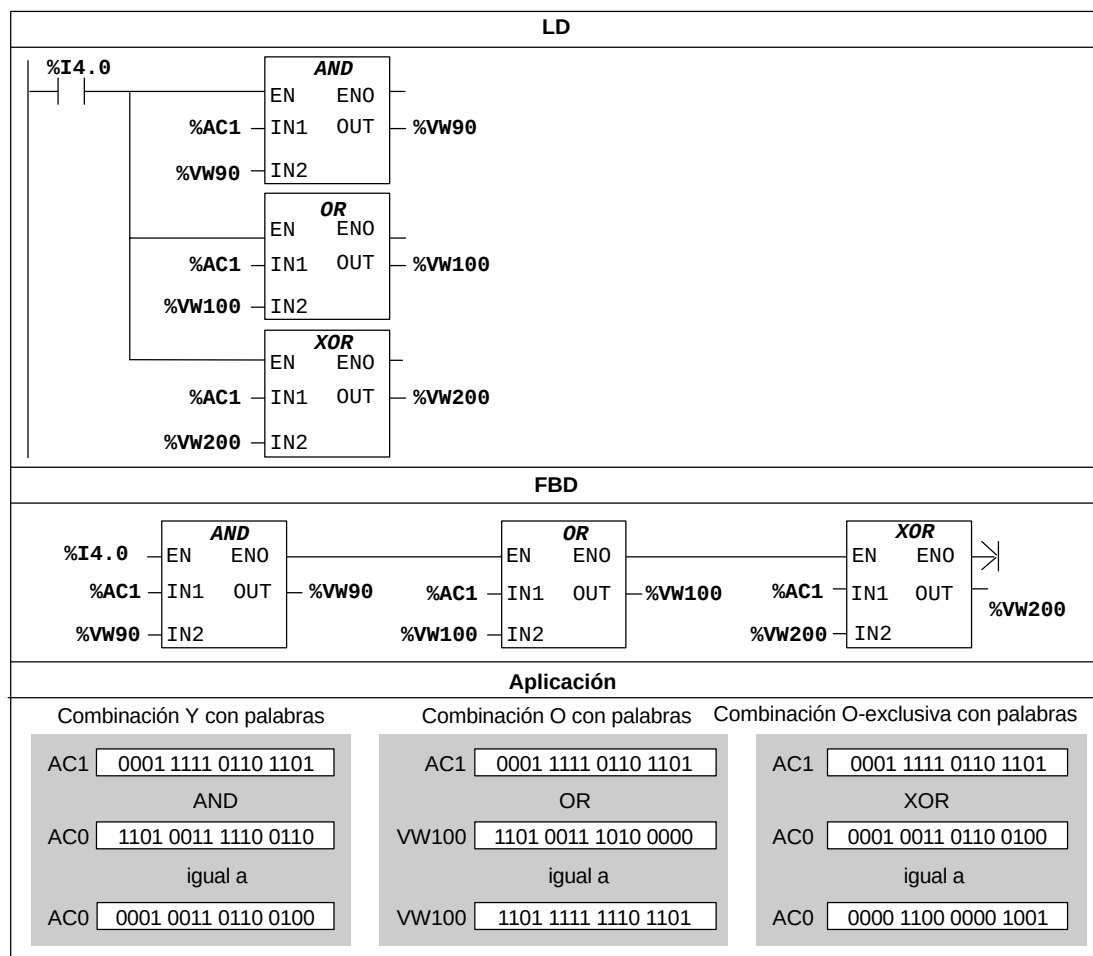
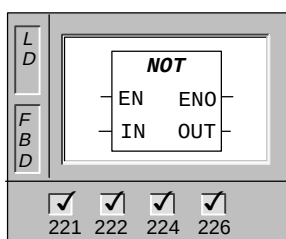


Figura 10-10 Ejemplo de las operaciones AND, OR y XOR

NOT



La función **NOT** invierte los bits correspondientes de IN y carga el resultado en OUT.

Los tipos de datos de entrada y salida pueden variar pero deben ser del mismo tipo.

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales: SM1.0 (cero)

Entradas/salidas	Operandos	Tipos de datos
IN	VB, IB, QB, MB, SB, SMB, LB, VW, IW, QW, MW, SW, SMW, AIW, T, C, LW, VD, ID, QD, MD, SD, SMD, LD, HC, AC, constante, *VD, *AC, *LD	BYTE, WORD DWORD
OUT	VB, IB, QB, MB, SB, SMB, LB, VW, IW, QW, MW, SW, SMW, T, C, LW, VD, ID, QD, MD, SD, SMD, LD, AC, *VD, *AC, *LD	BYTE, WORD DWORD

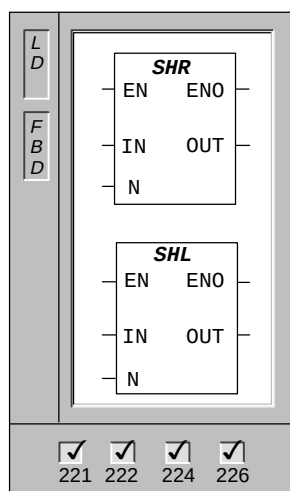
10.9 Operaciones de desplazamiento y rotación (IEC)

En la tabla 10-7 se indican las páginas donde se describen las operaciones de desplazamiento (IEC) no normalizadas a las que hace referencia el presente apartado.

Tabla 10-7 Operaciones IEC no normalizadas

Descripción	Página
Registro de desplazamiento	9-127

Desplazar a la derecha, Desplazar a la izquierda



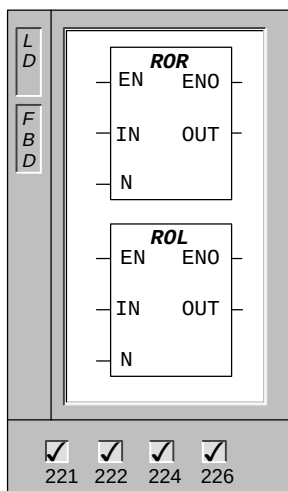
La función **Desplazar a la derecha** desplaza el valor indicado por la variable IN hacia la derecha tantas posiciones como indique N. El resultado se deposita en la variable indicada por OUT. Cada bit se rellena con un cero cuando es desplazado hacia la derecha. El bit de signo se desplaza cuando se utilizan tipos de datos con signo.

La función **Desplazar a la izquierda** desplaza el valor indicado por la variable IN hacia la izquierda tantas posiciones como indique N. El resultado se deposita en la variable indicada por OUT. Cada bit se rellena con un cero cuando es desplazado hacia la izquierda. El bit de signo se desplaza cuando se utilizan tipos de datos con signo.

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Entradas/salidas	Operandos	Tipos de datos
IN	VB, IB, QB, MB, SB, SMB, LB, VW, IW, QW, MW, SW, SMW, LW, T, C, AIW, VD, ID, QD, MD, SD, SMD, LD, HC, AC, constante, *VD, *LD, *AC	BYTE, WORD DWORD
N	VB, IB, QB, MB, SB, SMB, LB, AC, constante, *VD, *LD, *AC	Byte
OUT	VB, IB, QB, MB, SB, SMB, LB, VW, IW, QW, MW, SW, SMW, LW, T, C, VD, ID, QD, MD, SD, SMD, LD, AC, *VD, *LD, *AC	BYTE, WORD DWORD

Rotar a la derecha, Rotar a la izquierda



Las operaciones **Rotar a la derecha** y **Rotar a la izquierda** rotan el valor de la entrada (IN) a la derecha y a la izquierda respectivamente, tantas posiciones como indique el valor de desplazamiento (N) y cargan el resultado en la salida (OUT).

La rotación es circular. En ROR, el bit cero se rota al bit más significativo. En ROL, el bit más significativo se rota al bit cero. El bit de signo se desplaza cuando se utilizan tipos de datos con signo.

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Entradas/salidas	Operandos	Tipos de datos
IN	VB, IB, QB, MB, SB, SMB, LB, VW, IW, QW, MW, SW, SMW, LW, T, C, AIW, VD, ID, QD, MD, SD, SMD, LD, HC, AC, constante, *VD, *LD, *AC	BYTE, WORD DWORD
N	VB, IB, QB, MB, SB, SMB, LB, AC, constante, *VD, *LD, *AC	Byte
OUT	VB, IB, QB, MB, SB, SMB, LB, VW, IW, QW, MW, SW, SMW, LW, T, C, VD, ID, QD, MD, SD, SMD, LD, AC, *VD, *LD, *AC	BYTE, WORD DWORD

Ejemplos de operaciones de rotación y desplazamiento

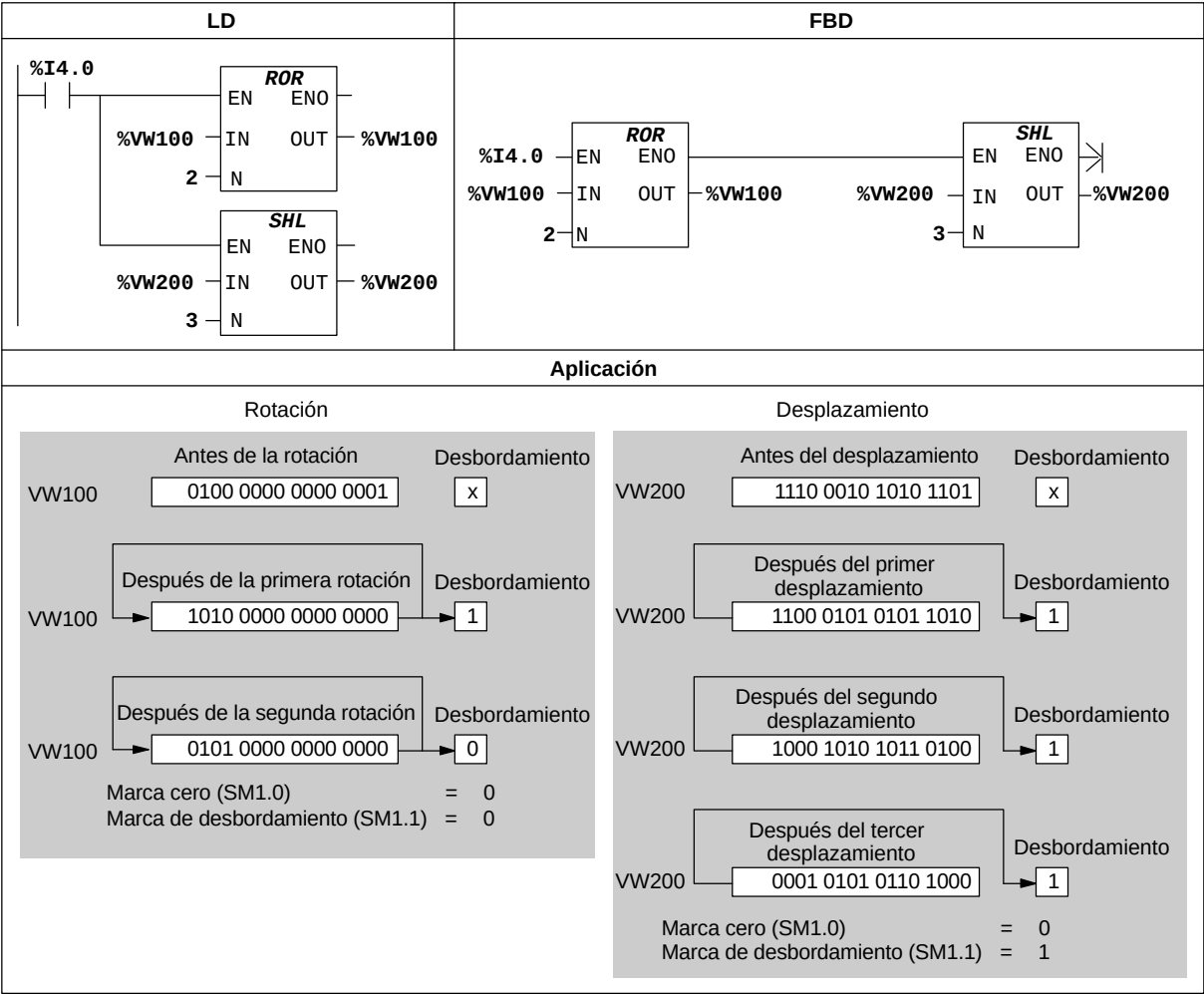


Figura 10-11 Ejemplo de las funciones de desplazamiento y rotación en LD y FBD

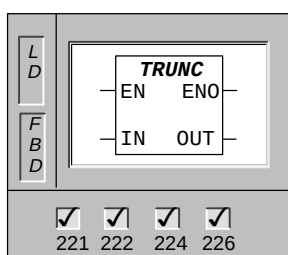
10.10 Operaciones de conversión (IEC)

En la tabla 10-8 se indican las páginas donde se describen las operaciones de conversión (IEC) no normalizadas a las que hace referencia el presente apartado.

Tabla 10-8 Operaciones de conversión (IEC) no normalizadas

Descripción	Página
Decodificar	9-135
Codificar	9-135
Segmento	9-137
Convertir de ASCII a hexadecimal, Convertir de hexadecimal a ASCII	9-139
Convertir de entero a ASCII	9-140
Convertir de entero doble a ASCII	9-142
Convertir de real a ASCII	9-143

Truncar



La función **Truncar** convierte un número real (IN) en un valor de entero doble y deposita el resultado en OUT. El resultado no se redondea.

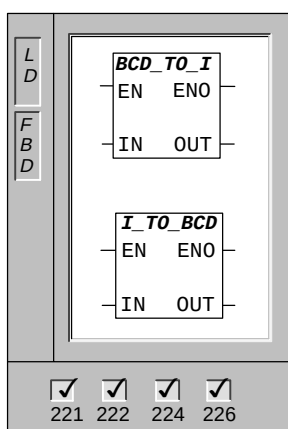
Condiciones de error que ponen ENO a 0:

SM1.1 (desbordamiento), SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Esta función afecta a las siguientes marcas especiales:
SM1.1 (desbordamiento)

Entradas/salidas	Operandos	Tipos de datos
IN	VD, ID, QD, MD, SD, SMD, LD, AC, constante, *VD, *AC, *LD	REAL
OUT	VD, ID, QD, MD, SD, SMD, LD, AC, *VD, *AC, *LD	DINT

Convertir de BCD a entero, Convertir de entero a BCD



La función **Convertir de BCD a entero** convierte el valor BCD (decimal codificado en binario) de entrada (IN) en un valor de entero y carga el resultado en la variable indicada por OUT.

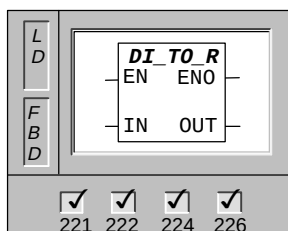
La función **Convertir de entero a BCD** convierte el valor de entero de entrada en un valor BCD (decimal codificado en binario) y carga el resultado en OUT.

Condiciones de error que ponen ENO a 0: SM1.6 (BCD), SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Estas funciones afectan a las siguientes marcas especiales:
SM1.6 (BCD no válido)

Entradas/salidas	Operandos	Tipos de datos
IN	VW, IW, QW, MW, SW, SMW, LW, T, C, AIW, AC, constante, *VD, *LD, *AC	WORD
OUT	VW, IW, QW, MW, SW, SMW, LW, T, C, AC, *VD, *AC, *LD	WORD

Convertir de entero doble a real

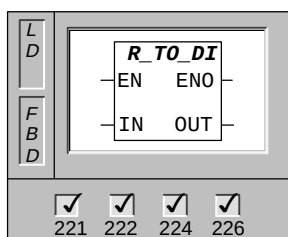


La función **Convertir de entero doble a real** convierte un entero de 32 bits con signo (IN) en un número real de 32 bits y carga el resultado en la variable indicada por OUT.

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Entradas/salidas	Operandos	Tipos de datos
IN	VD, ID, QD, MD, SD, SMD, LD, HC, AC, constante, *VD, *LD, *AC	DINT
OUT	VD, ID, QD, MD, SD, SMD, LD, AC, *VD, *LD, *AC	REAL

Convertir de real a entero doble

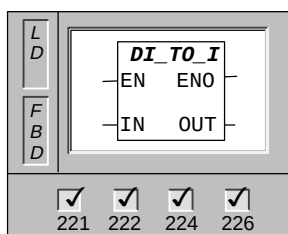


La función **Convertir de real a entero doble** convierte un valor de número real (N) en un valor de entero doble y carga el resultado en la variable indicada por OUT.

Condiciones de error que ponen ENO a 0:
SM1.1 (desbordamiento), SM4.3 (tiempo de ejecución),
0006 (direccionamiento indirecto)

Entradas/salidas	Operandos	Tipos de datos
IN	VD, ID, QD, MD, SD, SMD, LD, AC, constante, *VD, *LD, *AC	REAL
OUT	VD, ID, QD, MD, SD, SMD, LD, AC, *VD, *LD, *AC	DINT

Convertir de entero doble a entero



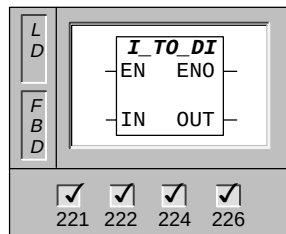
La función **Convertir de entero doble a entero** convierte un entero doble (IN) en un valor de entero y carga el resultado en la variable indicada por OUT.

Condiciones de error que ponen ENO a 0:
SM1.1 (desbordamiento), SM4.3 (tiempo de ejecución),
0006 (direccionamiento indirecto)

Esta función afecta a las siguientes marcas especiales:
SM1.1 (desbordamiento)

Entradas/salidas	Operandos	Tipos de datos
IN	VD, ID, QD, MD, SD, SMD, LD, HC, AC, constante, *VD, *LD, *AC	DINT
OUT	VW, IW, QW, MW, SW, SMW, LW, T, C, AC, *VD, *LD, *AC	INT

Convertir de entero a entero doble

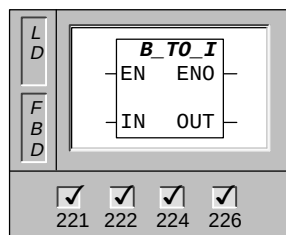


La función **Convertir de entero a entero doble** convierte un valor de entero (IN) en un valor de entero doble y carga el resultado en la variable indicada por OUT.

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Entradas/salidas	Operandos	Tipos de datos
IN	VW, IW, QW, MW, SW, SMW, LW, T, C, AIW, constante, AC, *VD, *LD, *AC	INT
OUT	VD, ID, QD, MD, SD, SMD, LD, AC, *VD, *LD, *AC	DINT

Convertir de byte a entero

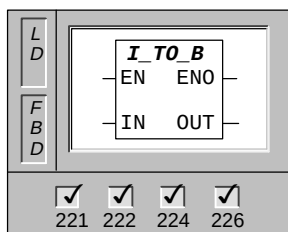


La función **Convertir de byte a entero** convierte el valor de byte (IN) en un valor de entero y carga el resultado en la variable indicada por OUT.

Condiciones de error que ponen ENO a 0: SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Entradas/salidas	Operandos	Tipos de datos
IN	VB, IB, QB, MB, SB, SMB, LB, AC, constante, *VD, *LD, *AC	Byte
OUT	VW, IW, QW, MW, SW, SMW, LW, T, C, AC, *VD, *LD, *AC	INT

Convertir de entero a byte



La función **Convertir de entero a byte** convierte un valor de entero (IN) en un valor de byte y carga el resultado en la variable indicada por OUT.

Condiciones de error que ponen ENO a 0:
SM1.1 (desbordamiento), SM4.3 (tiempo de ejecución),
0006 (direccionamiento indirecto)

Esta función afecta a las siguientes marcas especiales:
SM1.1 (desbordamiento)

Entradas/salidas	Operandos	Tipos de datos
IN	VW, IW, QW, MW, SW, SMW, LW, T, C, AIW, AC, constante, *VD, *LD, *AC	INT
OUT	VB, IB, QB, MB, SB, SMB, LB, AC, *VD, *LD, *AC	Byte

Ejemplos de conversión

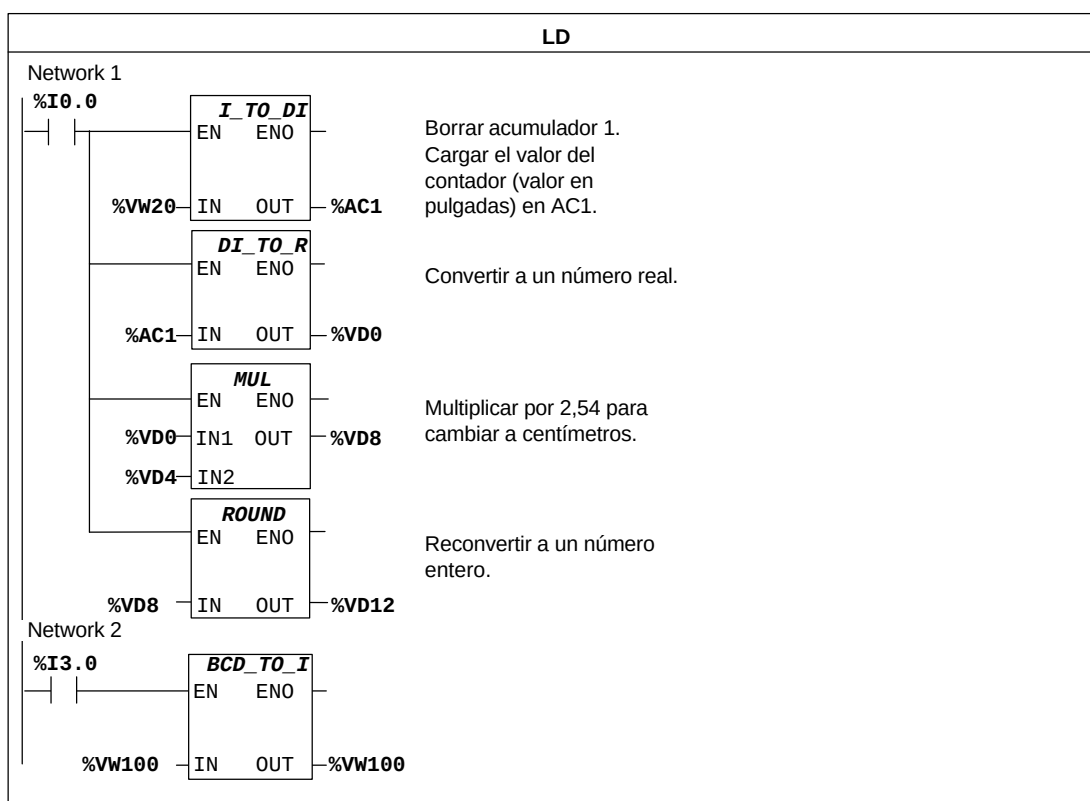


Figura 10-12 Ejemplo de una operación de conversión de un número real en LD

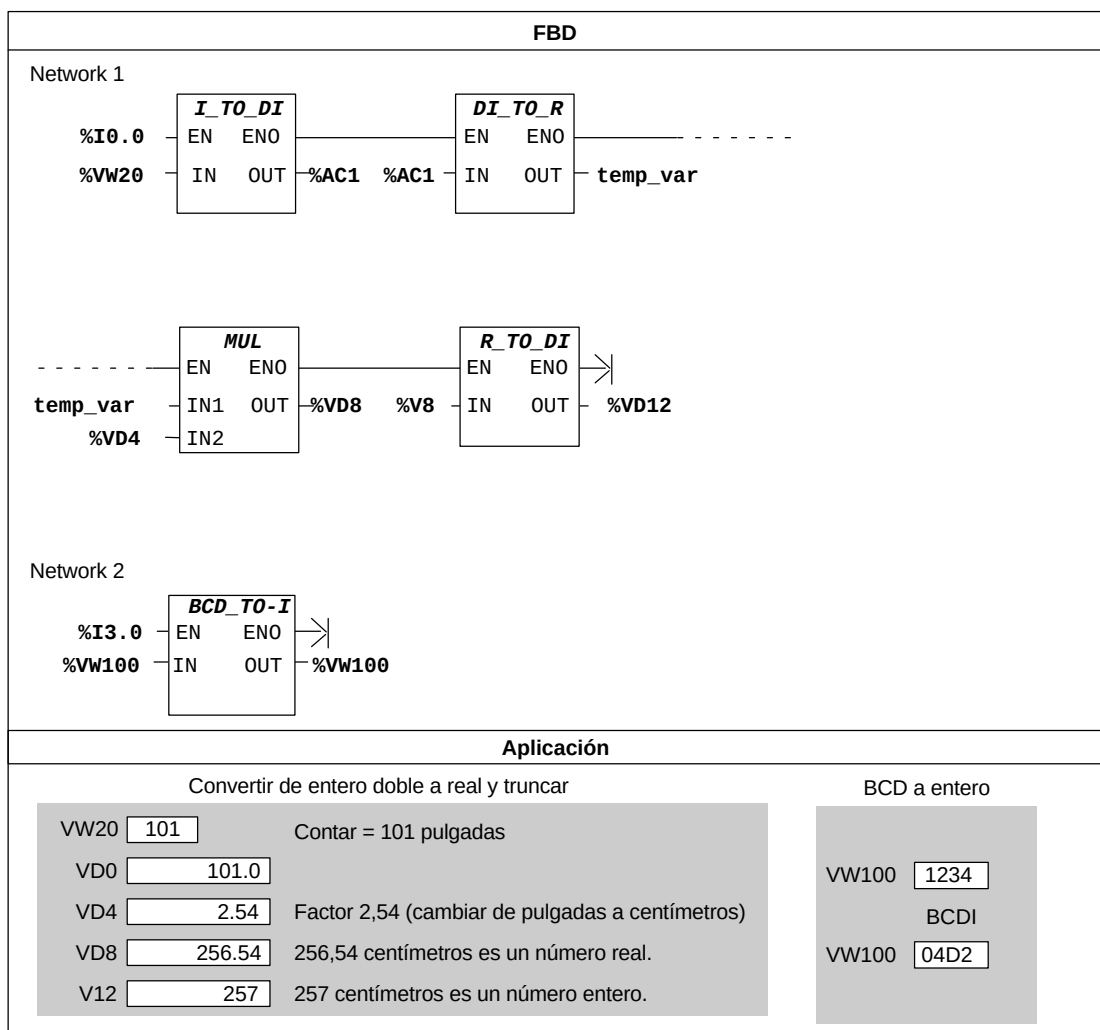


Figura 10-13 Ejemplo de una operación de conversión de un número real en FBD

Utilizar operaciones del protocolo USS para la comunicación con accionamientos

11

En este capítulo se describen las operaciones estándar del protocolo USS, con las que una CPU S7-200 puede controlar un accionamiento MicroMaster. Las operaciones del protocolo USS son una función de STEP 7-Micro/WIN 32 Toolbox.

STEP 7-Micro/WIN 32 Toolbox permite controlar fácilmente los accionamientos MicroMaster, incluyendo para ello subrutinas y rutinas de interrupción preconfiguradas, diseñadas especialmente para la comunicación mediante el protocolo USS. Estas rutinas están disponibles en forma de operaciones en la carpeta "Librerías" del árbol de operaciones de STEP 7-Micro/WIN. Estas nuevas operaciones permiten controlar el accionamiento físico y los parámetros de lectura/escritura.

Cuando se selecciona una operación del protocolo USS, se añaden automáticamente una o más subrutinas asociadas (USS1 hasta USS7), que, sin embargo, se pueden ignorar.

Índice del capítulo

Apartado	Descripción	Página
11.1	Requisitos de las operaciones del protocolo USS	11-2
11.2	Secuencia de programación	11-4
11.3	Operaciones del protocolo USS	11-5
11.4	Conectar los accionamientos	11-17
11.5	Configurar los accionamientos	11-18

11.1 Requisitos de las operaciones del protocolo USS

Para poder ejecutar las operaciones del protocolo USS se necesitan los siguientes recursos de la CPU:

- Tamaño: 1250 a 1750 bytes (dependiendo de las operaciones utilizadas)
- Puerto 0 (únicamente)
- 8 subrutinas y 3 rutinas de interrupción
- Tamaño de la memoria V: 400 bytes y un búfer de 16 bytes para algunas operaciones

Uso de recursos

Las operaciones del protocolo USS utilizan los siguientes recursos:

- Puerto 0: Si el puerto 0 se está utilizando para las operaciones del protocolo USS, no se puede usar para ninguna otra función, incluyendo la comunicación con STEP 7-Micro/WIN. La operación USS_INIT controla la asignación del puerto 0 al protocolo USS o al protocolo PPI. Cuando el puerto 0 se haya asignado al protocolo USS, sólo se podrá utilizar nuevamente para la comunicación con STEP 7-Micro/WIN tras reasignarlo nuevamente a través de una operación USS_INIT diferente, o colocando el selector de la CPU en STOP. Si se interrumpe la comunicación con los accionamientos, éstos se pararán. Al desarrollar programas para aplicaciones del protocolo USS, es recomendable utilizar una CPU 226 o un módulo de ampliación EM 277 PROFIBUS-DP en combinación con una tarjeta PROFIBUS CP incorporada en el PC. Ello permite utilizar un segundo puerto de comunicación, con objeto de poder observar la aplicación mediante STEP 7-Micro/WIN mientras se está ejecutando el protocolo USS.
- Las operaciones del protocolo USS afectarán a todas las direcciones de marcas especiales asociadas a la comunicación Freeport por el puerto 0.
- Espacio en el programa de usuario: Además del espacio ocupado por cada una de las operaciones, las rutinas que soportan el protocolo USS ocupan un espacio adicional en el programa de usuario de hasta 1750 bytes.
- Memoria V: Un bloque de 400 bytes de la memoria V (que comienza en una dirección asignada por el usuario) está reservado para las variables USS. Para algunas operaciones se necesita un búfer de comunicación de 16 bytes (que comienza en una dirección asignada por el usuario). Es aconsejable asignar un búfer unívoco para cada instancia de las operaciones del protocolo USS.
- El protocolo USS necesita 9 subrutinas y 3 rutinas de interrupción.

Configuración de la tabla de símbolos globales

Tras haberse introducido la dirección del primer símbolo, la tabla calcula automáticamente y asigna los demás símbolos contenidos en la tabla. La figura 11-1 muestra la ficha "Protocolo USS" de la tabla de símbolos. Asigne a la tabla una dirección inicial de la memoria V, la cual ocupará 400 bytes.

	Nombre	Dirección	Comentario
1	USS_LOW_V	VD100	(DWORD) Introduzca aquí la dirección inicial de los símbolos del protocolo USS.
2	USS_HIGH_V	VD104	
3	USS_TIME_OUT	VW108	
4	USS_I_LINE	VW110	
5	USS_DAT_PTR0	VD112	
6	USS_ACT_DRV0	VD116	
7	USS_CUR_DRV0	VD120	
8	USS_P0	V124.0	
9	USS_REQ_SNT0	V124.1	
10	USS_RETRY_P0	VB125	
11	USS_ERR_CODE	VB126	
12	USS_CS_D0	VD127	
13	USS_CS_W0	VW127	
14	USS_CS_W2	VW129	
15	USS_CS_B0	VB127	
16	USS_CS_B1	VB128	

Introduzca la dirección inicial.

USR1 Protocolo USS Símbolos UOP

Figura 11-1 Configuración de la tabla de símbolos

Tiempo de comunicación con los accionamientos

La comunicación con los accionamientos es asíncrona al ciclo de la CPU. Por lo general, transcurrirán varios ciclos de la CPU mientras se completa una transacción de comunicación con un accionamiento. Esto depende del número de accionamientos presentes, de la velocidad de transferencia y del tiempo de ciclo de la CPU. La tabla 11-1 muestra los tiempos de las transacciones de comunicación. Una vez que una operación USS_INIT le asigne el puerto 0 al protocolo USS, la CPU muestrea con regularidad todos los accionamientos activos en los intervalos que figuran en la tabla 11-1. El parámetro de timeout de todos los accionamientos se deberá activar para que acepte el tiempo de muestreo.

Tabla 11-1 Tiempo de comunicación con los accionamientos

Velocidad de transferencia (en bit/s)	Intervalo entre los muestreos de los accionamientos activos (en ms)
1200	(460 máx./230 típ.) * número de accionamientos
2400	(240 máx./120 típ.) * número de accionamientos
4800	(130 máx./65 típ.) * número de accionamientos
9600	(80 máx./40 típ.) * número de accionamientos
19200	(50 máx./25 típ.) * número de accionamientos

Restricciones

Sólo se puede activar una operación READ_PM ó WRITE_PM a la vez. Antes de que la lógica pueda iniciar una nueva operación, la salida DONE de cada operación deberá señalar que se ha finalizado la ejecución. Utilice sólo una operación DRV_CTRL por accionamiento.

11.2 Secuencia de programación

A continuación se indica la secuencia de programación cuando se utilicen las operaciones del protocolo USS.

1. Disponga la operación USS_INIT en el programa de usuario. Así se añaden automáticamente al programa varias subrutinas y rutinas de interrupción ocultas. La operación USS_INIT sólo se deberá llamar en un ciclo para iniciar o modificar los parámetros de comunicación USS. En la página 11-5 se describe la operación USS_INIT.
2. Asigne una dirección de la memoria V a la primera dirección de la tabla de símbolos globales USS. Todas las demás direcciones se asignarán automáticamente. Se necesitan 400 bytes consecutivos en total. La figura 11-1 muestra la ficha "Protocolo USS" de la tabla de símbolos.
3. Disponga en el programa sólo una operación DRV_CTRL por cada accionamiento activo. Se puede añadir un número cualquiera de operaciones READ_PM y WRITE_PM, pero sólo una de éstas podrá estar activada.
4. Configure los parámetros de los accionamientos para que concuerden con la velocidad de transferencia y la dirección utilizada en el programa. En el apartado 11.5 se describe cómo configurar los accionamientos.
5. Conecte el cable de comunicación entre la CPU y los accionamientos. Es de suma importancia que todos los equipos controlados que se encuentren unidos al accionamiento (como p.ej. la CPU) se conecten a la misma toma de tierra o al mismo punto neutro que el accionamiento, utilizando para ello un cable corto y grueso.



Cuidado

Al interconectar equipos con potenciales de referencia diferentes pueden circular corrientes indeseadas por el cable de conexión.

Las corrientes indeseadas pueden causar errores de comunicación o deteriorar los equipos.

Para evitar corrientes indeseadas, asegúrese de que todos los equipos que se deban conectar con un cable de comunicación compartan un circuito de referencia, o bien estén aislados entre sí. Para obtener más información al respecto, consulte el tema "Reglas de puesta a tierra de referencia de potencial para circuitos aislados" en el apartado 2.3.

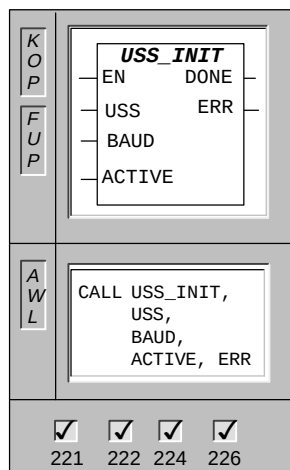
El blindaje deberá conectarse a tierra o al terminal 1 del conector de 9 pines. Es recomendable conectar a tierra el terminal 2–0V del accionamiento MicroMaster.

Nota

Si no puede leer todas las variables de los bloques de las operaciones USS, elija el comando de menú **Ver ► Zoom** y agrande el ancho de la retícula.

11.3 Operaciones del protocolo USS

USS_INIT



La operación **USS_INIT** sirve para activar e inicializar, o bien para desactivar la comunicación con los accionamientos MicroMaster. Antes de poder utilizar cualquier otra operación del protocolo USS, la operación USS_INIT se deberá ejecutar sin errores. La operación se finaliza y el bit DONE se pone a 1 inmediatamente, antes de continuar con la siguiente operación.

La operación se ejecuta en cada ciclo cuando está activada la entrada EN. La operación USS_INIT se deberá ejecutar una sola vez para cada cambio del estado de la comunicación. Por consiguiente, la entrada EN se debe activar cuando se detecte un flanco positivo. Una vez iniciado el protocolo USS, se deberá desactivar ejecutando una nueva operación USS_INIT antes de poder modificar los parámetros de inicialización.

El valor de la entrada USS selecciona el protocolo de comunicación. 1 asigna el puerto 0 al protocolo USS y lo activa. 0 asigna el puerto 0 al protocolo PPI y desactiva el protocolo USS.

BAUD ajusta la velocidad de transferencia a 1200, 2400, 4800, 9600 ó 19.200 bit/s.

ACTIVE indica qué accionamientos están activos. Algunos accionamientos soportan únicamente direcciones comprendidas entre 0 y 30. La figura 11-2 muestra la descripción y el formato de la entrada del accionamiento activo. Todos los accionamientos activos (ACTIVE) se muestrean automáticamente en segundo plano, con objeto de controlar el accionamiento, recoger el estado y evitar timeouts en el enlace serie del accionamiento. Consulte la tabla 11-1 (v. pág. 11-3) para calcular el tiempo entre los muestreos de estado. En el apartado 11.5 se indica cómo configurar el parámetro de timeout del enlace serie (P093).

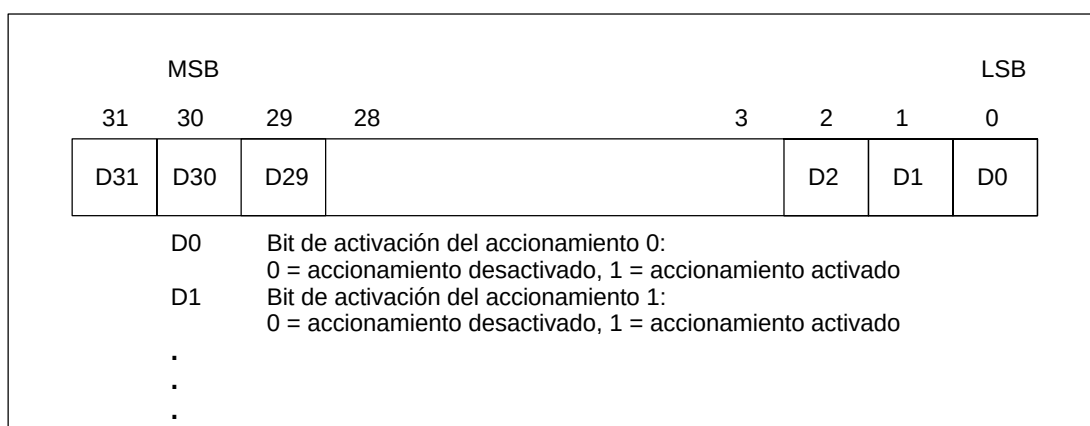


Figura 11-2 Descripción y formato de los accionamientos activos

Una vez finalizada la operación USS_INIT, se activa la salida DONE. El byte de salida ERR contiene el resultado de la ejecución de la operación. En la tabla 11-6 (v. pág. 11-16) figuran los errores que pueden ocurrir al ejecutar la operación.

La tabla 11-2 muestra los operandos y los tipos de datos de las subrutinas USS.

Tabla 11-2 Operandos y tipos de datos de la subrutina USS_INIT

Entradas/salidas	Operandos	Tipos de datos
USS	VB, IB, QB, MB, SB, SMB, LB, AC, constante, *VD, *AC, *LD	BYTE
BAUD	VW, IW, QW, MW, SW, SMW, LW, T, C, AIW, constante, AC *VD, *AC, *LD	WORD
ACTIVE	VD, ID, QD, MD, SD, SMD, LD, AC, constante, *VD, *AC, *LD	DWORD
DONE	I, Q, M, S, SM, T, C, V, L	BOOL
ERR	VB, IB, QB, MB, SB, SMB, LB, AC, *VD, *AC, *LD	BYTE

La figura 11-3 muestra cómo utilizar la subrutina USS_INIT en KOP, FUP y AWL.

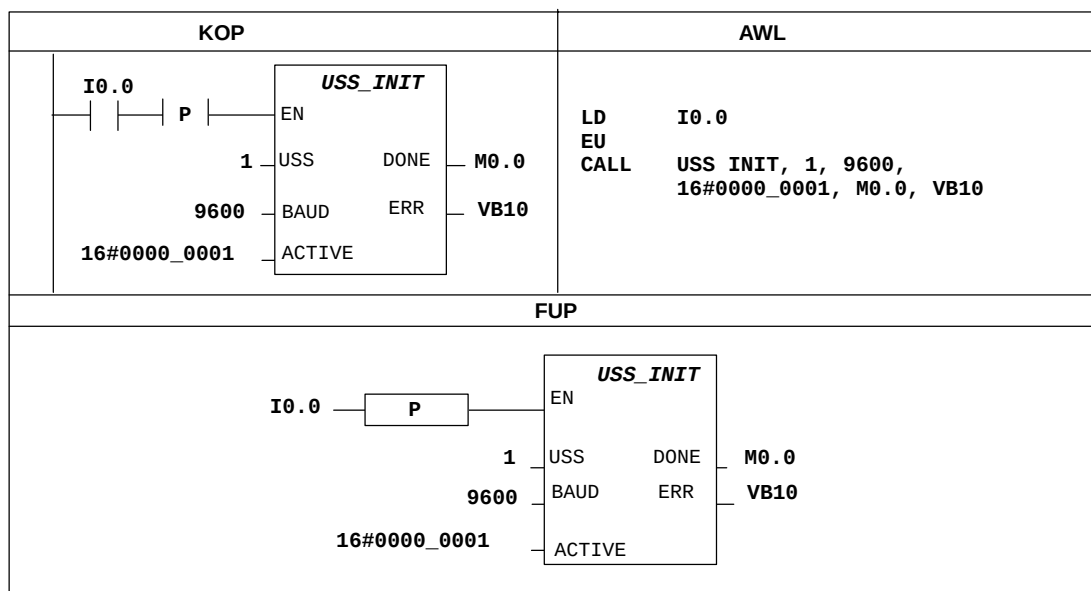
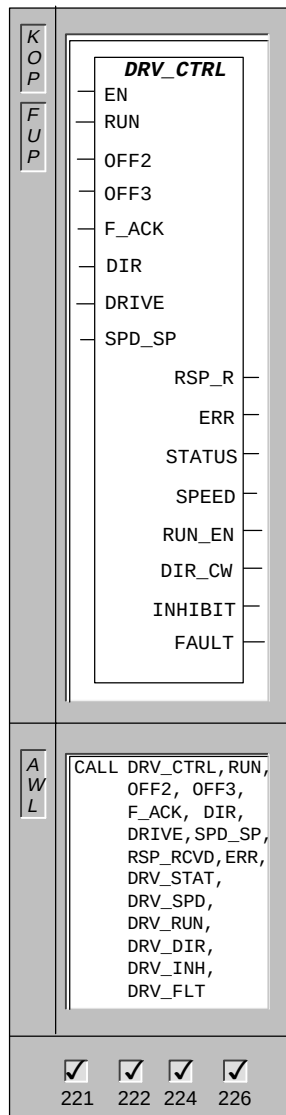


Figura 11-3 Utilizar la subrutina USS_INIT en KOP, FUP y AWL

DRV_CTRL



La operación **DRV_CTRL** sirve para controlar un accionamiento MicroMaster ACTIVO. Esta operación deposita en un búfer de comunicación los comandos seleccionados. Los comandos del búfer se transmiten al accionamiento direccionado (parámetro DRIVE), si el accionamiento se ha seleccionado en el parámetro ACTIVE de la operación USS_INIT. Utilice sólo una operación DRV_CTRL por accionamiento.

El bit EN deberá estar puesto a 1 para poder activar la operación DRV_CTRL. Por lo general, esta operación siempre está activada.

RUN (RUN/STOP) indica si el accionamiento está activado (1) o desactivado (0). Si está activado el bit RUN, el accionamiento MicroMaster recibirá un comando para que se ponga en marcha a la velocidad indicada y en el sentido deseado. Para que el accionamiento se pueda poner en marcha, se deberán cumplir los siguientes requisitos:

- DRIVE deberá estar seleccionado como ACTIVE en USS_INIT.
- OFF1 y OFF2 deberán estar puestos a 0.
- FAULT e INHIBIT deberán estar puestos a 0.

Si está desactivado el bit RUN, el accionamiento MicroMaster recibirá un comando para que disminuya la velocidad hasta que se pare el motor.

El bit OFF2 sirve para que el accionamiento MicroMaster pueda parar por inercia. El bit OFF3 le ordena al accionamiento MicroMaster que pare rápidamente.

El bit F_ACK indica que se ha detectado un fallo en el accionamiento. El accionamiento borra el fallo (FAULT) cuando F_ACK pasa de bajo a alto.

El bit DIR indica en qué sentido debe girar el accionamiento (0 = giro a la izquierda, 1 = giro a la derecha).

La entrada DRIVE es la dirección del accionamiento MicroMaster a la que se debe enviar el comando DRV_CTRL. Las direcciones válidas están comprendidas entre 0 y 31.

SPD_SP representa la velocidad del accionamiento indicada como porcentaje de la velocidad máxima (–200,0% a 200,0%). Si SPD_SP es un valor negativo, se invertirá el sentido de giro del accionamiento.

Nota

Utilice sólo una operación DRV_CTRL por accionamiento.

El bit RSP_R confirma que se ha recibido una respuesta del accionamiento. Todos los accionamientos ACTIVOS se muestrean para recoger las informaciones de estado más actuales. Cada vez que la CPU recibe una respuesta del accionamiento, el bit RSP_R se activa durante un ciclo, actualizándose todos los valores siguientes.

ERR es un byte de error que contiene el resultado de la última petición de comunicación con el accionamiento. En la tabla 11-6 (v. pág. 11-16) figuran los errores que pueden ocurrir al ejecutar la operación.

STATUS es el valor sin procesar de la palabra de estado que envía el accionamiento. La figura 11-4 muestra los bits de estado de la palabra de estado estándar y de la realimentación principal.

SPEED representa la velocidad del accionamiento indicada como porcentaje de la velocidad máxima (–200,0% a 200,0%).

Nota

Algunos accionamientos indican la velocidad sólo como valor positivo. Si la velocidad es negativa, el accionamiento indicará la velocidad como valor positivo, pero invertirá el bit DIR_CW (bit de sentido).

RUN_EN indica si el accionamiento está en marcha (1) o parado (0).

DIR_CW representa el sentido de giro del accionamiento (0 = hacia la izquierda, 1 = hacia la derecha).

INHIBIT indica el estado del bit de desactivación del accionamiento (0 = activado, 1 = desactivado). Para borrar el bit de desactivación, el bit FAULT deberá estar puesto a 0 (OFF) y las entradas RUN, OFF2 y OFF3 también deberán estar desactivadas.

FAULT representa el estado del bit de fallo (0 = sin fallo, 1 = fallo). El accionamiento visualiza el código del fallo. (Consulte a este respecto el manual del accionamiento). Para borrar el bit FAULT, corrija el fallo y active el bit F_ACK.

La tabla 11-3 muestra los operandos y los tipos de datos de la subrutina DRV_CTRL.

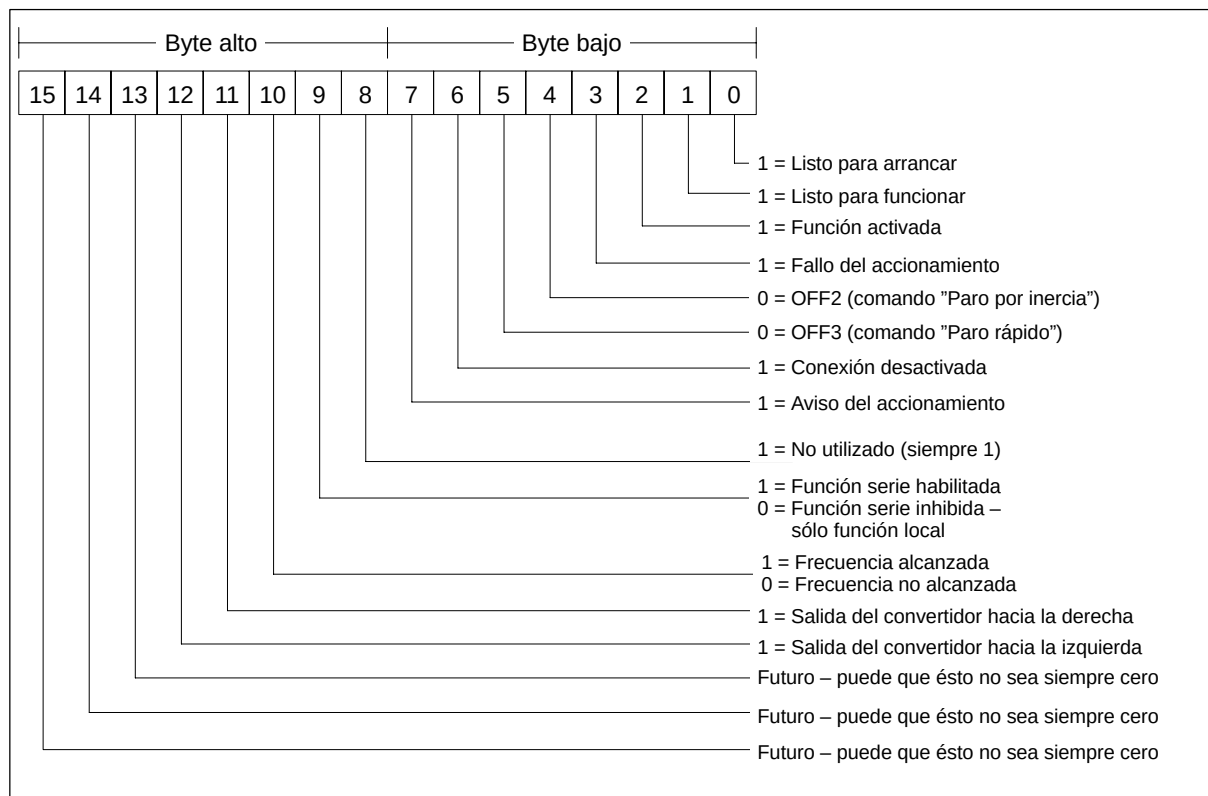


Figura 11-4 Bits de estado de la palabra de estado estándar y de la realimentación principal

Tabla 11-3 Operandos y tipos de datos de la subrutina DRV_CTRL

Entradas/salidas	Operandos	Tipos de datos
RUN	I, Q, M, S, SM, T, C, V, L, circulación de corriente	BOOL
OFF2	I, Q, M, S, SM, T, C, V, L, circulación de corriente	BOOL
OFF3	I, Q, M, S, SM, T, C, V, L, circulación de corriente	BOOL
F_ACK	I, Q, M, S, SM, T, C, V, L, circulación de corriente	BOOL
DIR	I, Q, M, S, SM, T, C, V, L, circulación de corriente	BOOL
DRIVE	VB, IB, QB, MB, SB, SMB, LB, AC, constante, *VD, *AC, *LD	BYTE
SPD_SP	VD, ID, QD, MD, SD, SMD, LD, AC, *VD, *AC, *LD, constante	REAL
RSP_R	I, Q, M, S, SM, T, C, V, L	BOOL
ERR	VB, IB, QB, MB, SB, SMB, LB, AC, *VD, *AC, *LD	BYTE
STATUS	VW, T, C, IW, QW, SW, MW, SMW, LW, AC, AQW, *VD, *AC, *LD	WORD
SPEED	VD, ID, QD, MD, SD, SMD, LD, AC, *VD, *AC, *LD	REAL
RUN_EN	I, Q, M, S, SM, T, C, V, L	BOOL
DIR_CW	I, Q, M, S, SM, T, C, V, L	BOOL

Tabla 11-3 Operandos y tipos de datos de la subrutina DRV_CTRL

Entradas/salidas	Operandos	Tipos de datos
INHIBIT	I, Q, M, S, SM, T, C, V, L	BOOL
FAULT	I, Q, M, S, SM, T, C, V, L	BOOL

La figura 11-5 muestra cómo utilizar la subrutina DRV_CTRL en KOP, FUP y AWL.

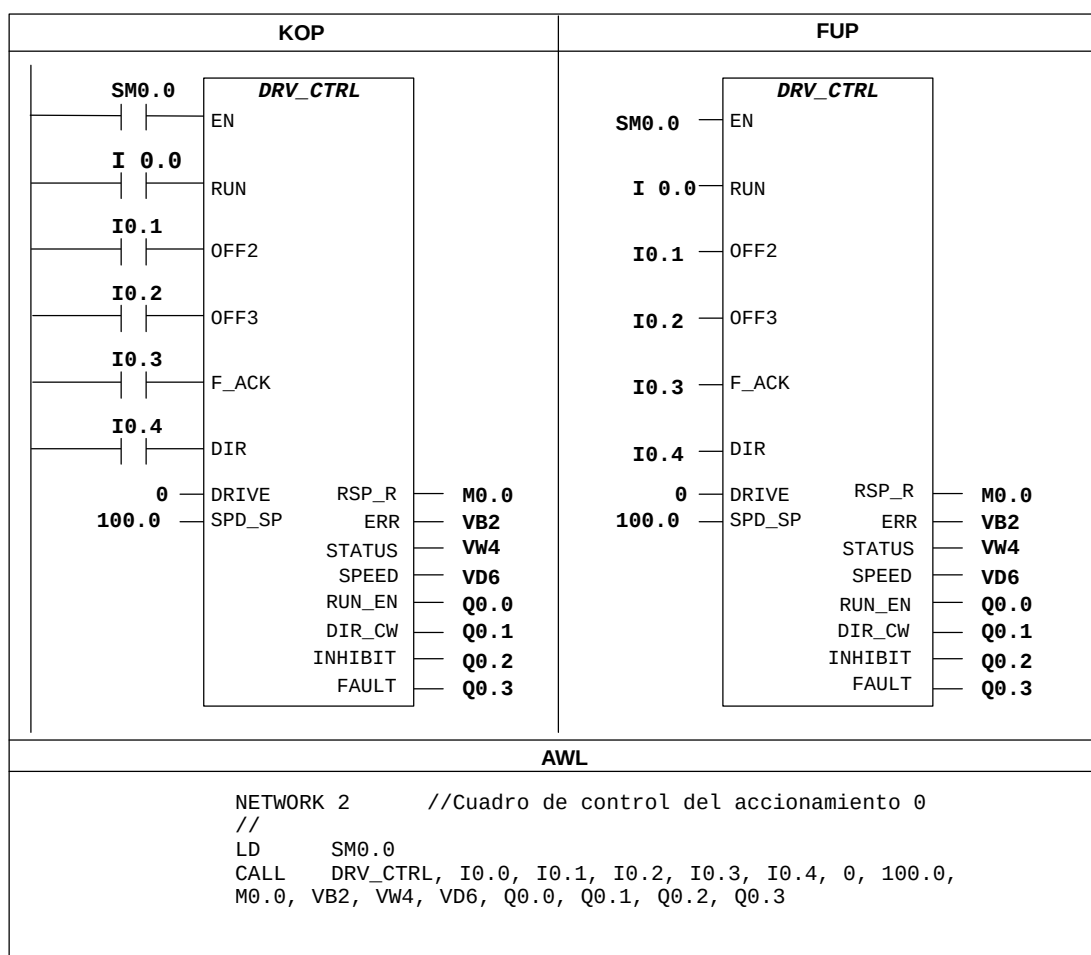
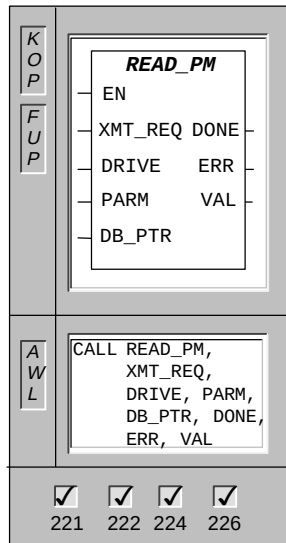


Figura 11-5 Utilizar la subrutina DRV_CTRL en KOP, FUP y AWL

READ_PM



La operación **READ_PM** lee los parámetros del accionamiento. La transacción **READ_PM** se finaliza cuando el accionamiento MicroMaster confirma que ha recibido el comando, o bien cuando se detecta un error. El programa se continúa ejecutando mientras se está esperando una respuesta.

El bit **EN** deberá estar puesto a 1 para poder transmitir una petición, debiendo permanecer activado hasta que se active el bit **DONE**, lo cual indica que se ha finalizado el proceso. La petición **READ_PM** se transmite al accionamiento MicroMaster en cada ciclo, en caso de estar activada la entrada **XMT_REQ**. Por consiguiente, la entrada **XMT_REQ** se debe activar cuando se detecte un flanco positivo, puesto que en cada flanco positivo de la entrada **EN** se envía una petición.

La entrada **DRIVE** es la dirección del accionamiento MicroMaster a la que se debe enviar el comando **READ_PM**. Las direcciones válidas de los accionamientos están comprendidas entre 0 y 31.

PARM representa el número de parámetro.

La dirección del búfer de 16 bytes se le debe suministrar a la entrada **DB_PTR**. La operación **READ_PM** utiliza el búfer para almacenar los resultados del comando enviado al accionamiento MicroMaster.

Cuando se finaliza la operación **READ_PM**, se activa la salida **DONE**. El byte de salida **ERR** contiene el resultado de la ejecución de la operación. En la tabla 11-6 (v. pág. 11-16) figuran los errores que pueden ocurrir al ejecutar la operación.

VAL es el valor del parámetro retornado.

Nota

Sólo se puede activar una operación **READ_PM** ó **WRITE_PM** a la vez.

La tabla 11-4 muestra los operandos y los tipos de datos de la subrutina **READ_PM**.

Tabla 11-4 Operandos y tipos de datos de la subrutina **READ_PM**

Entradas/salidas	Operandos	Tipos de datos
XMT_REQ	I, Q, M, S, SM, T, C, V, L, circulación de corriente (dependiendo de la detección de un flanco positivo)	BOOL
DRIVE	VB, IB, QB, MB, SB, SMB, LB, AC, constante, *VD, *AC, *LD	BYTE
PARM	VW, IW, QW, MW, SW, SMW, LW, T, C, AIW, constante, AC *VD, *AC, *LD	WORD
DB_PTR	&VB	DWORD
DONE	I, Q, M, S, SM, T, C, V, L	BOOL

Tabla 11-4 Operandos y tipos de datos de la subrutina READ_PM

Entradas/salidas	Operandos	Tipos de datos
ERR	VB, IB, QB, MB, SB, SMB, LB, AC, *VD, *AC, *LD	BYTE
VAL	VW, T, C, IW, QW, SW, MW, SMW, LW, AC, AQW, *VD, *AC, *LD	WORD

La figura 11-6 muestra cómo utilizar la subrutina READ_PM en KOP, FUP y AWL.

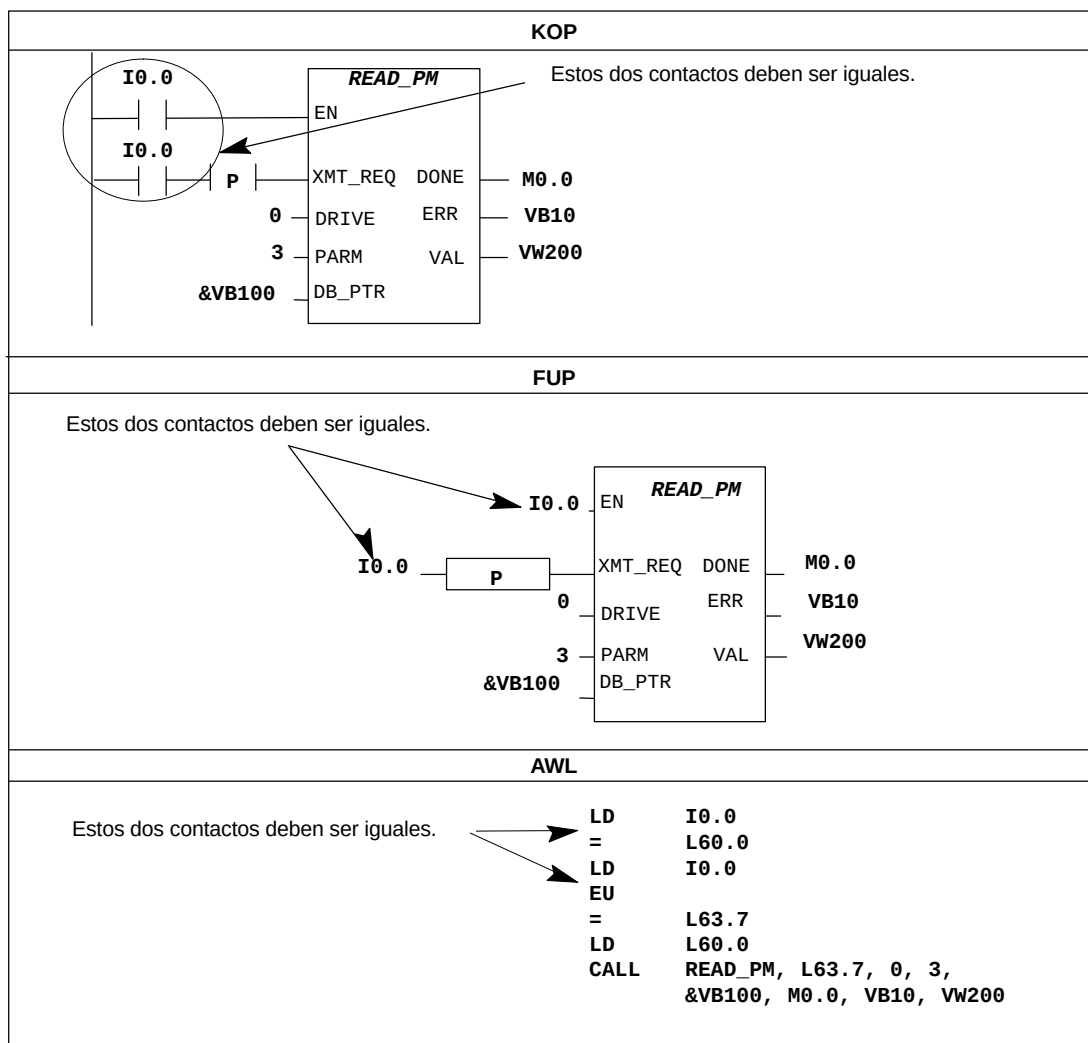
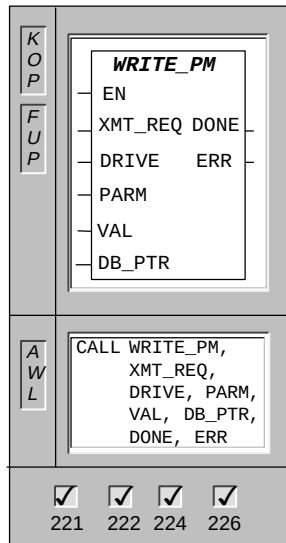


Figura 11-6 Utilizar la subrutina READ_PM en KOP, FUP y AWL

WRITE_PM



La operación **WRITE_PM** escribe un parámetro del accionamiento en una dirección específica. La transacción **WRITE_PM** se finaliza cuando el accionamiento MicroMaster confirma que ha recibido el comando, o bien cuando se detecta un error. El programa se continúa ejecutando mientras se está esperando una respuesta.

El bit **EN** deberá estar puesto a 1 para poder transmitir una petición, debiendo permanecer activado hasta que se active el bit **DONE**, lo cual indica que se ha finalizado el proceso. La petición **WRITE_PM** se transmite al accionamiento MicroMaster en cada ciclo, en caso de estar activada la entrada **XMT_REQ**. Por consiguiente, la entrada **XMT_REQ** se debe activar cuando se detecte un flanco positivo, puesto que en cada flanco positivo de la entrada **EN** se envía una petición.

La entrada **DRIVE** es la dirección del accionamiento MicroMaster a la que se debe enviar el comando **WRITE_PM**. Las direcciones válidas de los accionamientos están comprendidas entre 0 y 31.

PARM representa el número de parámetro. **VAL** es el valor del parámetro a escribir.

La dirección del búfer de 16 bytes se le debe suministrar a la entrada **DB_PTR**. La operación **WRITE_PM** utiliza el búfer para almacenar los resultados del comando enviado al accionamiento MicroMaster.

Cuando se finaliza la operación **WRITE_PM**, se activa la salida **DONE**. El byte de salida **ERR** contiene el resultado de la ejecución de la operación. En la tabla 11-6 (v. pág. 11-16) figuran los errores que pueden ocurrir al ejecutar la operación.

Nota

Sólo se puede activar una operación **READ_PM** ó **WRITE_PM** a la vez.

Cuidado

Cuando la operación WRITE_PM se utiliza para actualizar el juego de parámetros depositado en la EEPROM del accionamiento, se deberá tener en cuenta que no se exceda el número máximo de ciclos de escritura en la EEPROM (aprox. 50.000).

En caso de excederse el número máximo de ciclos de escritura, se corromperán los datos almacenados y se perderán los datos subsiguientes. El número de ciclos de lectura es ilimitado.

Si es necesario escribir con frecuencia en los parámetros del accionamiento, es recomendable ajustar P971 (control de almacenamiento en la EEPROM) a cero.

La tabla 11-5 muestra los operandos y los tipos de datos de la subrutina WRITE_PM.

Tabla 11-5 Operandos y tipos de datos de la subrutina WRITE_PM

Entradas/salidas	Operandos	Tipos de datos
XMT_REQ	I, Q, M, S, SM, T, C, V, L, circulación de corriente (dependiendo de la detección de un flanco positivo)	BOOL
DRIVE	VB, IB, QB, MB, SB, SMB, LB, AC, constante, *VD, *AC, *LD	BYTE
PARM	VW, IW, QW, MW, SW, SMW, LW, T, C, AIW, constante, AC *VD, *AC, *LD	WORD
VAL	VW, IW, QW, MW, SW, SMW, LW, T, C, AIW, constante, AC *VD, *AC, *LD	WORD
DB_PTR	&VB	DWORD
DONE	I, Q, M, S, SM, T, C, V, L	BOOL
ERR	VB, IB, QB, MB, SB, SMB, LB, AC, *VD, *AC, *LD	BYTE

La figura 11-7 muestra cómo utilizar la subrutina WRITE_PM en KOP, FUP y AWL.

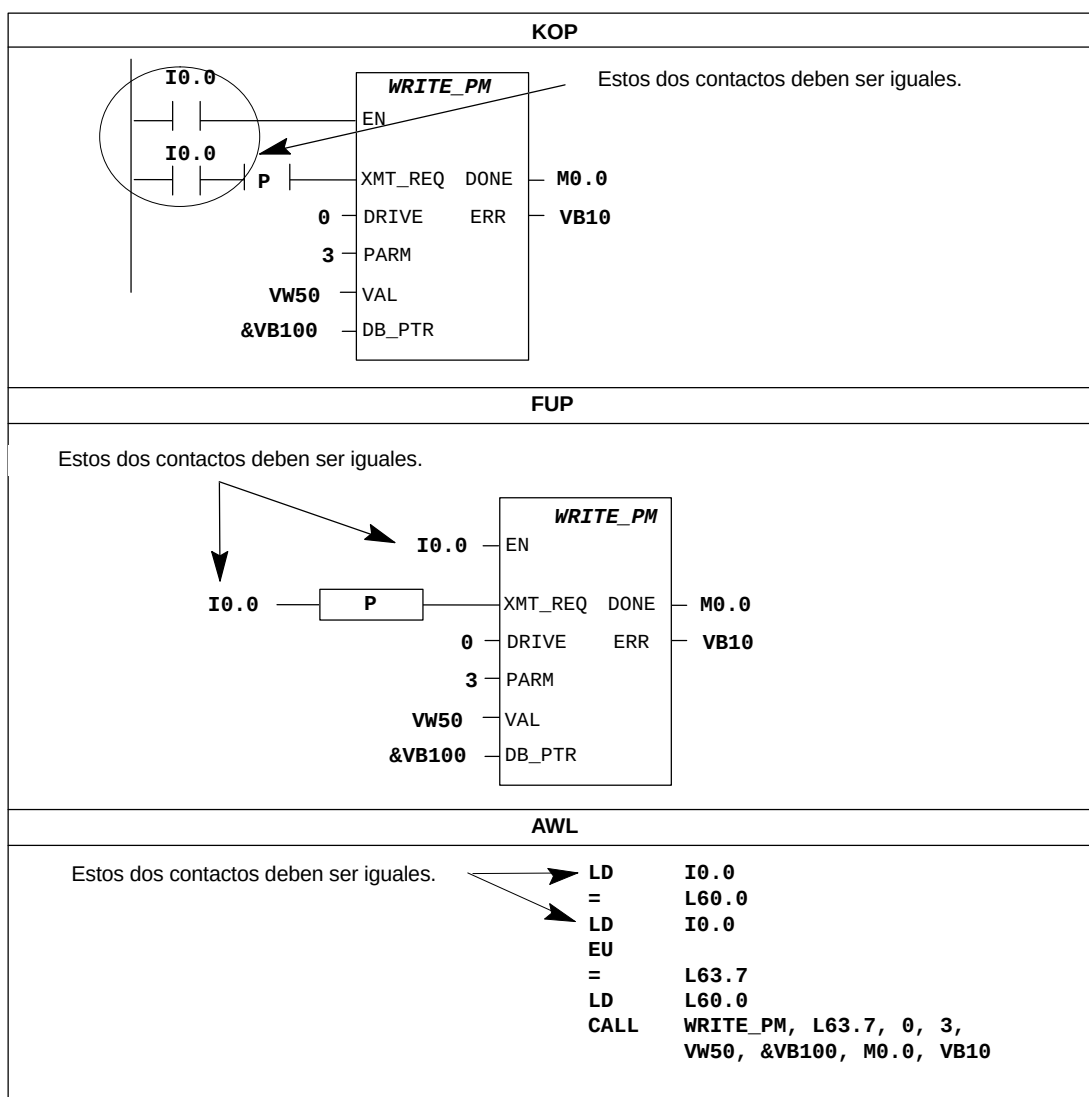


Figura 11-7 Utilizar la subrutina WRITE_PM en KOP, FUP y AWL

Tabla 11-6 Errores de ejecución de las operaciones del protocolo USS

Número de error	Explicación
0	Sin error.
1	El accionamiento no ha respondido.
2	Se ha detectado un error de suma de verificación en la respuesta del accionamiento.
3	Se ha detectado un error de paridad en la respuesta del accionamiento.
4	Una interferencia del programa de usuario ha causado un error.
5	Se ha intentado ejecutar un comando no válido.
6	Se ha indicado una dirección no válida del accionamiento.
7	El puerto de comunicación no se ha ajustado para el protocolo USS.
8	El puerto de comunicación está ocupado procesando una operación.
9	La velocidad del accionamiento está fuera de margen.
10	La longitud de respuesta del accionamiento no es correcta.
11	El primer carácter de la respuesta del accionamiento no es correcto.
12	El carácter de longitud de la respuesta del accionamiento no es correcto.
13	No ha respondido el accionamiento correcto.
14	La dirección DB_PTR indicada no es correcta.
15	El número de parámetro indicado no es correcto.
16	Se ha seleccionado un protocolo no válido.
17	Está activado el protocolo USS. No es posible cambiar.
18	Se ha indicado una velocidad de transferencia no válida.
19	No se ha establecido la comunicación: el accionamiento no está ACTIVO.
20	El parámetro o el valor de la respuesta del accionamiento no es correcto.

11.4 Conectar los accionamientos

El cable PROFIBUS y los conectores estándar sirven para conectar la CPU al accionamiento MicroMaster. Para obtener informaciones más detalladas acerca de los conectores de bus, consulte el capítulo 7. La figura 11-8 muestra cómo polarizar y cerrar el cable de interconexión.



Cuidado

Al interconectar equipos con potenciales de referencia diferentes pueden circular corrientes indeseadas por el cable de enlace.

Las corrientes indeseadas pueden causar errores de comunicación o deteriorar los equipos.

Para evitar corrientes indeseadas, asegúrese de que todos los equipos que se deban conectar con un cable de comunicación compartan un circuito de referencia, o bien estén aislados entre sí. Para obtener más información al respecto, consulte el tema “Reglas de puesta a tierra de referencia de potencial para circuitos aislados” en el apartado 2.3.

El blindaje deberá conectarse a tierra o al terminal 1 del conector de 9 pines. Es recomendable conectar a tierra el terminal 2=0V del accionamiento MicroMaster.

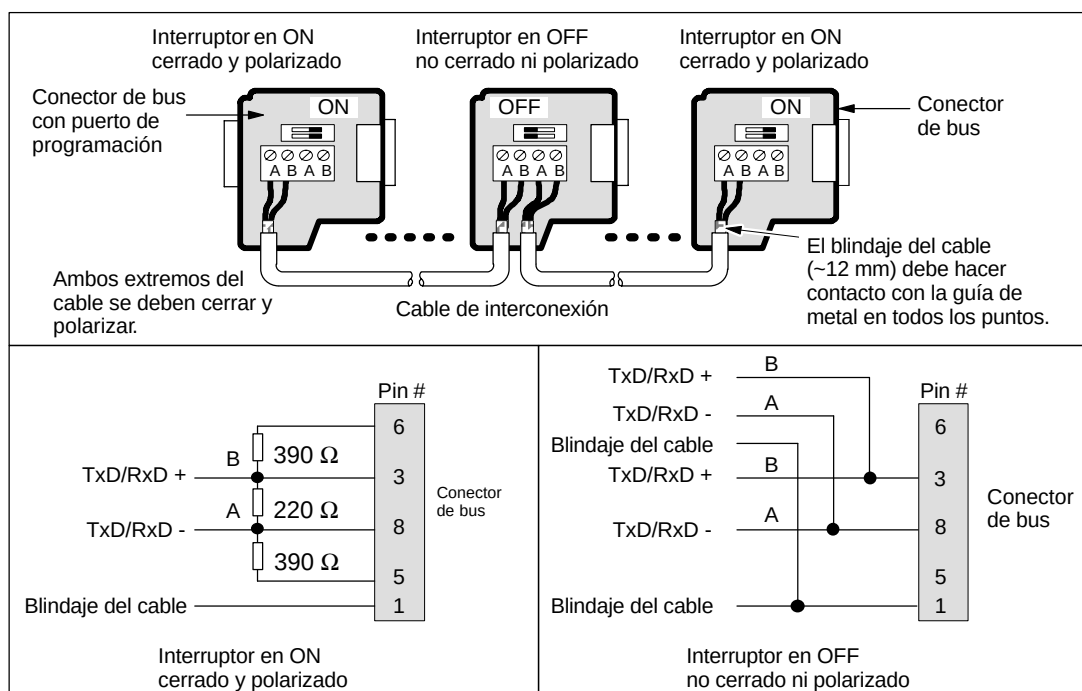


Figura 11-8 Polarizar y cerrar el cable de interconexión

11.5 Configurar los accionamientos

Antes de conectar un accionamiento a la CPU, es preciso verificar que tenga los siguientes parámetros de sistema. Los parámetros se configuran mediante el teclado del accionamiento.

Para configurar los parámetros del accionamiento, siga los pasos siguientes:

1. Reponga los ajustes de fábrica del accionamiento (opcional). Pulse la tecla P: se visualizará P000. Pulse las teclas con flecha arriba o abajo hasta que se visualice P944. Pulse P para introducir el parámetro.

P944=1

2. Habilite el acceso de lectura/escritura a todos los parámetros. Pulse la tecla P. Pulse las teclas con flecha arriba o abajo hasta que se visualice P009. Pulse P para introducir el parámetro.

P009=3

3. Compruebe los ajustes del motor. Los ajustes varían dependiendo del (de los) motor(es) utilizados. Pulse la tecla P. Pulse las teclas con flecha arriba o abajo hasta que se visualicen los ajustes del motor. Pulse P para introducir el parámetro.

P081=Frecuencia nominal del motor (Hz)

P082=Velocidad nominal del motor (r.p.m.)

P083=Intensidad nominal del motor (A)

P084=Tensión nominal del motor (V)

P085=Potencia nominal del motor (kW/HP)

4. Ajuste el modo de control (local o remoto). Pulse la tecla P. Pulse las teclas con flecha arriba o abajo hasta que se visualice P910. Pulse P para introducir el parámetro.

P910=1 Modo de control remoto

5. Ajuste la velocidad de transferencia del puerto serie RS-485. Pulse la tecla P. Pulse las teclas con flecha arriba o abajo hasta que se visualice P092. Pulse P para introducir el parámetro. Pulse las teclas con flecha arriba o abajo hasta que se visualice el número correspondiente a la velocidad de transferencia del puerto serie RS-485. Pulse P para aceptar el ajuste.

P092	3	(1200 bit/s)
	4	(2400 bit/s)
	5	(4800 bit/s)
	6	(9600 bit/s – ajuste estándar)
	7	(19200 bit/s)

6. Introduzca la dirección del esclavo. Todos los accionamientos (31 como máximo) se pueden controlar a través del bus. Pulse la tecla P. Pulse las teclas con flecha arriba o abajo hasta que se visualice P091. Pulse P para introducir el parámetro. Pulse las teclas con flecha arriba o abajo hasta que se visualice la dirección del esclavo deseada. Pulse P para aceptar el ajuste.

P091=0 a 31.

7. Tiempo(s) de aceleración (opcional). Este es el tiempo (indicado en segundos) que el motor necesita para acelerar hasta la frecuencia máxima. Pulse la tecla P. Pulse las teclas con flecha arriba o abajo hasta que se visualice P002. Pulse P para introducir el parámetro. Pulse las teclas con flecha arriba o abajo hasta que se visualice el tiempo de aceleración deseado. Pulse P para aceptar el ajuste.

P002=0 a 650,00

8. Tiempo(s) de desaceleración (opcional). Este es el tiempo (indicado en segundos) que el motor necesita para desacelerar hasta parar por completo. Pulse la tecla P. Pulse las teclas con flecha arriba o abajo hasta que se visualice P003. Pulse P para introducir el parámetro. Pulse las teclas con flecha arriba o abajo hasta que se visualice el tiempo de desaceleración deseado. Pulse P para aceptar el ajuste.

P003=0 a 650,00

9. Timeout de enlace serie. Este es el intervalo máximo admisible entre la recepción de dos telegramas de datos. Esta función se utiliza para desactivar el inversor en caso de un fallo de comunicación.

La temporización comienza tras haberse recibido un telegrama de datos válido. Si no se recibe otro telegrama de datos dentro del período de tiempo indicado, el inversor se desactivará, visualizándose el código de error F008. Ajustando el valor a cero se desconecta el control. Los intervalos de muestreo del estado de los accionamientos se pueden calcular utilizando la tabla 11-1.

Pulse la tecla P. Pulse las teclas con flecha arriba o abajo hasta que se visualice P093. Pulse P para introducir el parámetro. Pulse las teclas con flecha arriba o abajo hasta que se visualice el timeout de enlace serie deseado. Pulse P para aceptar el ajuste.

P093=0 a 240 (0 es el ajuste estándar. El tiempo se indica en segundos.)

10. Consigna nominal del enlace serie. Este valor puede variar, pero equivale típicamente a 50 Hz ó 60 Hz, definiendo el correspondiente valor del 100% de las PVs (variables del proceso) o de las SPs (consignas). Pulse la tecla P. Pulse las teclas con flecha arriba o abajo hasta que se visualice P094. Pulse P para introducir el parámetro. Pulse las teclas con flecha arriba o abajo hasta que se visualice la consigna nominal del enlace serie que se desea. Pulse P para aceptar el ajuste.

P094=0–400,00

11. Compatibilidad con el protocolo USS (opcional). Pulse la tecla P. Pulse las teclas con flecha arriba o abajo hasta que se visualice P095. Pulse P para introducir el parámetro. Pulse las teclas con flecha arriba o abajo hasta que se visualice el número correspondiente a la compatibilidad deseada con el protocolo USS. Pulse P para aceptar el ajuste.

P095 = 0 0,1 Resolución en Hz (estándar)
 1 0,01 Resolución en Hz

12. Control de almacenamiento en la EEPROM (opcional). Pulse la tecla P. Pulse las teclas con flecha arriba o abajo hasta que se visualice P971. Pulse P para introducir el parámetro. Pulse las teclas con flecha arriba o abajo hasta que se visualice el número correspondiente al control deseado de almacenamiento en la EEPROM. Pulse P para aceptar el ajuste.

P971 = 0 Los cambios de la parametrización (incluyendo P971) se pierden cuando se desconecta la alimentación.
 1 (estándar) Los cambios de la parametrización se conservan durante períodos cuando se desconecta la alimentación.

13. Display de operación. Pulse P para salir del modo de parametrización.

Programa de ejemplo del protocolo USS

Las figuras 11-9 a 11-11 muestran programas de ejemplo del protocolo USS en KOP, FUP y AWL.

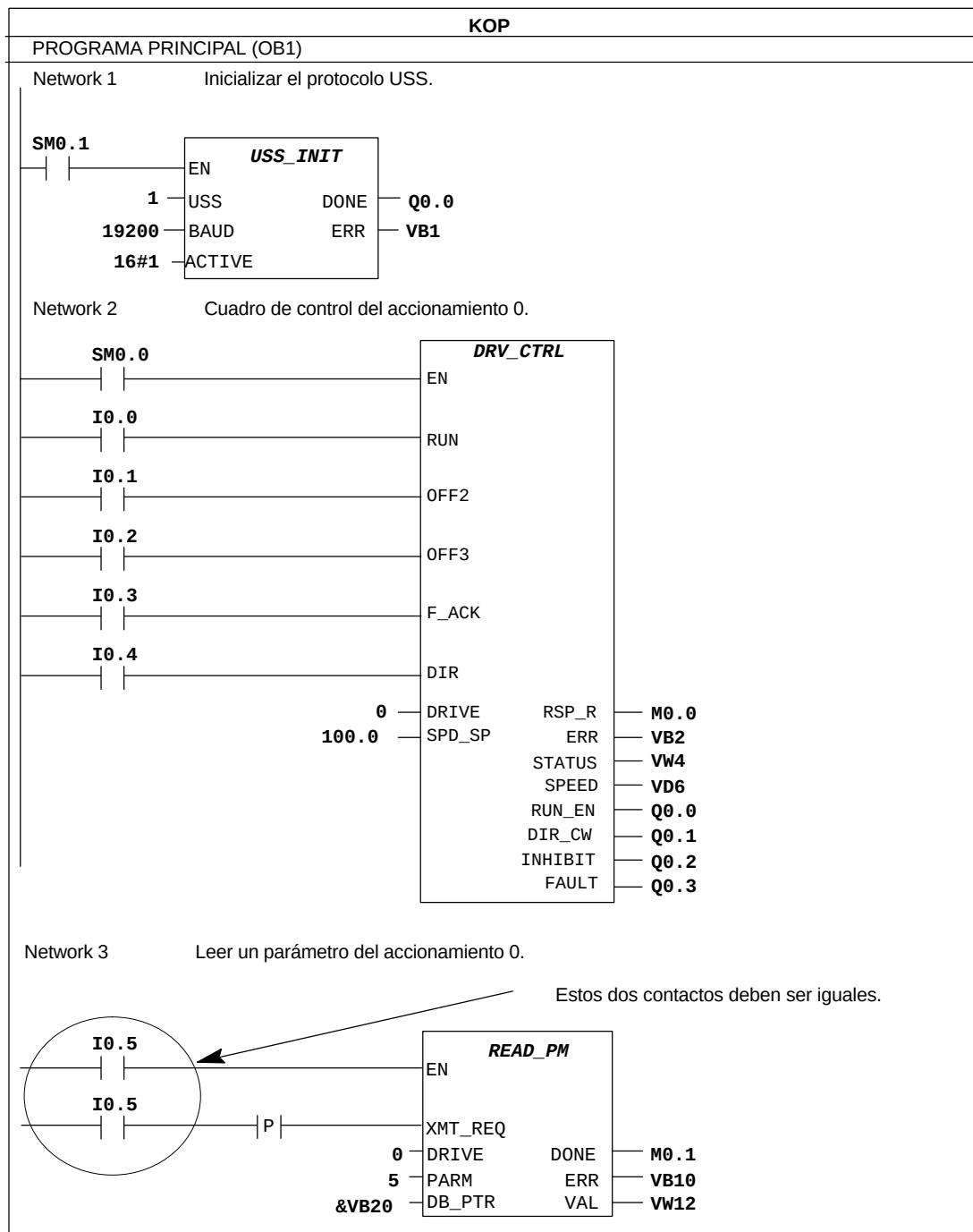


Figura 11-9 Ejemplo de operaciones del protocolo USS en KOP (SIMATIC)

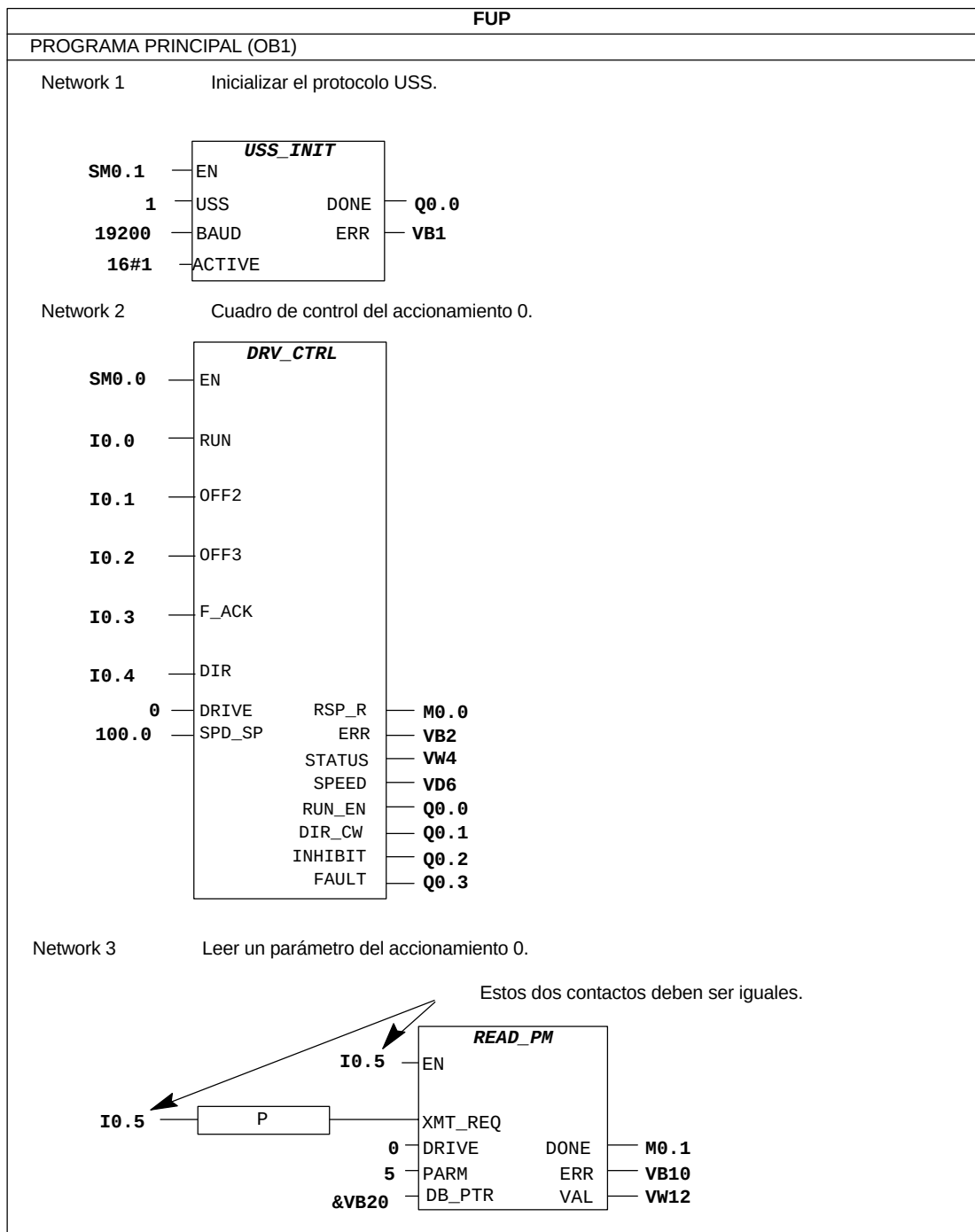


Figura 11-10 Ejemplo de operaciones del protocolo USS en FUP (SIMATIC)

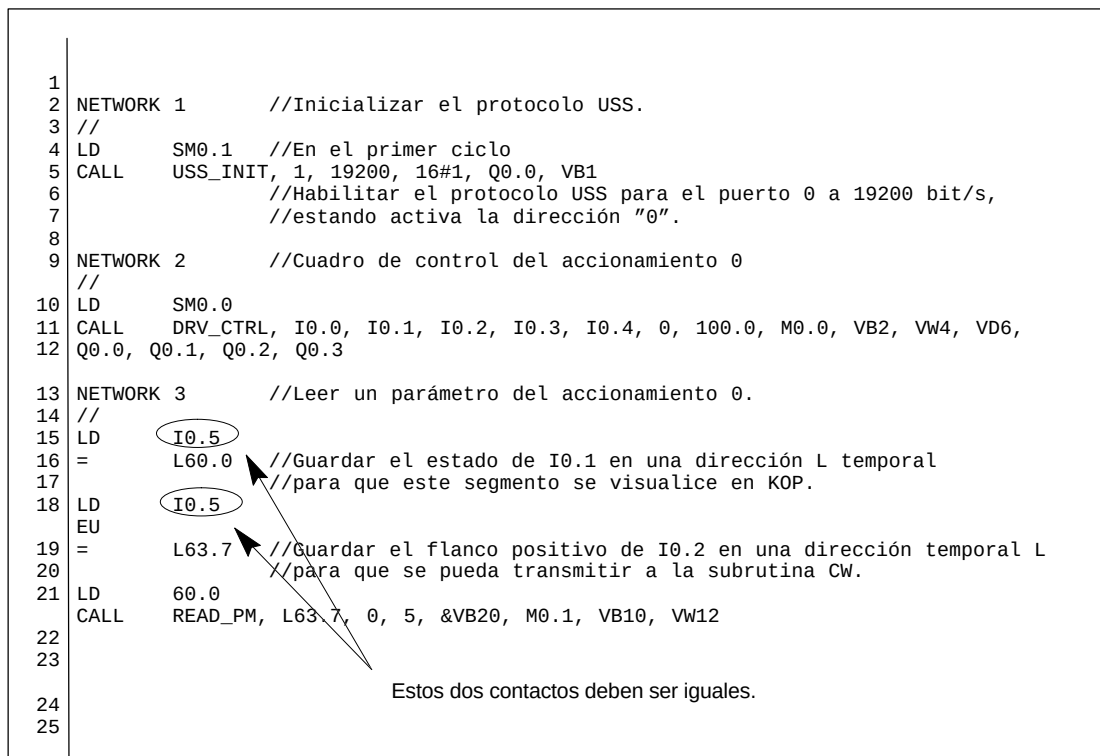


Figura 11-11 Ejemplo de operaciones del protocolo USS en AWL (SIMATIC)

Datos técnicos S7-200

A

Índice del capítulo

Apartado	Descripción	Página
A.1	Datos técnicos generales	A-2
A.2	Datos técnicos de la CPU 221 (firmware 1.1)	A-6
A.3	Datos técnicos de la CPU 222 (firmware 1.1)	A-11
A.4	Datos técnicos de la CPU 224 (firmware 1.1)	A-16
A.5	Datos técnicos de la CPU 226	A-21
A.6	Datos técnicos del módulo de ampliación EM 221 de entradas digitales	A-26
A.7	Datos técnicos de los módulos de ampliación EM 222 de salidas digitales	A-28
A.8	Datos técnicos de los módulos de ampliación EM 223, 4 entradas digitales/4 salidas digitales	A-30
A.9	Datos técnicos de los módulos de ampliación EM 223, 8 entradas digitales/8 salidas digitales	A-33
A.10	Datos técnicos de los módulos de ampliación EM 223, 16 entradas digitales/16 salidas digitales	A-36
A.11	Datos técnicos de los módulos de ampliación EM 231, EM 232 y EM 235 de entradas y/o salidas analógicas	A-39
A.12	Datos técnicos del módulo de ampliación EM 277 PROFIBUS-DP	A-50
A.13	Datos técnicos del módulo de ampliación EM 231 Termopar y RTD	A-67
A.14	Procesador de comunicaciones CP 243-2	A-85
A.15	Cartuchos opcionales	A-88
A.16	Cable de módulo de ampliación	A-89
A.17	Cable PC/PPI	A-90
A.18	Simulador de entradas	A-93

A.1 Datos técnicos generales

Homologaciones nacionales e internacionales

Las características técnicas y las pruebas realizadas con los productos de la gama S7-200 se basan en las homologaciones nacionales e internacionales que se indican a continuación. En la tabla A-1 se indica la conformidad específica con estas homologaciones.

- Underwriters Laboratories, Inc.: UL 508 registrado (Industrial Control Equipment)
- Canadian Standards Association: CSA C22.2 nº 142, certificado (Process Control Equipment)
- Factory Mutual Research: FM clase I, categoría 2, grupo A, B, C, y D Hazardous Locations, T4A Clase I, sección 2, IIC, T4. Las indicaciones de grupo y de temperatura corresponden a un entorno de operación con una temperatura ambiente de 40° C.
- VDE 0160: equipos electrónicos de uso en instalaciones de corriente eléctrica
- Directiva de Baja Tensión de la Comunidad Europea 73/23/CEE (EN 61131-2): Autómatas programables – requisitos del equipo
- Directiva EMC de la Comunidad Europea (CE) 89/336/CEE

Normas de emisión electromagnética:

EN 50081-1: entornos residenciales, comerciales y semi-industriales

EN 50081-2: entornos industriales

Normas de inmunidad electromagnética:

EN 50082-2: entornos industriales

Datos técnicos

La tabla A-1 muestra los datos técnicos generales de las CPUs S7-200 y los módulos de ampliación.

Tabla A-1 Datos técnicos de la gama S7-200

Condiciones ambientales — Transporte y almacenamiento	
IEC 68–2–2, ensayo Bb, calor seco e IEC 68–2–1, ensayo Ab, Frío	–40° C a +70° C
IEC 68–2–30, ensayo Dd, calor húmedo	25° C a 55° C, 95% humedad
IEC 68–2–31, vuelco	100 mm, 4 gotas, desembalado
IEC 68–2–32, caída libre	1m, 5 veces, embalado para embarque
Condiciones ambientales — Funcionamiento	
Condiciones ambientales (aire de entrada 25 mm debajo de la unidad)	0° C a 55° C montaje horizontal 0° C a 45° C montaje vertical 95% humedad no condensante
IEC 68–2–14 Ensayo Nb	5° C a 55° C, 3° C/minuto
IEC 68–2–27 Choque mecánico	15 G, 11 ms impulso, 6 choques en c/u de 3 ejes
IEC 68–2–6 Vibración sinusoidal	0,30 mm pico a pico 10 a 57 Hz; 2G montaje en armario eléctrico, 1G montaje en perfil soporte, 57 a 150 Hz; 10 barridos por eje, 1 octava/minuto
EN 60529, IP20 Protección mecánica	Protege los dedos contra el contacto con alto voltaje, según pruebas realizadas con sondas estándar. Se requiere protección externa contra polvo, impurezas, agua y objetos extraños de menos de 12,5 mm de diámetro.
Compatibilidad electromagnética — Inmunidad ¹ según EN50082–2 ¹	
EN 61000–4–2 (IEC 801–2) Descargas electrostáticas	8 kV descarga en el aire a todas las superficies y al interface de comunicación
EN 50140 (IEC 801–3) Campos electromagnéticos radiados	80 MHz a 1 GHz 10 V/m, 80% modulación con señal de 1 kHz
EN 50141 Perturbaciones conducidas	0,15 MHz a 80 GHz 10 V/m, 80% modulación con señal de 1 kHz
EN 50204 Inmunidad a radioteléfonos digitales	900 MHz ± 5 MHz, 10 V/m, 50% ciclo de trabajo, frecuencia de repetición 200 Hz
EN 61000–4–4 (IEC 801–4) Transitorios eléctricos rápidos	2 kV, 5 kHz con red de unión a la alimentación AC y DC 2 kV, 5 kHz con abrazadera de unión a las E/S digitales y a la comunicación
EN 61000–4–5 (IEC 801–5) Inmunidad a ondas de choque	2 kV asimétrico, 1 kV simétrico 5 impulsos positivos/5 impulsos negativos 0°, +90°, –90° decalaje de fase (para los circuitos de DC 24 V se necesita una protección externa contra sobrecargas)
VDE 0160 Sobrevoltaje no periódico	a AC 85 V línea, 90° decalaje de fase, aplicar cresta de 390 V, impulso de 1,3 ms a AC 180 V línea, 90° decalaje de fase, aplicar cresta de 750 V, impulso de 1,3 ms

Tabla A-1 Datos técnicos de la gama S7-200 (continuación)

Compatibilidad electromagnética — Emisiones conducidas y radiadas según EN50081 –1 2 y –2	
EN 55011, clase A, grupo 1, conducida ¹ 0,15 a 0,5 MHz 0,15 a 5 MHz 5 MHz a 30 MHz	< 79 dB (μV) casi cresta; < 66 dB (μV) promedio < 73 dB (μV) casi cresta; < 60 dB (μV) promedio < 73 dB (μV) casi cresta; < 60 dB (μV) promedio
EN 55011, clase A, grupo 1, radiada ¹ 30 MHz a 230 MHz 230 MHz a 1 GHz	30 dB (μV/m) casi cresta; medido a 30 m 37 dB (μV/m) casi cresta; medido a 30 m
EN 55011, clase B, grupo 1, conducida ² 0,15 a 0,5 MHz 0,15 a 5 MHz 5 MHz a 30 MHz	< 66 dB (μV) decremento casi cresta con frecuencia logarítmica a 56 dB (μV) < 56 dB (μV) decremento promedio con frecuencia logarítmica a 46 dB (μV) < 56 dB (μV) casi cresta; < 46 dB (μV) promedio < 60 dB (μV) casi cresta; < 50 dB (μV) promedio
EN 55011, clase B, grupo 1, radiada ² 30 MHz a 230 kHz 230 MHz a 1 GHz	30 dB (μV/m) casi cresta; medido a 10 m 37 dB (μV/m) casi cresta; medido a 10 m
Prueba de aislamiento a hipervoltajes	
24 V/5 V circuitos nominales 115/230 V circuitos a tierra 115/230 V circuitos hasta 115/230 V circuitos 230 V circuitos hasta 24 V/5V circuitos 115 V circuitos hasta 24 V/5V circuitos	AC 500 V (límites de aislamiento óptico) AC 1.500 V AC 1.500 V AC 1.500 V AC 1.500 V

- 1 La unidad deberá montarse en un soporte metálico puesto a tierra. El S7-200 deberá ponerse a tierra directamente a través del soporte metálico. Los cables se deberán conducir a lo largo de los soportes metálicos.
- 2 La unidad deberá montarse en una caja metálica puesta a tierra. La línea de alimentación de corriente alterna se deberá equipar con un filtro SIEMENS B84115–E–A30 o similar, teniendo el cable una longitud máxima de 25 cm entre los filtros y el S7-200. El cableado de la alimentación DC 24 V y de la alimentación de sensores se deberá apantallar.

Vida útil de los relés

La figura A-1 muestra los datos típicos de rendimiento de los relés suministrados por el comercio especializado. El rendimiento real puede variar dependiendo de la aplicación.

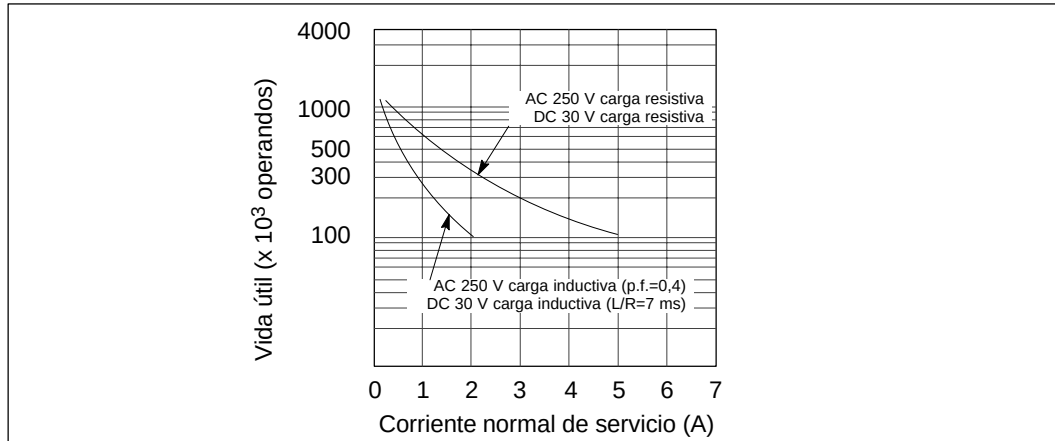


Figura A-1 Vida útil de los relés

A.2 Datos técnicos de la CPU 221 (firmware 1.1)

Tabla A-2 Datos técnicos de la CPU 221 DC/DC/DC y de la CPU 221 AC/DC/Relé

Descripción Nº de referencia	CPU 221 DC/DC/DC 6ES7 211-0AA21-0XB0	CPU 221 AC/DC/Relé 6ES7 211-0BA21-0XB0
Tamaño físico		
Dimensiones (l x a x p)	90 mm x 80 mm x 62 mm	90 mm x 80 mm x 62 mm
Peso	270 g	310 g
Pérdida de corriente (disipación)	3 W	6 W
Características de la CPU		
Entradas digitales integradas	6 entradas	6 entradas
Salidas digitales integradas	4 salidas	4 salidas
Contadores rápidos (valor de 32 bits)		
Total	4 contadores rápidos	4 contadores rápidos
Nº de contadores de fase simple	4, con una frecuencia de reloj de 20 kHz c/u	4, con una frecuencia de reloj de 20 kHz c/u
Nº de contadores de dos fases	2, con una frecuencia de reloj de 20 kHz c/u	2, con una frecuencia de reloj de 20 kHz c/u
Salidas de impulsos	2 a una frecuencia de impulsos de 20 kHz	2 a una frecuencia de impulsos de 20 kHz
Potenciómetros analógicos	1 con resolución de 8 bits	1 con resolución de 8 bits
Interrupciones temporizadas	2 con resolución de 1 ms	2 con resolución de 1 ms
Interrupciones de flanco	4 flancos positivos y/o 4 flancos negativos	4 flancos positivos y/o 4 flancos negativos
Tiempos de filtración de entradas	7 márgenes de 0,2 ms a 12,8 ms	7 márgenes de 0,2 ms a 12,8 ms
Captura de impulsos	6 entradas de captura de impulsos	6 entradas de captura de impulsos
Tamaño del programa (almacenado permanentemente)	2048 palabras	2048 palabras
Tamaño del bloque de datos:		
Almacenamiento permanente	1024 palabras	1024 palabras
Respaldos por condensador de alto rendimiento o pila	1024 palabras	1024 palabras
E/S de ampliación digitales (máx.)	10 E/S	10 E/S
Marcas internas	256 bits	256 bits
Almacenamiento permanente al apagar	112 bits	112 bits
Respaldos por condensador de alto rendimiento o pila	256 bits	256 bits
Temporizadores (total)	256 temporizadores	256 temporizadores
Respaldos por condensador de alto rendimiento o pila	64 temporizadores	64 temporizadores
1 ms	4 temporizadores	4 temporizadores
10 ms	16 temporizadores	16 temporizadores
100 ms	236 temporizadores	236 temporizadores
Contadores (total)	256 contadores	256 contadores
Respaldos por condensador de alto rendimiento o pila	256 contadores	256 contadores
Velocidad de ejecución booleana	0,37 µs por operación	0,37 µs por operación
Velocidad de ejecución de Transferir palabra	34 µs por operación	34 µs por operación
Velocidad de ejecución de temporizadores/contadores	50 µs a 64 µs por operación	50 µs a 64 µs por operación
Velocidad de ejecución de aritmética de precisión simple	46 µs por operación	46 µs por operación
Velocidad de ejecución de aritmética en coma flotante	100 µs a 400 µs por operación	100 µs a 400 µs por operación
Tiempo de respaldo por el condensador de alto rendimiento	típ. 50 h, mín. 8 h a 40° C	típ. 50 h, mín. 8 h a 40° C

Tabla A-2 Datos técnicos de la CPU 221 DC/DC/DC y de la CPU 221 AC/DC/Relé (continuación)

Descripción Nº de referencia	CPU 221 DC/DC/DC 6ES7 211-0AA21-0XB0	CPU 221 AC/DC/Relé 6ES7 211-0BA21-0XB0
Comunicación integrada		
Nº de puertos	1 puerto	1 puerto
Puerto eléctrico	RS-485	RS-485
Aislamiento (señal externa a circuito lógico)	Sin aislamiento	Sin aislamiento
Velocidades de transferencia PPI/MPI	9,6, 19,2 y 187,5 kbit/s	9,6, 19,2 y 187,5 kbit/s
Velocidades de transferencia Freeport	0,3, 0,6, 1,2, 2,4, 4,8, 9,6, 19,2 y 38,4 kbit/s	0,3, 0,6, 1,2, 2,4, 4,8, 9,6, 19,2 y 38,4 kbit/s
Longitud máx. del cable por segmento hasta 38,4 kbit/s	1200 m	1200 m
187,5 kbit/s	1000 m	1000 m
Nº máximo de estaciones		
Por segmento	32 estaciones	32 estaciones
Por red	126 estaciones	126 estaciones
Nº máximo de maestros	32 maestros	32 maestros
Modo maestro PPI (NETR/NETW)	Sí	Sí
Enlaces MPI	4 en total; 2 reservados: 1 para PG y 1 OP	4 en total; 2 reservados: 1 para PG y 1 OP
Cartuchos opcionales		
Cartucho de memoria (almacenamiento permanente)	Programa, datos y configuración	Programa, datos y configuración
Cartucho de pila (tiempo de respaldo de datos)	típ. 200 días	típ. 200 días
Cartucho de reloj (precisión del reloj)	2 minutos por mes a 25° C 7 minutos por mes 0° C a 55° C	2 minutos por mes a 25° C 7 minutos por mes 0° C a 55° C
Alimentación		
Tensión de línea (margen admisible)	DC 20,4 a 28,8 V	AC 85 a 264 V 47 a 63 Hz
Corriente de entrada (sólo CPU) / carga máx.	80/900 mA a DC 24 V	15/60 mA a AC 240 V 30/120 mA a AC 120 V
Extra-corriente de serie (máx.)	10 A a DC 28,8 V	20 A a AC 264 V
Aislamiento (corriente de entrada a lógica)	Sin aislamiento	AC 1500 V
Tiempo de retardo (desde la pérdida de corriente de entrada)	mín. 10 ms de DC 24 V	80 ms de AC 240 V, 20 ms de AC 120 V
Fusible interno (no reemplazable por el usuario)	2 A, 250 V, de acción lenta	2 A, 250 V, de acción lenta
Alimentación para sensores DC 24 V		
Margen de tensión	DC 15,4 a 28,8 V	DC 20,4 a 28,8 V
Corriente máxima	180 mA	180 mA
Rizado corriente parásita	Igual que línea de entrada	Menos de 1 V pico a pico (máx.)
Corriente límite	600 mA	600 mA
Aislamiento (alimentación de sensor a circuito lógico)	Sin aislamiento	Sin aislamiento

Tabla A-2 Datos técnicos de la CPU 221 DC/DC/DC y de la CPU 221 AC/DC/Relé (continuación)

Descripción Nº de referencia	CPU 221 DC/DC/DC 6ES7 211-0AA21-0XB0	CPU 221 AC/DC/Relé 6ES7 211-0BA21-0XB0
Características de las entradas		
Nº de entradas integradas	6 entradas	6 entradas
Tipo de entrada	Sumidero de corriente/fuente (tipo 1 IEC con sumidero de corriente)	Sumidero de corriente/fuente (tipo 1 IEC con sumidero de corriente)
Tensión de entrada		
Tensión máx. continua admisible	DC 30 V	DC 30 V
Sobretensión transitoria	DC 35 V, 0,5 s	DC 35 V, 0,5 s
Valor nominal	DC 24 V a 4 mA, nominal	DC 24 V a 4 mA, nominal
Señal 1 lógica (mín.)	mín. DC 15 V a 2,5 mA	mín. DC 15 V a 2,5 mA
Señal 0 lógica (máx.)	máx. DC 5 V a 1 mA	máx. DC 5 V a 1 mA
Aislamiento (campo a circuito lógico)		
Separación galvánica	AC 500 V, 1 minuto	AC 500 V, 1 minuto
Grupos de aislamiento de:	4 y 2 E/S	4 y 2 E/S
Tiempos de retardo de las entradas		
Entradas filtradas y entradas de interrupción	0,2 a 12,8 ms, seleccionable por el usuario	0,2 a 12,8 ms, seleccionable por el usuario
Entradas de reloj de los contadores rápidos		
Fase simple		
Nivel 1 lógico = DC 15 V a DC 30 V	20 kHz	20 kHz
Nivel 1 lógico = DC 15 V a DC 26 V	30 kHz	30 kHz
Contadores A/B		
Nivel 1 lógico = DC 15 V a DC 30 V	10 kHz	10 kHz
Nivel 1 lógico = DC 15 V a DC 26 V	20 kHz	20 kHz
Conexión de sensor de proximidad de 2 hilos (Bero)		
Corriente de fuga admisible	máx. 1 mA	máx. 1 mA
Longitud del cable		
No apantallado (no HSC)	300 m	300 m
Apantallado	500 m	500 m
Entradas HSC, apantalladas	50 m	50 m
Nº de entradas ON simultáneamente		
40° C	6	6
55° C	6	6
Características de las salidas		
Nº de salidas integradas	4 salidas	4 salidas
Tipo de salida	Estado sólido-MOSFET	Relé, contacto de baja potencia
Tensión de salida		
Margen admisible	DC 20,4 a 28,8 V	DC 5 a 30 V ó AC 5 a 250 V
Valor nominal	DC 24 V	–
Señal 1 lógica a corriente máxima	mín. DC 20 V	–
Señal 0 lógica a 10 K Ω de carga	máx. DC 0,1 V	–
Corriente de salida		
Señal 1 lógica	0,75 A	2,00 A
Nº de grupos de salidas	1	2
Nº de salidas ON (máx.)	4	4
Por grupo – montaje horizontal (máx.)	4	3 y 1
Por grupo – montaje vertical (máx.)	4	3 y 1
Corriente máx. por común/grupo	3,0 A	6,0 A
Carga LEDs	5,0 W	30 W DC/200 W AC
Resistencia estado ON (resistencia contactos)	0,3 Ω	0,2 Ω , máx. si son nuevas
Corriente de derivación por salida	máx. 10 μ A	–
Sobrecorriente momentánea	máx. 8 A, 100 ms	7A al estar cerrados los contactos
Protección contra sobrecargas	no	no

Tabla A-2 Datos técnicos de la CPU 221 DC/DC/DC y de la CPU 221 AC/DC/Relé (continuación)

Descripción Nº de referencia	CPU 221 DC/DC/DC 6ES7 211-0AA21-0XB0	CPU 221 AC/DC/Relé 6ES7 211-0BA21-0XB0
Aislamiento		
Separación galvánica	AC 500 V, 1 minuto	–
Resistencia de aislamiento	–	100 M Ω , mín. si son nuevas
Aislamiento bobina a contacto	–	AC 500 V, 1 minuto
Aislamiento entre contactos abiertos	–	AC 750 V, 1 minuto
En grupos de:	4 E/S	3 y 1 E/S
Carga inductiva, apriete		
Repetición disipación de energía < 0.5 L I ² x frecuencia de conmutación	1 W, en todos los canales	–
Límites tensión de bloqueo	L+ menos 48 V	–
Retardo de las salidas		
OFF a ON (Q0.0 y Q0.1)	máx. 2 μ s	–
ON a OFF (Q0.0 y Q0.1)	máx. 10 μ s	–
OFF a ON (Q0.2 y Q0.3)	máx. 15 μ s	–
ON a OFF (Q0.2 y Q0.3)	máx. 100 μ s	–
Frecuencia de conmutación (salida de impulsos)		
Q0.0 y Q0.1	máx. 20 kHz	máx. 1 Hz
Relé		
Retardo de conmutación	–	máx. 10 ms
Vida útil mecánica (sin carga)	–	10.000.000 ciclos abiertos/cerrados
Vida útil contactos a carga nominal	–	100.000 ciclos abiertos/cerrados
Longitud del cable		
No apantallado	150 m	150 m
Apantallado	500 m	500 m

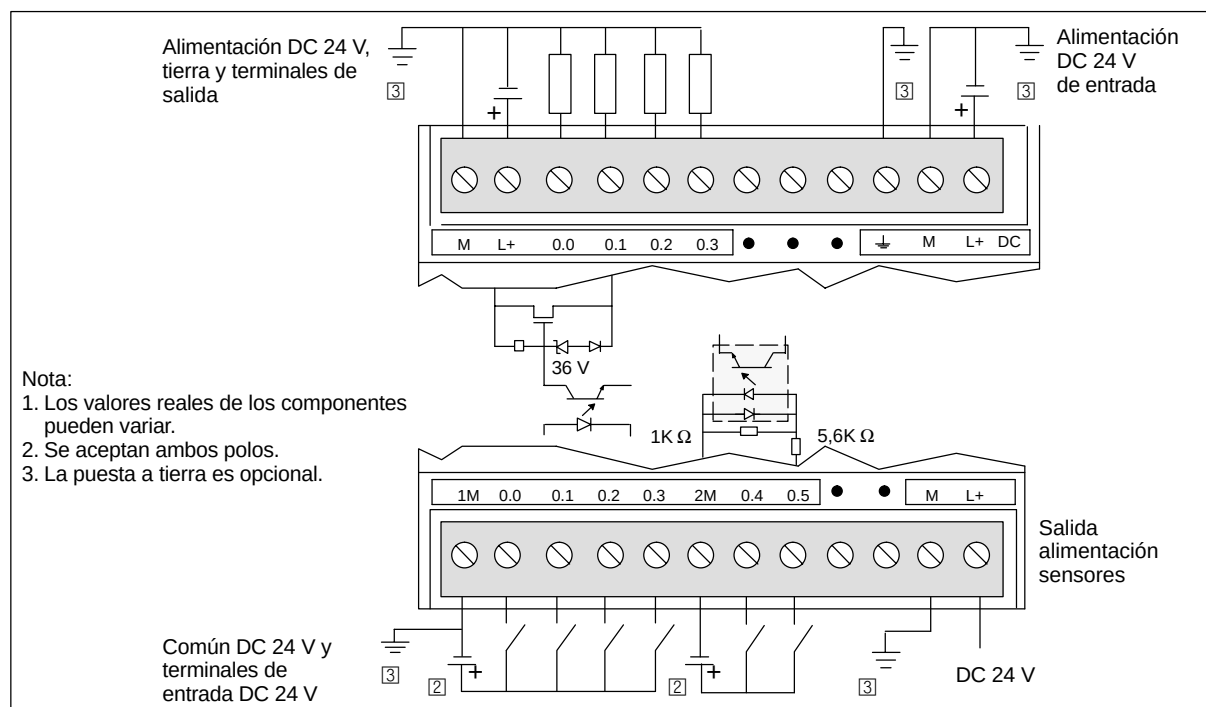


Figura A-2 Identificación de terminales de conexión para la CPU 221 DC/DC/DC

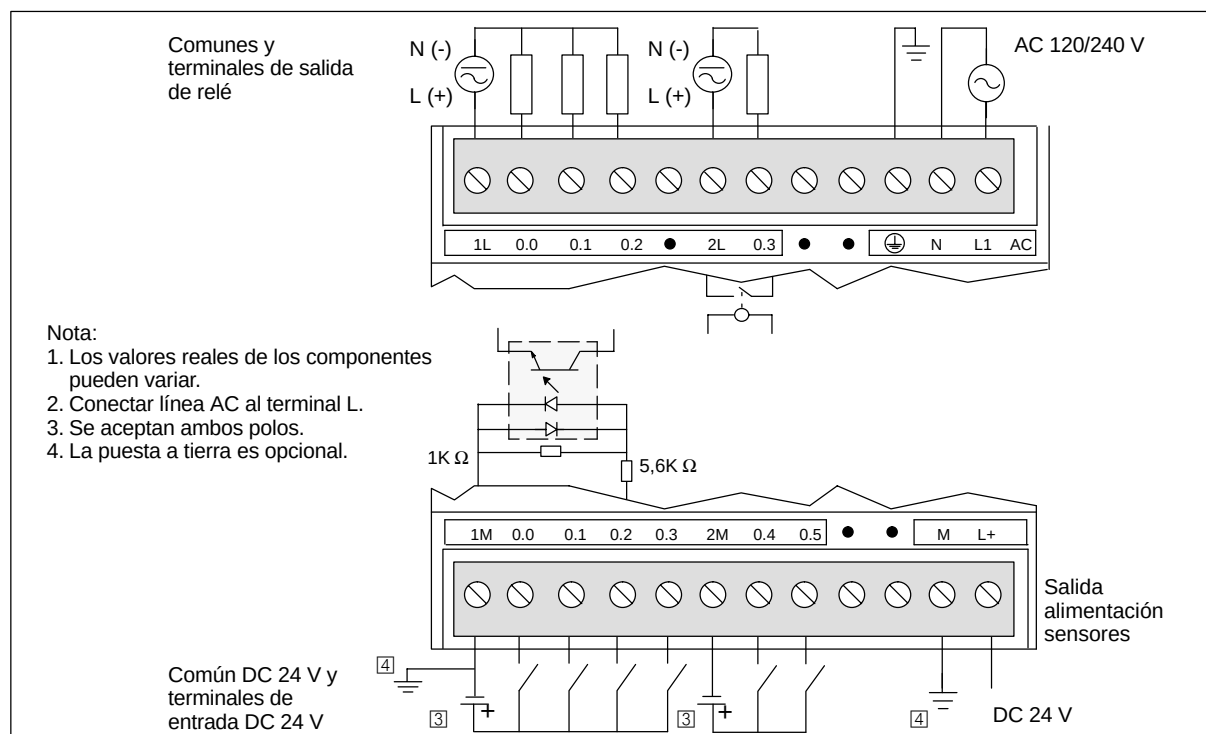


Figura A-3 Identificación de terminales de conexión para la CPU 221 AC/DC/Relé

A.3 Datos técnicos de la CPU 222 (firmware 1.1)

Tabla A-3 Datos técnicos de la CPU 222 DC/DC/DC y de la CPU 222 AC/DC/Relé

Descripción Nº de referencia	CPU 222 DC/DC/DC 6ES7 212-1AB21-0XB0	CPU 222 AC/DC/Relé 6ES7 212-1BB21-0XB0
Tamaño físico		
Dimensiones (l x a x p)	90 mm x 80 mm x 62 mm	90 mm x 80 mm x 62 mm
Peso	270 g	310 g
Pérdida de corriente (disipación)	5 W	7 W
Características de la CPU		
Entradas digitales integradas	8 entradas	8 entradas
Salidas digitales integradas	6 salidas	6 salidas
Contadores rápidos (valor de 32 bits)		
Total	4 contadores rápidos	4 contadores rápidos
Contadores de fase simple	4, con una frecuencia de reloj máx. de 30 kHz c/u	4, con una frecuencia de reloj máx. de 30 kHz c/u
Contadores de dos fases	2, con una frecuencia de reloj máx. de 20 kHz c/u	2, con una frecuencia de reloj máx. de 20 kHz c/u
Salidas de impulsos	2 a una frecuencia de impulsos de 20 kHz	2 a una frecuencia de impulsos de 20 kHz
Potenciómetros analógicos	1 con resolución de 8 bits	1 con resolución de 8 bits
Interrupciones temporizadas	2 con resolución de 1 ms	2 con resolución de 1 ms
Interrupciones de flanco	4 flancos positivos y/o 4 flancos negativos	4 flancos positivos y/o 4 flancos negativos
Tiempos de filtración de entradas	7 márgenes de 0,2 ms a 12,8 ms	7 márgenes de 0,2 ms a 12,8 ms
Captura de impulsos	8 entradas de captura de impulsos	8 entradas de captura de impulsos
Tamaño del programa (almacenado permanentemente)	2048 palabras	2048 palabras
Tamaño del bloque de datos	1024 palabras	1024 palabras
Almacenamiento permanente	1024 palabras	1024 palabras
Respaldo por condensador de alto rendimiento o pila	1024 palabras	1024 palabras
Nº de módulos de ampliación	2 módulos	2 módulos
E/S de ampliación digitales (máx.)	256 E/S	256 E/S
E/S analógicas (máx.)	16 entradas y 16 salidas	16 entradas y 16 salidas
Marcas internas	256 bits	256 bits
Almacenamiento permanente al apagar	112 bits	112 bits
Respaldo por condensador de alto rendimiento o pila	256 bits	256 bits
Temporizadores (total)	256 temporizadores	256 temporizadores
Respaldo por condensador de alto rendimiento o pila	64 temporizadores	64 temporizadores
1 ms	4 temporizadores	4 temporizadores
10 ms	16 temporizadores	16 temporizadores
100 ms	236 temporizadores	236 temporizadores
Contadores (total)	256 contadores	256 contadores
Respaldo por condensador de alto rendimiento o pila	256 contadores	256 contadores
Velocidad de ejecución booleana	0,37 µs por operación	0,37 µs por operación
Velocidad de ejecución de Transferir palabra	34 µs por operación	34 µs por operación
Velocidad de ejecución de temporizadores/contadores	50 µs a 64 µs por operación	50 µs a 64 µs por operación
Velocidad de ejecución de aritmética de precisión simple	46 µs por operación	46 µs por operación
Velocidad de ejecución de aritmética en coma flotante	100 µs a 400 µs por operación	100 µs a 400 µs por operación
Tiempo de respaldo por el condensador de alto rendimiento	típ. 50 h, mín. mín. 8 h a 40° C	típ. 50 h, mín. mín. 8 h a 40° C

Tabla A-3 Datos técnicos de la CPU 222 DC/DC/DC y de la CPU 222 AC/DC/Relé (continuación)

Descripción Nº de referencia	CPU 222 DC/DC/DC 6ES7 212-1AB21-0XB0	CPU 222 AC/DC/Relé 6ES7 212-1BB21-0XB0
Comunicación integrada		
Nº de puertos	1 puerto	1 puerto
Puerto eléctrico	RS-485	RS-485
Aislamiento (señal externa a circuito lógico)	Sin aislamiento	Sin aislamiento
Velocidades de transferencia PPI/MPI	9,6, 19,2 y 187,5 kbit/s	9,6, 19,2 y 187,5 kbit/s
Velocidades de transferencia Freeport	0,3, 0,6, 1,2, 2,4, 4,8, 9,6, 19,2 y 38,4 kbit/s	0,3, 0,6, 1,2, 2,4, 4,8, 9,6, 19,2 y 38,4 kbit/s
Longitud máx. del cable por segmento hasta 38,4 kbit/s 187,5 kbit/s	1200 m 1000 m	1200 m 1000 m
Nº máximo de estaciones		
Por segmento	32 estaciones	32 estaciones
Por red	126 estaciones	126 estaciones
Nº máximo de maestros	32 maestros	32 maestros
Modo maestro PPI (NETR/NETW)	Sí	Sí
Enlaces MPI	4 en total; 2 reservados: 1 para PG y 1 OP	4 en total; 2 reservados: 1 para PG y 1 OP
Cartuchos opcionales		
Cartucho de memoria (almacenamiento permanente)	Programa, datos y configuración	Programa, datos y configuración
Cartucho de pila (tiempo de respaldo de datos)	típ. 200 días	típ. 200 días
Cartucho de reloj (precisión del reloj)	2 minutos por mes a 25° C 7 minutos por mes 0° C a 55° C	2 minutos por mes a 25° C 7 minutos por mes 0° C a 55° C
Alimentación		
Tensión de línea (margen admisible)	DC 20,4 a 28,8 V	AC 85 V a 264 V, 47 a 63 Hz
Corriente de entrada (sólo CPU) / carga máx.	85/500 mA a DC 24 V	20/70 mA a AC 240 V 40/140 mA a AC 120 V
Extra-corriente de serie (máx.)	10 A a DC 28,8 V	20 A a AC 264 V
Aislamiento (corriente de entrada a lógica)	Sin aislamiento	AC 1500 V
Tiempo de retardo (desde la pérdida de corriente de entrada)	mín. 10 ms de DC 24 V	80 ms de AC 240 V, 20 ms de AC 120 V
Fusible interno (no reemplazable por el usuario)	2 A, 250 V, de acción lenta	2 A, 250 V, de acción lenta
+5 alimentación para módulos de ampliación (máx.)	340 mA	340 mA
Alimentación para sensores DC 24 V		
Margen de tensión	DC 15,4 a 28,8 V	DC 20,4 a 28,8 V
Corriente máxima	180 mA	180 mA
Rizado corriente parásita	Igual que línea de entrada	Menos de 1 V pico a pico (máx.)
Corriente límite	600 mA	600 mA
Aislamiento (alimentación de sensor a circuito lógico)	Sin aislamiento	Sin aislamiento

Tabla A-3 Datos técnicos de la CPU 222 DC/DC/DC y de la CPU 222 AC/DC/Relé (continuación)

Descripción Nº de referencia	CPU 222 DC/DC/DC 6ES7 212-1AB21-0XB0	CPU 222 AC/DC/Relé 6ES7 212-1BB21-0XB0
Características de las entradas		
Nº de entradas integradas	8 entradas	8 entradas
Tipo de entrada	Sumidero de corriente/fuente (tipo 1 IEC con sumidero de corriente)	Sumidero de corriente/fuente (tipo 1 IEC con sumidero de corriente)
Tensión de entrada		
Tensión máx. continua admisible	DC 30 V	DC 30 V
Sobretensión transitoria	DC 35 V, 0,5 s	DC 35 V, 0,5 s
Valor nominal	DC 24 V a 4 mA, nominal	DC 24 V a 4 mA, nominal
Señal 1 lógica (mín.)	mín. DC 15 V a 2,5 mA	mín. DC 15 V a 2,5 mA
Señal 0 lógica (máx.)	máx. DC 5 V a 1 mA	máx. DC 5 V a 1 mA
Aislamiento (campo a circuito lógico)		
Separación galvánica	AC 500 V, 1 minuto	AC 500 V, 1 minuto
Grupos de aislamiento de:	4 E/S	4 E/S
Tiempos de retardo de las entradas		
Entradas filtradas y entradas de interrupción	0,2 a 12,8 ms, seleccionable por el usuario	0,2 a 12,8 ms, seleccionable por el usuario
Entradas de reloj de los contadores rápidos		
Fase simple		
Nivel 1 lógico = DC 15 V a DC 30 V	máx. 20 kHz	máx. 20 kHz
Nivel 1 lógico = DC 15 V a DC 26 V	máx. 30 kHz	máx. 30 kHz
Contadores A/B		
Nivel 1 lógico = DC 15 V a DC 30 V	máx. 10 kHz	máx. 10 kHz
Nivel 1 lógico = DC 15 V a DC 26 V	máx. 20 kHz	máx. 20 kHz
Conexión de sensor de proximidad de 2 hilos (Bero)		
Corriente de fuga admisible	máx. 1 mA	máx. 1 mA
Longitud del cable		
No apantallado (no HSC)	300 m	300 m
Apantallado	500 m	500 m
Entradas HSC, apantalladas	50 m	50 m
Nº de entradas ON simultáneamente		
40° C	8	8
55° C	8	8
Características de las salidas		
Nº de salidas integradas	6 salidas	6 salidas
Tipo de salida	Estado sólido-MOSFET	Relé, contacto de baja potencia
Tensión de salida		
Margen admisible	DC 20,4 a 28,8 V	DC 5 a 30 V ó AC 5 a 250 V
Valor nominal	DC 24 V	—
Señal 1 lógica a corriente máxima	mín. DC 20 V	—
Señal 0 lógica a 10 K Ω de carga	máx. DC 0,1 V	—
Corriente de salida		
Señal 1 lógica	0,75 A	2,00 A
Nº de grupos de salidas	1	2
Nº de salidas ON (máx.)	6	6
Por grupo – montaje horizontal (máx.)	6	3
Por grupo – montaje vertical (máx.)	6	3
Corriente máx. por común/grupo	4,5 A	6 A
Carga LEDs	5 W	30 W DC/ 200 W AC
Resistencia estado ON (resistencia contactos)	0,3 Ω	0,2 Ω , máx. si son nuevas
Corriente de derivación por salida	máx. 10 μ A	—
Sobrecorriente momentánea	máx. 8 A, 100 ms	7A al estar cerrados los contactos
Protección contra sobrecargas	no	no

Tabla A-3 Datos técnicos de la CPU 222 DC/DC/DC y de la CPU 222 AC/DC/Relé (continuación)

Descripción Nº de referencia	CPU 222 DC/DC/DC 6ES7 212-1AB21-0XB0	CPU 222 AC/DC/Relé 6ES7 212-1BB21-0XB0
Aislamiento		
Separación galvánica	AC 500 V, 1 minuto	—
Resistencia de aislamiento	—	100 M Ω , mín. si son nuevas
Aislamiento bobina a contacto	—	AC 500 V, 1 minuto
Aislamiento entre contactos abiertos	—	AC 750 V, 1 minuto
En grupos de:	6 E/S	3 E/S
Carga inductiva, apriete		
Repetición disipación de energía $< 0.5 LI^2 \times$ frecuencia de conmutación	1 W, en todos los canales	—
Límites tensión de bloqueo	L+ menos 48V	—
Retardo de las salidas		
OFF a ON (Q0.0 y Q0.1)	máx. 2 μ s	—
ON a OFF (Q0.0 y Q0.1)	máx. 10 μ s	—
OFF a ON (Q0.2 hasta Q0.5)	máx. 15 μ s	—
ON a OFF (Q0.2 hasta Q0.5)	máx. 100 μ s	—
Frecuencia de conmutación (salida de impulsos)		
Q0.0 y I0.1	máx. 20 kHz	máx. 1 Hz
Relé		
Retardo de conmutación	—	máx. 10 ms
Vida útil mecánica (sin carga)	—	10.000.000 ciclos abiertos/cerrados
Vida útil contactos a carga nominal	—	100.000 ciclos abiertos/cerrados
Longitud del cable		
Apantallado	150 m	150 m
No apantallado	500 m	500 m

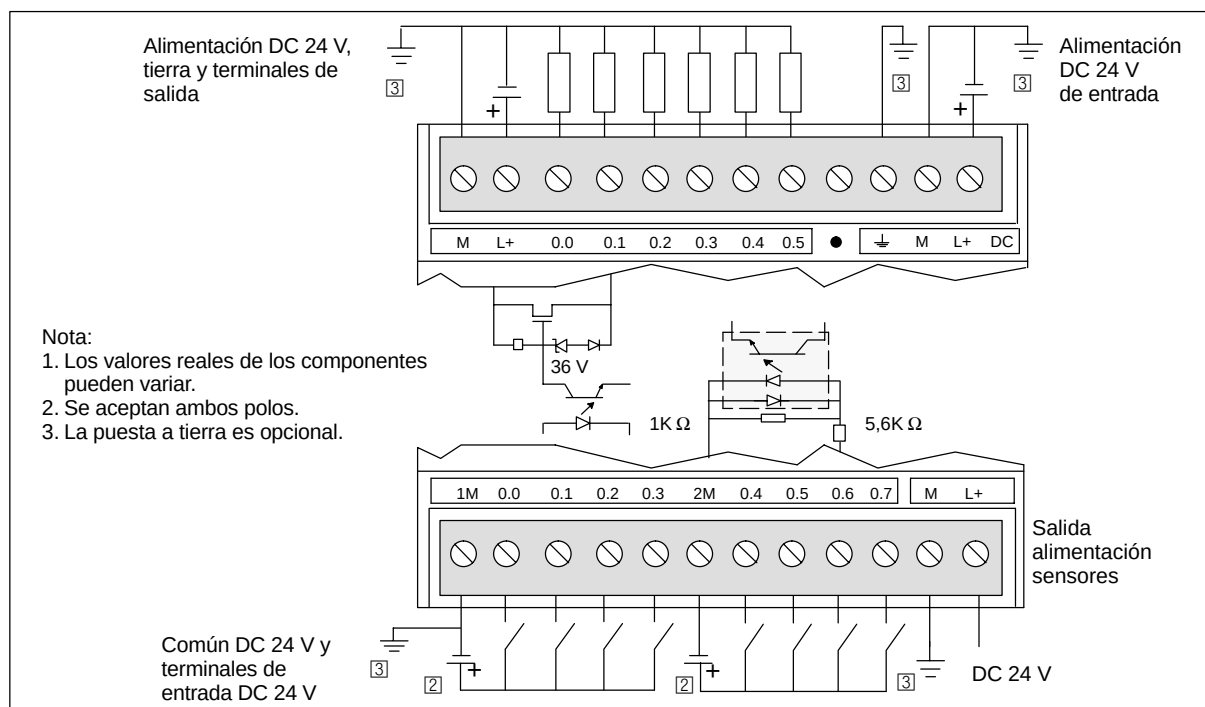


Figura A-4 Identificación de terminales de conexión para la CPU 222 DC/DC/DC

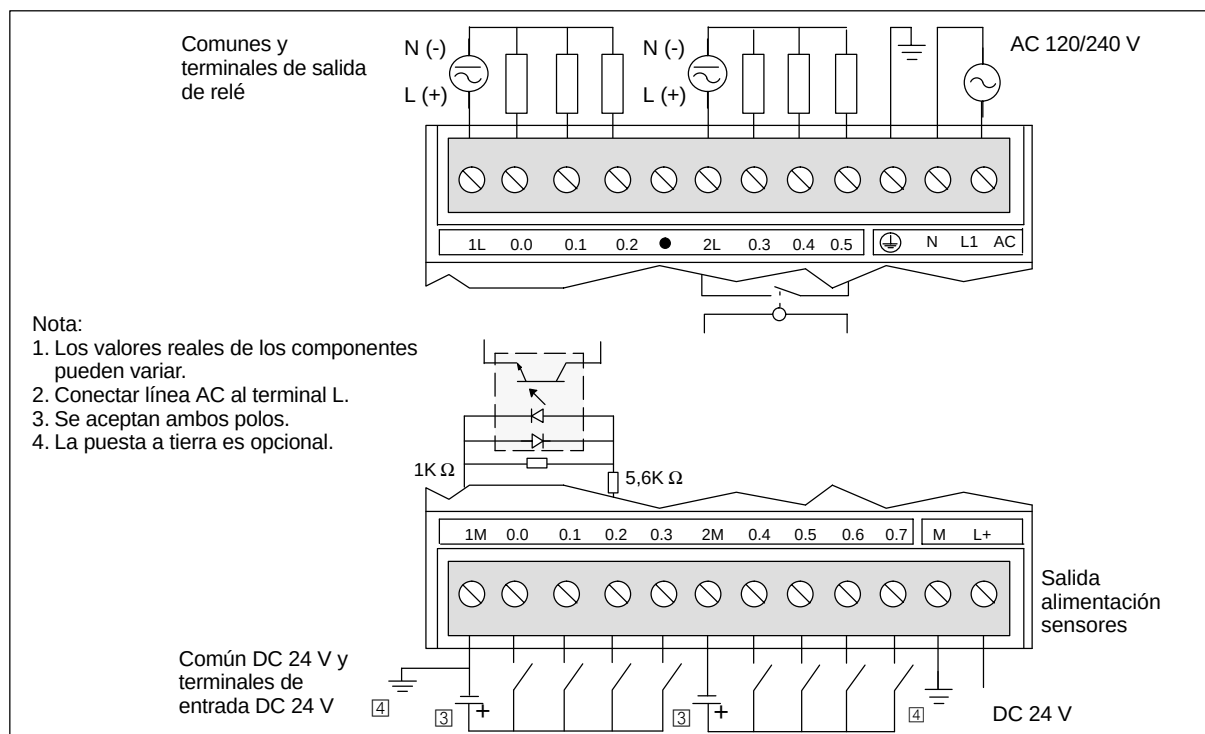


Figura A-5 Identificación de terminales de conexión para la CPU 222 AC/DC/Relé

A.4 Datos técnicos de la CPU 224 (firmware 1.1)

Tabla A-4 Datos técnicos de la CPU 224 DC/DC/DC y de la CPU 224 AC/DC/Relé

Descripción Nº de referencia	CPU 224 DC/DC/DC 6ES7 214-1AD21-0XB0	CPU 224 AC/DC/Relé 6ES7 214-1BD21-0XB0
Tamaño físico		
Dimensiones (l x a x p)	120.5 mm x 80 mm x 62 mm	120.5 mm x 80 mm x 62 mm
Peso	360 g	410 g
Pérdida de corriente (disipación)	7 W	10 W
Características de la CPU		
Entradas digitales integradas	14 entradas	14 entradas
Salidas digitales integradas	10 salidas	10 salidas
Contadores rápidos (valor de 32 bits)		
Total	6 contadores rápidos	6 contadores rápidos
Contadores de fase simple	6, con una frecuencia de reloj máx. de 30 kHz c/u	6, con una frecuencia de reloj máx. de 30 kHz c/u
Contadores de dos fases	4, con una frecuencia de reloj máx. de 20 kHz c/u	4, con una frecuencia de reloj máx. de 20 kHz c/u
Salidas de impulsos	2 a una frecuencia de impulsos de 20 kHz	2 a una frecuencia de impulsos de 20 kHz
Potenciómetros analógicos	2 con resolución de 8 bits	2 con resolución de 8 bits
Interrupciones temporizadas	2 con resolución de 1 ms	2 con resolución de 1 ms
Interrupciones de flanco	4 flancos positivos y/o 4 flancos negativos	4 flancos positivos y/o 4 flancos negativos
Tiempos de filtración de entradas	7 márgenes de 0,2 ms a 12,8 ms	7 márgenes de 0,2 ms a 12,8 ms
Capturar impulsos	14 entradas de captura de impulsos	14 entradas de captura de impulsos
Reloj de tiempo real (precisión del reloj)	2 minutos por mes a 25° C 7 minutos por mes 0° C a 55° C	2 minutos por mes a 25° C 7 minutos por mes 0° C a 55° C
Tamaño del programa (almacenado permanentemente)	4096 palabras	4096 palabras
Tamaño del bloque de datos (almacenado permanentemente):	2560 palabras	2560 palabras
Almacenamiento permanente	2560 palabras	2560 palabras
Respaldo por condensador de alto rendimiento o pila	2560 palabras	2560 palabras
Nº de módulos de ampliación	7 módulos	7 módulos
E/S de ampliación digitales (máx.)	256 E/S	256 E/S
E/S analógicas (máx.)	32 entradas y 32 salidas	32 entradas y 32 salidas
Marcas internas	256 bits	256 bits
Almacenamiento permanente al apagar	112 bits	112 bits
Respaldo por condensador de alto rendimiento o pila	256 bits	256 bits
Temporizadores (total)	256 temporizadores	256 temporizadores
Respaldo por condensador de alto rendimiento o pila	64 temporizadores	64 temporizadores
1 ms	4 temporizadores	4 temporizadores
10 ms	16 temporizadores	16 temporizadores
100 ms	236 temporizadores	236 temporizadores
Contadores (total)	256 contadores	256 contadores
Respaldo por condensador de alto rendimiento o pila	256 contadores	256 contadores
Velocidad de ejecución booleana	0,37 µs por operación	0,37 µs por operación
Velocidad de ejecución de Transferir palabra	34 µs por operación	34 µs por operación
Velocidad de ejecución de temporizadores/contadores	50 µs a 64 µs por operación	50 µs a 64 µs por operación
Velocidad de ejecución de aritmética de precisión simple	46 µs por operación	46 µs por operación
Velocidad de ejecución de aritmética en coma flotante	100 µs a 400 µs por operación	100 µs a 400 µs por operación
Tiempo de respaldo por el condensador de alto rendimiento	típ. 190 h, mín. 120 h a 40° C	típ. 190 h, mín. 120 h a 40° C

Tabla A-4 Datos técnicos de la CPU 224 DC/DC/DC y de la CPU 224 AC/DC/Relé (continuación)

Descripción Nº de referencia	CPU 224 DC/DC/DC 6ES7 214-1AD21-0XB0	CPU 224 AC/DC/Relé 6ES7 214-1BD21-0XB0
Comunicación integrada		
Nº de puertos	1 puerto	1 puerto
Puerto eléctrico	RS-485	RS-485
Aislamiento (señal externa a circuito lógico)	Sin aislamiento	Sin aislamiento
Velocidades de transferencia PPI/MPI	9,6, 19,2 y 187,5 kbit/s	9,6, 19,2 y 187,5 kbit/s
Velocidades de transferencia Freeport	0,3, 0,6, 1,2, 2,4, 4,8, 9,6, 19,2 y 38,4 kbit/s	0,3, 0,6, 1,2, 2,4, 4,8, 9,6, 19,2 y 38,4 kbit/s
Longitud máx. del cable por segmento hasta 38,4 kbit/s	1200 m	1200 m
187,5 kbit/s	1000 m	1000 m
Nº máximo de estaciones		
Por segmento	32 estaciones	32 estaciones
Por red	126 estaciones	126 estaciones
Nº máximo de maestros	32 maestros	32 maestros
Modo maestro PPI (NETR/NETW)	Sí	Sí
Enlaces MPI	4 en total; 2 reservados: 1 para PG y 1 OP	4 en total; 2 reservados: 1 para PG y 1 OP
Cartuchos opcionales		
Cartucho de memoria (almacenamiento permanente)	Programa, datos y configuración	Programa, datos y configuración
Cartucho de pila (tiempo de respaldo de datos)	típ. 200 días	típ. 200 días
Alimentación		
Tensión de línea (margen admisible)	DC 20,4 a 28,8 V	AC 85 a 264 V 47 a 63 Hz
Corriente de entrada (sólo CPU) / carga máx.	110/700 mA a DC 24 V	30/100 mA a AC 240 V 60/200 mA a AC 120 V
Extra-corriente de serie (máx.)	10 A a DC 28,8 V	20 A a AC 264 V
Aislamiento (corriente de entrada a lógica)	Sin aislamiento	AC 1500 V
Tiempo de retardo (desde la pérdida de corriente de entrada)	mín. 10 ms de DC 24 V	80 ms de AC 240 V, 20 ms de AC 120 V
Fusible interno (no reemplazable por el usuario)	2 A, 250 V, de acción lenta	2 A, 250 V, de acción lenta
+5 alimentación para módulos de ampliación (máx.)	660 mA	660 mA
Alimentación para sensores DC 24 V		
Margen de tensión	DC 15,4 a 28,8 V	DC 20,4 a 28,8 V
Corriente máxima	280 mA	280 mA
Rizado corriente parásita	Igual que línea de entrada	Menos de 1 V pico a pico (máx.)
Corriente límite	600 mA	600 mA
Aislamiento (alimentación de sensor a circuito lógico)	Sin aislamiento	Sin aislamiento

Tabla A-4 Datos técnicos de la CPU 224 DC/DC/DC y de la CPU 224 AC/DC/Relé (continuación)

Descripción Nº de referencia	CPU 224 DC/DC/DC 6ES7 214-1AD21-0XB0	CPU 224 AC/DC/Relé 6ES7 214-1BD21-0XB0
Características de las entradas		
Nº de entradas integradas	14 entradas	14 entradas
Tipo de entrada	Sumidero de corriente/fuente (tipo 1 IEC)	Sumidero de corriente/fuente (tipo 1 IEC)
Tensión de entrada		
Tensión máx. continua admisible	DC 30 V	DC 30 V
Sobretensión transitoria	DC 35 V, 0,5 s	DC 35 V, 0,5 s
Valor nominal	DC 24 V a 4 mA, nominal	DC 24 V a 4 mA, nominal
Señal 1 lógica (mín.)	mín. DC 15 V a 2,5 mA	mín. DC 15 V a 2,5 mA
Señal 0 lógica (máx.)	máx. DC 5 V a 1 mA	máx. DC 5 V a 1 mA
Aislamiento (campo a circuito lógico)		
Separación galvánica	AC 500 V, 1 minuto	AC 500 V, 1 minuto
Grupos de aislamiento de:	8 y 6 E/S	8 y 6 E/S
Tiempos de retardo de las entradas		
Entradas filtradas y entradas de interrupción	0,2 a 12,8 ms, seleccionable por el usuario	0,2 a 12,8 ms, seleccionable por el usuario
Entradas de reloj de los contadores rápidos		
Fase simple		
Nivel 1 lógico = DC 15 V a DC 30 V	20 kHz	20 kHz
Nivel 1 lógico = DC 15 V a DC 26 V	30 kHz	30 kHz
Contadores A/B		
Nivel 1 lógico = DC 15 V a DC 30 V	10 kHz	10 kHz
Nivel 1 lógico = DC 15 V a DC 26 V	20 kHz	20 kHz
Conexión de sensor de proximidad de 2 hilos (Bero)		
Corriente de fuga admisible	máx. 1 mA	máx. 1 mA
Longitud del cable		
No apantallado (no HSC)	300 m	300 m
Apantallado	500 m	50 m
Entradas HSC, apantalladas	50 m	50 m
Nº de entradas ON simultáneamente		
40° C	14	14
55° C	14	14
Características de las salidas		
Nº de salidas integradas	10 salidas	10 salidas
Tipo de salida	Estado sólido–MOSFET	Relé, contacto de baja potencia
Tensión de salida		
Margen admisible	DC 20,4 a 28,8 V	DC 5 a 30 V ó AC 5 a 250 V
Valor nominal	DC 24 V	–
Señal 1 lógica a corriente máxima	mín. DC 20 V	–
Señal 0 lógica a 10 K Ω de carga	máx. DC 0,1 V	–
Corriente de salida		
Señal 1 lógica	0,75 A	2.00 A
Nº de grupos de salidas	2	3
Nº de salidas ON (máx.)	10	10
Por grupo – montaje horizontal (máx.)	5	4/3/3
Por grupo – montaje vertical (máx.)	5	4/3/3
Corriente máx. por común/grupo	3,75 A	8 A
Carga LEDs	5 W	30 W DC/200 W AC
Resistencia estado ON (resistencia contactos)	0,3 Ω	0,2 Ω , máx. si son nuevas
Corriente de derivación por salida	máx. 10 μ A	–
Sobrecorriente momentánea	máx. 8 A, 100 ms	7A al estar cerrados los contactos
Protección contra sobrecargas	no	no

Tabla A-4 Datos técnicos de la CPU 224 DC/DC/DC y de la CPU 224 AC/DC/Relé (continuación)

Descripción Nº de referencia	CPU 224 DC/DC/DC 6ES7 214-1AD21-0XB0	CPU 224 AC/DC/Relé 6ES7 214-1BD21-0XB0
Aislamiento (campo a circuito lógico)		
Separación galvánica	AC 500 V, 1 minuto	–
Resistencia de aislamiento	–	100 M Ω , mín. si son nuevas
Aislamiento bobina a contacto	–	AC 500 V, 1 minuto
Aislamiento entre contactos abiertos	–	AC 750 V, 1 minuto
En grupos de:	5 E/S	4 E/S / 3 E/S / 3 E/S
Carga inductiva, apriete		
Repetición disipación de energía < 0.5 LI ² x frecuencia de conmutación	1 W, en todos los canales	–
Límites tensión de bloqueo	L+ menos 48V	–
Retardo de las salidas		
OFF a ON (Q0.0 y Q0.1)	máx. 2 μ s	–
ON a OFF (Q0.0 y Q0.1)	máx. 10 μ s	–
OFF a ON (Q0.2 hasta Q1.1)	máx. 15 μ s	–
ON a OFF (Q0.2 hasta Q1.1)	máx. 100 μ s	–
Frecuencia de conmutación (salida de impulsos)		
Q0.0 y I0.1	máx. 20 kHz	máx. 1 Hz
Relé		
Retardo de conmutación	–	máx. 10 ms
Vida útil mecánica (sin carga)	–	10.000.000 ciclos abiertos/cerrados
Vida útil contactos a carga nominal	–	100.000 ciclos abiertos/cerrados
Longitud del cable		
No apantallado	150 m	150 m
Apantallado	500 m	500 m

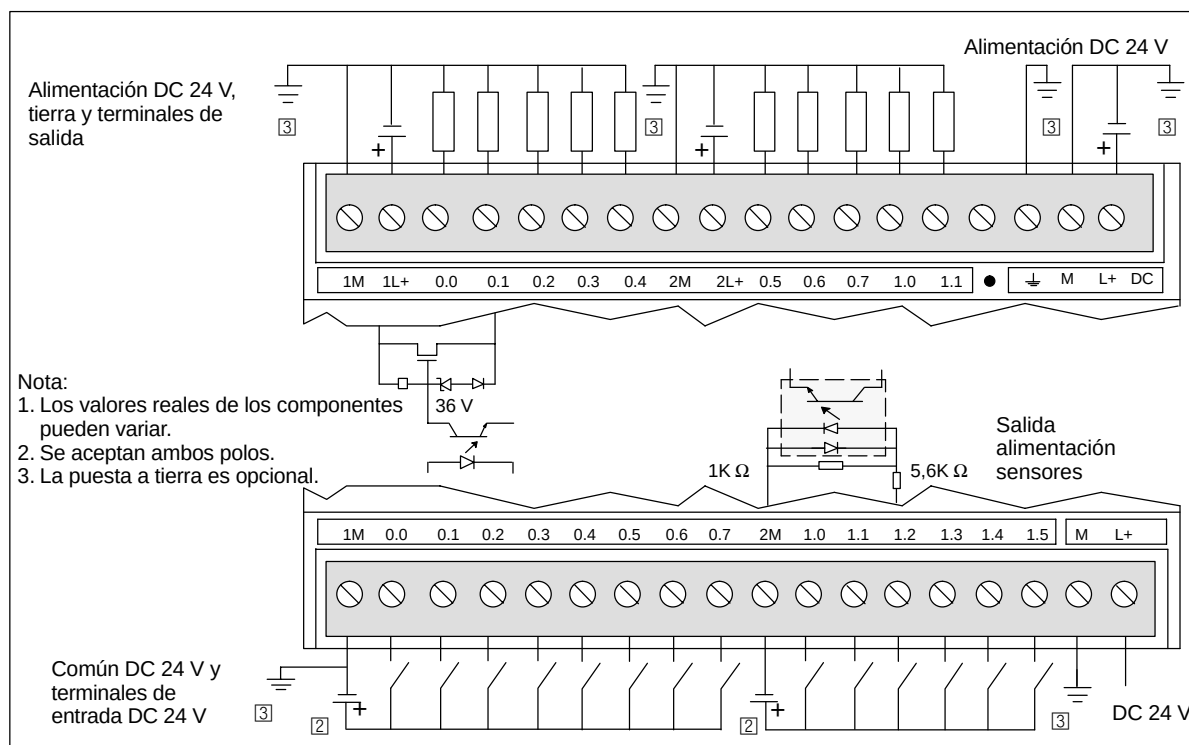


Figura A-6 Identificación de terminales de conexión para la CPU 224 DC/DC/DC

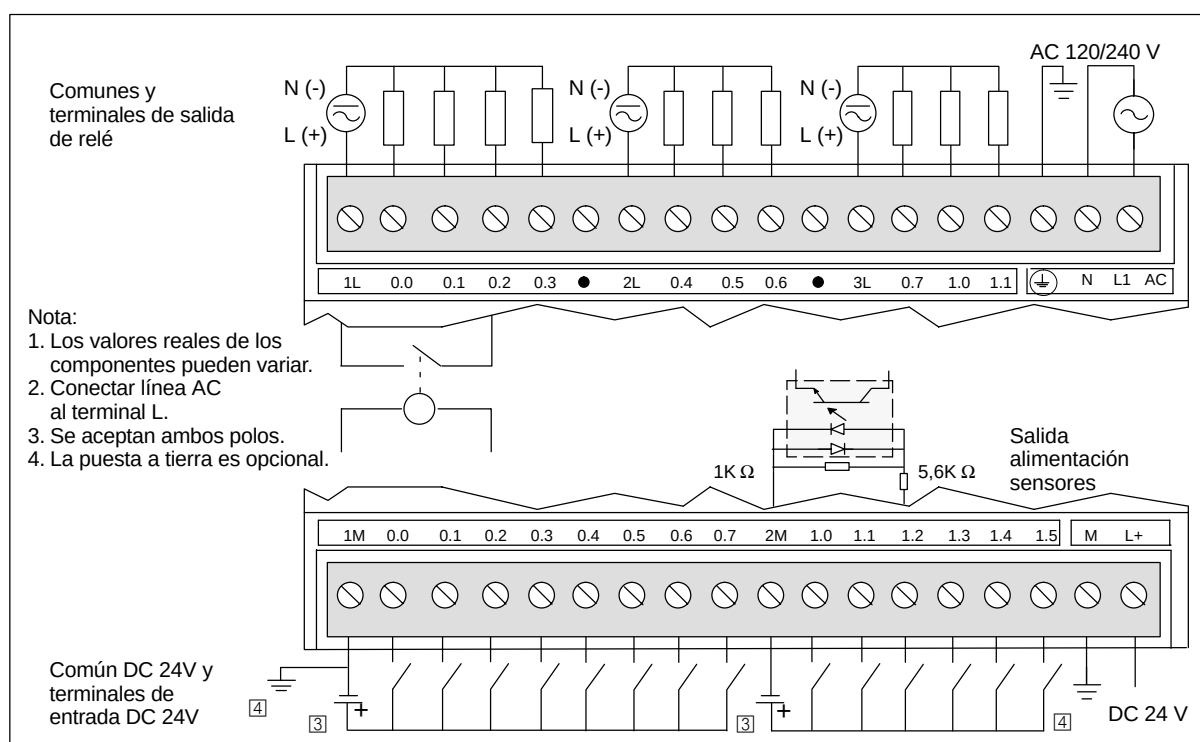


Figura A-7 Identificación de terminales de conexión para la CPU 224 AC/DC/Relé

A.5 Datos técnicos de la CPU 226

Tabla A-5 Datos técnicos de la CPU 226 DC/DC/DC y de la CPU 226 AC/DC/Relé

Descripción Nº de referencia	CPU 226 DC/DC/DC 6ES7 216-2AD21-0XB0	CPU 226 AC/DC/Relé 6ES7 216-2BD21-0XB0
Tamaño físico		
Dimensiones (l x a x p)	196 mm x 80 mm x 62 mm	196 mm x 80 mm x 62 mm
Peso	550 g	660 g
Pérdida de corriente (disipación)	11 W	17 W
Características de la CPU		
Entradas digitales integradas	24 entradas	24 entradas
Salidas digitales integradas	16 salidas	16 salidas
Contadores rápidos (valor de 32 bits)		
Total	6 contadores rápidos	6 contadores rápidos
Contadores de fase simple	6, con una frecuencia de reloj máx. de 30 kHz c/u	6, con una frecuencia de reloj máx. de 30 kHz c/u
Contadores de dos fases	4, con una frecuencia de reloj máx. de 20 kHz c/u	4, con una frecuencia de reloj máx. de 20 kHz c/u
Salidas de impulsos	2 a una frecuencia de impulsos de 20 kHz	2 a una frecuencia de impulsos de 20 kHz
Potenciómetros analógicos	2 con resolución de 8 bits	2 con resolución de 8 bits
Interrupciones temporizadas	2 con resolución de 1 ms	2 con resolución de 1 ms
Interrupciones de flanco	4 flancos positivos y/o 4 flancos negativos	4 flancos positivos y/o 4 flancos negativos
Tiempos de filtración de entradas	7 márgenes de 0,2 ms a 12,8 ms	7 márgenes de 0,2 ms a 12,8 ms
Capturar impulsos	14 entradas de captura de impulsos	14 entradas de captura de impulsos
Reloj de tiempo real (precisión del reloj)	2 minutos por mes a 25° C 7 minutos por mes 0° C a 55° C	2 minutos por mes a 25° C 7 minutos por mes 0° C a 55° C
Tamaño del programa (almacenado permanentemente)	4096 palabras	4096 palabras
Tamaño del bloque de datos (almacenado permanentemente):	2560 palabras	2560 palabras
Almacenamiento permanente	2560 palabras	2560 palabras
Respaldo por condensador de alto rendimiento o pila	2560 palabras	2560 palabras
Nº de módulos de ampliación	7 módulos	7 módulos
E/S de ampliación digitales (máx.)	256 E/S	256 E/S
E/S analógicas (máx.)	32 entradas y 32 salidas	32 entradas y 32 salidas
Marcas internas	256 bits	256 bits
Almacenamiento permanente al apagar	112 bits	112 bits
Respaldo por condensador de alto rendimiento o pila	256 bits	256 bits
Temporizadores (total)	256 temporizadores	256 temporizadores
Respaldo por condensador de alto rendimiento o pila	64 temporizadores	64 temporizadores
1 ms	4 temporizadores	4 temporizadores
10 ms	16 temporizadores	16 temporizadores
100 ms	236 temporizadores	236 temporizadores
Contadores (total)	256 contadores	256 contadores
Respaldo por condensador de alto rendimiento o pila	256 contadores	256 contadores
Velocidad de ejecución booleana	0,37 µs por operación	0,37 µs por operación
Velocidad de ejecución de Transferir palabra	34 µs por operación	34 µs por operación
Velocidad de ejecución de temporizadores/contadores	50 µs a 64 µs por operación	50 µs a 64 µs por operación
Velocidad de ejecución de aritmética de precisión simple	46 µs por operación	46 µs por operación
Velocidad de ejecución de aritmética en coma flotante	100 µs a 400 µs por operación	100 µs a 400 µs por operación
Tiempo de respaldo por el condensador de alto rendimiento	típ. 190 h, mín. 120 h a 40° C	típ. 190 h, mín. 120 h a 40° C

Tabla A-5 Datos técnicos de la CPU 226 DC/DC/DC y de la CPU 226 AC/DC/Relé (continuación)

Descripción Nº de referencia	CPU 226 DC/DC/DC 6ES7 216-2AD21-0XB0	CPU 226 AC/DC/Relé 6ES7 216-2BD21-0XB0
Comunicación integrada		
Nº de puertos	2 puertos	2 puertos
Puerto eléctrico	RS-485	RS-485
Aislamiento (señal externa a circuito lógico)	Sin aislamiento	Sin aislamiento
Velocidades de transferencia PPI/MPI	9,6, 19,2 y 187,5 kbit/s	9,6, 19,2 y 187,5 kbit/s
Velocidades de transferencia Freeport	0,3, 0,6, 1,2, 2,4, 4,8, 9,6, 19,2 y 38,4 kbit/s	0,3, 0,6, 1,2, 2,4, 4,8, 9,6, 19,2 y 38,4 kbit/s
Longitud máx. del cable por segmento hasta 38,4 kbit/s	1200 m	1200 m
187,5 kbit/s	1000 m	1000 m
Nº máximo de estaciones		
Por segmento	32 estaciones	32 estaciones
Por red	126 estaciones	126 estaciones
Nº máximo de maestros	32 maestros	32 maestros
Modo maestro PPI (NETR/NETW)	Sí	Sí
Enlaces MPI	4 en total; 2 reservados: 1 para PG y 1 OP	4 en total; 2 reservados: 1 para PG y 1 OP
Cartuchos opcionales		
Cartucho de memoria (almacenamiento permanente)	Programa, datos y configuración	Programa, datos y configuración
Cartucho de pila (tiempo de respaldo de datos)	típ. 200 días	típ. 200 días
Alimentación		
Tensión de línea (margen admisible)	DC 20,4 a 28,8 V	AC 85 a 264 V 47 a 63 Hz
Corriente de entrada (sólo CPU) / carga máx.	150/1050 mA	40/160 mA a AC 240 V 80/320 mA a AC 120 V
Extra-corriente de serie (máx.)	10 A a DC 28,8 V	20 A a AC 264 V
Aislamiento (corriente de entrada a lógica)	Sin aislamiento	AC 1500 V
Tiempo de retardo (desde la pérdida de corriente de entrada)	mín. 10 ms de DC 24 V	80 ms de AC 240 V, 20 ms de AC 120 V
Fusible interno (no reemplazable por el usuario)	3 A, 250 V, de acción lenta	2 A, 250 V, de acción lenta
+5 alimentación para módulos de ampliación (máx.)	1000 mA	1000 mA
Alimentación para sensores DC 24 V		
Margen de tensión	DC 15,4 a 28,8 V	DC 20,4 a 28,8 V
Corriente máxima	400 mA	400 mA
Rizado corriente parásita	Igual que línea de entrada	Menos de 1 V pico a pico (máx.)
Corriente límite	Aprox. 1,5 A	Aprox. 1,5 A
Aislamiento (alimentación de sensor a circuito lógico)	Sin aislamiento	Sin aislamiento

Tabla A-5 Datos técnicos de la CPU 226 DC/DC/DC y de la CPU 226 AC/DC/Relé (continuación)

Descripción Nº de referencia	CPU 226 DC/DC/DC 6ES7 216-2AD21-0XB0	CPU 226 AC/DC/Relé 6ES7 216-2BD21-0XB0
Características de las entradas		
Nº de entradas integradas	24 entradas	24 entradas
Tipo de entrada	Sumidero de corriente/fuente (tipo 1 IEC)	Sumidero de corriente/fuente (tipo 1 IEC)
Tensión de entrada		
Tensión máx. continua admisible	DC 30 V	DC 30 V
Sobretensión transitoria	DC 35 V, 0,5 s	DC 35 V, 0,5 s
Valor nominal	DC 24 V a 4 mA, nominal	DC 24 V a 4 mA, nominal
Señal 1 lógica (mín.)	mín. DC 15 V a 2,5 mA	mín. DC 15 V a 2,5 mA
Señal 0 lógica (máx.)	máx. DC 5 V a 1 mA	máx. DC 5 V a 1 mA
Aislamiento (campo a circuito lógico)		
Separación galvánica	AC 500 V, 1 minuto	AC 500 V, 1 minuto
Grupos de aislamiento de:	13 E/S y 11 E/S	13 E/S y 11 E/S
Tiempos de retardo de las entradas		
Entradas filtradas y entradas de interrupción	0,2 a 12,8 ms, seleccionable por el usuario	0,2 a 12,8 ms, seleccionable por el usuario
Entradas de reloj de los contadores rápidos		
Fase simple		
Nivel 1 lógico = DC 15 V a DC 30 V	20 kHz	20 kHz
Nivel 1 lógico = DC 15 V a DC 26 V	30 kHz	30 kHz
Contadores A/B		
Nivel 1 lógico = DC 15 V a DC 30 V	10 kHz	10 kHz
Nivel 1 lógico = DC 15 V a DC 26 V	20 kHz	20 kHz
Conexión de sensor de proximidad de 2 hilos (Bero)		
Corriente de fuga admisible	máx. 1 mA	máx. 1 mA
Longitud del cable		
No apantallado (no HSC)	300 m	300 m
Apantallado	500 m	50 m
Entradas HSC, apantalladas	50 m	50 m
Nº de entradas ON simultáneamente		
40° C	24	24
55° C	24	24

Tabla A-5 Datos técnicos de la CPU 226 DC/DC/DC y de la CPU 226 AC/DC/Relé (continuación)

Descripción Nº de referencia	CPU 226 DC/DC/DC 6ES7 216-2AD21-0XB0	CPU 226 AC/DC/Relé 6ES7 216-2BD21-0XB0
Características de las salidas		
Nº de salidas integradas	16 salidas	16 salidas
Tipo de salida	Estado sólido–MOSFET	Relé, contacto de baja potencia
Tensión de salida		
Margen admisible	DC 20,4 a 28,8 V	DC 5 a 30 V ó AC 5 a 250 V
Valor nominal	DC 24 V	–
Señal 1 lógica a corriente máxima	mín. DC 20 V	–
Señal 0 lógica a 10 K Ω de carga	máx. DC 0,1 V	–
Corriente de salida		
Señal 1 lógica	0,75 A	2.00 A
Nº de grupos de salidas	2	3
Nº de salidas ON (máx.)	16	16
Por grupo – montaje horizontal (máx.)	8	4/5/7
Por grupo – montaje vertical (máx.)	8	4/5/7
Corriente máx. por común/grupo	6 A	10 A
Carga LEDs	5 W	30 W DC/200 W AC
Resistencia estado ON (resistencia contactos)	0,3 Ω	0,2 Ω , máx. si son nuevas
Corriente de derivación por salida	máx. 10 μ A	–
Sobrecorriente momentánea	máx. 8 A, 100 ms	7A al estar cerrados los contactos
Protección contra sobrecargas	no	no
Aislamiento (campo a circuito lógico)		
Separación galvánica	AC 500 V, 1 minuto	–
Resistencia de aislamiento	–	100 M Ω , mín. si son nuevas
Aislamiento bobina a contacto	–	AC 500 V, 1 minuto
Aislamiento entre contactos abiertos	–	AC 750 V, 1 minuto
En grupos de:	8 E/S	4 E/S / 5 E/S / 7 E/S
Carga inductiva, apriete		
Repetición disipación de energía < 0.5 L I ² x frecuencia de conmutación	1 W, en todos los canales	–
Límites tensión de bloqueo	L+ menos 48V	–
Retardo de las salidas		
OFF a ON (Q0.0 y Q0.1)	máx. 2 μ s	–
ON a OFF (Q0.0 y Q0.1)	máx. 10 μ s	–
OFF a ON (Q0.2 hasta Q1.7)	máx. 15 μ s	–
ON a OFF (Q0.2 hasta Q1.7)	máx. 100 μ s	–
Frecuencia de conmutación (salida de impulsos)		
Q0.0 y Q0.1	máx. 20 kHz	máx. 1 Hz
Relé		
Retardo de conmutación	–	máx. 10 ms
Vida útil mecánica (sin carga)	–	10.000.000 ciclos abiertos/cerrados
Vida útil contactos a carga nominal	–	100.000 ciclos abiertos/cerrados
Longitud del cable		
No apantallado	150 m	150 m
Apantallado	500 m	500 m

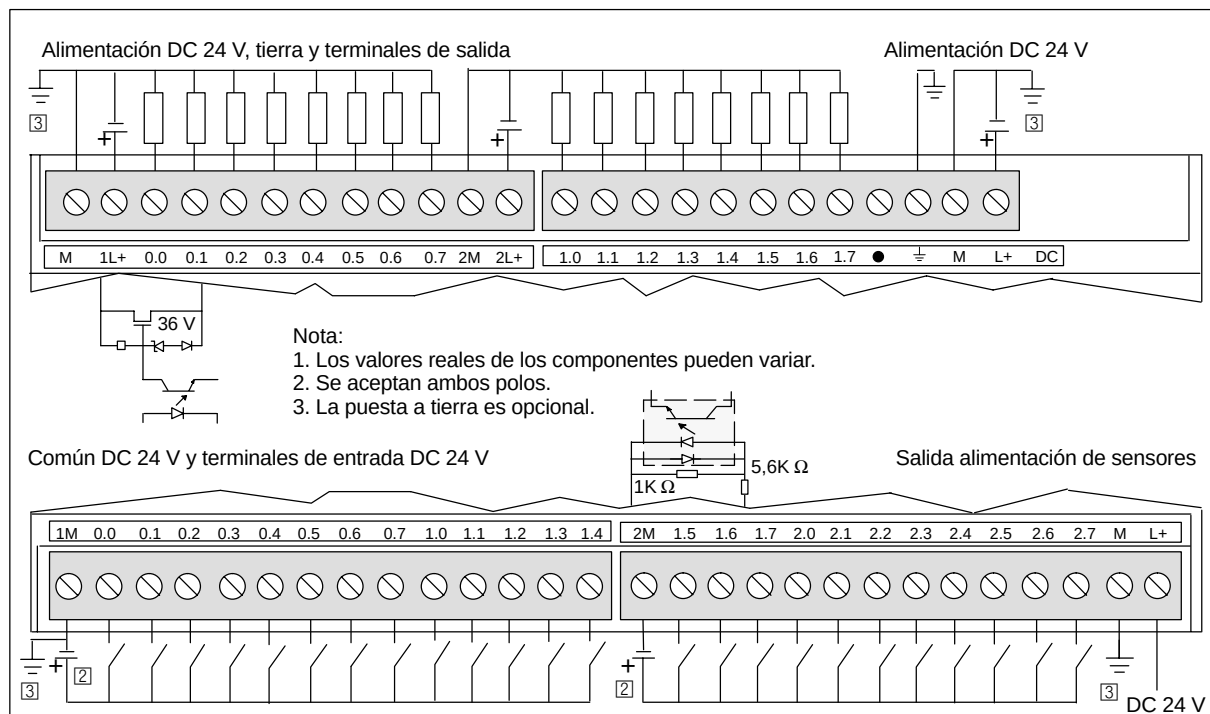


Figura A-8 Identificación de terminales de conexión para la CPU 226 DC/DC/DC

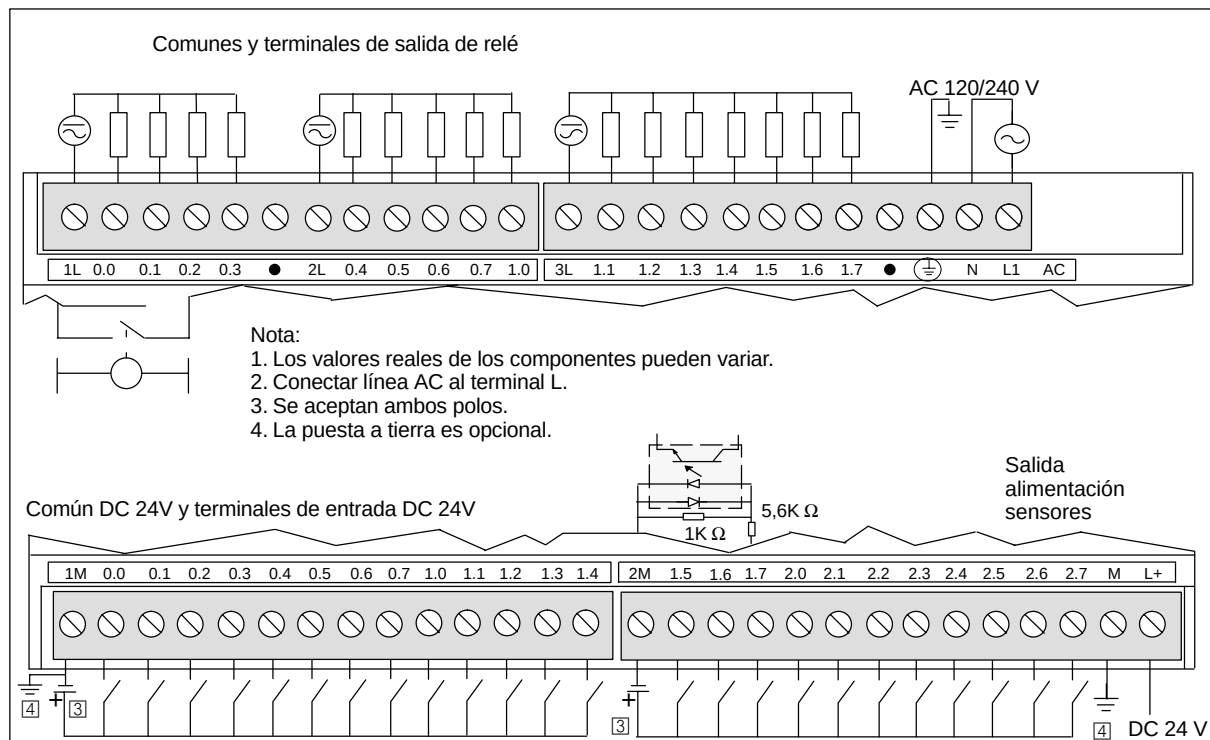


Figura A-9 Identificación de terminales de conexión para la CPU 226 AC/DC/Relé

A.6 Datos técnicos del módulo de ampliación EM 221 de entradas digitales

Tabla A-6 Datos técnicos del módulo de ampliación EM 221, 8 entradas digitales DC 24 V

Descripción Nº de referencia	EM 221, 8 entradas digitales x DC 24 V 6ES7 221-1BF20-0XA0
Tamaño físico	
Dimensiones (l x a x p)	46 x 80 x 62 mm
Peso	150 g
Pérdida de corriente (disipación)	2 W
Características de las entradas	
Nº de entradas integradas	8 entradas
Tipo de entrada	Sumidero de corriente/fuente (tipo 1 IEC con sumidero de corriente)
Tensión de entrada	
Tensión máx. continua admisible	DC 30 V
Sobretensión transitoria	DC 35 V, 0,5 s
Valor nominal	DC 24 V a 4 mA, nominal
Señal 1 lógica (mín.)	mín. DC 15 V a 2,5 mA
Señal 0 lógica (máx.)	máx. DC 5 V a 1 mA
Aislamiento	
Separación galvánica	AC 500 V, 1 minuto
Grupos de aislamiento de:	4 E/S
Tiempos de retardo de las entradas	
Máximo	4,5 ms
Conexión de sensor de proximidad de 2 hilos (Bero)	
Corriente de fuga admisible	máx. 1 mA
Longitud del cable	
No apantallado	300 m
Apantallado	500 m
Nº de entradas ON simultáneamente	
40° C	8
55° C	8
Consumo de corriente	
De +DC 5 V (del bus de ampliación)	30 mA

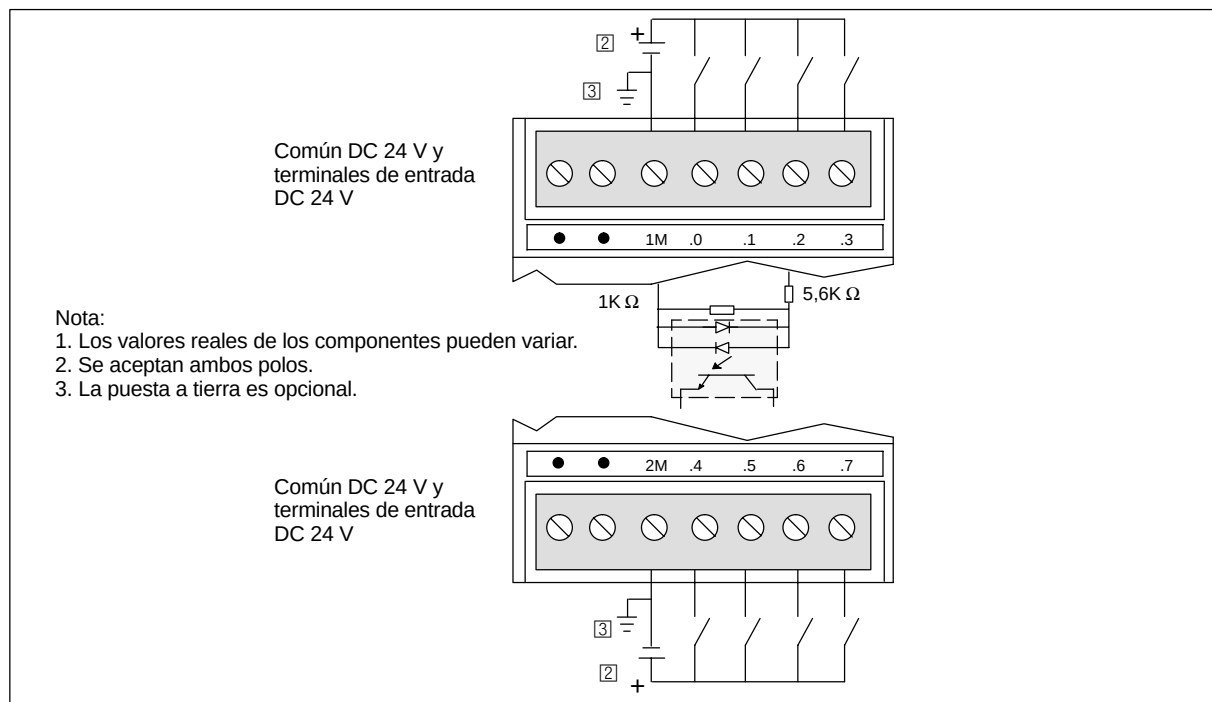


Figura A-10 Identificación de terminales de conexión para el EM 221, 8 entradas digitales x DC 24 V

A.7 Datos técnicos de los módulos de ampliación EM 222 de salidas digitales

Tabla A-7 Datos técnicos de los módulos de ampliación EM 222, salidas DC 24 V y salidas de relé

Descripción Nº de referencia	EM 222, 8 salidas digitales x DC 24 V 6ES7 222-1BF20-0XA0	EM 222, 8 salidas digitales x relé 6ES7 222-1HF20-0XA0
Tamaño físico		
Dimensiones (l x a x p)	46 x 80 x 62 mm	46 x 80 x 62 mm
Peso	150 g	170 g
Pérdida de corriente (disipación)	2 W	2 W
Características de la CPU		
Nº de salidas	8 salidas	8 salidas
Tipo de salida	Estado sólido-MOSFET	Relé, contacto de baja potencia
Tensión de salida		
Margen admisible	DC 20,4 a 28,8 V	DC 5 a 30 V ó AC 5 a 250 V
Valor nominal	DC 24 V	—
Señal 1 lógica a corriente máxima	mín. DC 20 V	—
Señal lógica 0 con 10 K Ω de carga	máx. DC 0,1 V	—
Corriente de salida		
Señal 1 lógica	0,75 A	2,00 A
Nº de grupos de salidas	2	2
Nº de salidas ON (máx.)	8	8
Por grupo – montaje horizontal (máx.)	4	4
Por grupo – montaje vertical (máx.)	4	4
Corriente máx. por común/grupo	3 A	8 A
Carga LEDs	5 W	30 W DC/200 W AC
Resistencia estado ON (resistencia contactos)	0,3 Ω	0,2 Ω , máx. si son nuevas
Corriente de derivación por salida	máx. 10 μ A	—
Sobrecorriente momentánea	máx. 8 A, 100 ms	7A al estar cerrados los contactos
Protección contra sobrecargas	no	no
Aislamiento		
Separación galvánica	AC 500 V, 1 minuto	—
Resistencia de aislamiento	—	100 M Ω , mín. si son nuevas
Aislamiento bobina a contacto	—	AC 500 V, 1 minuto
Aislamiento entre contactos abiertos	—	AC 750 V, 1 minuto
En grupos de:	4 E/S	4 E/S
Carga inductiva, apriete		
Repetición disipación de energía < 0.5 L I ² x frecuencia de conmutación	1 W, en todos los canales	—
Límites tensión de bloqueo	L+ menos 48 V	—
Retardo de las salidas		
OFF a ON	máx. 50 μ s	—
ON a OFF	máx. 200 μ s	—
Relé		
Retardo de conmutación	—	máx. 10 ms
Vida útil mecánica (sin carga)	—	10.000.000 ciclos abiertos/cerrados
Vida útil contactos a carga nominal	—	100.000 ciclos abiertos/cerrados
Longitud del cable		
No apantallado	150 m	150 m
Apantallado	500 m	500 m
Consumo de corriente		
De +DC 5 V (del bus de ampliación)	50 mA	40 mA
De L+	—	9 mA por salida en ON
L+ margen tensión aliment. bobina	—	DC 20,4 a 28,8 V

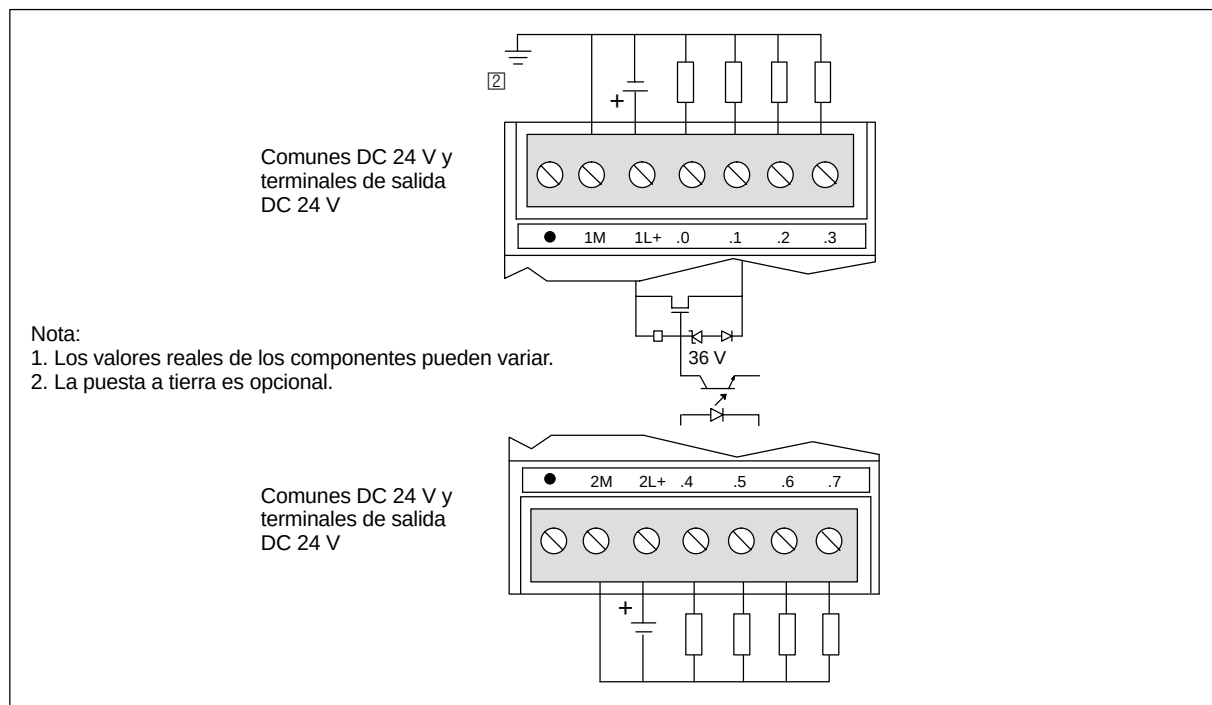


Figura A-11 Identificación de terminales de conexión para el EM 222, 8 salidas digitales x DC 24 V

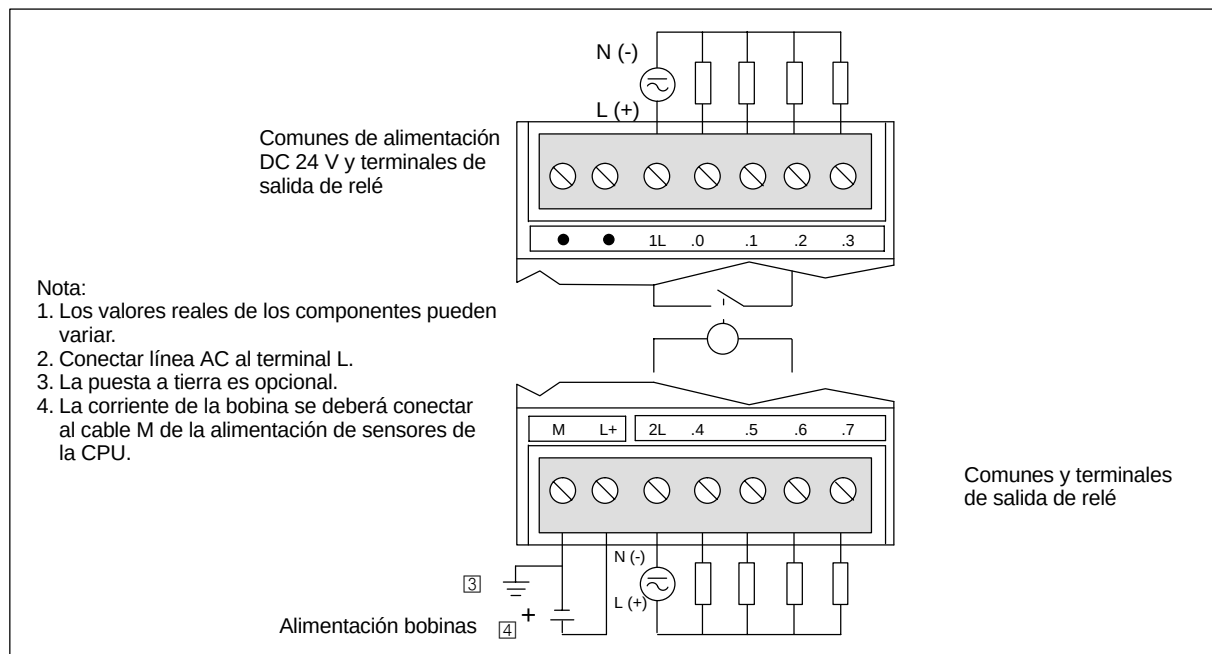


Figura A-12 Identificación de terminales de conexión para el EM 222, 8 salidas digitales x relé

A.8 Datos técnicos de los módulos de ampliación EM 223, 4 entradas digitales/4 salidas digitales

Tabla A-8 Datos técnicos de los módulos de ampliación EM 223, 4 entradas/4 salidas x DC 24 V
y EM 223, 4 entradas x DC 24 V/4 salidas de relé

Descripción Nº de referencia	EM 223, 4 entradas digitales/4 salidas digitales X DC 24 V 6ES7 223-1BF20-0XA0	EM 223, 4 entradas digitales/4 salidas digitales de relé X DC 24 V 6ES7 223-1HF20-0XA0
Tamaño físico		
Dimensiones (l x a x p)	46 mm x 80 mm x 62 mm	46 mm x 80 mm x 62 mm
Peso	160 g	170 g
Pérdida de corriente (disipación)	2 W	2 W
Características de las entradas		
Nº de entradas	4 entradas	4 entradas
Tipo de entrada	Sumidero de corriente/fuente (tipo 1 IEC con sumidero de corriente)	Sumidero de corriente/fuente (tipo 1 IEC con sumidero de corriente)
Tensión de entrada		
Tensión máx. continua admisible	DC 30 V	DC 30 V
Sobretensión transitoria	DC 35 V, 0,5 s	DC 35 V, 0,5 s
Valor nominal	DC 24 V a 4 mA, nominal	DC 24 V a 4 mA, nominal
Señal 1 lógica (mín.)	mín. DC 15 V a 2,5 mA	mín. DC 15 V a 2,5 mA
Señal 0 lógica (máx.)	máx. DC 5 V a 1 mA	máx. DC 5 V a 1 mA
Aislamiento		
Separación galvánica	AC 500 V, 1 minuto	AC 500 V, 1 minuto
Grupos de aislamiento de:	4 E/S	4 E/S
Tiempos de retardo de las entradas		
Máximo	4,5 ms	4,5 ms
Conexión de sensor de proximidad de dos hilos (Bero)		
Máximo	máx. 1 mA	máx. 1 mA
Longitud del cable		
No apantallado	300 m	300 m
Apantallado	500 m	500 m
Nº de entradas ON simultáneamente		
40° C	4	4
55° C	4	4

Tabla A-8 Datos técnicos de los módulos de ampliación EM 223, 4 entradas/4 salidas x DC 24 V y EM 223, 4 entradas x DC 24 V/4 salidas de relé (continuación)

Descripción Nº de referencia	EM 223, 4 entradas digitales/4 salidas digitales X DC 24 V 6ES7 223-1BF20-0XA0	EM 223, 4 entradas digitales/4 salidas digitales de relé X DC 24 V 6ES7 223-1HF20-0XA0
Características de las salidas		
Nº de salidas integradas	4 salidas	4 salidas
Tipo de salida	Estado sólido–MOSFET	Relé, contacto de baja potencia
Tensión de salida		
Margen admisible	DC 20,4 a 28,8 V	DC 5 a 30 V ó AC 5 a 250 V
Valor nominal	DC 24 V	–
Señal 1 lógica a corriente máxima	mín. DC 20 V	–
Señal 0 lógica con 10K Ω de carga	máx. DC 0,1 V	–
Corriente de salida		
Señal 1 lógica	0,75 A	2,00 A
Nº de grupos de salidas	1	1
Nº de salidas ON (máx.)	4	4
Por grupo – montaje horizontal (máx.)	4	4
Por grupo – montaje vertical (máx.)	4	4
Corriente máx. por común/grupo	3 A	8 A
Carga LEDs	5 W	30 W DC/200 W AC
Resistencia estado ON (resistencia contactos)	0,3 Ω	0,2 Ω , máx. si son nuevas
Corriente de derivación por salida	máx. 10 μ A	–
Sobrecorriente momentánea	máx. 8 A, 100 ms	7A al estar cerrados los contactos
Protección contra sobrecargas	no	no
Aislamiento		
Separación galvánica	AC 500 V, 1 minuto	–
Resistencia de aislamiento	–	100 M Ω , mín. si son nuevas
Aislamiento bobina a contacto	–	AC 500 V, 1 minuto
Aislamiento entre contactos abiertos	–	AC 750 V, 1 minuto
En grupos de:	4 E/S	4 E/S
Carga inductiva, apriete		
Repetición disipación de energía < 0.5 L I ² x frecuencia de conmutación	1 W, en todos los canales	–
Límites tensión de bloqueo	L+ menos 48V	–
Retardo de las salidas		
OFF a ON	máx. 50 μ s	–
ON a OFF	máx. 200 μ s	–
Relé		
Retardo de conmutación	–	máx. 10 ms
Vida útil mecánica (sin carga)	–	10.000.000 ciclos abiertos/cerrados
Vida útil contactos a carga nominal	–	100.000 ciclos abiertos/cerrados
Longitud del cable		
No apantallado	150 m	150 m
Apantallado	500 m	500 m
Consumo de corriente		
De +DC 5 V (del bus de ampliación)	40 mA	40 mA
De L+	–	9 mA por salida en ON
L+ margen tensión aliment. bobina	–	DC 20,4 a 28,8 V

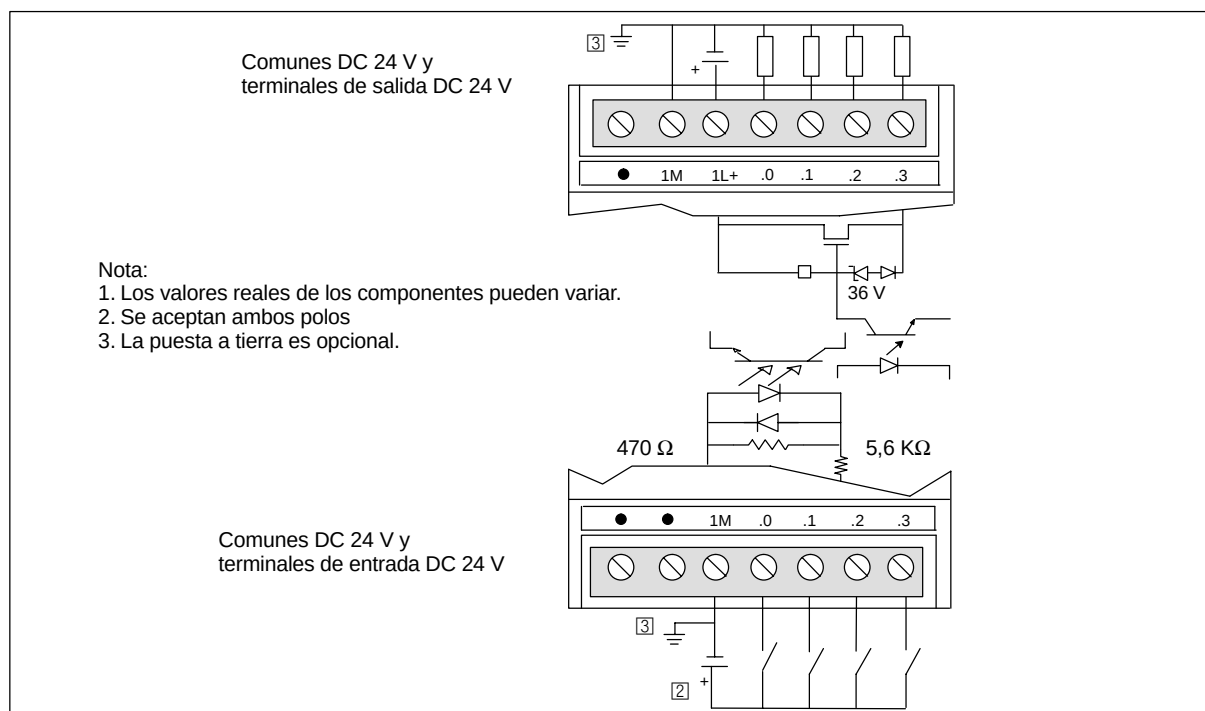


Figura A-13 Identificación de terminales de conexión para el EM 223, 4 entradas digitales x DC 24 V / 4 salidas digitales x DC 24 V

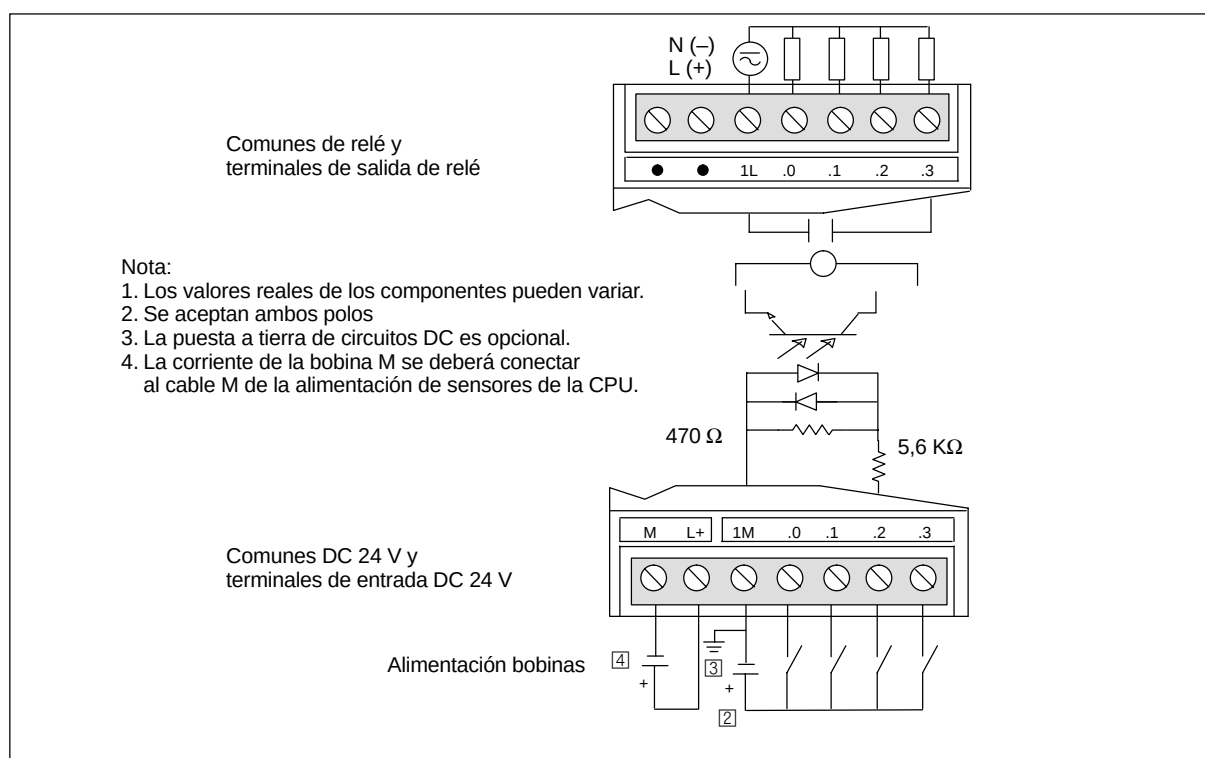


Figura A-14 Identificación de terminales de conexión para el EM 223, 4 entradas digitales x DC 24 V / 4 salidas de relé

A.9 Datos técnicos de los módulos de ampliación EM 223, 8 entradas digitales/8 salidas digitales

Tabla A-9 Datos técnicos de los módulos de ampliación EM 223, 8 entradas/8 salidas x DC 24 V y EM 223, 8 entradas x DC 24 V/8 salidas de relé

Descripción Nº de referencia	EM 223, 8 entradas digitales/8 salidas digitales X DC 24 V 6ES7 223-1BH20-0XA0	EM 223, 8 entradas digitales/8 salidas digitales de relé X DC 24 V 6ES7 223-1PH20-0XA0
Tamaño físico		
Dimensiones (l x a x p)	71,2 mm x 80 mm x 62 mm	71,2 mm x 80 mm x 62 mm
Peso	200 g	300 g
Pérdida de corriente (disipación)	3 W	3 W
Características de las entradas		
Nº de entradas	8 entradas	8 entradas
Tipo de entrada	Sumidero de corriente/fuente (tipo 1 IEC con sumidero de corriente)	Sumidero de corriente/fuente (tipo 1 IEC con sumidero de corriente)
Tensión de entrada		
Tensión máx. continua admisible	DC 30 V	DC 30 V
Sobretensión transitoria	DC 35 V, 0,5 s	DC 35 V, 0,5 s
Valor nominal	DC 24 V a 4 mA, nominal	DC 24 V a 4 mA, nominal
Señal 1 lógica (mín.)	mín. DC 15 V a 2,5 mA	mín. DC 15 V a 2,5 mA
Señal 0 lógica (máx.)	máx. DC 5 V a 1 mA	máx. DC 5 V a 1 mA
Aislamiento		
Separación galvánica	AC 500 V, 1 minuto	AC 500 V, 1 minuto
Grupos de aislamiento de:	4 E/S	4 E/S
Tiempos de retardo de las entradas		
Máximo	4,5 ms	4,5 ms
Conexión de sensor de proximidad de dos hilos (Bero)		
Máximo	máx. 1 mA	máx. 1 mA
Longitud del cable		
No apantallado	300 m	300 m
Apantallado	500 m	500 m
Nº de entradas ON simultáneamente		
40° C	8	8
55° C	8	8
Características de las salidas		
Nº de salidas integradas	8 salidas	8 salidas
Tipo de salida	Estado sólido-MOSFET	Relé, contacto de baja potencia
Tensión de salida		
Margen admisible	DC 20,4 a 28,8 V	DC 5 a 30 V ó AC 5 a 250 V
Valor nominal	DC 24 V	—
Señal 1 lógica a corriente máxima	mín. DC 20 V	—
Señal 0 lógica con 10K Ω de carga	máx. DC 0,1 V	—

Tabla A-9 Datos técnicos de los módulos de ampliación EM 223, 8 entradas/8 salidas x DC 24 V y EM 223, 8 entradas x DC 24 V/8 salidas de relé (continuación)

Descripción Nº de referencia	EM 223, 8 entradas digitales/8 salidas digitales X DC 24 V 6ES7 223-1BH20-0XA0	EM 223, 8 entradas digitales/8 salidas digitales de relé X DC 24 V 6ES7 223-1PH20-0XA0
Corriente de salida		
Señal 1 lógica	0,75 A	2,00 A
Nº de grupos de salidas	2	2
Nº de salidas ON (máx.)	8	8
Por grupo – montaje horizontal (máx.)	4	4
Por grupo – montaje vertical (máx.)	4	4
Corriente máx. por común/grupo	3 A	8 A
Carga LEDs	5 W	30 W DC/200 W AC
Resistencia estado ON (resistencia contactos)	0,3Ω	0,2 Ω, máx. si son nuevas
Corriente de derivación por salida	máx. 10 µA	–
Sobrecorriente momentánea	máx. 8 A, 100 ms	7A al estar cerrados los contactos
Protección contra sobrecargas	no	no
Aislamiento		
Separación galvánica	AC 500 V, 1 minuto	–
Resistencia de aislamiento	–	100 M Ω, mín. si son nuevas
Aislamiento bobina a contacto	–	AC 500 V, 1 minuto
Aislamiento entre contactos abiertos	–	AC 750 V, 1 minuto
En grupos de:	4 E/S	4 E/S
Carga inductiva, apriete		
Repetición disipación de energía < 0.5 L I ² x frecuencia de conmutación	1 W, en todos los canales	–
Límites tensión de bloqueo	L+ menos 48V	–
Retardo de las salidas		
OFF a ON	máx. 50 µs	–
ON a OFF	máx. 200 µs	–
Relé		
Retardo de conmutación	–	máx. 10 ms
Vida útil mecánica (sin carga)	–	100.000.000 ciclos abiertos/cerrados
Vida útil contactos a carga nominal	–	100.000 ciclos abiertos/cerrados
Longitud del cable		
No apantallado	150 m	150 m
Apantallado	500 m	500 m
Consumo de corriente		
De +DC 5 V (del bus de ampliación)	80 mA	80 mA
De L+	–	9 mA por salida en ON
L+ margen tensión aliment. bobina	–	DC 20,4 a 28,8 V

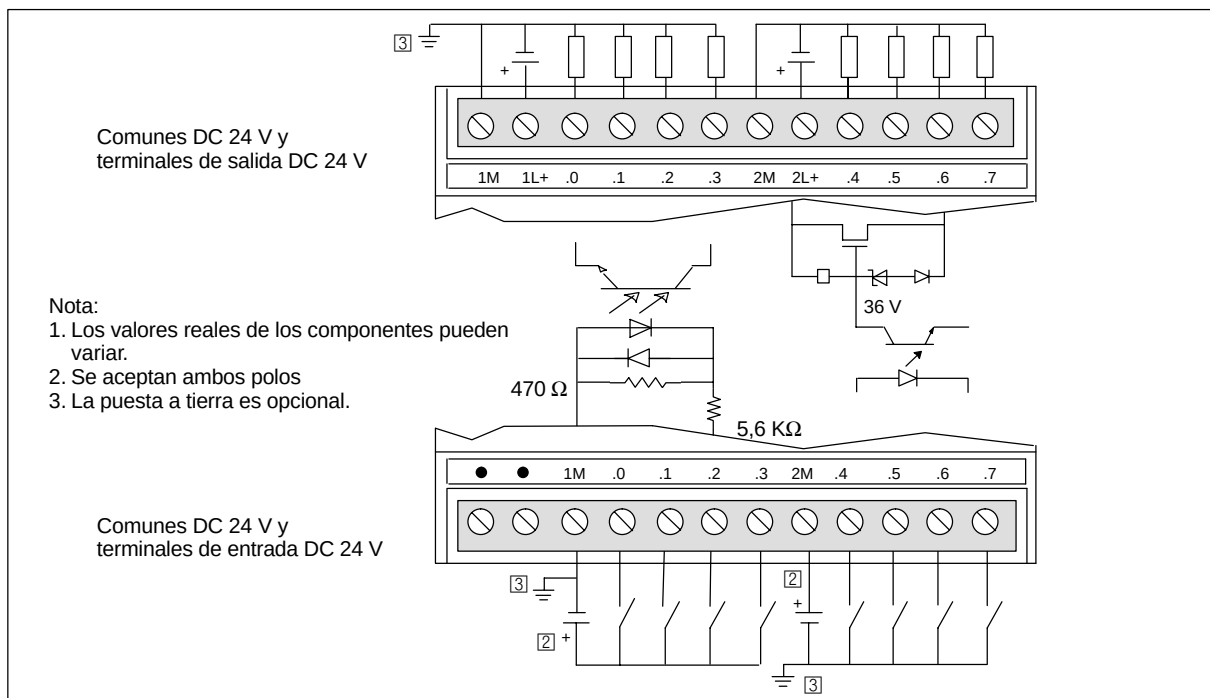


Figura A-15 Identificación de terminales de conexión para el EM 223, 8 entradas digitales x DC 24 V / 8 salidas digitales x DC 24 V

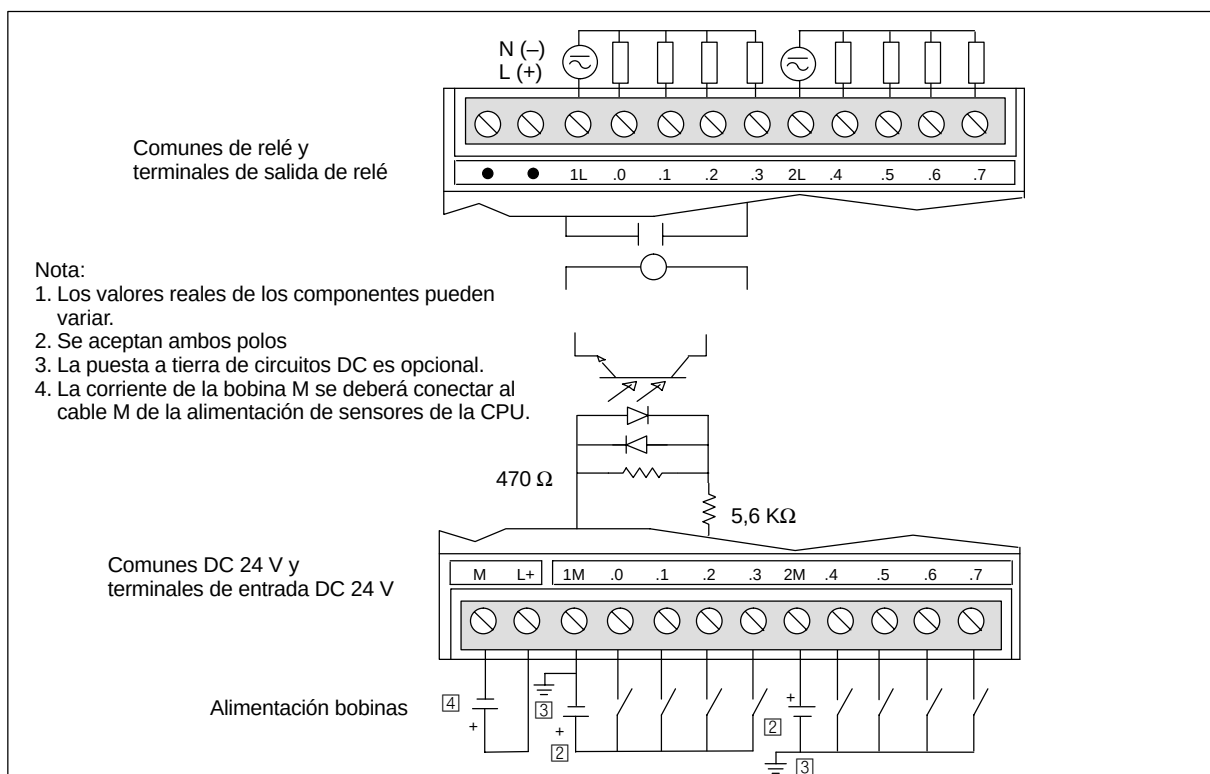


Figura A-16 Identificación de terminales de conexión para el EM 223, 8 entradas digitales DC 24 V / 8 salidas digitales de relé

A.10 Datos técnicos de los módulos de ampliación EM 223, 16 entradas digitales/16 salidas digitales

Tabla A-10 Datos técnicos de los módulos de ampliación EM 223, 16 entradas/16 salidas x DC 24 V y EM 223, 16 entradas x DC 24 V/16 salidas de relé

Descripción Nº de referencia	EM 223, 16 entradas digitales/ 16 salidas digitales X DC 24 V 6ES7 223-1BL20-0XA0	EM 223, 16 entradas digitales/ 16 salidas digitales de relé X DC 24 V 6ES7 223-1PL20-0XA0
Tamaño físico		
Dimensiones (l x a x p)	137,3 mm x 80 mm x 62 mm	137,3 mm x 80 mm x 62 mm
Peso	360 g	400 g
Pérdida de corriente (disipación)	6 W	6 W
Características de las entradas		
Nº de entradas	16 entradas	16 entradas
Tipo de entrada	Sumidero de corriente/fuente (tipo 1 IEC con sumidero de corriente)	Sumidero de corriente/fuente (tipo 1 IEC con sumidero de corriente)
Tensión de entrada		
Tensión máx. continua admisible	DC 30 V	DC 30 V
Sobretensión transitoria	DC 35 V, 0,5 s	DC 35 V, 0,5 s
Valor nominal	DC 24 V a 4 mA, nominal	DC 24 V a 4 mA, nominal
Señal 1 lógica (mín.)	mín. DC 15 V a 2,5 mA	mín. DC 15 V a 2,5 mA
Señal 0 lógica (máx.)	máx. DC 5 V a 1 mA	máx. DC 5 V a 1 mA
Aislamiento		
Separación galvánica	AC 500 V, 1 minuto	AC 500 V, 1 minuto
Grupos de aislamiento de:	8 E/S	8 E/S
Tiempos de retardo de las entradas		
Máximo	4,5 ms	4,5 ms
Conexión de sensor de proximidad de dos hilos (Bero)		
Máximo	máx. 1 mA	máx. 1 mA
Longitud del cable		
No apantallado	300 m	300 m
Apantallado	500 m	500 m
Nº de entradas ON simultáneamente		
40° C	16	16
55° C	16	16

Tabla A-10 Datos técnicos de los módulos de ampliación EM 223, 16 entradas/16 salidas x DC 24 V y EM 223, 16 entradas x DC 24 V/16 salidas de relé (continuación)

Descripción Nº de referencia	EM 223, 16 entradas digitales/ 16 salidas digitales X DC 24 V 6ES7 223-1BL20-0XA0	EM 223, 16 entradas digitales/ 16 salidas digitales de relé X DC 24 V 6ES7 223-1PL20-0XA0
Características de las salidas		
Nº de salidas integradas	16 salidas	16 salidas
Tipo de salida	Estado sólido-MOSFET	Relé, contacto de baja potencia
Tensión de salida		
Margen admisible	DC 20,4 a 28,8 V	DC 5 a 30 V ó AC 5 a 250 V
Valor nominal	DC 24 V	—
Señal 1 lógica a corriente máxima	mín. DC 20 V	—
Señal 0 lógica con 10K Ω de carga	máx. DC 0,1 V	—
Corriente de salida		
Señal 1 lógica	0,75 A	2.00 A
Nº de grupos de salidas	3	4
Nº de salidas ON (máx.)	16	16
Por grupo – montaje horizontal (máx.)	4/4/8	4
Por grupo – montaje vertical (máx.)	4/4/8	4
Corriente máx. por común/grupo	3/3/6 A	8 A
Carga LEDs	5 W	30 W DC/200 W AC
Resistencia estado ON (resistencia contactos)	0,3 Ω máx. 10 μ A	0,2 Ω , máx. si son nuevas —
Corriente de derivación por salida	máx. 8 A, 100 ms	7A al estar cerrados los contactos
Sobrecorriente momentánea	no	no
Protección contra sobrecargas		
Aislamiento		
Separación galvánica	AC 500 V, 1 minuto	—
Resistencia de aislamiento	—	100 M Ω , mín. si son nuevas
Aislamiento bobina a contacto	—	AC 500 V, 1 minuto
Aislamiento entre contactos abiertos	—	AC 750 V, 1 minuto
En grupos de:	4 E/S / 4 E/S / 8 E/S	4 E/S
Carga inductiva, apriete		
Repetición disipación de energía < 0.5 L I ² x frecuencia de conmutación	1 W, en todos los canales	—
Límites tensión de bloqueo	L+ menos 48V	—
Retardo de las salidas		
OFF a ON	máx. 50 μ s	—
ON a OFF	máx. 200 μ s	—
Relé		
Retardo de conmutación	—	máx. 10 ms
Vida útil mecánica (sin carga)	—	10.000.000 ciclos abiertos/cerrados
Vida útil contactos a carga nominal	—	100.000 ciclos abiertos/cerrados
Longitud del cable		
No apantallado	150 m	150 m
Apantallado	500 m	500 m
Consumo de corriente		
De +DC 5 V (del bus de ampliación)	160 mA	150 mA
De L+	—	9 mA por salida en ON
L+ margen tensión aliment. bobina	—	DC 20,4 a 28,8 V

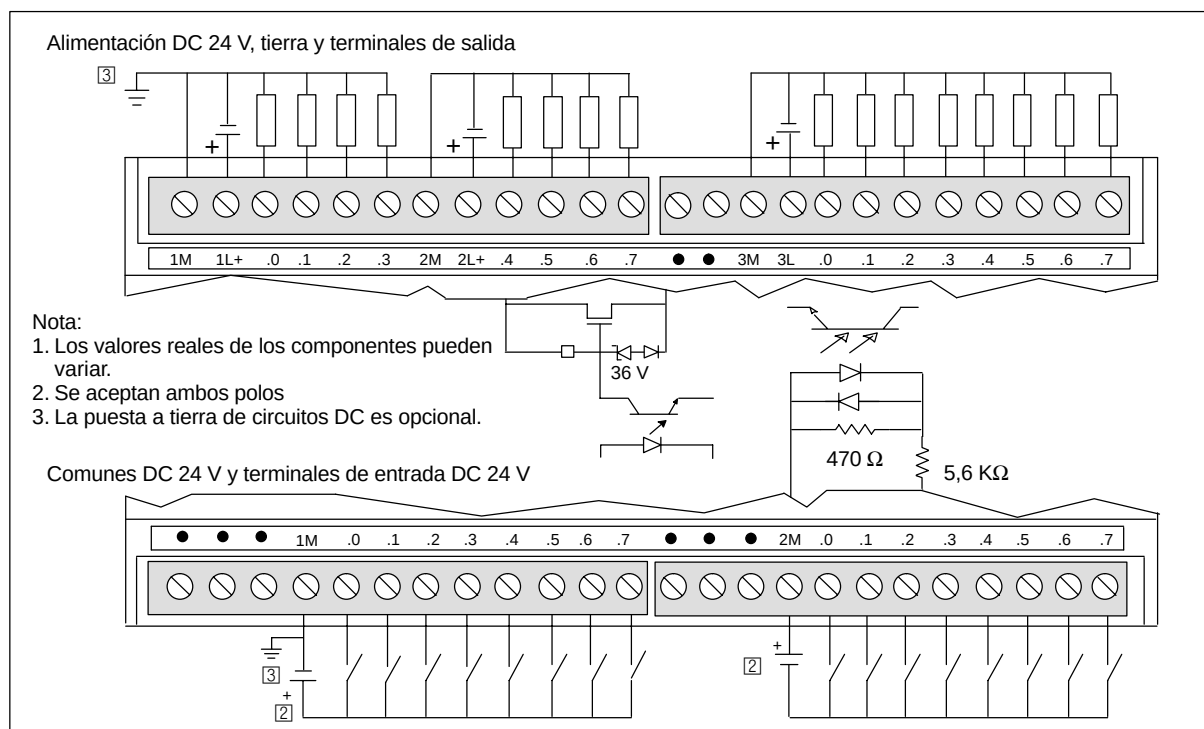


Figura A-17 Identificación de terminales de conexión para el EM 223, 16 entradas digitales x DC 24 V/
16 salidas digitales x DC 24 V

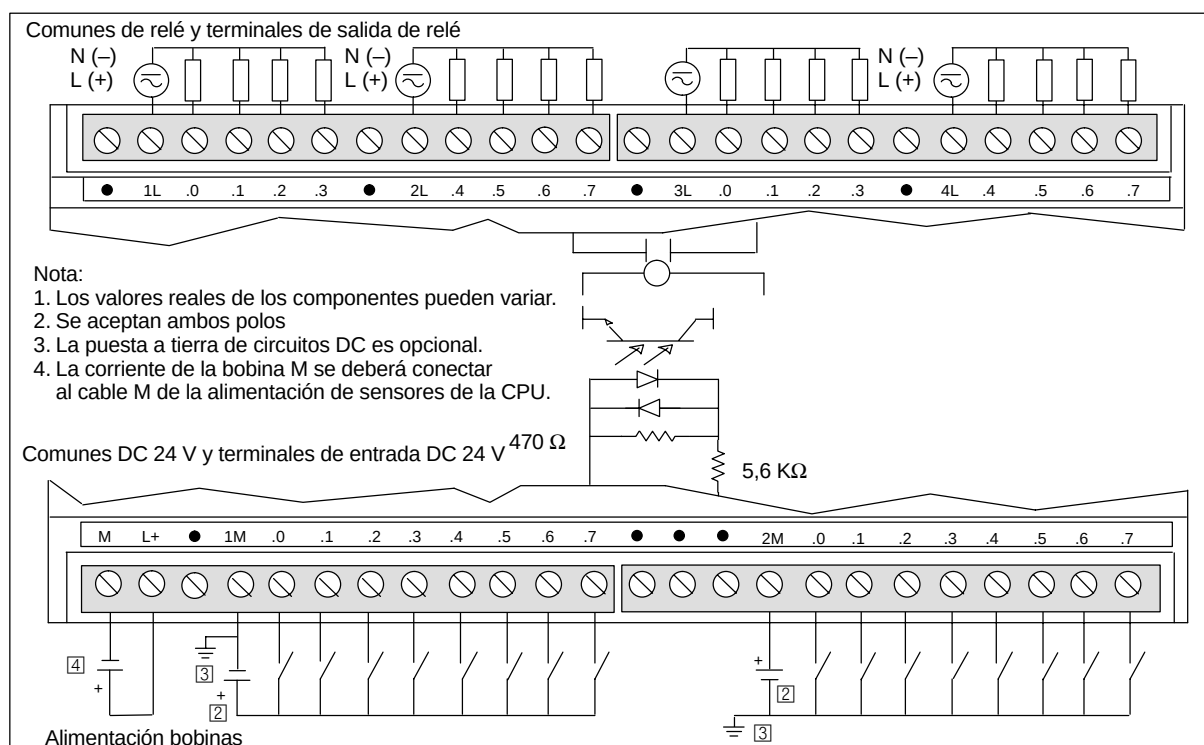


Figura A-18 Identificación de terminales de conexión para el EM 223, 16 entradas digitales DC 24 V/
16 salidas digitales de relé

A.11 Datos técnicos de los módulos de ampliación EM 231, EM 232 y EM 235 de entradas y/o salidas analógicas

Tabla A-11 Datos técnicos de los módulos de ampliación EM 231, EM 232 y EM 235 de entradas y/o salidas analógicas

Descripción Nº de referencia	EM 231, 4 entradas analógicas x 12 bits 6ES7 231-0HC20-0XA0	EM 232, 2 salidas analógicas x 12 bits 6ES7 232-0HB20-0XA0	EM 235, 4 entradas analógicas/4 salidas analógicas x 12 bits 6ES7 235-0KD20-0XA0	
	Datos de las entradas	Datos de las salidas	Datos de las entradas	Datos de las salidas
Datos generales				
Dimensiones (l x a x p)	71,2 mm x 80 mm x 62 mm	46 mm x 80 mm x 62 mm	71,2 mm x 80 mm x 62 mm	
Peso	183 g	148 g	186 g	
Pérdida de corriente (disipación)	2 W	2 W	2 W	
Cantidad de E/S físicas	4 entradas analógicas	2 salidas analógicas	4 entradas analógicas, 1 salida analógica	
Consumo de corriente				
De +DC 5 V (del bus de ampliación)	20 mA	20 mA	30 mA	
De L+	60 mA	70 mA (ambas salidas a 20 mA)	60 mA (salida a 20 mA)	
Margen de tensión L+, clase 2 o alimentación de sensores DC	20,4 a 28,8	20,4 a 28,8	20,4 a 28,8	
Indicador LED	Alimentación DC 24 V, ON = correcta, OFF = sin corriente DC 24 V	Alimentación DC 24 V, ON = correcta, OFF = sin corriente DC 24 V	Alimentación DC 24 V, ON = correcta, OFF = sin corriente DC 24 V	
Datos de las entradas analógicas				
Formato palabra de datos	(v. figura A-21)		(v. figura A-21).	
Bipolar, margen máx.	–32000 a +32000		–32000 a +32000	
Unipolar, margen máx.	0 a 32000		0 a 32000	
Impedancia de entrada	≥10 MΩ		≥ 10 MΩ	
Atenuación filtro de entrada	–3 db @ 3,1 kHz		–3 db @ 3,1 kHz	
Tensión de entrada máxima	DC 30 V		DC 30 V	
Corriente de entrada máxima	32 mA		32 mA	
Resolución	Convertidor A/D de 12 bits		Convertidor A/D de 12 bits	
Nº de entradas analógicas	4		4	
Aislamiento (campo a circuito lógico)	Ninguno		Ninguno	
Tipo de entrada	Diferencial		Diferencial	
Márgenes de las entradas				
Tensión (unipolar)	0 a 10 V, 0 a 5 V		0 a 10 V, 0 a 5 V, 0 a 1 V, 0 a 500 mV, 0 a 100 mV, 0 a 50 mV	
Tensión (bipolar)	±5 V, ± 2,5 V		± 10 V, ± 5 V, ± 2,5 V, ± 1 V, ± 500 mV, ± 250 mV, ± 100 mV, ± 50 mV, ± 25 mV	
Corriente	0 a 20 mA		0 a 20 mA	
Resolución de las entradas	(v. tabla A-5)	(v. tabla A-13)		
Tensión (unipolar)				
Tensión (bipolar)				
Corriente				
Tiempo de conversión analógica/digital	< 250 μs	< 250 μs		
Respuesta de salto de la entrada analógica	1,5 ms a 95%	1,5 ms a 95%		
Rechazo en modo común	40 dB, DC a 60 Hz	40 dB, DC a 60 Hz		
Tensión en modo común	Tensión de señal más tensión en modo común (debe ser ≤ 12 V)	Tensión de señal más tensión en modo común (debe ser ≤ 12 V)		

Descripción Nº de referencia	EM 231, 4 entradas analógicas x 12 bits 6ES7 231-0HC20-0XA0	EM 232, 2 salidas analógicas x 12 bits 6ES7 232-0HB20-0XA0	EM 235, 4 entradas analógicas/4 salidas analógicas x 12 bits 6ES7 235-0KD20-0XA0
	Datos de las entradas	Datos de las salidas	Datos de las entradas
Datos de las salidas analógicas			
Nº de salidas analógicas		2	1
Aislamiento (campo a circuito lógico)		Ninguno	Ninguno
Margen de señales Salida de tensión Salida de corriente		± 10 V 0 a 20 mA	± 10 V 0 a 20 mA
Resolución, margen máx. Tensión Corriente		12 bits 11 bits	12 bits 11 bits
Formato palabra de datos Tensión Corriente		–32000 a +32000 0 a +32000	–32000 a +32000 0 a +32000
Precisión Caso más desfavorable, 0° a 55° C Salida de tensión Salida de corriente Típico, 25° C Salida de tensión Salida de corriente		± 2% de margen máx. ± 2% de margen máx. ± 0,5 % de margen máx. ± 0,5 % de margen máx.	± 2% de margen máx. ± 2% de margen máx. ± 0,5 % de margen máx. ± 0,5 % de margen máx.
Tiempo de estabilización Salida de tensión Salida de corriente		100 µS 2 mS	100 µS 2 mS
Accionamiento máx. Salida de tensión Salida de corriente		mín. 5000 Ω máx. 500 Ω	mín. 5000 Ω máx. 500 Ω

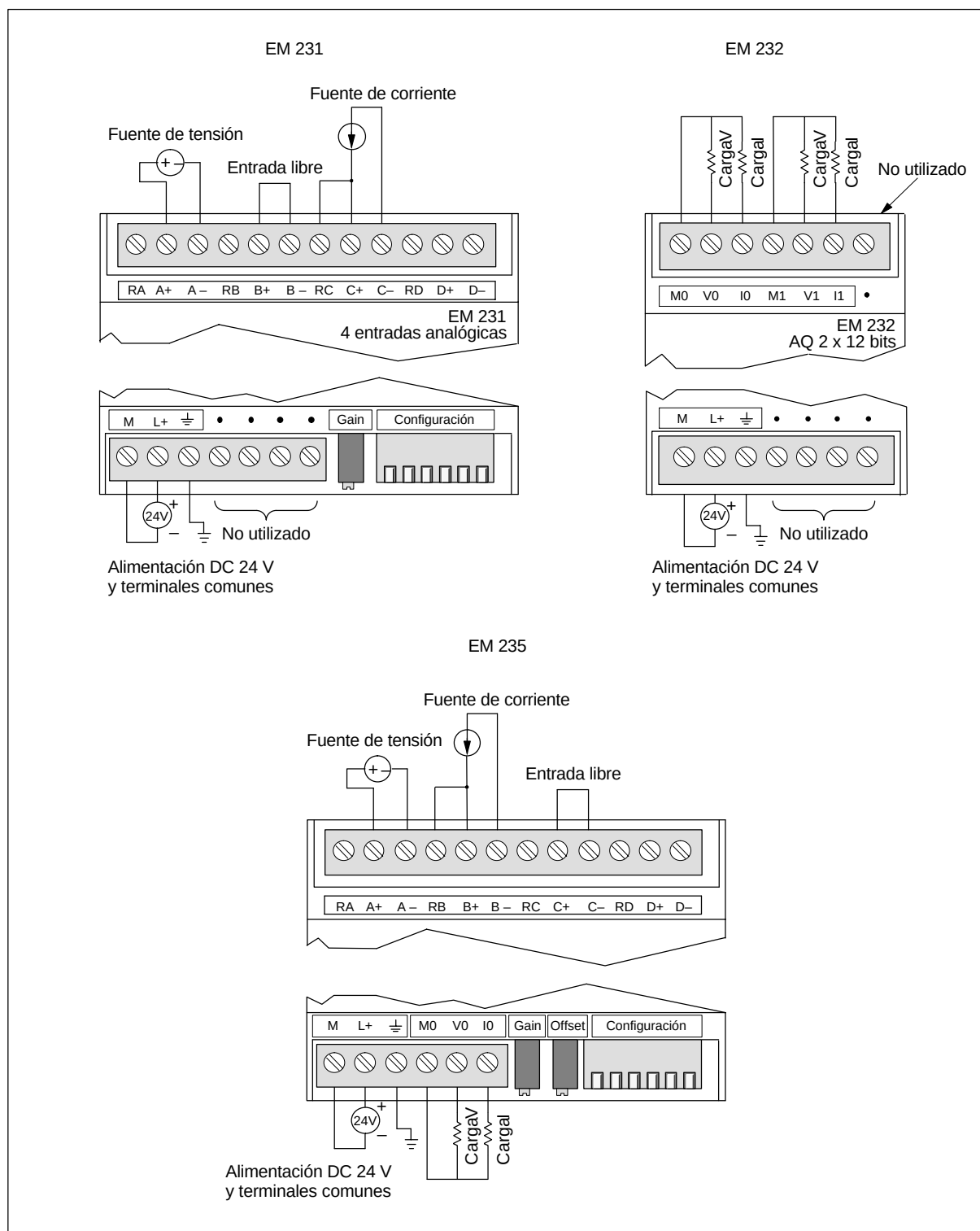


Figura A-19 Identificación de terminales de conexión para los módulos de ampliación EM 231, EM 232 y EM 235

Calibración de entradas

Los ajustes de calibración afectan a la fase de amplificación de la instrumentación que sigue al multiplexor analógico (v. fig. A-22). Por consiguiente, la calibración afecta a todos los canales de entrada del usuario. Cualquier variación de los valores de los circuitos de entrada que preceden al multiplexor analógico provocará diferencias mínimas entre los valores de los distintos canales que estén conectados a la misma señal, incluso después de la calibración.

Para cumplir las especificaciones contenidas en la presente hoja de datos, es preciso utilizar filtros de entradas analógicas para todas las entradas del módulo. Elija 64 o más muestreos para calcular el valor promedio.

Para calibrar una entrada, siga los pasos siguientes:

1. Desconecte la alimentación del módulo. Seleccione el margen de entrada deseado.
2. Conecte la alimentación de la CPU y del módulo. Espere unos 15 minutos para que el módulo pueda estabilizarse.
3. Mediante una fuente de tensión o de corriente, aplique a una de las entradas una señal de valor cero.
4. Lea el valor que la CPU ha recibido del correspondiente canal de entrada.
5. Con el potenciómetro OFFSET, seleccione el valor cero u otro valor digital.
6. Aplique una señal de margen máximo a una entrada. Lea el valor que ha recibido la CPU.
7. Con el potenciómetro GAIN, seleccione el valor 32000 u otro valor digital.
8. En caso necesario, vuelva a calibrar el desplazamiento (OFFSET) y la ganancia (GAIN).

Calibración y configuración de los módulos de ampliación EM 231 y EM 235

Como muestra la figura A-20, el potenciómetro de calibración y los interruptores DIP de configuración están ubicados a la derecha del bloque de terminales inferior del módulo.

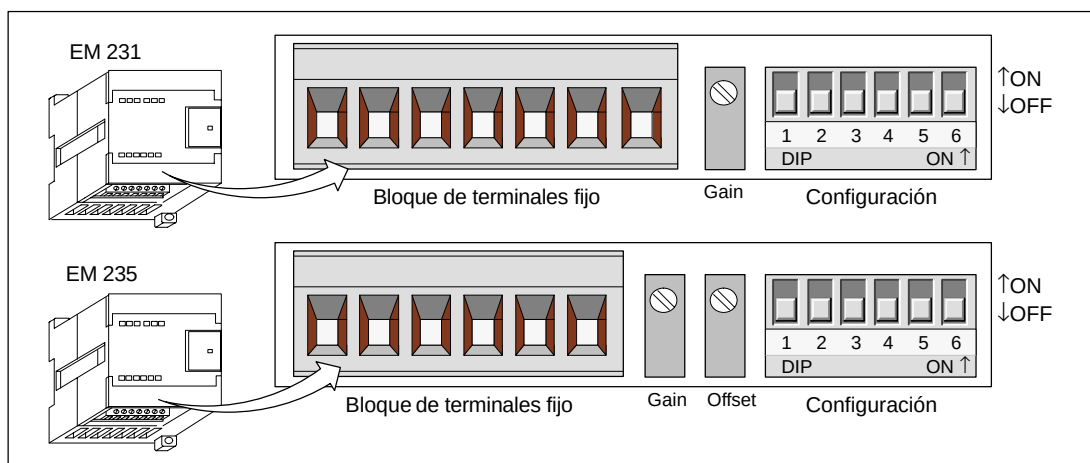


Figura A-20 Potenciómetros de calibración e interruptores DIP de configuración de los módulos de ampliación EM 231 y EM 235

Configuración del módulo de ampliación EM 231

La tabla A-12 muestra cómo configurar el módulo EM 231 utilizando los interruptores DIP. El margen de las entradas analógicas se selecciona con los interruptores 1, 2 y 3. Todas las entradas analógicas se activan en un mismo margen. En la tabla, ON está cerrado y OFF está abierto.

Tabla A-12 Tabla de interruptores de configuración del EM 231 para seleccionar el margen de las entradas analógicas

Unipolar			Margen de tensión	Resolución
Interruptor 1	Interruptor 2	Interruptor 3		
ON	OFF	ON	0 a 10 V	2,5 mV
	ON	OFF	0 a 5 V	1,25 mV
			0 a 20 mA	5 μA
Bipolar			Margen de tensión	Resolución
Interruptor 1	Interruptor 2	Interruptor 3		
OFF	OFF	ON	± 5 V	2,5 mV
	ON	OFF	± 2,5 V	1,25 mV

Configuración del módulo de ampliación EM 235

La tabla A-13 muestra cómo configurar el módulo EM 235 utilizando los interruptores DIP. El margen de las entradas analógicas y la resolución se seleccionan con los interruptores 1 a 6. Todas las entradas se activan en un mismo margen y formato. La tabla A-14 muestra cómo seleccionar el formato unipolar/bipolar (interruptor 6), la ganancia (interruptores 4 y 5) y la atenuación (interruptores 1, 2 y 3). En la tabla, ON está cerrado y OFF está abierto.

Tabla A-13 Tabla de interruptores de configuración del EM 235 para seleccionar el margen de las entradas analógicas y la resolución

Unipolar						Margen de tensión	Resolución
Interruptor 1	Interruptor 2	Interruptor 3	Interruptor 4	Interruptor 5	Interruptor 6		
ON	OFF	OFF	ON	OFF	ON	0 a 50 mV	12,5 µV
OFF	ON	OFF	ON	OFF	ON	0 a 100 mV	25 µV
ON	OFF	OFF	OFF	ON	ON	0 a 500 mV	125 µV
OFF	ON	OFF	OFF	ON	ON	0 a 1 V	250 µV
ON	OFF	OFF	OFF	OFF	ON	0 a 5 V	1,25 mV
ON	OFF	OFF	OFF	OFF	ON	0 a 20 mA	5 µA
OFF	ON	OFF	OFF	OFF	ON	0 a 10 V	2,5 mV
Bipolar						Margen de tensión	Resolución
Interruptor 1	Interruptor 2	Interruptor 3	Interruptor 4	Interruptor 5	Interruptor 6		
ON	OFF	OFF	ON	OFF	OFF	±25 mV	12,5 µV
OFF	ON	OFF	ON	OFF	OFF	±50 mV	25 µV
OFF	OFF	ON	ON	OFF	OFF	±100 mV	50 µV
ON	OFF	OFF	OFF	ON	OFF	±250 mV	125 µV
OFF	ON	OFF	OFF	ON	OFF	±500 mV	250 µV
OFF	OFF	ON	OFF	ON	OFF	±1 V	500 µV
ON	OFF	OFF	OFF	OFF	OFF	±2,5 V	1,25 mV
OFF	ON	OFF	OFF	OFF	OFF	±5 V	2,5 mV
OFF	OFF	ON	OFF	OFF	OFF	±10 V	5 mV

Tabla A-14 Tabla de interruptores de configuración del EM 235 para seleccionar el formato unipolar/bipolar, la ganancia y la atenuación

Interruptores de configuración del módulo de ampliación EM 235						Unipolar/ bipolar	Ganancia	Atenuación
Interruptor 1	Interruptor 2	Interruptor 3	Interruptor 4	Interruptor 5	Interruptor 6			
					ON	Unipolar		
					OFF	Bipolar		
			OFF	OFF			x1	
			OFF	ON			x10	
			ON	OFF			x100	
			ON	ON			no válido	
ON	OFF	OFF						0,8
OFF	ON	OFF						0.4
OFF	OFF	ON						0.2

Formato de la palabra de datos de entrada de los módulos de ampliación EM 231 y EM 235

La figura A-21 muestra la disposición del valor de 12 bits dentro de la palabra de entrada analógica de la CPU.

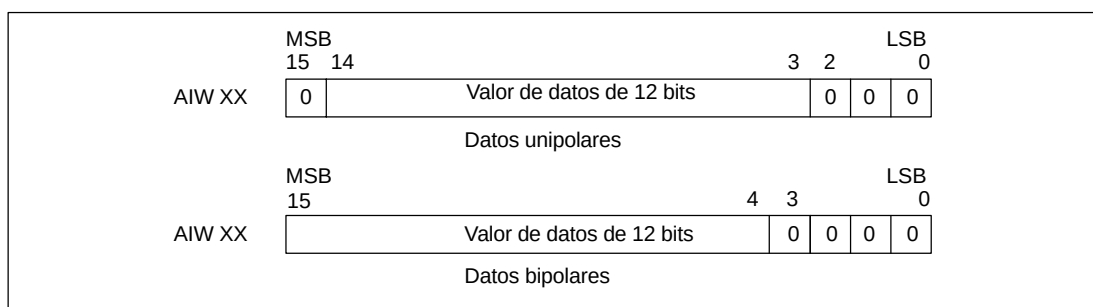


Figura A-21 Formato de la palabra de datos de entrada de los módulos de ampliación EM 231 y EM 235

Nota

Los 12 bits del valor de conversión analógica/digital (ADC) se justifican a la izquierda en el formato de palabra de datos. El MSB (bit más significativo) indica el signo, en tanto que cero indica un valor positivo de la palabra de datos. En formato unipolar, los tres ceros a la derecha modifican el valor de la palabra de datos en incrementos de 8 por cada cambio del valor ADC. En formato bipolar, los cuatro ceros a la derecha modifican el valor de la palabra de datos en incrementos de 16 por cada cambio del valor ADC.

Esquemas de conexiones de las entradas de los módulos de ampliación EM 231 y EM 235

La figura A-22 muestra los esquemas de conexiones de las entradas de los módulos de ampliación EM 231 y EM 235.

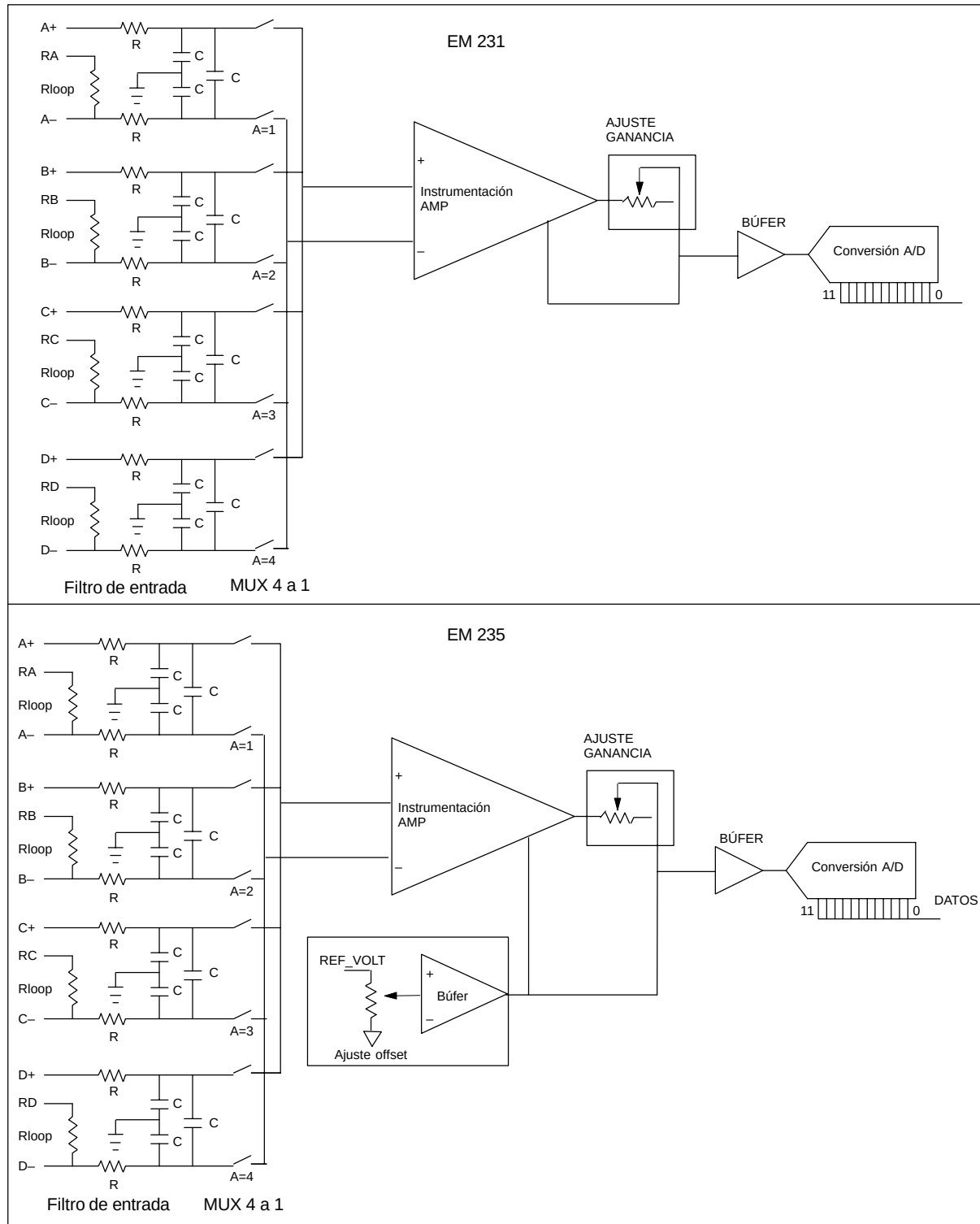


Figura A-22 Esquemas de conexiones de las entradas de los módulos de ampliación EM 231 y EM 235

Formato de la palabra de datos de salida de los módulos de ampliación EM 232 y EM 235

La figura A-23 muestra la disposición del valor de 12 bits dentro de la palabra de salida analógica de la CPU.

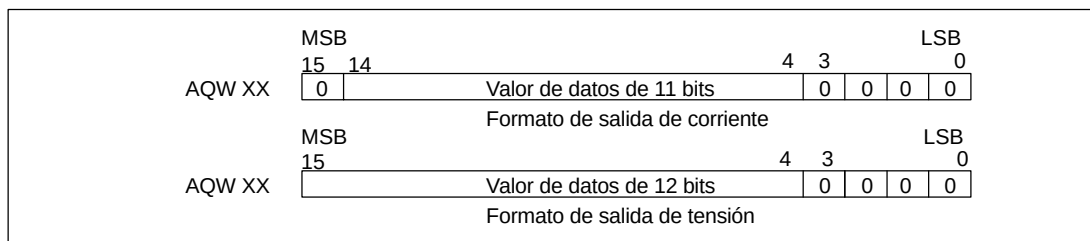


Figura A-23 Formato de la palabra de datos de salida de los módulos de ampliación EM 232 y EM 235

Nota

Los 12 bits del valor de conversión digital/analógica (DAC) se justifican a la izquierda en el formato de palabra de datos de salida. El MSB (bit más significativo) indica el signo, en tanto que cero indica un valor positivo de la palabra de datos. Los cuatro ceros a la derecha se truncan antes de cargarse en los registros DAC. Estos bits no tienen efecto alguno en el valor de señal de salida.

Esquemas de conexiones de las salidas de los módulos de ampliación EM 232 y EM 235

La figura A-24 muestra los esquemas de conexiones de las salidas de los módulos de ampliación EM 232 y EM 235.

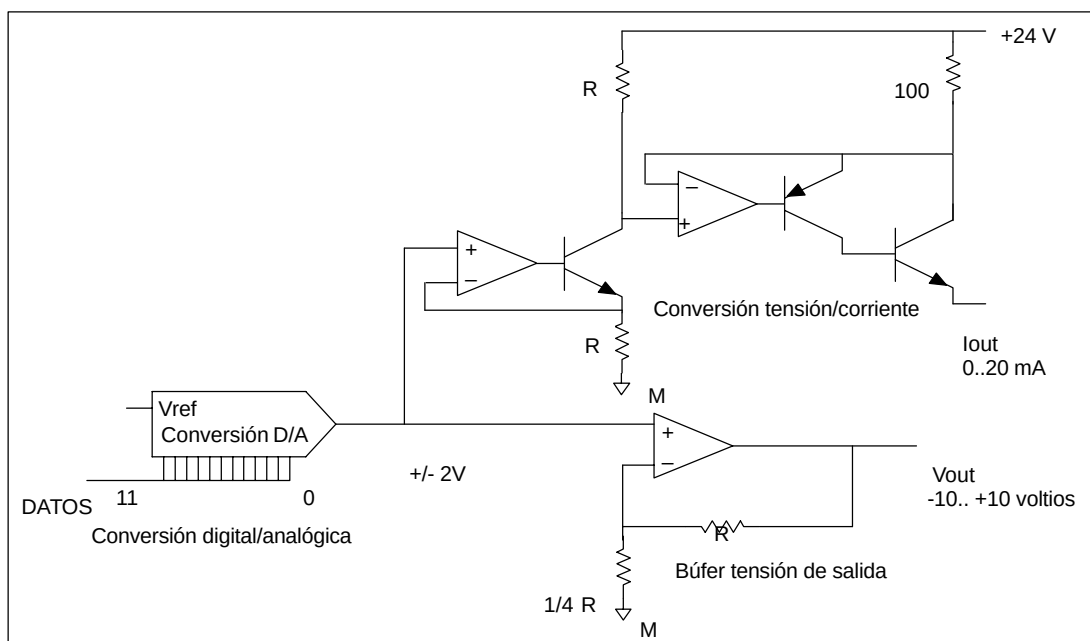


Figura A-24 Esquemas de conexiones de las salidas de los módulos de ampliación EM 232 y EM 235

Reglas de instalación

Tenga en cuenta las siguientes reglas para asegurar la precisión y la repetibilidad:

- Asegúrese de que la alimentación de sensores DC 24 V sea estable y esté exenta de interferencias.
- Utilice cables lo más cortos posible para la alimentación de sensores.
- Utilice cables dobles trenzados apantallados para el cableado de la alimentación de sensores.
- Conecte el apantallado sólo del lado de los sensores.
- Desvíe las entradas de los canales no utilizados, como muestra la figura A-19.
- Evite doblar excesivamente los cables.
- Conduzca los cables a través de canales.
- Evite colocar los cables de señales en paralelo con cables de alta tensión. Si los cables se deben cruzar, hágalo en ángulo recto.
- Verifique que las señales de entrada se encuentren dentro de los límites de tensión en modo común, aislando dichas señales o referenciándolas al hilo común externo de 24V del módulo analógico.

Nota

No es recomendable utilizar termopares junto con los módulos de ampliación EM 231 y EM 235.

Descripción y uso del módulo de entradas analógicas: precisión y repetibilidad

EM 231 y EM 235 son módulos de ampliación de entradas analógicas de 12 bits, rápidos y de bajo costo. Los módulos pueden convertir una entrada analógica a su correspondiente valor digital en 149 μ s. La conversión de la señal analógica se efectúa cada vez que el programa de usuario accede a la entrada analógica. Los tiempos mencionados se deben agregar al tiempo de ejecución básico de la operación utilizada para acceder a la entrada analógica.

Los módulos EM 231 y EM 235 proporcionan un valor digital no procesado (sin linealización ni filtraje) que corresponde a la tensión o a la corriente analógicas en los terminales de entrada del módulo. Puesto que se trata de módulos rápidos, la señal de entrada analógica puede cambiar rápidamente (incluyendo interferencias internas y externas). Las diferencias de un muestreo a otro, causadas por interferencias de una señal de entrada analógica que cambie constante o lentamente, se pueden reducir creando un promedio de una serie de muestreos. Cuanto mayor sea la cantidad de muestreos utilizados para calcular el promedio, tanto más lento será el tiempo de respuesta a cambios en la señal de entrada.

Los datos relativos a la repetibilidad describen las diferencias de un muestreo a otro en el caso de las señales de entrada que no cambien. Dichos datos definen el margen que contiene un 99% de todos los muestreos. La precisión media describe el valor promedio del error (la diferencia entre el valor promedio de los muestreos individuales y el valor exacto de la señal real de la entrada analógica). La repetibilidad se describe en la curva representada en la figura A-25. Esta figura muestra el margen de repetibilidad (que contiene un 99% de los muestreos), el valor promedio de los muestreos individuales y la precisión media. En la tabla A-15 se indican los datos relativos a la repetibilidad y la precisión media con respecto a los márgenes configurables.

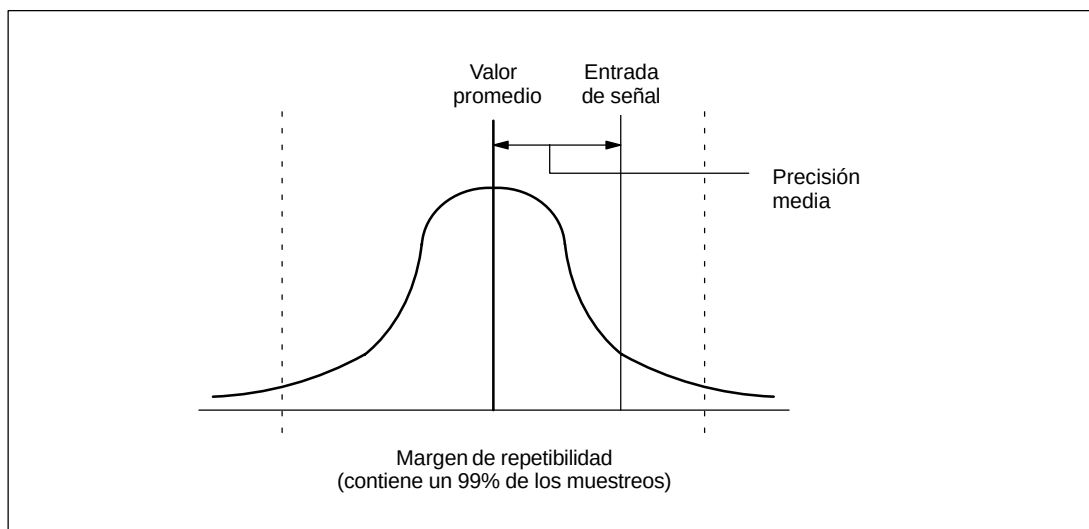


Figura A-25 Definición de la precisión

Tabla A-15 Datos de los módulos de ampliación EM 231 y EM 235

Margen máx.	Repetibilidad ¹		Precisión media ^{1,2,3,4}	
	% del margen máx.	Contajes	% del margen máx.	Contajes
Datos del módulo EM 231				
0 a 5 V	± 0,075%	± 24	± 0,01%	± 32
0 a 20 mA				
0 a 10 V		± 48	± 0,05%	
± 2,5 V				
± 5 V				
Datos del módulo EM 235				
0 a 50 mV	± 0,075%	± 24	± 0,25%	± 80
0 a 100 mV			± 0,2%	± 64
0 a 500 mV			± 0,05%	± 16
0 a 1 V				
0 a 5 V				
0 a 20 mA				
0 a 10 V		± 0,075%	± 48	± 0,25%
± 25 mV	± 0,2%			± 128
± 50 mV	± 0,1%			± 64
± 100 mV	± 0,05%			± 32
± 250 mV				
± 500 mV				
± 1 V				
± 2,5 V				
± 5 V				
± 10 V				

¹ Mediciones realizadas después de haber calibrado el margen de entrada seleccionado.

² El error de desplazamiento en la señal próxima a cero de la entrada analógica no se corrige y no se considera en los datos relativos a la precisión.

³ Al transferir de canal a canal se presenta un error de conversión debido al tiempo de estabilización finito del multiplexor analógico. El error máximo de transferencia es de 0,1 % de la diferencia entre canales.

⁴ La precisión media incluye los efectos de la falta de linealidad y de la deriva de 0 a 55° C.

Definición de los datos analógicos

- Precisión: desviación del valor previsto en una E/S determinada.
- Resolución: efecto de un cambio de LSB reflejado en la salida.

Homologaciones

Estos módulos cumplen las normas y prescripciones de las organizaciones que se nombran a continuación: UL 508 homologado (Industrial Control Equipment); CSA C22.2 número 142 certificado (Process Control Equipment); FM clase I, categoría 2, grupos A, B, C y D ubicaciones peligrosas, T4A; VDE 0160: equipos electrónicos utilizables en instalaciones eléctricas; Comunidad Europea (CE), Directiva de Baja Tensión 73/23/EEC, EN 61131-2: Autómatas programables – requisitos de los equipos; Comunidad Europea (CE) Directiva EMC 89/336/EEC.

A.12 Datos técnicos del módulo EM 277 PROFIBUS-DP

Tabla A-16 Datos técnicos del módulo EM 277 PROFIBUS-DP

Descripción Nº de referencia	EM 277 PROFIBUS-DP 6ES7 277-0AA20-0XA0
Tamaño físico	
Dimensiones (l x a x p)	71 mm x 80 mm x 62 mm
Peso	175 g
Pérdida de corriente (disipación)	2,5 W
Comunicación	
Nº de puertos	1 puerto
Puerto eléctrico	RS-485
Aislamiento (señal externa a circuito lógico)	AC 500 V (galvánica)
Velocidades de transferencia PROFIBUS-DP/MPI (ajustadas automáticamente)	9,6, 19,2, 45,45, 93,75, 187,5 y 500 kbit/s; 1, 1,5, 3, 6 y 12 Mbit/s
Protocolos	Esclavo PROFIBUS-DP y esclavo MPI
Longitud del cable	
Hasta 93,75 kbit/s	1200 m
187,5 kbit/s	1000 m
500 kbit/s	400 m
1 a 1,5 Mbit/s	200 m
3 a 12 Mbit/s	100 m
Capacidad de red	
Ajustes de la dirección de estación	0 – 99 (ajustado con interruptores rotativos)
Nº máximo de estaciones por segmento	32
Nº máximo de estaciones por red	126, hasta 99 estaciones EM 277
Enlaces MPI	6 en total, 2 reservados (1 para una PG y 1 para un OP)
Consumo de corriente	
+DC 5 V (del bus de ampliación)	150 mA
Corriente de entrada de DC 24 V necesaria	
Margen de tensión	DC 20,4 a 28,8 V (clase 2 o alimentación de sensores de la CPU)
Corriente máxima	
Módulo sólo con el puerto activo	30 mA
Agregar 90 mA a 5V de carga del puerto	60 mA
Agregar 120 mA a 24V de carga del puerto	180 mA
Rizado/corriente parásita (<10 MHz)	máx. 1 V de pico a pico
¹ Aislado (corriente de entrada a circuito lógico del módulo)	AC 500 V, 1 minuto
Corriente DC 5 V en el puerto de comunicación	
Corriente máx. por puerto	90 mA
Aislamiento del transformador del circuito lógico del módulo y de la corriente de entrada DC 24 V	AC 500 V, 1 minuto
Corriente DC 24 V en el puerto de comunicación	
Margen de tensión	DC 20,4 a 28,8 V
Corriente máx. por puerto	120 mA
Corriente límite	0,7 a 2,4 A
Aislamiento	No aislado, igual circuito que la entrada DC 24 V

¹ La fuente de alimentación de DC 24 V no le suministra corriente a la lógica del módulo. La fuente de alimentación de DC 24 VDC le suministra corriente al puerto de comunicación.

Compatibilidad

El módulo esclavo EM 277 PROFIBUS-DP es un módulo de ampliación inteligente diseñado para su utilización junto con las CPUs S7-200 que muestra la tabla A-17.

Tabla A-17 Compatibilidad del módulo EM 277 PROFIBUS-DP con las CPUs S7-200

CPU	Descripción	Nº de referencia
CPU 222, versión 1.10 o superior	CPU 222 DC/DC/DC	6ES7 212-1AB21-0XB0
	CPU 222 AC/DC/Relé	6ES7 212-1BB21-0XB0
CPU 224, versión 1.10 o superior	CPU 224 DC/DC/DC	6ES7 214-1AD21-0XB0
	CPU 224 AC/DC/Relé	6ES7 214-1BD21-0XB0
CPU 226, versión 1.00 o superior	CPU 226 DC/DC/DC	6ES7 216-2AD21-0XB0
	CPU 226 AC/DC/Relé	6ES7 216-2BD21-0XB0

Interruptores de direccionamiento y LEDs de estado

Los interruptores de direccionamiento y los LEDs de estado están ubicados en el lado frontal del módulo, como muestra la figura A-26. Los LEDs de estado del EM 277 figuran en la tabla A-20.

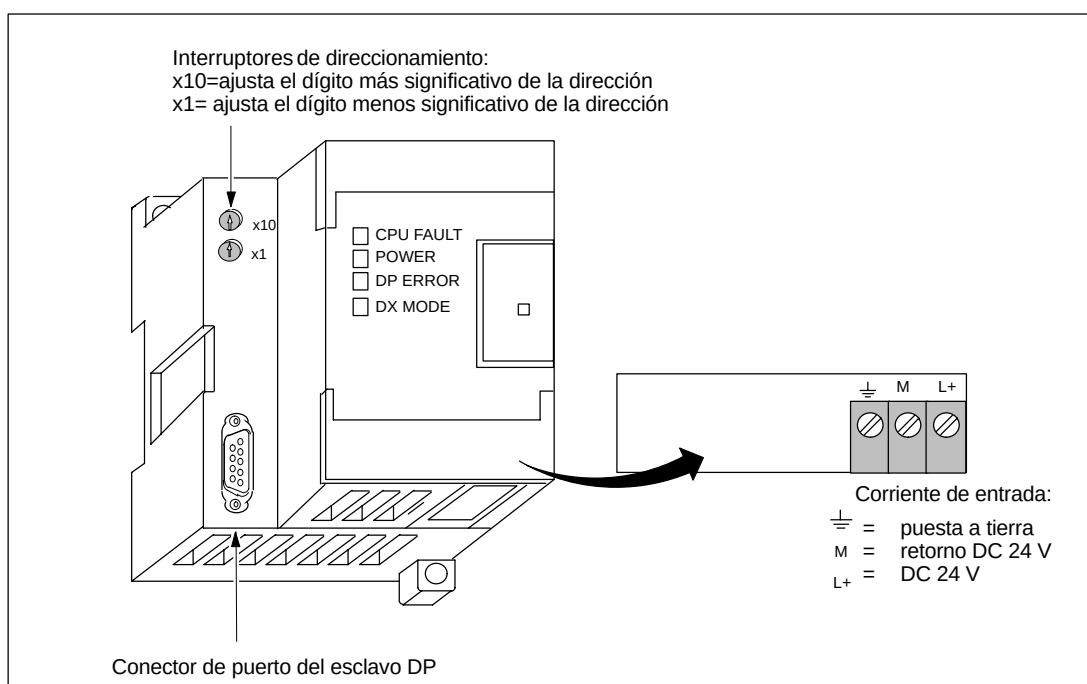


Figura A-26 Vista frontal del módulo EM 277 PROFIBUS-DP

Conector de puerto del esclavo DP

La figura A-27 muestra la asignación de pines del conector de puerto del esclavo DP.

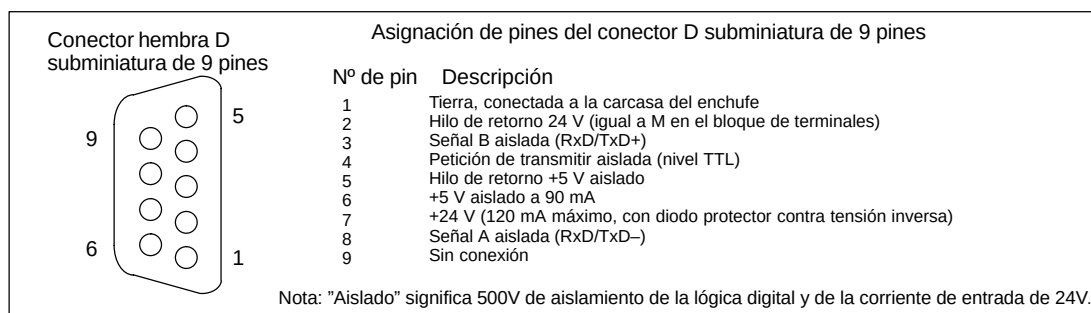


Figura A-27 Asignación de pines del conector de puerto del esclavo DP

Comunicación DP estándar

PROFIBUS-DP (o estándar DP) es un protocolo de telecomunicación definido en la norma europea EN 50170. Los dispositivos que cumplen con dicha norma son compatibles entre sí, aunque sean de diferentes fabricantes. DP es la abreviatura inglesa de Distributed Peripherals que significa periferia descentralizada (o periferia distribuida). PROFIBUS es la abreviatura de Process Field Bus.

El protocolo estándar DP está implementado en el módulo EM 277 PROFIBUS-DP como se define para las unidades esclavas en las siguientes normas relativas a los protocolos de comunicación:

- EN 50 170 (PROFIBUS) describe el acceso de bus y el protocolo de transferencia, indicando las propiedades del soporte de transferencia de datos.
- EN 50 170 (estándar DP) describe el intercambio de datos rápido y cíclico entre los maestros DP y los esclavos DP. En esta norma se definen también los procedimientos de configuración y parametrización, el intercambio de datos cíclico con las unidades periféricas descentralizadas y las funciones de diagnóstico asistidas.

La configuración de un maestro DP le permite reconocer las direcciones, los tipos de esclavos y las informaciones relativas a la parametrización que éstos necesitan. Al maestro se le indica también dónde depositar los datos que haya leído de los esclavos (entradas) y de dónde obtener los datos a escribir en los esclavos (salidas). El maestro DP establece la red e inicializa sus esclavos DP, escribiendo posteriormente los parámetros y la configuración de E/S en el esclavo. Luego lee las informaciones de diagnóstico del esclavo DP para verificar que éste haya aceptado los parámetros y la configuración de E/S. El maestro comienza entonces a intercambiar datos con el esclavo. En cada intercambio con el esclavo, escribe en las salidas y lee de las entradas. Dicho intercambio de datos continúa indefinidamente. Los esclavos pueden informar al maestro si se presenta una condición excepcional. Entonces, el maestro lee la información de diagnóstico del esclavo.

Una vez que un maestro DP haya escrito los parámetros y la configuración de E/S en un esclavo DP y éste los haya aceptado, el esclavo será propiedad del maestro. El esclavo sólo acepta peticiones de escritura de su respectivo maestro. Los demás maestros de la red pueden leer las entradas y salidas del esclavo, pero no escribir datos en él.

Utilizar el módulo de ampliación EM 277 para conectar una CPU S7-200 a la red en calidad de estación esclava DP

La CPU S7-200 se puede conectar a una red PROFIBUS-DP a través del módulo de ampliación EM 277 esclavo PROFIBUS-DP. El EM 277 se conecta a la CPU S7-200 a través del bus de E/S serie. La red PROFIBUS se conecta al módulo EM 277 PROFIBUS-DP por su puerto de comunicación DP. Éste último puede funcionar a una velocidad de transferencia cualquiera comprendida entre 9.600 bit/s y 12 Mbit/s (en la tabla A-16 se indican las velocidades de transferencia asistidas). En calidad de esclavo DP, el módulo EM 277 acepta varias configuraciones de E/S diferentes del maestro, pudiendo transferir diferentes cantidades de datos de y al maestro. Esta función permite adaptar a las exigencias de la aplicación la cantidad de datos que se deban transferir. A diferencia de numerosos dispositivos DP, el módulo EM 277 no se limita a transferir datos de E/S. El EM 277 transfiere datos de y a un bloque de la memoria de variables definida en la CPU S7-200. Gracias a ello puede intercambiar cualquier tipo de datos con él. Las entradas, los valores de los contadores y de los temporizadores, así como cualquier otro valor calculado se pueden enviar al maestro transfiriendo primero los datos a la memoria de variables de la CPU S7-200. De igual manera, los datos recibidos del maestro se almacenan en la memoria de variables de la CPU S7-200, pudiéndose transferir de allí a otras áreas de datos.

El puerto DP del módulo EM 277 PROFIBUS-DP se puede conectar a un maestro DP en la red, siendo posible comunicarse aún como esclavo MPI con otros maestros tales como unidades de programación (PGs) SIMATIC o CPUs S7-300/S7-400 en esa misma red.

La figura A-28 muestra una red PROFIBUS con una CPU 224 y un módulo EM 277 PROFIBUS-DP. En este ejemplo, la CPU 315-2 es el maestro DP que ha sido configurado por una unidad de programación SIMATIC con el software de programación STEP 7. La CPU 224 es un esclavo DP de la CPU 315-2. El módulo ET 200 es también un esclavo de la CPU 315-2. La CPU S7-400 se encuentra conectada a la red PROFIBUS y está leyendo datos de la CPU 224 mediante las operaciones XGET contenidas en el programa de usuario de la CPU S7-400.

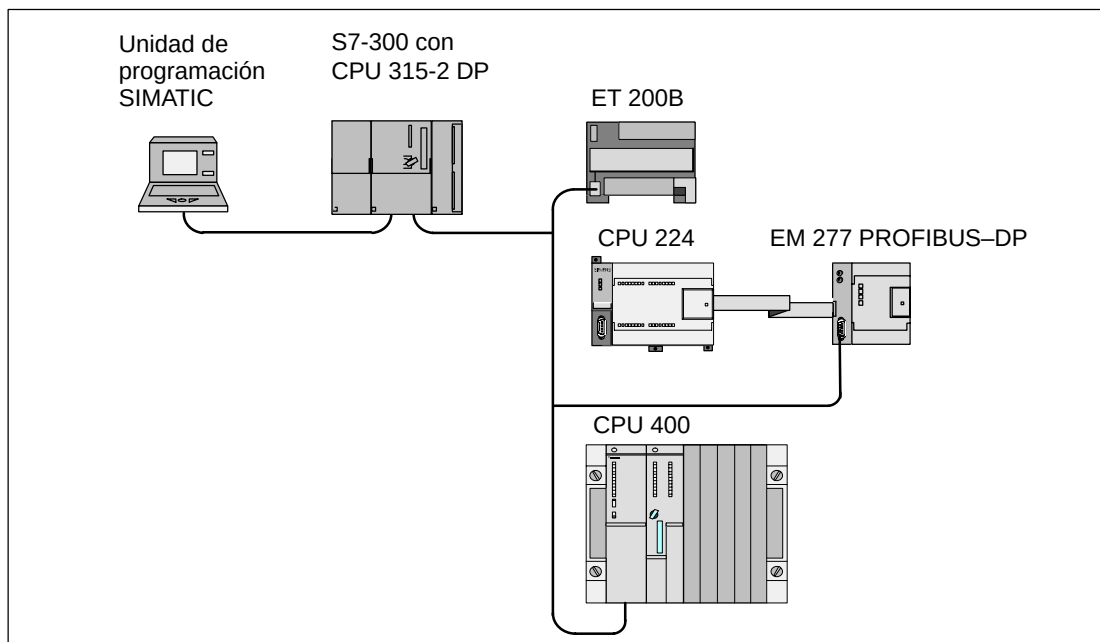


Figura A-28 Módulo EM 277 PROFIBUS-DP y CPU 224 en una red PROFIBUS

Configuración

Para utilizar el módulo EM 277 en calidad de esclavo DP es preciso ajustar la dirección de estación del puerto DP para que coincida con la dirección fijada en la configuración del maestro. La dirección de estación se ajusta con los interruptor, es rotativos del módulo EM 277. Tras haber efectuado un cambio con un interruptor es preciso desconectar la CPU y conectarla de nuevo para poder adoptar la nueva dirección del esclavo.

El maestro intercambia datos con cada uno de sus esclavos, enviando informaciones de su área de salidas al búfer de salida del esclavo en cuestión (o "buzón receptor"). El esclavo responde al mensaje del maestro retornando un búfer de entrada (o "buzón emisor") que el maestro almacena en un área de entradas (v. fig. A-29).

El maestro DP puede configurar el módulo EM 277 para que éste reciba datos de salida del maestro y retorne datos de entrada al mismo. Los búfers de salida y de entrada se almacenan en la memoria de variables (memoria V) de la CPU S7-200. Al configurar el maestro DP, se define la dirección de byte en la memoria V donde debe comenzar el búfer de salida como parte de la asignación de parámetros para el EM 277. Asimismo, se define la configuración de E/S como la cantidad de datos de salida a escribir en la CPU S7-200 y de datos de entrada a leer de la misma. El módulo EM 277 determina el tamaño de los búfers de entrada y de salida conforme a la configuración de E/S. El maestro DP escribe la asignación de parámetros y la configuración de E/S en el módulo EM 277 PROFIBUS DP. El EM 277 transfiere a la CPU la dirección de la memoria V, así como las longitudes de los datos de entrada y salida.

La figura A-29 muestra un ejemplo de la memoria V en una CPU 224, así como las áreas de direcciones de E/S de una CPU que actúa de maestro DP. En el ejemplo, el maestro DP ha definido una configuración de E/S compuesta por 16 bytes de salida y 16 bytes de entrada, así como un offset (desplazamiento) de 5000 bytes en la memoria V. La longitud de los búfers de salida y de entrada en la CPU 224, determinada conforme a la configuración de E/S, es de 16 bytes en ambos casos. El búfer de salida comienza en V5000, siguiéndole inmediatamente el búfer de entrada que comienza en V5016. Los datos de salida (del maestro) se depositan en la dirección V5000 de la memoria V. Los datos de entrada (al maestro) provienen de la dirección V5016 en la memoria V.

Nota

Al utilizar una unidad de datos (coherentes) de tres bytes o unidades de datos (coherentes) superiores a cuatro bytes, será preciso usar SFC14 para leer las entradas del esclavo DP y SFC15 para direccionar las salidas del mismo. Para obtener informaciones más detalladas al respecto, consulte el manual *Software de sistema para SIMATIC S7-300/400 – Funciones estándar y funciones de sistema*.

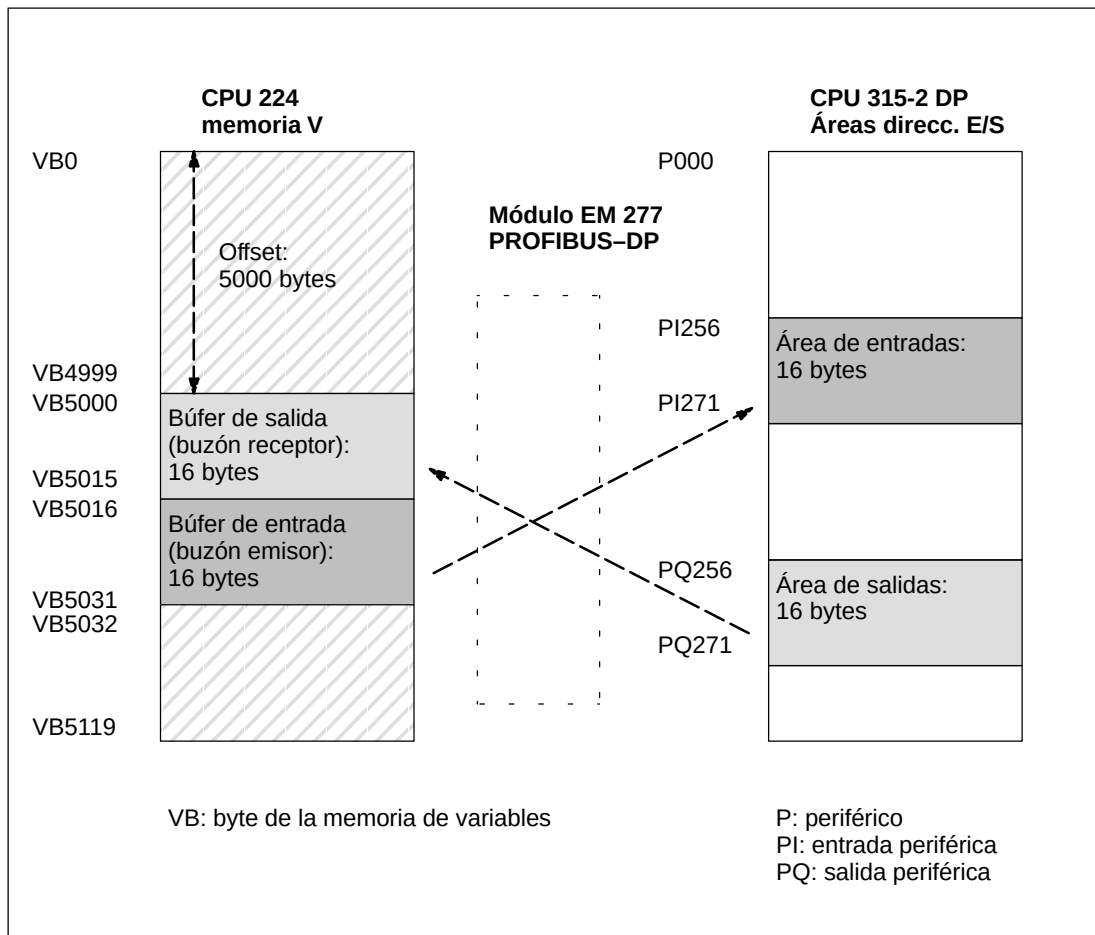


Figura A-29 Ejemplo: Memoria V de la CPU 224 y área de direcciones de E/S de un maestro PROFIBUS-DP

La tabla A-18 indica las configuraciones soportadas por el módulo EM 277 PROFIBUS-DP. La configuración estándar del módulo EM 277 es: dos palabras de entrada y dos palabras de salida.

Tabla A-18 Opciones de configuración del módulo EM 277

Configuración	Entradas al maestro	Salidas del maestro	Coherencia de datos
1	1 palabra	1 palabra	Coherencia de palabras
2	2 palabras	2 palabras	
3	4 palabras	4 palabras	
4	8 palabras	8 palabras	
5	16 palabras	16 palabras	
6	32 palabras	32 palabras	
7	8 palabras	2 palabras	
8	16 palabras	4 palabras	
9	32 palabras	8 palabras	
10	2 palabras	8 palabras	
11	4 palabras	16 palabras	
12	8 palabras	32 palabras	
13	2 bytes	2 bytes	Coherencia de bytes
14	8 bytes	8 bytes	
15	32 bytes	32 bytes	
16	64 bytes	64 bytes	
17	4 bytes	4 bytes	Coherencia de búfers
18	8 bytes	8 bytes	
19	12 bytes	12 bytes	
20	16 bytes	16 bytes	

La dirección de los búfers de entrada y de salida se puede configurar de manera que se almacenen en cualquier dirección de la memoria de variables de la CPU S7-200, siendo VB0 el ajuste estándar. La dirección de los búfers de entrada y salida forma parte de la parametrización que el maestro escribe en la CPU S7-200. El maestro se debe configurar para que reconozca a sus esclavos y para que escriba los parámetros necesarios y la configuración de E/S en cada uno de ellos.

Utilice las siguientes aplicaciones para configurar el maestro DP:

- Para los maestros SIMATIC S5, utilice el software de Windows COM PROFIBUS.
- Para los maestros SIMATIC S7, utilice el software de programación STEP 7.
- Para los maestros SIMATIC 505, utilice COM PROFIBUS y TISOFT2 o SoftShop.

Para obtener informaciones más detalladas acerca del software de configuración y programación, consulte los manuales correspondientes a los dispositivos en cuestión. Para obtener informaciones más detalladas acerca de las redes PROFIBUS y sus componentes, consulte el manual *Sistemas de periferia descentralizada ET 200*.

Coherencia de datos

PROFIBUS asiste tres tipos de coherencia de datos:

- La coherencia de bytes garantiza que éstos se transfieran en unidades enteras.
- La coherencia de palabras garantiza que otros procesos de la CPU no puedan interrumpir las transferencias de palabras. Por tanto, los dos bytes que componen una palabra se transferirán siempre juntos, sin que se puedan separar.
- La coherencia de búfers garantiza que el búfer de datos se transfiera en forma de una unidad, sin que ningún otro proceso de la CPU pueda interrumpir la transferencia.

La coherencia de palabras y de búfers obliga a la CPU a detener otros procesos (tales como las interrupciones incluidas en el programa de usuario) mientras los datos de la periferia descentralizada se transfieren o se manipulan en la CPU. La coherencia de palabras se deberá utilizar si los valores de datos a transferir son números enteros. La coherencia de búfers se deberá utilizar si dichos valores son palabras dobles o números reales (en coma flotante). También es recomendable utilizar la coherencia de búfers cuando un grupo de valores se refiere a un cálculo u objeto.

La coherencia de datos se ajusta como parte de la configuración de E/S en el maestro. La coherencia seleccionada se escribe en el esclavo DP como parte de la inicialización del mismo. Tanto el maestro DP como el esclavo DP utilizan la coherencia seleccionada para garantizar que los valores de datos (bytes, palabras o búfers) se transfieran sin interrupciones entre el maestro y el esclavo. La figura A-30 muestra los diversos tipos de coherencias.

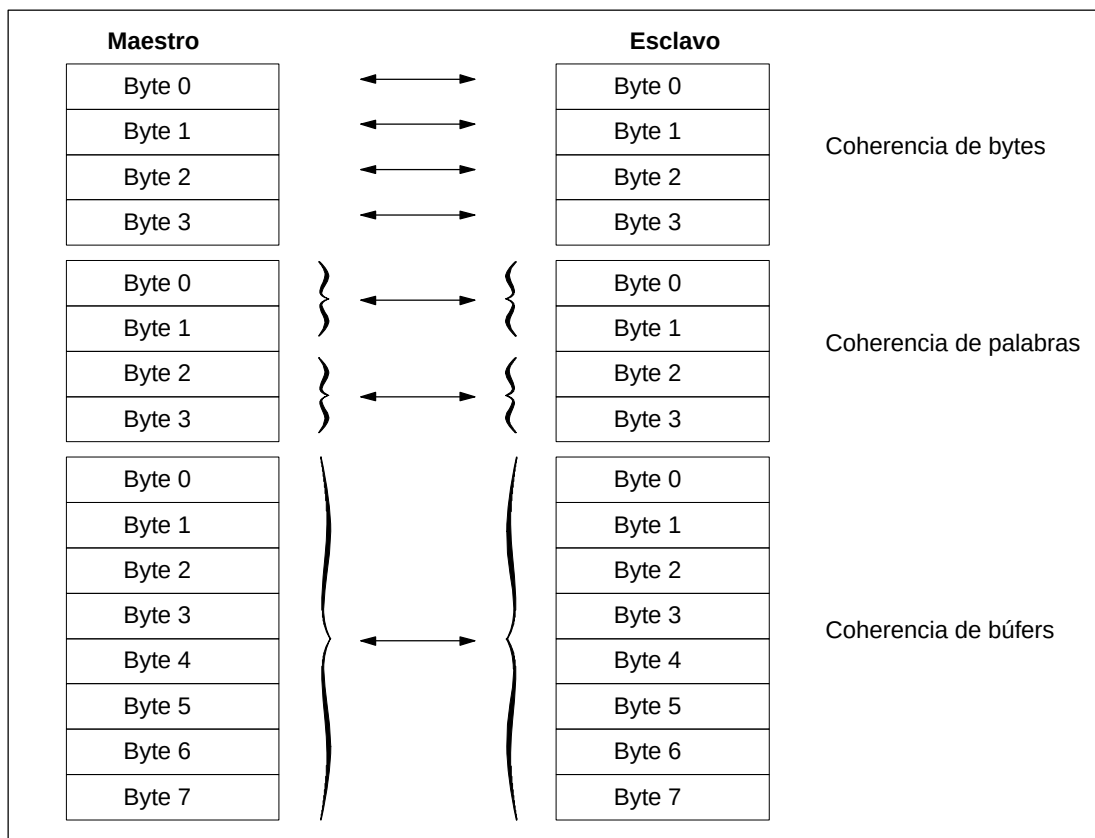


Figura A-30 Coherencia de bytes, palabras y búfers

Consideraciones relativas al programa de usuario

Una vez que un maestro DP haya configurado correctamente el módulo EM 277 PROFIBUS-DP, ambos equipos pasan al modo de intercambio de datos. En modo de intercambio de datos, el maestro escribe los datos de salida en el módulo EM 277 PROFIBUS-DP y éste último responde con los datos de entrada de la CPU S7-200 más recientes. El módulo EM 277 actualiza continuamente sus entradas de la CPU S7-200 para facilitarle los datos de entrada más recientes al maestro DP. El módulo transmite entonces los datos de salida a la CPU S7-200. Los datos de salida del maestro se depositan en la memoria V (en el búfer de salida) que comienza en la dirección indicada por el maestro DP durante la inicialización. Los datos de entrada se leen de las direcciones de la memoria V (el búfer de entrada) que le siguen inmediatamente a los datos de salida.

La dirección inicial de los búfers de datos en la memoria V y sus longitudes se deben conocer cuando se cree el programa de usuario para la CPU S7-200. El programa de usuario contenido en dicha CPU debe transferir los datos de salida que envía el maestro desde el búfer de salida a las áreas de datos donde se utilizarán. Igualmente, los datos de entrada dirigidos al maestro se deben transferir de diversas áreas de datos al búfer de entrada para poder enviarlos de allí al maestro.

Los datos de salida que envía el maestro DP se depositan en la memoria V inmediatamente después de haberse ejecutado la correspondiente parte del ciclo del programa de usuario. Los datos de entrada (dirigidos al maestro) se copian de la memoria V en el EM 277 para transmitirlos simultáneamente al maestro. Los datos de salida sólo se escriben en la memoria V cuando el maestro suministra nuevos datos. Los datos de entrada se transmiten al maestro en el siguiente intercambio de datos con él.

Las marcas especiales SMB200 a SMB249 proporcionan informaciones de estado sobre el módulo EM 277 esclavo PROFIBUS-DP si se trata del primer módulo inteligente de la cadena de módulos de ampliación. Si el EM 277 es el segundo módulo de la cadena, el estado del EM 277 se obtiene de las marcas especiales SMB250 a SMB299. Estas marcas tienen valores estándar si no se ha establecido la comunicación DP con un maestro. Una vez que un maestro haya escrito los parámetros y la configuración de E/S en el módulo EM 277 PROFIBUS-DP, dichas marcas adoptan la configuración ajustada por el maestro DP. Antes de utilizar la información contenida en las marcas SMB225 a SMB229 (v. tabla A-19) o los datos del búfer de la memoria V, compruebe la marca SMB224 para verificar que el EM 277 se encuentre actualmente en modo de intercambio de datos con el maestro.

Nota

Los tamaños o las direcciones de los búfers de E/S del módulo EM 277 PROFIBUS-DP no se pueden configurar escribiendo en las marcas especiales SMB225 a SMB229 ó SMB275 a SMB279. Sólo el maestro DP puede configurar el módulo EM 277 PROFIBUS-DP para que pueda trabajar en modo DP.

Tabla A-19 Bytes de marcas SMB200 a SMB299

DP es el primer módulo inteligente	DP es el segundo módulo inteligente	Descripción
SMB200 a SMB215:	SMB250 a SMB265	Nombre del módulo (16 caracteres ASCII) “EM 277 ProfibusDP”
SMB216 a SMB219	SMB266 a SMB269	Número de versión del software (4 caracteres ASCII) xxxx
SMW220	SMW270	Código de error 16#0000 Sin error 16#0001 Sin corriente de usuario 16#0002 a 16#FFFF Reservados
SMB222	SMB272	La dirección del módulo DP esclavo se ajusta con los interruptores de direccionamiento (0 – 99 decimal)
SMB223	SMB273	Reservados
SMB224	SMB274	Byte de estado del protocolo estándar DP MSB LSB 000000S1S0 S1 S0 Descripción del byte de estado del estándar DP 0 0 Comunicación DP no iniciada desde el arranque 0 1 Error de configuración/parametrización detectado 1 0 Modo de intercambio de datos activado 1 1 Modo de intercambio de datos desactivado
SMB225	SMB275	Protocolo estándar DP – dirección del maestro del esclavo (0 a 126)
SMW226	SMW276	Protocolo estándar DP – dirección en la memoria V del búfer de salida (desplazamiento de VB0)
SMB228	SMB278	Protocolo estándar DP – número de bytes de los datos de salida
SMB229	SMB279	Protocolo estándar DP – número de bytes de los datos de entrada
SMB230 a SMB249	SMB280 a SMB299	Reservados – se borran durante el arranque

Nota: SMB225 a SMB229 y SMB275 a SMB279 se actualizan cada vez que el módulo esclavo DP acepta datos de configuración y/o parametrización. Estas direcciones se actualizan aun si se detecta un error de configuración y/o parametrización. Las direcciones se borran durante cada arranque.

LEDs de estado del módulo de ampliación EM 277 PROFIBUS-DP

El módulo EM 277 PROFIBUS-DP dispone de cuatro LEDs en el panel frontal que indican el estado del puerto DP:

- Después del arranque de la CPU S7-200, el LED DX MODE permanecerá apagado mientras que no se intente establecer la comunicación DP.
- Una vez iniciada correctamente la comunicación DP (el módulo EM 277 PROFIBUS-DP pasa a modo de intercambio de datos con el maestro), el LED DX MODE se encenderá en verde y permanecerá encendido hasta que se desactive el modo de intercambio de datos.
- Si se interrumpe la comunicación DP, lo que obliga al módulo EM 277 a salir del modo de intercambio de datos, el LED DX MODE se apagará (OFF) y el LED DP ERROR se encenderá en rojo. Dicho estado persistirá hasta que la CPU S7-200 se desconecte o hasta que se restablezca el intercambio de datos.
- Si se detecta un error en la configuración de E/S o en la parametrización que el maestro DP está escribiendo en el módulo EM 227, el LED DP ERROR parpadea en rojo.
- Si no se suministra corriente DC 24 V, el LED POWER permanece apagado (OFF).

La tabla A-20 resume los diversos estados de los LEDs del módulo EM 277.

Tabla A-20 Indicadores de estado (LEDs) del módulo EM 277 PROFIBUS-DP

LED	OFF	Rojo	Parpadeante (rojo)	Verde
CPU FAULT	Módulo OK	Fallo interno del módulo	—	—
POWER	Sin alimentación DC 24 V	—	—	Alimentación DC 24 V OK
DP ERROR	No se ha presentado ningún error.	Modo de intercambio de datos desactivado	Error de configuración/parametrización	—
DX MODE	Modo de intercambio de datos desactivado	—	—	Modo de intercambio de datos activado

Nota: Si el módulo EM 277 PROFIBUS-DP actúa exclusivamente de esclavo MPI, sólo estará activado (verde).

Funciones de configuración adicionales

El módulo EM 277 PROFIBUS-DP puede servir de interface de comunicación con otros maestros MPI, así esté actuando o no de esclavo PROFIBUS-DP. El módulo permite conectar una CPU S7-300/400 a una CPU S7-200 utilizando las operaciones XGET/XPUT de la S7-300/400. STEP 7-Micro/WIN y un procesador de comunicaciones (p.ej. CP5611) con el juego de parámetros MPI ó PROFIBUS, o bien un panel de operador (OP) o el TD 200 (versión 2.0 o superior, nº de referencia 6ES7 272-0AA20-0YA0) se pueden utilizar para la comunicación con la CPU S7-200 a través del módulo EM 277 PROFIBUS-DP.

Además del maestro DP, es posible conectar seis equipos (como máximo) al módulo EM 277 PROFIBUS-DP. Una conexión se reserva para una unidad de programación (PG) y otra, para un panel de operador (OP). Las cuatro conexiones restantes pueden ser utilizadas por cualquier maestro MPI. Para que el módulo EM 277 PROFIBUS-DP se pueda comunicar con varios maestros, todos los maestros deberán utilizar una misma velocidad de transferencia. La figura A-31 muestra un ejemplo de una configuración de red.

Si el módulo EM 277 PROFIBUS-DP se utiliza para la comunicación MPI, el maestro MPI debe utilizar la dirección de estación del módulo para todos los mensajes que se envíen a la CPU S7-200 conectada al módulo. Los mensajes MPI que se envíen al módulo EM 277 PROFIBUS-DP se transmiten a la CPU S7-200.

El EM 277 PROFIBUS-DP es un módulo esclavo, por lo que no se puede utilizar para la comunicación entre CPUs S7-200 utilizando las operaciones NETR y NETW. El módulo EM 277 PROFIBUS-DP no se puede usar para la comunicación en modo Freeport, que es una de las funciones de los puertos de comunicación de las CPUs S7-200.

La figura A-31 muestra un ejemplo de una red PROFIBUS-DP/MPI.

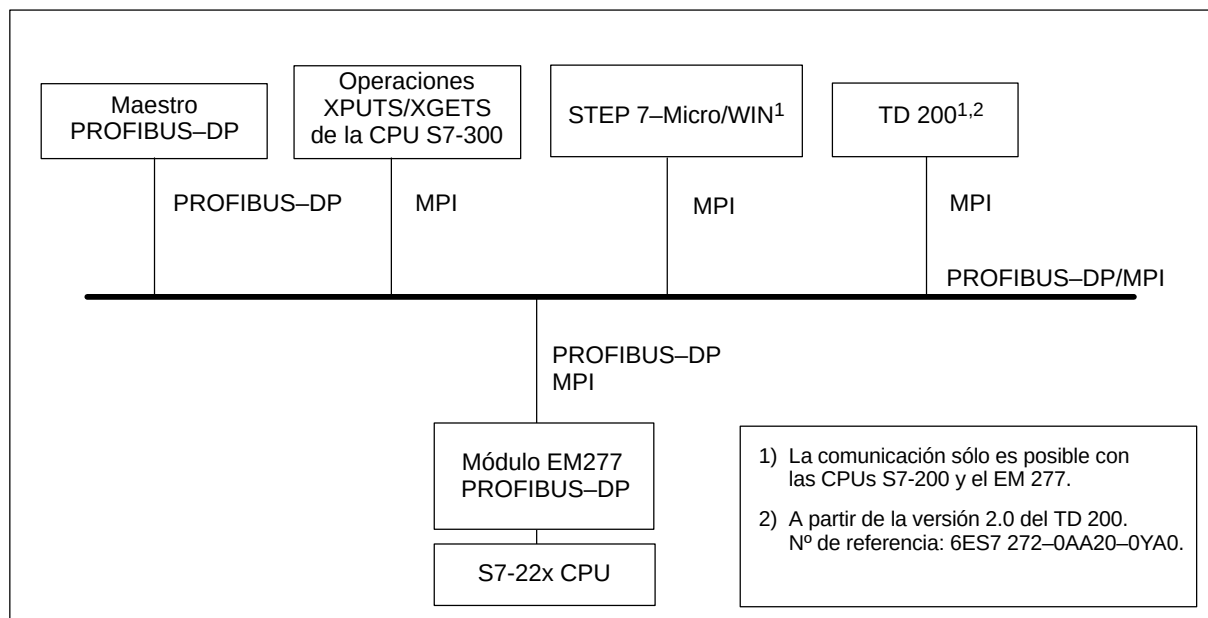


Figura A-31 Red PROFIBUS-DP/MPI

Archivo de datos maestros de los dispositivos GSD

Los diversos dispositivos PROFIBUS tienen diferentes propiedades de rendimiento. Estas propiedades difieren con respecto a la funcionalidad (p.ej. el número de señales de E/S y de mensajes de diagnóstico) o a los parámetros de bus, tales como la velocidad de transferencia y el tiempo de vigilancia. Los parámetros varían de un dispositivo a otro y de un fabricante a otro, documentándose por lo general en un manual técnico. Para facilitar la configuración de las redes PROFIBUS, las propiedades de rendimiento de los diversos dispositivos se indican en un archivo de datos maestros (archivo GSD). Las herramientas de configuración basadas en los archivos GSD permiten integrar fácilmente los dispositivos de diferentes fabricantes en una misma red.

Los archivos GSD ofrecen una descripción detallada de las propiedades de un dispositivo en un formato definido exactamente. Estos archivos son preparados para cada tipo de dispositivo por el correspondiente fabricante, poniéndolos a disposición del usuario de equipos PROFIBUS. El archivo GSD permite que el sistema de configuración lea las propiedades de un dispositivo PROFIBUS y utilice dichas informaciones al configurar la red.

Las versiones más recientes de COM PROFIBUS o del software STEP 7 incluyen archivos de configuración para el módulo EM 277 PROFIBUS-DP. Si su versión del software no incluye un archivo de configuración para el EM 277, puede descargar el archivo GSD más reciente (SIEM089D.GSD) visitando el sitio web: www.profibus.com.

Si utiliza un maestro no SIMATIC, consulte la documentación suministrada por el fabricante en cuestión para configurar el maestro utilizando el archivo GSD.

Listado del archivo GSD para el módulo EM 277 PROFIBUS-DP

```

;=====
; GSD File for the EM 277 PROFIBUS-DP with a DPC31
; MLFB   : 6ES7 277-0AA20-0XA0
; DATE   : 07-Oct-1999
;=====
#Profibus_DP
;General parameters
GSD_Revision      = 1
Vendor_Name       = "Siemens"
Model_Name        = "EM 277 PROFIBUS-DP"
Revision          = "V1.00"
Ident_Number      = 0x089D
Protocol_Ident    = 0
Station_Type      = 0
FMS_supp          = 0
Hardware_Release  = "1.00"
Software_Release  = "1.00"
9.6_supp          = 1
19.2_supp         = 1
45.45_supp        = 1
93.75_supp        = 1
187.5_supp        = 1
500_supp          = 1
1.5M_supp         = 1
3M_supp           = 1
6M_supp           = 1
12M_supp          = 1
MaxTsdr_9.6       = 60
MaxTsdr_19.2      = 60
MaxTsdr_45.45     = 250
MaxTsdr_93.75     = 60

```

```

MaxTsdr_187.5      = 60
MaxTsdr_500        = 100
MaxTsdr_1.5M       = 150
MaxTsdr_3M         = 250
MaxTsdr_6M         = 450
MaxTsdr_12M        = 800
Redundancy          = 0
Repeater_Ctrl_Sig  = 2
24V_Pins            = 2

; Slave-Specification:
OrderNumber="6ES7 277-0AA20-0XA0"
Periphery="SIMATIC S5"
Slave_Family=10@Tdf@SIMATIC

Freeze_Mode_supp    = 1
Sync_Mode_supp      = 1
Set_Slave_Add_Supp  = 0
Auto_Baud_supp      = 1
Min_Slave_Intervall = 1
Fail_Safe           = 0
Max_Diag_Data_Len   = 6
Modul_Offset        = 0
Modular_Station     = 1
Max_Module          = 1
Max_Input_len       = 128
Max_Output_len      = 128
Max_Data_len        = 256

; UserPrmData-Definition
ExtUserPrmData=1 "I/O Offset in the V-memory"
Unsigned16 0 0-5119
EndExtUserPrmData
; UserPrmData: Length and Preset:
User_Prm_Data_Len=3
User_Prm_Data= 0,0,0
Max_User_Prm_Data_Len=3
Ext_User_Prm_Data_Const(0)=0x00,0x00,0x00
Ext_User_Prm_Data_Ref(1)=1

; Module Definition List
Module = "2 Bytes Out/ 2 Bytes In      -" 0x31
EndModule
Module = "8 Bytes Out/ 8 Bytes In      -" 0x37
EndModule
Module = "32 Bytes Out/ 32 Bytes In    -" 0xC0,0x1F,0x1F
EndModule
Module = "64 Bytes Out/ 64 Bytes In    -" 0xC0,0x3F,0x3F
EndModule
Module = "1 Word Out/ 1 Word In        -" 0x70
EndModule
Module = "2 Word Out/ 2 Word In        -" 0x71
EndModule
Module = "4 Word Out/ 4 Word In        -" 0x73
EndModule
Module = "8 Word Out/ 8 Word In        -" 0x77
EndModule
Module = "16 Word Out/ 16 Word In      -" 0x7F
EndModule
Module = "32 Word Out/ 32 Word In      -" 0xC0,0x5F,0x5F
EndModule
Module = "2 Word Out/ 8 Word In        -" 0xC0,0x41,0x47
EndModule

```

```

Module = "4 Word Out/ 16 Word In      -" 0xC0,0x43,0x4F
EndModule
Module = "8 Word Out/ 32 Word In      -" 0xC0,0x47,0x5F
EndModule
Module = "8 Word Out/ 2 Word In       -" 0xC0,0x47,0x41
EndModule
Module = "16 Word Out/ 4 Word In      -" 0xC0,0x4F,0x43
EndModule
Module = "32 Word Out/ 8 Word In      -" 0xC0,0x5F,0x47
EndModule
Module = "4 Byte buffer I/O          -" 0xB3
EndModule
Module = "8 Byte buffer I/O          -" 0xB7
EndModule
Module = "12 Byte buffer I/O         -" 0xBB
EndModule
Module = "16 Byte buffer I/O         -" 0xBF
EndModule

```

Programa de ejemplo para la comunicación DP con una CPU 224

La figura A-32 muestra un programa de ejemplo en AWL para una CPU 224 que utiliza la información del puerto DP en la memoria de marcas especiales. La figura A-33 muestra ese mismo programa en KOP. El programa averigua la dirección de los búfers DP mediante SMW226 y lee los tamaños de los mismos de SMB228 y SMB229. Estas informaciones se utilizan en el programa para copiar los datos del búfer de salida DP a la imagen de proceso de las salidas de la CPU 224. De forma similar, los datos contenidos en la imagen del proceso de las entradas de la CPU 224 se copian en el búfer de entrada de la memoria V.

```
// Programa de ejemplo DP
// Los datos de configuración DP en el área de marcas proporcionan
// la configuración del esclavo DP. El programa utiliza
// los datos siguientes:
//
// SMW220 Estado de error del módulo DP
// SMB224 Estado DP
// SMB225 Dirección del maestro
// SMW226 Offset en la memoria de variables para los datos de salida
// SMB228 Número de bytes de los datos de salida
// SMB229 Número de bytes de los datos de entrada
// VD1000 Puntero de datos de salida
// VD1004 Puntero de datos de entrada
//
NETWORK 1
//
// Calcular el puntero de datos de salida a la memoria V.
//
LDB= SMB224, 2 // Si (en modo de intercambio de datos)
MOVD &VB0, VD1000 // El búfer de salida es un offset de VB0.
ITD SMW226, AC0 // Sumar offset Vmem para obtener el
+D AC0, VD1000 // offset del búfer de salida

NETWORK 2
//
// Calcular el puntero de datos de entrada a la memoria V.
//
LDB= SMB224, 2 // Si (en modo de intercambio de datos)
MOVD VD1000, VD1004 // Obtener la dirección del puntero de salida
BTI SMB228, AC0 // Cargar el número de bytes de salida
ITD AC0, AC0 // en el puntero de salida para obtener
+D AC0, VD1004 // el puntero de entrada inicial.

NETWORK 3
//
// Determinar la cantidad de datos a copiar.
//
LDB= SMB224, 2 // Si (en modo de intercambio de datos)
MOVB SMB228, VB1008 // Obtener el número de bytes de salida a
// copiar
MOVB SMB229, VB1009 // Obtener el número de bytes de entrada a
// copiar

NETWORK 4
//
// Transferir las salidas del maestro a las salidas de la CPU.
// Copiar las entradas de la CPU en las entradas del maestro.
//
LDB= SMB224, 2 // Si (en modo de intercambio de datos)
BMB *VD1000, QB0, VB1008 // Copiar las salidas del maestro a las
// salidas de la CPU
BMB IB0, *VD1004, VB1009 // Copiar las entradas de la CPU en las
// entradas del maestro
```

Figura A-32 Programa de ejemplo AWL para la comunicación DP con una CPU 224

A.13 Datos técnicos de los módulos de ampliación EM 231 Termopar y RTD

Tabla A-21 Datos técnicos de los módulos EM 231,1 Termopar y RTD

Descripción Nº de referencia	EM 231 4 AI x termopar 6ES7 231-7PD20-0XA0	EM 231 2 AI x termorresistencia RTD 6ES7 231-7PB20-0XA0
Datos generales		
Dimensiones (l x a x p)	71,2 mm x 80 mm x 62 mm	71,2 mm x 80 mm x 62 mm
Peso	210 g	210 g
Pérdida de corriente (disipación)	1,8 W	1,8 W
Cantidad de E/S físicas	4 entradas analógicas	2 entradas analógicas
Consumo de corriente De +DC 5 V (del bus de ampliación) De L+ Margen de tensión L+, clase 2 o alimentación de sensores DC	87 mA 60 mA 20,4 a 28,8 VDC	87 mA 60 mA 20,4 a 28,8 VDC
Indicador LED	Estado de la alimentación DC 24 V: ON = correcta, OFF = sin corriente DC 24 V SF: ON = fallo del módulo, PARPADEANTE = error de señal de entrada, OFF = sin fallos	Estado de la alimentación DC 24 V: ON = correcta, OFF = sin corriente DC 24 V SF: ON = fallo del módulo, PARPADEANTE = error de señal de entrada, OFF = sin fallos
Datos de las entradas analógicas		
Aislamiento Campo a circuito lógico Campo a DC 24 V DC 24 V a circuito lógico	AC 500 V AC 500 V AC 500 V	AC 500 V AC 500 V AC 500 V
Margen de entrada en modo común (de canal de entrada a canal de entrada)	AC 120 V	0
Rechazo en modo común	> 120 dB @ AC 120 V	> 120 dB @ AC 120 V
Tipo de entrada	Termopar flotante	RTD referenciado a tierra
Márgenes de las entradas	Tipos de termopar (seleccione uno): S, T, R, E, N, K, J Margen de tensión +/- 80 mV	Tipos de RTD (seleccione uno): Pt -100Ω, 200Ω, 500Ω, 1000Ω (con α = 3850 ppm, 3920 ppm, 3850,55 ppm, 3916 ppm, 3902 ppm) Pt -10000Ω (α = 3850 ppm) Cu -9,035Ω (α = 4720 ppm) Ni -10Ω, 120Ω, 1000Ω (con α = 6720 ppm, 6178 ppm) R -150Ω, 300Ω, 600Ω FS
Resolución de las entradas Temperatura Tensión Resistencia	0,1° C / 0,1° F Signo más (+) de 15 bits	0,1° C / 0,1° F Signo más (+) de 15 bits
Principio de medición	Sigma-delta	Sigma-delta
Tiempo de actualización del módulo: Todos los canales	405 mS	405 ms (700 ms para Pt10000)
Longitud del hilo	máx. 100 metros hasta el sensor	máx. 100 metros hasta el sensor
Resistencia de bucle del hilo	máx. 100Ω	máx. 20Ω, 2,7Ω para Cu
Supresión de interferencias	85 dB @ 50 Hz/60 Hz/ 400 Hz	85 dB @ 50 Hz/60 Hz/400 Hz
Formato palabra de datos	Tensión -27648 a + 27648	Resistencia -27648 a +27648
Disipación máxima del sensor		1 mW
Impedancia de entrada	>1 MΩ	>10 MΩ
Tensión de entrada máxima	DC 30 V	DC 30 V (detección), DC 5 V (fuente)
Resolución	Signo más (+) de 15 bits	Signo más (+) de 15 bits
Atenuación filtro de entrada	-3 dB @ 21 kHz	-3 dB @ 3.6 kHz
Error básico	0,1% FS (tensión)	0,1% FS (resistencia)
Repetibilidad	0,05% FS	0,05% FS
Error en la unión fría	±1,5 ° C	

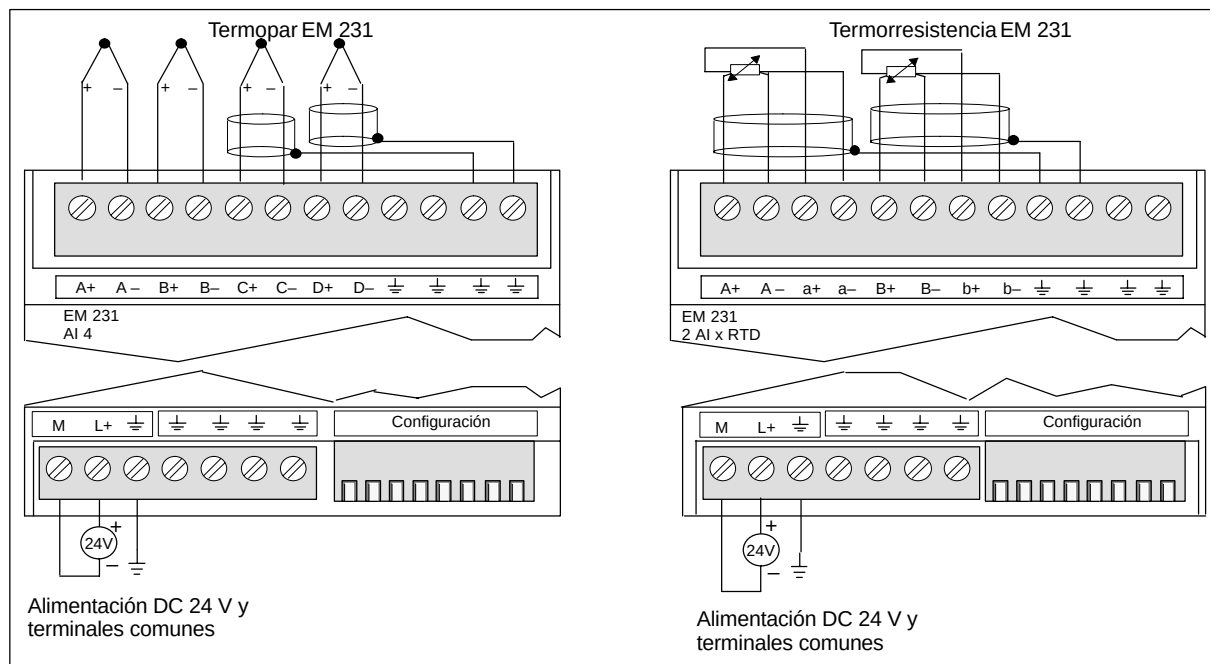


Figura A-34 Identificación de terminales de conexión para los módulos de ampliación EM 231 Termopar y RTD

Compatibilidad

Los módulos RTD y termopar se han diseñado para que se utilicen junto con las CPUs S7-200 222, 224 y 226.

Estos módulos ofrecen un rendimiento máximo al instalarse en entornos de temperatura estable. Por ejemplo, el módulo termopar EM 231 tiene circuitos especiales de compensación de unión fría que miden la temperatura en los conectores del módulo, exigiendo que se modifiquen las mediciones para compensar las diferencias entre la temperatura de referencia y la temperatura del módulo. Si la temperatura ambiente cambia rápidamente en el lugar donde está instalado el módulo termopar EM 231, se presentan errores adicionales. Para lograr una precisión y repetibilidad máximas, Siemens recomienda que los módulos termopar y RTD S7-200 se monten en lugares que tengan una temperatura ambiente estable.

Módulo EM 231 Termopar

El módulo EM 231 Termopar incorpora un interface aislado para conectar siete tipos de termopares a la gama S7-200, a saber: J, K, E, N, S, T y R. El módulo sirve para conectar el S7-200 a señales analógicas de nivel bajo en un margen de ± 80 mV. Los interruptores DIP se deben utilizar para seleccionar el tipo de termopar, la detección de hilos abiertos, la escala de temperatura, la compensación de temperatura en la unión fría y el sentido de la saturación térmica. Todos los termopares conectados al módulo deben ser del mismo tipo.

Configurar el módulo EM 231 Termopar

Como muestra la figura A-35, los interruptores DIP de configuración están ubicados en el lado inferior del módulo. Para que los ajustes de los interruptores DIP tengan efecto, es preciso desconectar y conectar nuevamente la CPU y/o la fuente de alimentación externa de 24 V.

El interruptor DIP 4 está reservado para el uso futuro. Ajuste el interruptor DIP 4 en la posición 0 (hacia abajo). En las tablas A-22 a A-26 figuran otros ajustes de los interruptores DIP.

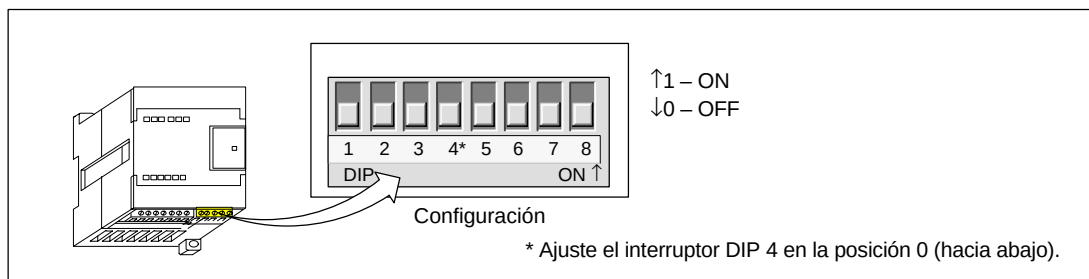


Figura A-35 Configurar los interruptores DIP del módulo EM 231 Termopar

Seleccionar el tipo de termopar Seleccione el tipo de termopar ajustando los interruptores DIP 1, 2 y 3, como muestra la tabla A-42.

Tabla A-22 Seleccionar el tipo de termopar

Tipo de termopar	Interruptor 1	Interruptor 2	Interruptor 3
J (estándar)	0	0	0
K	0	0	1
T	0	1	0
E	0	1	1
R	1	0	0
S	1	0	1
N	1	1	0
+/-80 mV	1	1	1

Seleccionar el sentido de la saturación térmica del sensor Seleccione el sentido de la saturación térmica (sentido ascendente o descendente de la escala) ajustando el interruptor DIP 5, como muestra la tabla A-23.

Tabla A-23 Seleccionar el sentido de la saturación térmica del sensor

Sentido de saturación térmica	Interruptor 5
Sentido ascendente de la escala (+3276,7 grados)	0
Sentido descendente de la escala (-3276,8 grados)	1

Seleccionar la detección de hilos abiertos La detección de hilos abiertos se efectúa inyectando una corriente de 25 μA en los terminales de entrada. El interruptor de detección de hilos abiertos habilita o inhibe la fuente de corriente. La detección de hilos abiertos se efectúa siempre, aunque esté inhibida la fuente de corriente. El módulo EM 231 Termopar detecta los hilos abiertos si la señal de entrada excede aproximadamente $\pm 200 \text{ mV}$. Cuando se detecta un hilo abierto, la lectura del módulo se ajusta al valor seleccionado por el sentido de saturación térmica del sensor. Habilite o inhiba la fuente de corriente de los hilos abiertos ajustando el interruptor DIP 6 a la función deseada, como muestra la tabla A-24.

Tabla A-24 Seleccionar la detección de hilos abiertos

Hilo abierto	Interruptor 6
Habilitar la fuente de corriente del hilo abierto	0
Inhibir la fuente de corriente del hilo abierto	1

Nota

- Es posible que la fuente de corriente del hilo abierto interfiera con señales de algunas fuentes de bajo nivel, tales como simuladores de termopares.
- Las tensiones de entrada superiores a aprox. $\pm 200 \text{ mV}$ dispararán la detección de hilos abiertos, aunque esté inhibida la fuente de corriente.

Seleccionar la escala de temperatura El módulo EM 231 Termopar puede indicar la temperatura en grados centígrados o Fahrenheit. La conversión de grados centígrados a Fahrenheit se efectúa en el módulo. Utilice el interruptor DIP 7 para seleccionar la escala de temperatura, como muestra la tabla A-25.

Tabla A-25 Seleccionar la escala de temperatura

Escala	Interruptor 7
Escala Celsius ($^{\circ}\text{C}$)	0
Escala Fahrenheit ($^{\circ}\text{F}$)	1

Seleccionar la compensación de la temperatura en la unión fría La compensación de la temperatura en la unión fría se debe habilitar cuando se están utilizando termopares. Si no está habilitada dicha compensación, las conversiones del módulo serán erróneas, debido a la tensión que se crea cuando el hilo del termopar está unido al conector del módulo. La compensación de la temperatura en la unión fría se inhibe automáticamente al seleccionarse el margen de ± 80 mV. Utilice el interruptor DIP 8 para habilitar o inhibir la compensación de la temperatura en la unión fría, como muestra la tabla A-26.

Tabla A-26 Seleccionar la compensación de la temperatura en la unión fría

Habilitar la compensación de la temperatura en la unión fría	Interruptor 8
Habilitar la compensación de la temperatura en la unión fría	0
Inhibir la compensación de la temperatura en la unión fría	1

Nota

- Al estar cambiando la temperatura ambiente, el error del módulo podría ser superior a lo indicado en los datos técnicos.
- Si se excede el margen de temperatura ambiente del módulo, ello podría provocar un error de la compensación de la temperatura en la unión fría.

Cableado del módulo EM 231 Termopar

Conecte los hilos de los termopares directamente al módulo EM 231 Termopar. Utilice hilos blindados para obtener el mayor nivel posible de inmunidad a interferencias. Si se usan hilos blindados, el apantallamiento se deberá conectar a tierra mediante los pines 1 a 4 del conector de señales. Esta tierra es la misma que está presente en los pines 3 a 7 del conector de corriente. Si no se utiliza un canal de entrada del termopar, es recomendable cortocircuitar las entradas de canal no utilizadas, o bien conectarlas en paralelo a otro canal. Ello evita la indicación de errores de bloqueo (resultantes de entradas flotantes) en los canales válidos.

La fuente de alimentación externa se debe conectar a los pines 1 y 2 del conector de corriente. El pin 3 del conector de corriente se debe conectar a una toma a tierra cercana (v. fig. A-36).

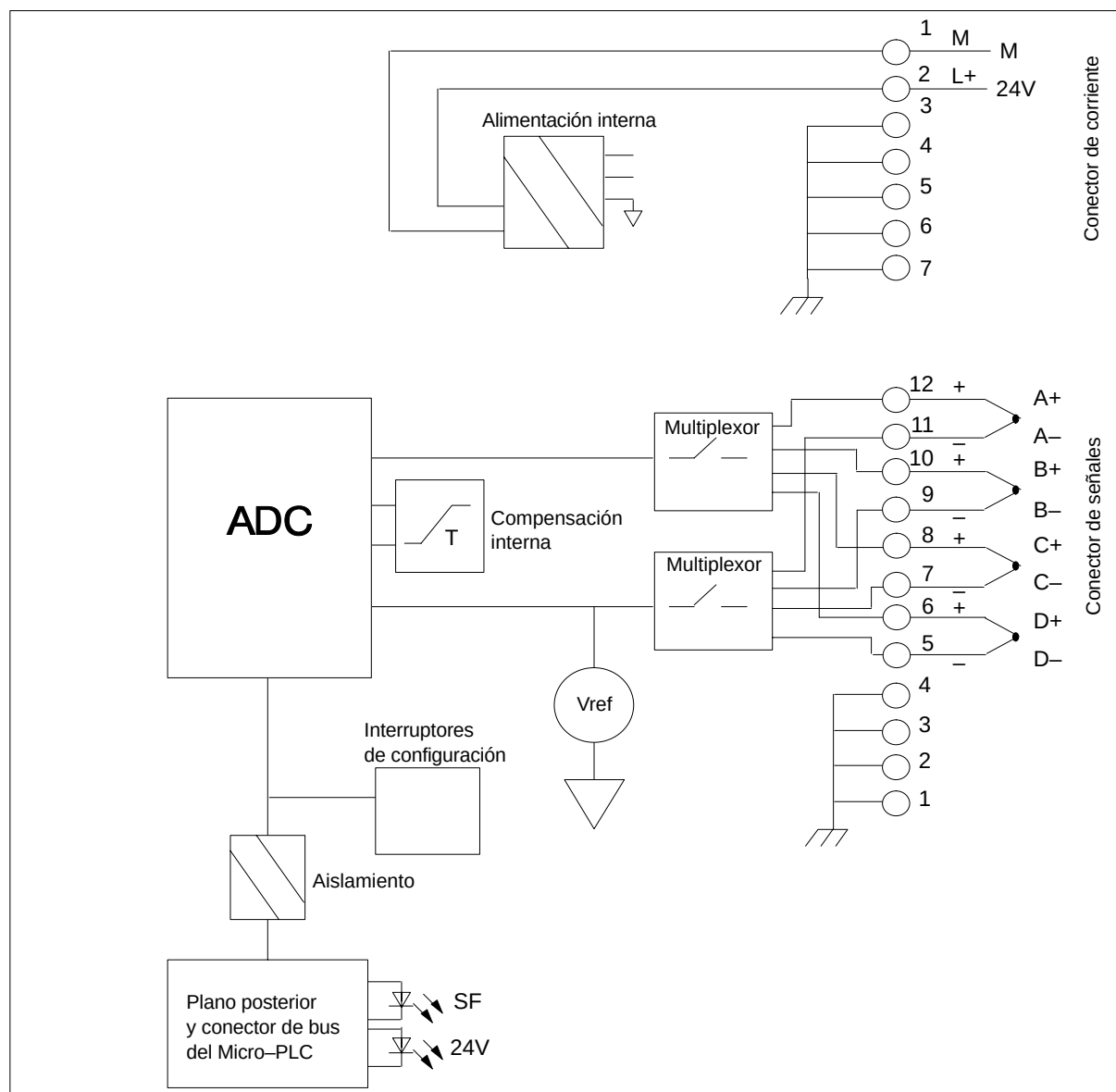


Figura A-36 Esquema de circuitos del módulo termopar

Indicadores de estado del módulo termopar

El módulo termopar le suministra a la CPU palabras de estado que indican la temperatura o condiciones de error. Los bits de estado muestran los errores de margen y el fallo de la alimentación externa o interna. Los LEDs indican el estado del módulo. El programa de usuario debe incorporar lógica para detectar las condiciones de error y reaccionar de forma apropiada conforme a la aplicación. La tabla A-27 muestra los indicadores de estado del módulo EM 231 Termopar.

Tabla A-27 Indicadores de estado del módulo EM 231 Termopar

Error	Datos de canal	LED SF	LED 24 V	Bit de estado (error de margen) ¹	Bit de estado (fallo de la alimentación de 24 V) ²
Sin errores	Datos de conversión	OFF	ON	0	0
Falta la alimentación de 24 V	32766	OFF	OFF	0	1
Detección de hilos abiertos y fuente de corriente habilitadas	–32768/32767	PARPADEANTE	ON	1	0
Entrada fuera de margen	–32768/32767	PARPADEANTE	ON	1	0
Error de diagnóstico ³	0000	ON	OFF	0	*

¹ El bit de error de margen es el bit 3 del byte de registro de errores del módulo (SMB9 en el caso del módulo 1, SMB11 en el caso del módulo 2, etc.)

² El bit de fallo de alimentación es el bit 2 del byte de registro de errores del módulo (SMB 9, SMB 11, etc. Consulte el Anexo C).

³ Los errores de diagnóstico causan un error de configuración del módulo. El bit de fallo del módulo se puede activar o no antes del error de configuración del módulo.

Nota

El formato de datos del canal es de palabra de 16 bits (complemento a 2). La temperatura se indica en unidades de 0,1 grados. (Por ejemplo, si se mide una temperatura de 100,2 grados, se indicará 1002). Los datos de tensión se escalan a 27648. Por ejemplo, –60,0 mV se indica como –20736 ($= -60 \text{ mV} / 80 \text{ mV} * 27648$).

Si la CPU ha leído los datos, los cuatro canales se actualizarán cada 405 milisegundos. Si la CPU no lee los datos dentro de un intervalo de actualización, el módulo indicará los datos antiguos hasta que se produzca la siguiente actualización (después de que la CPU haya leído los datos). Para mantener los datos de canal actualizados, es recomendable que el programa de la CPU lea los datos al menos con la misma frecuencia de actualización del módulo.

Nota

Si se utiliza el módulo termopar, es preciso desactivar el filtro de entradas analógicas de la CPU. La filtración de entradas analógicas puede provocar que las condiciones de error no se detecten a tiempo.

Nociones básicas de los termopares

Los termopares se forman cuando se unen dos metales distintos que, al calentarse, generan una fuerza electromotriz. La tensión generada es proporcional a la temperatura de unión. Se trata de una tensión pequeña; un microvoltio puede representar varios grados. La base de la medición de temperatura utilizando termopares consiste en medir la tensión de un termopar, compensar las uniones adicionales y linealizar posteriormente el resultado.

Cuando un termopar se conecta al módulo EM 231, dos hilos de distintos metales se unen al conector de señales del módulo. El punto donde estos dos hilos se unen al módulo constituye el termopar sensor. Dos termopares adicionales se forman cuando los dos hilos diferentes se unen al conector de señales. La temperatura del conector genera una tensión que se suma a la del termopar sensor. Si no se corrige esta tensión, la temperatura indicada será diferente a la temperatura del sensor. La compensación de la temperatura en la unión fría se utiliza para compensar el termopar del conector. Las tablas de termopares se basan en una temperatura de unión de referencia que, por lo general, equivale a cero grados centígrados. La compensación de temperatura en la unión fría del módulo compensa el conector a cero grados centígrados, restableciendo la tensión sumada por los termopares del conector. La temperatura del módulo se mide internamente, convirtiéndose a un valor a sumar a la conversión del sensor. La conversión del sensor corregida se linealiza entonces utilizando las tablas.

Márgenes del módulo termopar

Las tablas A-28 y A-29 muestran los márgenes de temperatura y la precisión de los termopares.

Tabla A-28 Márgenes de temperatura (°C) y precisión de los termopares

Palabra de datos (1 dígito = 0,1 °C)		Tipo J	Tipo K	Tipo T	Tipo E	Tipos R, S	Tipo N	±80 mV	
DEC	Hex								
32767	7FFF	>1200,0 °C	>1372,0 °C	>400,0 °C	>1000,0 °C	>1768,0 °C	>1300,0 °C	>94,071 mV	DE
↑	↑							↑	↑
32511	7EFF							94,071 mV	SM
:	:							80,0029 mV	
27649	6C01							80 mV	MN
27648	6C00					↑			
:	:								
17680	4510		↑			1768,0 °C			
:	:						↑		
13720	3598		1372,0 °C						
:	:		sobre margen						
13000	32C8	↑	1300,0 °C				1300,0 °C		
:	:								
12000	2EE0	1200,0 °C			↑				
:	:								
10000	2710			↑	1000,0 °C				
:	:								
4000	0FA0			400,0 °C		400,0 °C			
:	:								
1	0001	0,1 °C	0,1 °C	0,1 °C	0,1 °C	0,1 °C	0,1 °C	0,0029 mV	
0	0000	0,0 °C	0,0 °C	0,0 °C	0,0 °C	0,0 °C	0,0 °C	0,0 mV	
-1	FFFF	-0,1 °C	-0,1 °C	-0,1 °C	-0,1 °C	-0,1 °C	-0,1 °C	-0,0029 mV	
:	:								
-500	FE0C					bajo margen			
-1500	FA24	-150,0 °C				-50,0 °C			
:	:								
-2000	F830	bajo margen	-200,0 °C						
:	:								
-2100	F7CC	-210,0 °C							
:	:		bajo margen						
-2550	F60A			-255,0 °C	-255,0 °C				
:	:			bajo margen	bajo margen				
-2700	F574	↓	-270,0 °C	-270,0 °C	-270,0 °C		-270,0 °C		
:	:								
-27648	9400		↓	↓	↓		↓	-80 mV	
-27649	93FF							-80,0029 mV	
:	:								
-32512	8100							-94,071 mV	
↓	↓							↓	
-32768	8000	<-210,0 °C	<-270,0 °C	<-270,0 °C	<-270,0 °C	<-50,0 °C	<-270,0 °C	<-94,071 mV	DD
Precisión en todo el margen		± 0,1%	± 0,3%	± 0,6%	± 0,1%	± 0,6%	± 0,1%	± 0,1%	
Precisión (margen nominal sin unión fría)		± 1,5 °C	± 1,7 °C	± 1,4 °C	± 1,3 °C	± 3,7 °C	± 1,6 °C	± 0,10%	
Error en la unión fría		± 1,5 °C	± 1,5 °C	± 1,5 °C	± 1,5 °C	± 1,5 °C	± 1,5 °C	N/A	
*DE = desbordamiento por exceso; SM = sobre margen; MN = margen nominal; BM = bajo margen; DD = desbordamiento por defecto									
↑ indica que todos los valores analógicos superiores a éste e inferiores al valor umbral de conductos abiertos suministran los valores de datos de desbordamiento por exceso, 32767 (0x7FFF).									
↓ indica que todos los valores analógicos inferiores a éste y superiores al valor umbral de conductos abiertos suministran los valores de datos de desbordamiento por defecto, -32768 (0x8000).									

Tabla A-29 Márgenes de temperatura (°F) y precisión de los termopares

Palabra de datos (1 dígito = 0,1°F)		Tipo J	Tipo K	Tipo T	Tipo E	Tipos R, S	Tipo N	±80 mV	
DEC	Hex								
32767	7FFF	>2192,0 °F	>2502,0 °F	>752,0 °F	>1832,0 °F	>3214,0 °F	>2372,0 °F	>94,071 mV	DE
↑	↑					↑		↑	↑
32511	7EFF							94,071 mV	SM
32140	7D90					3214,0 °F		80,0029 mV	
27649	6C01					2764,8 °F		80 mV	MN
27648	6C00		↑						
:	:								
25020	61B8		2502,0 °F				↑		
:	:		sobre margen						
23720	5CA8	↑	2372,0 °F				2372,0 °F		
:	:								
21920	55A0	2192,0 °F			↑				
:	:								
18320	4790			↑	1832,0 °F				
:	:								
7520	1D60			752,0 °F		752,0 °F			
:	:								
320	0140					bajo margen	32,0 °F		
:	:								
1	0001	0,1 °F	0,1 °F	0,1 °F	0,1 °F	0,1 °F	0,1 °F	0,0029 mV	
0	0000	0,0 °F	0,0 °F	0,0 °F	0,0 °F	0,0 °F	0,0 °F	0,0 mV	
-1	FFFF	-0,1 °F	-0,1 °F	-0,1 °F	-0,1 °F	-0,1 °F	-0,1 °F	-0,0029 mV	
:	:								
-580	FDBC					-58,0 °F			
:	:								
-2380	F6B4	-238,0 °F							
:	:								
-3280	F330	bajo margen	-328,0 °F				bajo margen		
:	:								
-3460	F27C	-346,0 °F							
:	:		bajo margen						
-4270	EF52			-427,0 °F	-427,0 °F				
:	:			bajo margen	bajo margen				
-4540	EE44	↓	-454,0 °F	-454,0 °F	-454,0 °F		-454,0 °F		
:	:								
-27648	9400		↓	↓	↓		↓	-80 mV	
-27649	93FF							-80,0029 mV	
:	:								
-32512	8100							-94,071 mV	SM
↓	↓							↓	↓
-3268	8000	<-346,0 °F	<-454,0 °F	<-454,0 °F	<-454,0 °F	<-58,0 °F	<-454,0 °F	<-94,07 mV	DD

*DE = desbordamiento por exceso; SM = sobre margen; MN = margen nominal; BM = bajo margen; DD = desbordamiento por defecto
↑ indica que todos los valores analógicos superiores a éste e inferiores al valor umbral de conductos abiertos suministran los valores de datos de desbordamiento por exceso, 32767 (0x7FFF).
↓ indica que todos los valores analógicos inferiores a éste y superiores al valor umbral de conductos abiertos suministran los valores de datos de desbordamiento por defecto, -32768 (0x8000).

Módulo EM 231 RTD

El módulo EM 231 RTD incorpora un interface aislado para conectar diversas RTDs (termo-resistencias) a la gama S7-200, sirviendo también para medir tres diferentes márgenes de resistencia con un sistema de automatización S7-200. Los interruptores DIP se deben utilizar para seleccionar el tipo de RTD, el esquema de cableado, la escala de temperatura y el sentido de la saturación térmica. Las dos RTDs conectadas al módulo deben ser de un mismo tipo.

Configurar el módulo EM 231 RTD

Como muestra la figura A-37, los interruptores DIP de configuración están ubicados en el lado inferior del módulo. Para que los ajustes de los interruptores DIP tengan efecto, es preciso desconectar y conectar nuevamente la CPU y/o la fuente de alimentación externa de 24 V.

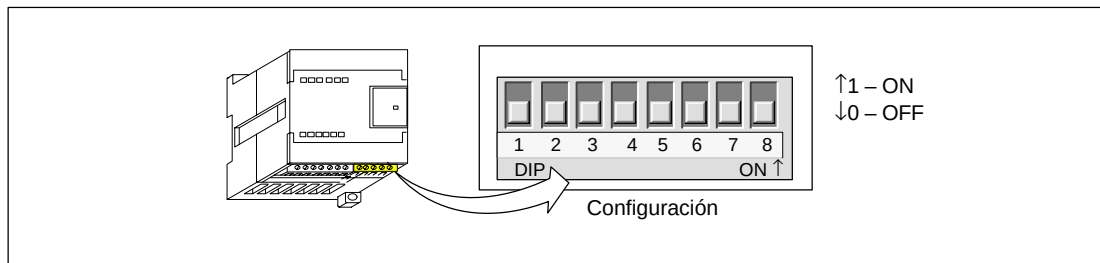


Figura A-37 Configurar los interruptores DIP del módulo RTD

Seleccionar el tipo de RTD Seleccione el tipo de RTD ajustando los interruptores DIP 1, 2, 3, 4 y 5, de manera que correspondan a las RTDs que muestra la tabla A-30.

Tabla A-30 Seleccionar el tipo de RTD

Tipo de RTD y alfa	Interruptor 1	Interruptor 2	Interruptor 3	Interruptor 4	Interruptor 5
100Ω Pt 0,003850 (estándar)	0	0	0	0	0
200Ω Pt 0,003850	0	0	0	0	1
500Ω Pt 0,003850	0	0	0	1	0
1000Ω Pt 0,003850	0	0	0	1	1
100Ω Pt 0,003920	0	0	1	0	0
200Ω Pt 0,003920	0	0	1	0	1
500Ω Pt 0,003920	0	0	1	1	0
1000Ω Pt 0,003920	0	0	1	1	1
100Ω Pt 0,00385055	0	1	0	0	0
200Ω Pt 0,00385055	0	1	0	0	1
500Ω Pt 0,00385055	0	1	0	1	0
1000Ω Pt 0,00385055	0	1	0	1	1
100Ω Pt 0,003916	0	1	1	0	0
200Ω Pt 0,003916	0	1	1	0	1
500Ω Pt 0,003916	0	1	1	1	0
1000Ω Pt 0,003916	0	1	1	1	1
100Ω Pt 0,00302	1	0	0	0	0
200Ω Pt 0,003902	1	0	0	0	1
500Ω Pt 0,003902	1	0	0	1	0
1000Ω Pt 0,003902	1	0	0	1	1
SPARE	1	0	1	0	0
100Ω Ni 0,00672	1	0	1	0	1
120Ω Ni 0,00672	1	0	1	1	0
1000Ω Ni 0,00672	1	0	1	1	1
100Ω Ni 0,006178	1	1	0	0	0
120Ω Ni 0,006178	1	1	0	0	1
1000Ω Ni 0,006178	1	1	0	1	0
10000Ω Pt 0,003850	1	1	0	1	1
10Ω Cu 0,004270	1	1	1	0	0
Resistencia 150Ω FS	1	1	1	0	1
Resistencia 300Ω FS	1	1	1	1	0
Resistencia 600Ω FS	1	1	1	1	1

Seleccionar el sentido de la saturación térmica del sensor Utilice el interruptor DIP 6 para ajustar el sentido de la saturación térmica del sensor, como muestra la tabla A-31.

Tabla A-31 Seleccionar el sentido de la saturación térmica del sensor

Sentido de saturación térmica	Interruptor 6
Sentido ascendente de la escala (+3276,7 grados)	0
Sentido descendente de la escala (−3276,8 grados)	1

Seleccionar la escala de temperatura El módulo RTD puede indicar temperaturas en grados centígrados o Fahrenheit. La conversión de grados centígrados a Fahrenheit se efectúa en el módulo. Utilice el interruptor DIP 7 para seleccionar la escala de temperatura, como muestra la tabla A-32.

Tabla A-32 Seleccionar la escala de temperatura

Escala	Interruptor 7
Escala Celsius (°C)	0
Escala Fahrenheit (°F)	1

Seleccionar el esquema de cableado Utilice el interruptor DIP 8 para seleccionar el esquema de cableado, como muestra la tabla A-33.

Tabla A-33 Esquema de cableado

Esquema de cableado	Interruptor 8
3 hilos	0
2 ó 4 hilos	1

Cableado del módulo EM 231 RTD

El módulo EM 231 RTD se puede cablear directamente al sistema de automatización S7-200. Alternativamente, es posible utilizar cables de extensión. Utilice hilos blindados para obtener el mayor nivel posible de inmunidad a interferencias. Si se usan hilos blindados, el apantallamiento se deberá conectar a tierra mediante los pines 1 a 4 del conector de señales. Esta tierra es la misma que está presente en los pines 3 a 7 del conector de corriente. Si no desea usar un canal de entrada RTD, es recomendable que cablee una resistencia a las entradas de canal no utilizadas. Ello evita la indicación de errores de bloqueo (resultantes de entradas flotantes) en los canales válidos.

La fuente de alimentación externa se debe conectar a los pines 1 y 2 del conector de corriente. El pin 3 del conector de corriente se debe conectar a una toma a tierra cercana (v. fig. A-38).

El módulo RTD se puede conectar al sensor de las tres maneras que muestra la figura A-39. La más precisa es la conexión a 4 hilos. La menos precisa es la conexión a 2 hilos, siendo sólo recomendable si en la aplicación se pueden ignorar los errores debidos al cableado.

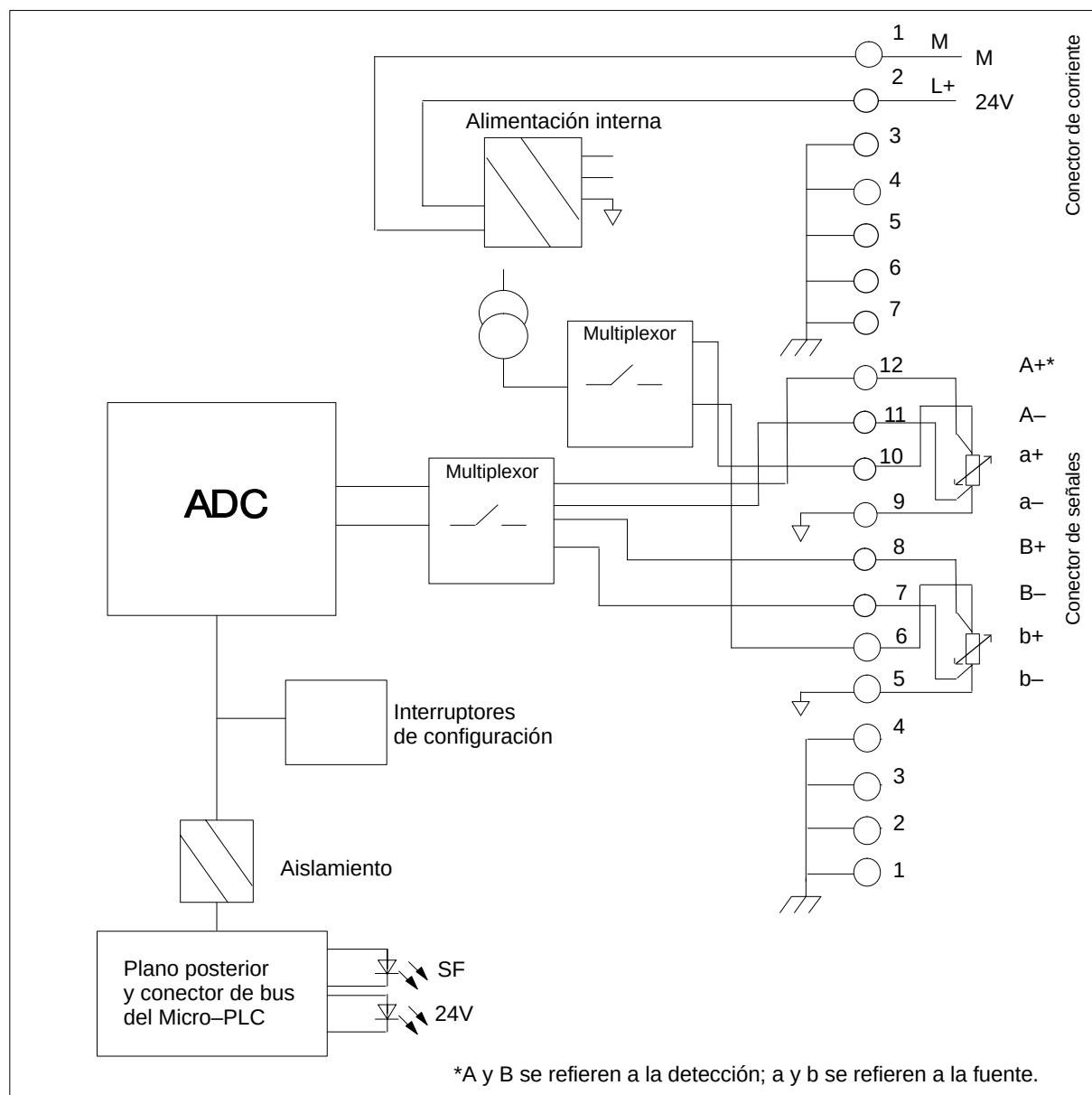


Figura A-38 Esquema de circuitos del módulo RTD

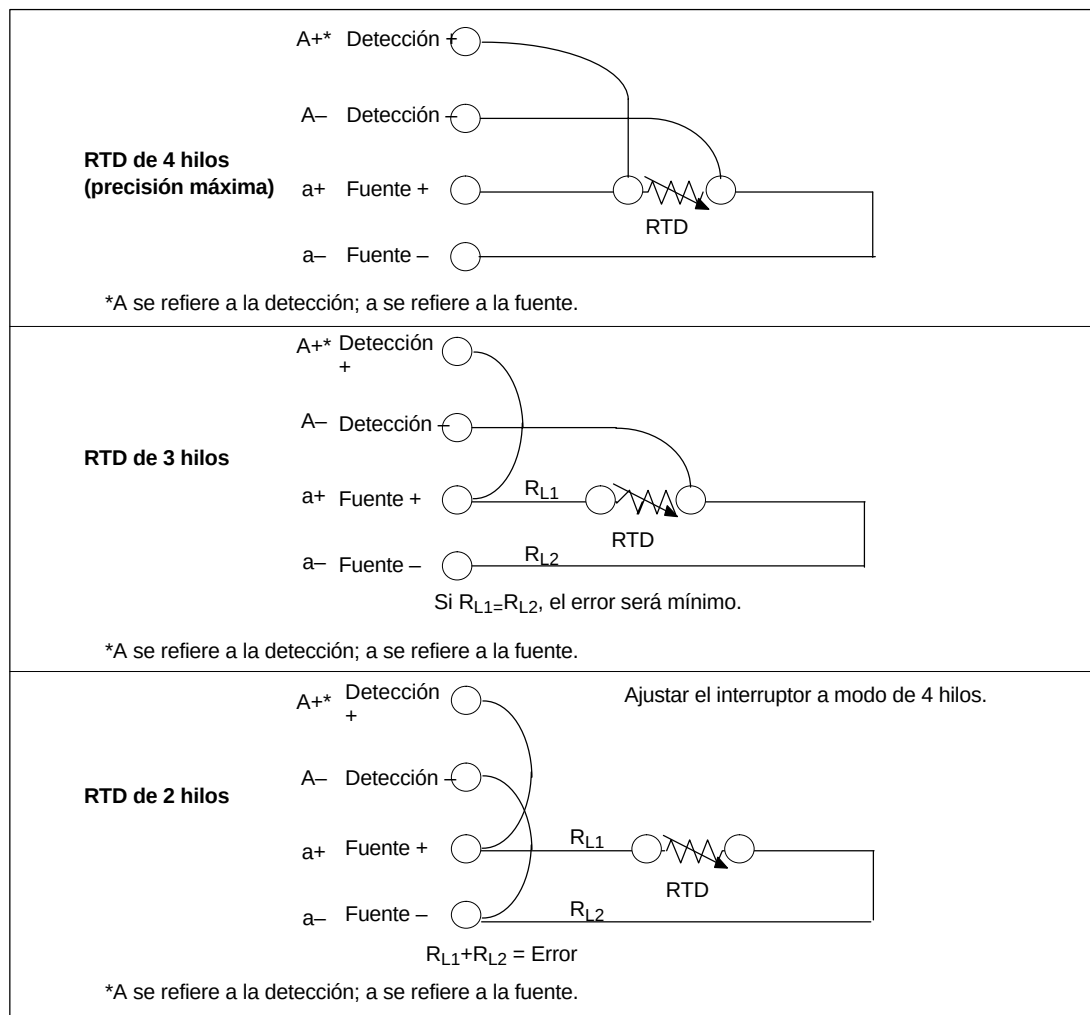


Figura A-39 Cablear el módulo RTD al sensor con 4, 3 y 2 hilos

Indicadores de estado del módulo EM 231 RTD

El módulo RTD le suministra a la CPU palabras de estado que indican la temperatura o condiciones de error. Los bits de estado muestran los errores de margen y el fallo de la alimentación externa o interna. Los LEDs indican el estado del módulo. El programa de usuario debe incorporar lógica para detectar las condiciones de error y reaccionar de forma apropiada conforme a la aplicación. La tabla A-34 muestra los indicadores de estado del módulo EM 231 RTD.

Nota

El formato de datos del canal es de palabra de 16 bits (complemento a 2). La temperatura se indica en unidades de 0,1 grados. (Por ejemplo, si se mide una temperatura de 100,2 grados, se indicará 1002). Los datos de resistencia se escalan a 27648. Por ejemplo, 75% de la resistencia de escala total se indica como 20736 ($=225\Omega / 300\Omega \cdot 27648$).

Tabla A-34 Indicadores de estado del módulo EM 231 RTD

Error	Datos de canal	LED SF	LED 24 V	Bit de estado (error de margen) ¹	Bit de estado (fallo de la alimentación de 24 V) ²
Sin errores	Datos de conversión	OFF	ON	0	0
Falta la alimentación de 24 V	32766	OFF	OFF	0	1
El interruptor detecta un hilo abierto	–32768/32767	PARPA-DEANTE	ON	1	0
Entrada fuera de margen	–32768/32767	PARPA-DEANTE	ON	1	0
Error de diagnóstico ³	0000	ON	OFF	0	*

¹ El bit de error de margen es el bit 3 del byte de registro de errores del módulo (SMB9 en el caso del módulo 1, SMB11 en el caso del módulo 2, etc.)

² El bit de fallo de alimentación es el bit 2 del byte de registro de errores del módulo (SMB 9, SMB 11, etc. Consulte el Anexo C).

³ Los errores de diagnóstico causan un error de configuración del módulo. El bit de fallo del módulo se puede activar o no antes del error de configuración del módulo.

Si la CPU ha leído los datos, los datos de canal se actualizarán cada 405 milisegundos. Si la CPU no lee los datos dentro de un intervalo de actualización, el módulo indicará los datos antiguos hasta que se produzca la siguiente actualización (después de que la CPU haya leído los datos). Para mantener los datos de canal actualizados, es recomendable que el programa de la CPU lea los datos al menos con la misma frecuencia de actualización del módulo.

Nota

Si se utiliza el módulo RTD, es preciso desactivar el filtro de entradas analógicas de la CPU. La filtración de entradas analógicas puede provocar que las condiciones de error no se detecten a tiempo.

El software interno del módulo RTD detecta si hay hilos abiertos. Las entradas fuera de margen se declaran y los hilos abiertos se indican en calidad de datos de saturación térmica. La detección de hilos abiertos tarda como mínimo tres ciclos del módulo o más, dependiendo del hilo (de los hilos) que esté(n) abierto(s). Los hilos abiertos Fuente+ y/o Fuente– se detectan en el tiempo mínimo. La detección de los hilos abiertos Detección+ y/o Detección– puede tardar 5 segundos o más. En los hilos abiertos se pueden presentar datos válidos al azar, detectándose éstos de forma intermitente, especialmente en entornos con muchas interferencias. Las interferencias eléctricas también pueden prolongar el tiempo necesario para detectar hilos abiertos. Tras haberse suministrado datos válidos, en el programa de aplicación es recomendable utilizar cerrojos para las indicaciones de hilos abiertos o fuera de margen.

Márgenes del módulo EM 231 RTD

Las tablas A-35 y A-36 muestran los márgenes de temperatura y la precisión de los módulos RTD.

Tabla A-35 Márgenes de temperatura (°C) y precisión de los módulos RTD

Palabra de sistema (1 dígito = 0,1 °C)		Pt10000	Pt100, Pt200, Pt500, Pt1000	Ni100, Ni120, Ni1000	Cu9,035	0 – 150Ω	0 – 300Ω	0 – 160Ω		
Decimal	Hex									
32767	7FFF									
32766	7FFE									
32511	7EFF									
29649	6C01									
27648	6C00									
25000	61A8									
18000	4650									
15000	3A98									
13000	32C8	↑	↑							
10000	2710	1000,0°C								
8500	2134									
6000	1770	600,0°C								
3120	0C30			↑	312,0°C					
2950	0B86			295,0°C						
2600	0A28									
2500	09C4			250,0°C						
1	0001	0,1°C	0,1°C	0,1°C	0,1°C	0,005Ω	0,011Ω	0,022Ω	MN	
0	0000	0,0°C	0,0°C	0,0°C	0,0°C	0,000Ω	0,000Ω	0,000Ω		
-1	FFFF	-0,1°C	-0,1°	-0,1°C	-0,1°C	(no son posibles los valores negativos)				
-600	FDA8			-60,0°C						
-1050	FBE6			-105,0°C						
-2000	F830	-200,0°C		-200,0°						
-2400	F6A0									
-2430	F682	-243,0°C		-243,0°C						
		↓	↓							
-5000	EC78									
-6000	E890									
-10500	D6FC									
-12000	VD120									
-20000	4E20									
-32767	8001									
-32768	8000									
Precisión en todo el margen		± 0,4%	± 0,1%	± 0,2%	± 0,5%	± 0,1%	± 0,1%	± 0,1%		
Precisión (margen nominal)		± 4°C	± 1°C	± 0,6°C	± 2,8°C	± 0,15Ω	± 0,3Ω	± 0,6Ω		
*DE = desbordamiento por exceso; SM = sobre margen; MN = margen nominal; BM = bajo margen; DD = desbordamiento por defecto										
↑ ó ↓ indican que todos los valores analógicos que excedan los límites suministran el valor seleccionado de saturación térmica, 32767 (0x7FFF) ó -32768 (0x8000).										

Tabla A-36 Márgenes de temperatura (°F) y precisión de los módulos RTD

Palabra de sistema (1 dígito = 0,1 °F)		PT1000	PT100, Pt200, Pt500, Pt1000	Ni100, Ni120, Ni1000	Cu9,035	
Decimal	Hexadecimal					
32767	7FFF					
32766	7FFE					
		↑	↑			
18320	4790	1832,0°F	1832,0 °F			
15620	3D04	1562,0°F				
11120	2B70	1112,0°F				
5936	1730	↑ 593,6°F				
5630	15FE	563,0°F				
5000	1388	500,0°F				
4820	12D4	482,0°F				
1	0001	0,1°F	0,1°F	0,1°F	0,1°F	Margen normal
0	0000	0,0°F	0,0°F	0,0°F	0,0°F	
-1	FFFF	-0,1°F	-0,1°F	-0,1°F	-0,1°F	
-760	FD08	-76,0°F				
-1570	F9DE	-157,0°F				
		↓				
-3280	F330	-328,0°F	-328,0°F	-328,0°F		
-4000	F060	-400,0°F		-400,0°F		
-4054	F02A	-405,4°F	-405,4°F	↓		
		↓	↓			
-5000	EC78					
-6000	E890					
-10500	D6FC					
-32767	8001					
-32768	8000					

↑

sobre margen

Margen normal

bajo margen

↓

↑ ó ↓ indican que todos los valores analógicos que excedan los límites suministran el valor seleccionado de saturación térmica, 32767 (0x7FFF) ó -32768 (0x8000).

A.14 Procesador de comunicaciones CP 243-2

Descripción Nº de referencia	Procesador de comunicaciones CP 243-2 6GK7 243-2AX00-0XA0
Perfil del maestro AS-Interface	M0/M1
Interfaces – Asignación al área de direcciones de la CPU – Conexión al AS-Interface	Equivale a 2 módulos de E/S (8 DI/8 DO y 8 AI/8 AQ) Conexión de terminales
Consumo de corriente – vía AS-Interface – vía bus posterior	máx. 100 mA 220 mA a DC 5 V (típ.)
Pérdida de corriente	aprox. 2 W
Condiciones ambientales admisibles – Temperatura de servicio Montaje horizontal Montaje vertical – Temperatura de transporte/almacenamiento – Humedad relativa	0° C a +55° C 0° C a +45° C –40° C a +70° C 95% a +25° C
Construcción – Formato del módulo – Dimensiones (l x a x p) en mm – Peso	Módulo de ampliación S7-22x 71,2 x 80 x 62 aprox. 250 g

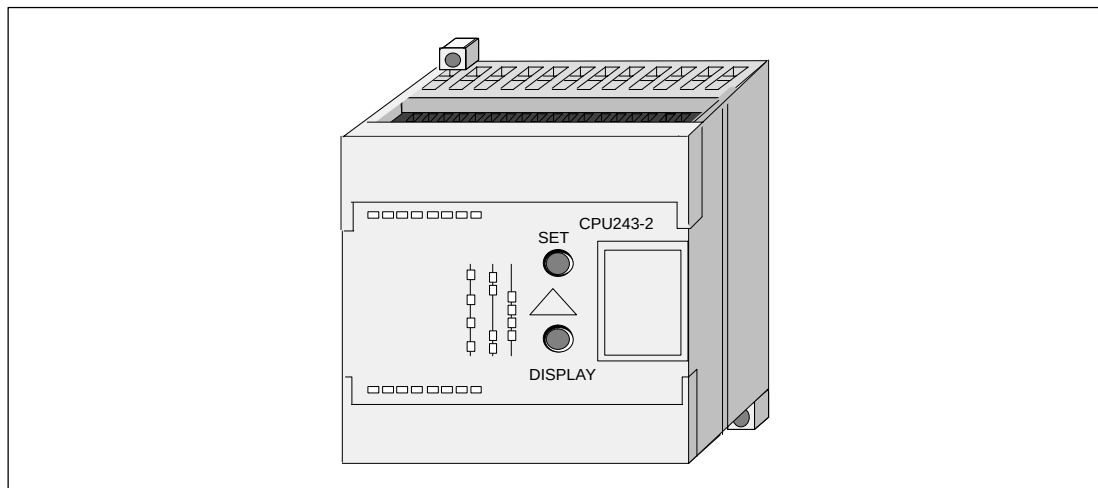


Figura A-40 Procesador de comunicaciones CP 243-2

Resumen breve

El CP 243-2 es el maestro AS-Interface de la gama de CPUs 22x del S7-200. Este nuevo procesador de comunicaciones ofrece la siguiente funcionalidad:

- Permite conectar hasta 31 esclavos AS-Interface.
- Soporta todas las funciones del maestro AS-Interface.
- Los LEDs de la placa frontal muestran el estado de operación y la disponibilidad de los esclavos conectados.
- Los LEDs de la placa frontal muestran los errores que se hayan detectado (incluyendo fallos de tensión del AS-Interface y errores de configuración).
- Carcasa compacta conforme al diseño innovador de la gama SIMATIC S7-200.

Aplicación

El procesador de comunicaciones CP 243-2 es el maestro AS-Interface diseñado exclusivamente para las CPUs S7-200 22x. Conectando la CPU S7-200 a un AS-Interface se incrementa considerablemente el número de entradas y salidas digitales disponibles (máx. 124 entradas digitales y 124 salidas digitales en un AS-Interface por cada CP). La CPU S7-200 puede controlar simultáneamente dos CPs 243-2 como máximo.

Diseño

El CP 243-2 se conecta a la CPU S7-200 de la misma manera que un módulo de ampliación. Dispone de:

- Dos terminales para la conexión directa al cable del AS-Interface.
- LEDs en la placa frontal que muestran el estado de operación y la disponibilidad de todos los esclavos conectados y activados.
- Dos botones para visualizar la información de estado de los esclavos, para cambiar el modo de operación y para adoptar la configuración existente como configuración SET.

Operación

En la imagen del proceso de la CPU S7-200, el CP 243-2 ocupa un byte de entrada digital (byte de estado), un byte de salida digital (byte de control), así como 8 palabras de entrada analógica y 8 palabras de salida analógica. Por consiguiente, el CP 243-2 ocupa dos posiciones de módulo lógicas. Los bytes de estado y de control se pueden utilizar para ajustar el modo del CP 243-2 mediante el programa de usuario. Dependiendo del modo, el CP 243-2 almacena los datos de E/S del esclavo AS-Interface o los valores de diagnóstico, o bien habilita las llamadas del maestro (p.ej. para cambiar la dirección de un esclavo) en el área de direcciones analógicas del S7-200.

Todos los esclavos AS-Interface conectados se pueden configurar pulsando un botón. No es necesario efectuar ningún otro ajuste de configuración del CP.

Cuidado

Si se utiliza el módulo CP 243-2, es preciso desactivar el filtro de entradas analógicas de la CPU.

En caso contrario, se destruirán los datos digitales y los errores no se indicarán como valores binarios de la palabra analógica.

Verifique que esté desactivado el filtro de entradas analógicas de la CPU.

Funciones

El CP 243-2 es el maestro AS-Interface de la categoría de maestros M1, por lo que soporta todas las funciones indicadas. Gracias a ello, el AS-Interface puede controlar hasta 31 esclavos digitales, asignando direcciones repetidas (A–B).

El CP 243-2 se puede ajustar a dos modos diferentes:

- Modo estándar: acceso a los datos de E/S del esclavo AS-Interface.
- Modo extendido: llamadas del maestro (p.ej. para escribir parámetros) o petición de valores de diagnóstico.

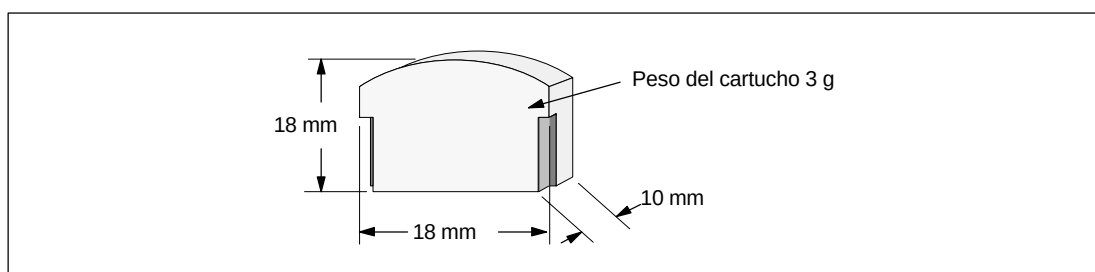
Diseñado para la industria

- Aplicaciones más flexibles y variadas de la gama SIMATIC S7-200, gracias al considerable aumento de las entradas y salidas, tanto analógicas como digitales.
- Ahorro de tiempo gracias a la posibilidad de configurar con tan sólo pulsar un botón.
- Reducción del tiempo de parada y de mantenimiento en caso de presentarse un error, gracias a los LEDs:
 - Estado del CP
 - Visualización de la disponibilidad de todos los esclavos conectados
 - Vigilancia de la tensión de la red del AS-Interface

A.15 Cartuchos opcionales

Nº de referencia	Función del cartucho
6ES7 291 8GE20 0XA0	Programa de usuario
6ES7 297 1AA20 0XA0	Reloj de tiempo real con pila
6ES7 291 8BA20 0XA0	Cartucho de pila

Cartuchos opcionales	
Almacenamiento en el cartucho de memoria	Programa, datos y configuración
Cartucho de pila (tiempo de respaldo de datos)	típ. 200 días
Precisión del cartucho de reloj	2 minutos por mes @ 25°C 7 minutos por mes @ 0°C a 55°C



Características generales	
Pila	3 V, 30 mA hora, Renata CR 1025
Tamaño	9,9 x 2,5 mm
Tipo	Litio < 0,6 g
Vida útil de almacenaje	10 años

A.16 Cable de módulo de ampliación

Nº de referencia 6ES7 290-6AA20-0XA0

Características generales	
Longitud del cable	0,8 m
Peso	25 g
Tipo de conector	Cinta de 10 pines

Instalación típica del cable de conexión de E/S de ampliación

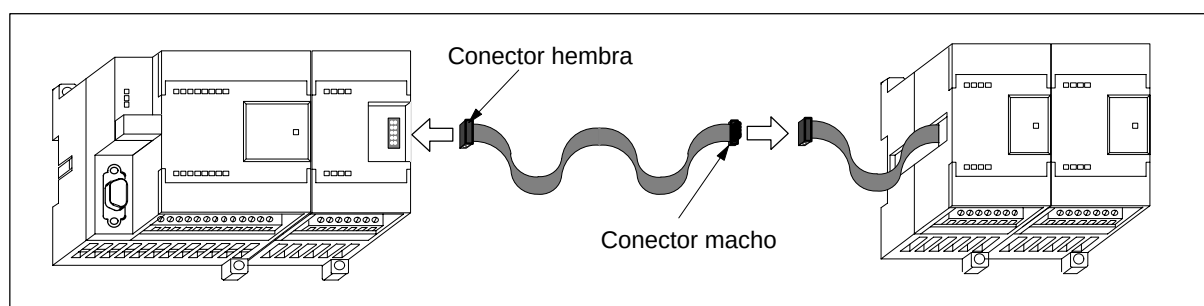


Figura A-41 Instalación típica del cable de conexión de E/S de ampliación

Nota

En una cadena de CPU/módulo de ampliación es aconsejable utilizar un solo cable de ampliación.

A.17 Cable PC/PPI

Nº de referencia 6ES7 901-3BF20-0XA0

Características generales	
Tensión de alimentación	DC 14,4 a 28,8 V
Corriente de alimentación @ 24 V alimentación nominal	máx. 50 mA RMS
Retardo al cambio de sentido: bit de arranque flanco recibido en RS-232 a bit de arranque flanco enviado a RS-485	máx. 1,2 µS
Retardo del cambio de sentido: bit de parada flanco recibido en RS-232 a transmisión inhibida en RS-485	1,4 caracteres por máx. (1,4 x 11/bit/s) = 1,6 ms @ 9600 bit/s
Retardo de propagación	máx. 4 µS, RS-485 a RS-232, máx. 1,2 µS, RS-232 a RS-485
Aislamiento	DC 500 V
Características eléctricas del enlace RS-485	
Margen de la tensión en modo común	-7 V a +12 V, 1 segundo 3 V RMS continuo
Impedancia de entrada del receptor	mín. 5,4K Ω, incluyendo cierre
Cierre/polarización	10K Ω a +5V en B, pin 3 PROFIBUS 10K Ω a GND en A, pin 8 PROFIBUS
Umbral/sensibilidad del receptor	+/- 0,2 V, 60 mV (histéresis típ.)
Tensión de salida diferencial del transmisor	mín. 2 V @ $R_L = 100 \Omega$ mín. 1,5 V @ $R_L = 54 \Omega$
Características eléctricas del enlace RS-232	
Impedancia de entrada del receptor	mín. 3K Ω
Umbral/sensibilidad del receptor	mín. 0,8 V bajo, máx. 2,4 V alto, histéresis típica: 0,5 V
Tensión de salida del transmisor	mín. +/- 5 V @ $R_L = 3K \Omega$

Dimensiones del cable PC/PPI

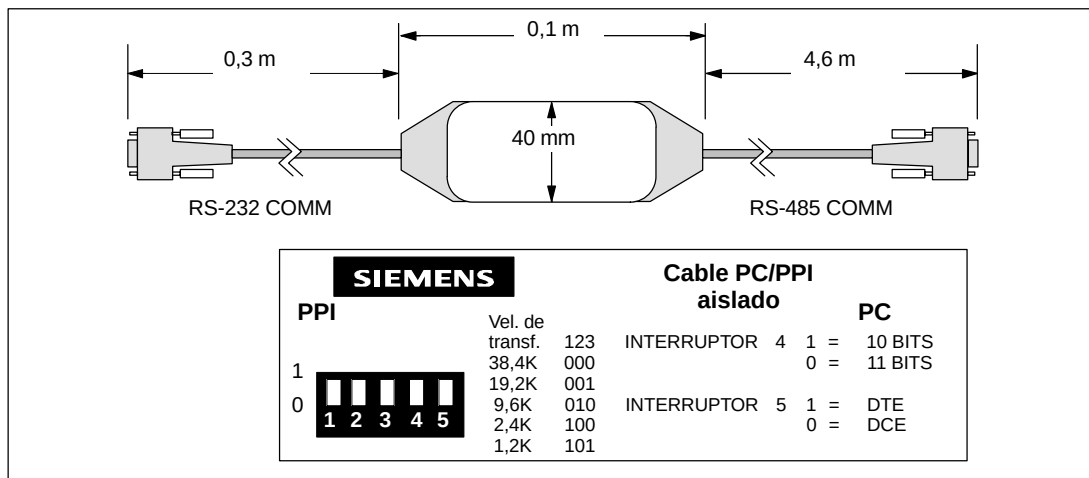


Figura A-42 Dimensiones del cable PC/PPI

Tabla A-37 Posición de los interruptores DIP en el cable PC/PPI para seleccionar la velocidad de transferencia

Velocidad de transferencia (en bit/s)	Interruptor DIP (1 = arriba)
38400	000
19200	001
9600	010
4800	011
2400	100
1200	101
600	110

Tabla A-38 Utilización de módems con el cable PC/PPI

Tipo de módem	Interruptor DIP (1 = arriba)
Módem de 11 bits	0
Módem de 10 bits	1

Tabla A-39 Asignación de pines del cable PC/PPI

Asignación de pines	Interruptor DIP (1 = arriba)
DCE	0
DTE	1

Tabla A-40 Asignación de pines para un conector de RS-485 a RS-232 DCE

Asignación de pines del conector RS-485		Asignación de pines del conector RS-232 DCE	
Nº de pin	Descripción de la señal	Nº de pin	Descripción de la señal
1	Tierra (RS-485)	1	Data Carrier Detect (DCD) (no utilizado)
2	Hilo de retorno 24 V (tierra RS-485)	2	Receive Data (RD) (salida del cable PC/PPI)
3	Señal B (RxD/TxD+)	3	Transmit Data (TD) (entrada al cable PC/PPI)
4	RTS (nivel TTL)	4	Data Terminal Ready (DTR) (no utilizado)
5	Tierra (RS-485)	5	Tierra (RS-232)
6	+5 V (con resistor en serie de 100 Ω)	6	Data Set Ready (DSR) (no utilizado)
7	Alimentación 24 V	7	Request To Send (RTS) (no utilizado)
8	Señal A (RxD/TxD-)	8	Clear To Send (CTS) (no utilizado)
9	Selección de protocolo	9	Ring Indicator (RI) (no utilizado)

Tabla A-41 Asignación de pines para un conector de RS-485 a RS-232 DTE

Asignación de pines del conector RS-485		Asignación de pines del conector RS-232 DTE ¹	
Nº de pin	Descripción de la señal	Nº de pin	Descripción de la señal
1	Tierra (RS-485)	1	Data Carrier Detect (DCD) (no utilizado)
2	Hilo de retorno 24 V (tierra RS-485)	2	Receive Data (RD) (entrada al cable PC/PPI)
3	Señal B (RxD/TxD+)	3	Transmit Data (TD) (salida del cable PC/PPI)
4	RTS (nivel TTL)	4	Data Terminal Ready (DTR) (no utilizado)
5	Tierra (RS-485)	5	Tierra (RS-232)
6	+5 V (con resistor en serie de 100 Ω)	6	Data Set Ready (DSR) (no utilizado)
7	Alimentación 24 V	7	Request To Send (RTS) (salida del cable PC/PPI)
8	Señal A (RxD/TxD-)	8	Clear To Send (CTS) (no utilizado)
9	Selección de protocolo	9	Ring Indicator (RI) (no utilizado)

¹ Para los módems se debe efectuar una conversión de conector hembra a conectar macho y de 9 pines a 25 pines.

A.18 Simuladores de entradas

Tabla A-42 Datos técnicos de los simuladores de entradas

Nº de referencia	Simulador de 8 entradas 6ES7 274-1XF00-0XA0	Simulador de 14 entradas 6ES7 274-1XH00-0XA0	Simulador de 24 entradas 6ES7 274-1XK00-0XA0
Dimensiones (l x a x p)	61 x 36 x 22 mm	91 x 36 x 22 mm	147 x 36 x 25 mm
Peso	0,02 kg	0,03 kg	0,04 kg
Entradas	8	14	24

Instalación

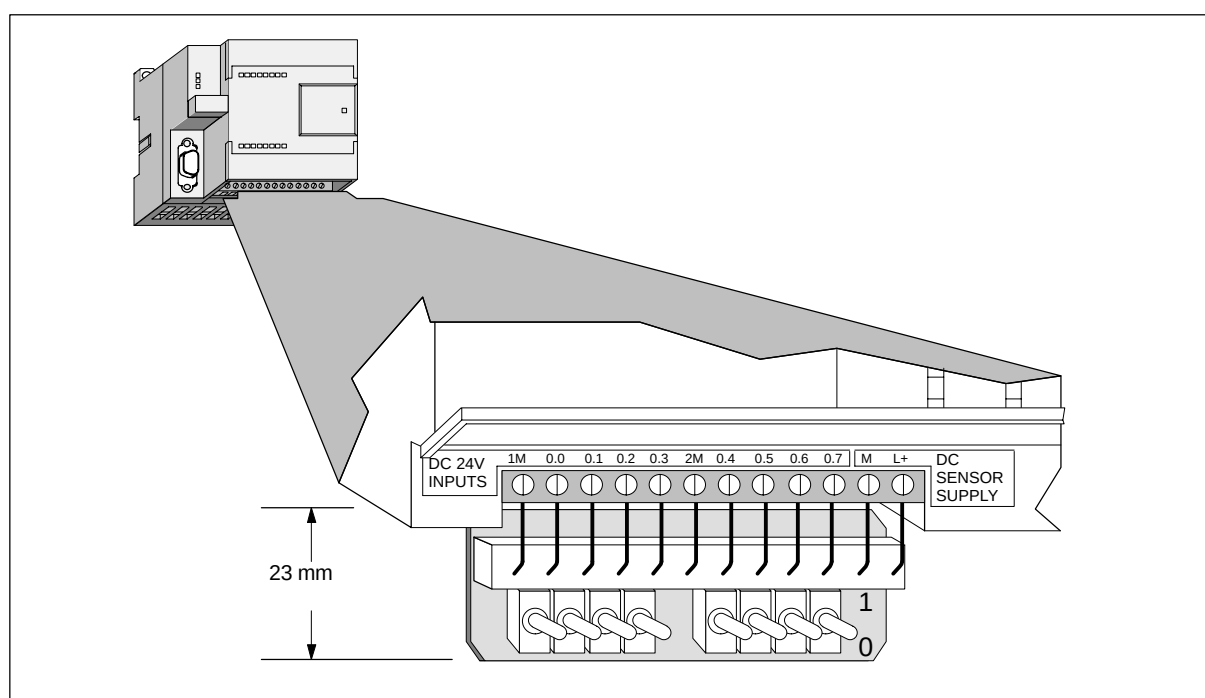


Figura A-43 Instalación del simulador de entradas



Cuidado

Estos simuladores de entradas no están aprobados para su utilización en emplazamientos peligrosos ("hazardous locations") conforme a la clase I, categoría 2 o según la clase I, sección 2. Los interruptores pueden producir chispas.

No utilice los simuladores de entradas en emplazamientos peligrosos ("hazardous locations") conforme a la clase I, categoría 2 o según la clase I, sección 2.

Códigos de error

La información relativa a los códigos de error permite identificar rápidamente los problemas que se hayan presentado en la CPU S7-200.

Índice del capítulo

Apartado	Descripción	Página
B.1	Códigos de errores fatales y mensajes	B-2
B.2	Errores de programación del tiempo de ejecución	B-3
B.3	Violación de reglas de compilación	B-4

B.1 Códigos de errores fatales y mensajes

Cuando ocurre un error fatal, la CPU detiene la ejecución del programa. Dependiendo de la gravedad del error, es posible que la CPU no pueda ejecutar todas las funciones, o incluso ninguna de ellas. El objetivo del tratamiento de errores fatales es conducir a la CPU a un estado seguro, en el que se puedan analizar y eliminar las condiciones que hayan causado el error.

Cuando la CPU detecta un error fatal:

- Cambia a modo STOP.
- Se encienden los indicadores "SF" (fallo del sistema) y "STOP".
- Se desactivan las salidas.

La CPU permanece en dicho estado hasta que se elimine la causa del error fatal. La tabla B-1 muestra una lista con las descripciones de los códigos de errores fatales que se pueden leer de la CPU.

Tabla B-1 Códigos de errores fatales y mensajes

Código de error	Descripción
0000	No hay errores fatales.
0001	Error de suma de verificación en el programa de usuario.
0002	Error de suma de verificación en el programa KOP compilado.
0003	Error de tiempo en la vigilancia del tiempo de ciclo (watchdog).
0004	Error EEPROM interno.
0005	Error EEPROM interno de suma de verificación en el programa de usuario.
0006	Error EEPROM interno de suma de verificación en los parámetros de configuración.
0007	Error EEPROM interno de suma de verificación en los datos forzados.
0008	Error EEPROM interno de suma de verificación en los valores predeterminados de la imagen de proceso de las salidas.
0009	Error EEPROM interno de suma de verificación en los datos de usuario, DB1.
000A	Error en el cartucho de memoria.
000B	Error de suma de verificación del cartucho de memoria en el programa de usuario.
000C	Error de suma de verificación del cartucho de memoria en los parámetros de configuración.
000D	Error de suma de verificación del cartucho de memoria en los datos forzados.
000E	Error de suma de verificación del cartucho de memoria en los valores predeterminados de la imagen de proceso de las salidas.
000F	Error de suma de verificación del cartucho de memoria en los datos de usuario, DB1.
0010	Error interno de software.
0011	Error en el direccionamiento indirecto del contacto de comparación.
0012	Valor en coma flotante no válido en el contacto de comparación.
0013	Cartucho de memoria vacío o programa no apto para esta CPU.

B.2 Errores de programación del tiempo de ejecución

Durante la ejecución normal del programa se pueden presentar errores no fatales (p.ej. errores de direccionamiento). La CPU genera entonces un código de error no fatal de tiempo de ejecución. La tabla B-2 muestra una lista con las descripciones de los errores no fatales.

Tabla B-2 Errores de programación del tiempo de ejecución

Código de error	Error de programación del tiempo de ejecución (no fatal)
0000	Sin error.
0001	Cuadro HSC habilitado antes de ejecutar el cuadro HDEF.
0002	Interrupción de entrada asignada a una entrada que ya está asociada a un contador rápido (conflicto).
0003	Entrada asignada a un contador rápido que ya está asociado a una interrupción de entrada u otro contador rápido (conflicto).
0004	Se ha intentado ejecutar una operación ENI, DISI, SPA o HDEF en una rutina de interrupción.
0005	Antes de finalizar el primer HSC/PLC se ha intentado ejecutar un segundo HSC con el mismo número (HSC/PLS de la rutina de interrupción en conflicto con HSC/PLC del programa principal).
0006	Error de direccionamiento indirecto.
0007	Error en datos para operación TODW (Escribir en reloj de tiempo real) o TODR (Leer del reloj de tiempo real).
0008	Excedida la profundidad máxima de anidado para subrutina de usuario.
0009	Ejecución de una operación XMT ó RCV simultáneamente con otra operación XMT o RCV en el puerto 0.
000A	Se ha intentado redefinir un HSC ejecutando otra operación HDEF para el mismo HSC.
000B	Ejecución simultánea de las operaciones XMT/RCV en el puerto 1.
000C	Falta cartucho de reloj.
000D	Intento de redefinir la salida de impulsos mientras está activada.
000E	El número de segmento del perfil PTO se ha puesto a 0.
0091	Error de margen (con información sobre direcciones): verificar las áreas de operandos.
0092	Error en el campo de contaje de una operación (con información sobre el contaje): verificar el valor máximo de contaje.
0094	Error de margen al escribir en la memoria no volátil (con información sobre direcciones).
009A	Intento de cambiar a modo Freeport en una interrupción de usuario.

B.3 Violación de reglas de compilación

Al cargar un programa en la CPU, ésta lo compila. Si durante la compilación se detecta una violación de las reglas (p.ej. una operación no válida), la CPU detendrá el proceso de carga, generando entonces un código de error no fatal (de violación de las reglas de compilación). En la tabla B-3 se describen los códigos de error generados al violarse las reglas de compilación.

Tabla B-3 Violación de reglas de compilación

Código de error	Error de compilación (no fatal)
0080	El programa es demasiado grande para que la CPU pueda generar un código ejecutable. Por favor, reduzca el tamaño del programa.
0081	Rebase negativo de la pila: dividir el segmento en varios segmentos.
0082	Operación no válida: comprobar la nemotécnica de la operación.
0083	Falta MEND u operación no admisible en el programa principal: agregar la operación MEND o borrar la operación incorrecta.
0084	Reservado
0085	Falta FOR: agregar la operación FOR o borrar la operación NEXT.
0086	Falta NEXT: agregar la operación NEXT o borrar la operación FOR.
0087	Falta meta (LBL, INT, SBR): agregar la meta apropiada.
0088	Falta RET u operación no admisible en una subrutina: agregar RET al final de la subrutina o borrar la operación incorrecta.
0089	Falta RETI u operación no admisible en una rutina de interrupción: agregar RETI al final de la rutina de interrupción o borrar la operación incorrecta.
008A	Reservado
008B	Reservado
008C	Meta doble (LBL, INT, SBR): cambiar el nombre de una de las metas.
008D	Meta no válida (LBL, INT, SBR): asegurarse de que el número admisible de metas no se haya excedido.
0090	Parámetro no válido: comprobar los parámetros admisibles para la operación.
0091	Error de margen (con información sobre direcciones): verificar las áreas de operandos.
0092	Error en el campo de contaje de una operación (con información sobre el contaje): verificar el valor máximo de contaje.
0093	Excedida la profundidad de anidado FOR/NEXT.
0095	Falta la operación LSCR (cargar SCR).
0096	Falta la operación SCRE (fin de SCR) u operación no admisible antes de la operación SCRE.
0097	Reservado
0098	Edición no válida en modo RUN.
0099	Demasiados segmentos ocultos.

Marcas especiales (SM)

Las marcas especiales (SM) ofrecen una serie de funciones de estado y control. Sirven para intercambiar informaciones entre la CPU y el programa, pudiéndose utilizar en formato de bits, bytes, palabras o palabras dobles.

SMB0: Bits de estado

Como muestra la tabla C-1, SMB0 contiene ocho bits de estado que la CPU S7-200 actualiza al final de cada ciclo.

Tabla C-1 Byte de marcas SMB0 (SM0.0 a SM0.7)

Bits de marcas	Descripción (sólo lectura)
SM0.0	Este bit siempre está activado.
SM0.1	Este bit se activa en el primer ciclo. Se utiliza p.ej. para llamar una subrutina de inicialización.
SM0.2	Este bit se activa durante un ciclo si se pierden los datos remanentes. Se puede utilizar como marca de error o como mecanismo para llamar a una secuencia especial de arranque.
SM0.3	Este bit se activa durante un ciclo cuando se pasa a modo RUN tras conectarse la alimentación. Se puede utilizar durante el tiempo de calentamiento de la instalación antes del funcionamiento normal.
SM0.4	Este bit ofrece un reloj que está activado durante 30 segundos y desactivado durante 30 segundos, siendo el tiempo de ciclo de 1 minuto. Ofrece un retardo fácil de utilizar o un tiempo de reloj de 1 minuto.
SM0.5	Este bit ofrece un reloj que está activado durante 0,5 segundos y desactivado durante 0,5 segundos, siendo el tiempo de ciclo de 1 segundo. Ofrece un reloj que está activado durante 0,5 segundos y desactivado durante 0,5 segundos, siendo el tiempo de ciclo de 1 minuto.
SM0.6	Este bit es un reloj de ciclo que está activado en un ciclo y desactivado en el ciclo siguiente. Se puede utilizar como entrada de conteo de ciclos.
SM0.7	Este bit indica la posición del selector de modos de operación (OFF = TERM; ON = RUN). Si el bit se utiliza para habilitar el modo Freeport cuando el selector esté en posición RUN, se podrá habilitar la comunicación normal con la unidad de programación cambiando el selector a TERM.

SMB1: Bits de estado

Como muestra la tabla C-2, SMB1 contiene varios indicadores de los posibles errores. Estos bits son activados y desactivados por operaciones durante el tiempo de ejecución.

Tabla C-2 Byte de marcas SMB1 (SM1.0 a SM1.7)

Bits de marcas	Descripción (sólo lectura)
SM1.0	Este bit se activa al ejecutarse ciertas operaciones si el resultado lógico es cero.
SM1.1	Este bit se activa al ejecutarse ciertas operaciones si se produce un desbordamiento o si se detecta un valor numérico no válido.
SM1.2	Este bit se activa si el resultado de una operación aritmética es negativo.
SM1.3	Este bit se activa si se intenta dividir por cero.
SM1.4	Este bit se activa si la operación Registrar valor en tabla intenta sobrepasar el límite de llenado de la tabla.
SM1.5	Este bit se activa si las operaciones FIFO o LIFO intentan leer de una tabla vacía.
SM1.6	Este bit se activa si se intenta convertir un valor no BCD en un valor binario.
SM1.7	Este bit se activa si un valor ASCII no se puede convertir en un valor hexadecimal válido.

SMB2: Búfer de recepción de caracteres en modo Freeport

SMB2 es el búfer de recepción de caracteres en modo Freeport. Como muestra la tabla C-3, cada carácter recibido en dicho modo se deposita en este búfer, fácilmente accesible desde el programa KOP.

Tabla C-3 Byte de marcas SMB2

Byte de marcas	Descripción (sólo lectura)
SMB2	Este byte contiene todos los caracteres recibidos de los puertos 0 ó 1 en modo Freeport.

SMB3: Error de paridad en modo Freeport

SMB3 se utiliza para el modo Freeport y contiene un bit de error de paridad que se activa si se detecta un error de este tipo en un carácter recibido. Como muestra la tabla C-4, SM3.0 se activa si se detecta un error de paridad. Utilice esta marca para rechazar el mensaje.

Tabla C-4 Byte de marcas SMB3 (SM3.0 a SM3.7)

Bits de marcas	Descripción (sólo lectura)
SM3.0	Error de paridad del puerto 0 ó 1 (0 = sin error; 1 = error)
SM3.1 a SM3.7	Reservados

SMB4: Desbordamiento de la cola de espera

Como muestra la tabla C-5, SMB4 contiene los bits de desbordamiento de la cola de espera, un indicador de estado que muestra las interrupciones habilitadas o inhibidas y una marca de transmisor en vacío. Los bits de desbordamiento de la cola de espera indican que las interrupciones se están presentando más rápidamente de lo que se pueden procesar, o bien que se inhibieron mediante la operación Inhibir todos los eventos de interrupción (DISI).

Tabla C-5 Byte de marcas SMB4 (SM4.0 a SM4.7)

Bits de marcas	Descripción (sólo lectura)
SM4.0 ¹	Este bit se activa si se desborda la cola de espera para las interrupciones de comunicación.
SM4.1 ¹	Este bit se activa si se desborda la cola de espera para las interrupciones de E/S.
SM4.2 ¹	Este bit se activa si se desborda la cola de espera para las interrupciones temporizadas.
SM4.3	Este bit se activa si se detecta un error de programación del tiempo de ejecución.
SM4.4	Este bit refleja el estado de habilitación de las interrupciones. Se activa cuando se habilitan las interrupciones.
SM4.5	Este bit se activa si el transmisor está en vacío (puerto 0).
SM4.6	Este bit se activa si el transmisor está en vacío (puerto 1).
SM4.7	Este bit se activa al forzarse un valor.

¹ Utilice los bits de estado 4.0, 4.1 y 4.2 sólo en rutinas de interrupción. Estos bits se desactivan cuando se vacía la cola de espera, retornando entonces el control al programa principal.

SMB5: Estado de las entradas y salidas

Como muestra la tabla C-6, SMB5 contiene los bits de estado acerca de las condiciones de error detectadas en las entradas y salidas (E/S). Estos bits indican los errores de E/S detectados.

Tabla C-6 Byte de marcas SMB5 (SM5.0 a SM5.7)

Bits de marcas	Descripción (sólo lectura)
SM5.0	Este bit se activa si se presenta algún error de E/S.
SM5.1	Este bit se activa si se han conectado demasiadas entradas y salidas digitales al bus E/S.
SM5.2	Este bit se activa si se han conectado demasiadas entradas y salidas analógicas al bus E/S.
SM5.3	Este bit se activa si se han conectado demasiados módulos de ampliación inteligentes al bus E/S.
SM5.4 a SM5.6	Reservados.
SM5.7	Este bit se activa si se presenta un fallo de bus estándar DP.

SMB6: Identificador de la CPU

Como muestra la tabla C-7, SMB6 es el identificador de la CPU. SM6.4 a SM6.7 indican el tipo de CPU. SM6.0 a SM6.3 están reservados para un uso futuro.

Tabla C-7 Byte de marcas SMB6

Bits de marcas	Descripción (sólo lectura)							
Formato	MSB 7 X X X X r r r r LSB 0 Identificador de la CPU							
SM6.4 a SM6.7	xxxx = 0000 = CPU 212/CPU 222 0010 = CPU 214/CPU 224 0110 = CPU 221 1000 = CPU 215 1001 = CPU 216/CPU 226							
SM6.0 a SM6.3	Reservados							

SMB7: Reservado

SMB7 está reservado para un uso futuro.

SMB8 a SMB21: Registro de errores e identificadores de los módulos de ampliación

SMB8 a SMB21 están organizados en pares de bytes para los módulos de ampliación 0 a 6. Como muestra la tabla C-8, el byte de número par de cada pareja de bytes constituye el registro del identificador de módulo. Este byte indica el tipo de módulo, el tipo de E/S y el número de entradas y salidas. El byte de número impar de cada pareja de bytes constituye el registro de errores del módulo. Este byte indica los errores de configuración y de alimentación de las E/S del correspondiente módulo de ampliación.

Tabla C-8 Bytes de marcas SMB8 a SMB21

Byte de marcas	Descripción (sólo lectura)															
Formato	Byte de número par: ID del módulo								Byte de número impar: Registro de errores del módulo							
	MSB				LSB				MSB				LSB			
	7				0				7				0			
	M	t	t	A	i	i	Q	Q	C	0	0	b	r	P	f	t
	M: Módulo presente				0 = presente 1 = no presente				C: Error de configuración							
	tt: 00 Módulo de ampliación no inteligente								b: Fallo de bus o error de paridad							
	01 Módulo inteligente								r: Área excedida							
	10 Reservado								P: Error alimentac. usuario							
	11 Reservado								f: Fusible fundido							
	a: Tipo de E/S				0 = digital 1 = analógica				t: Bloque de terminales suelto							
ii: 00 Sin entradas																
01 2 AI u 8 DI																
10 4 AI ó 16 DI																
11 8 AI ó 32 DI																
QQ: 00 Sin salidas																
01 2 AQ u 8 DQ																
10 4 AQ ó 16 DQ																
11 8 AQ ó 32 DQ																
SMB8	Identificador del módulo 0															
SMB9	Registro de errores del módulo 0															
SMB10	Identificador del módulo 1															
SMB11	Registro de errores del módulo 1															
SMB12	Identificador del módulo 2															
SMB13	Registro de errores del módulo 2															
SMB14	Identificador del módulo 3															
SMB15	Registro de errores del módulo 3															
SMB16	Identificador del módulo 4															
SMB17	Registro de errores del módulo 4															
SMB18	Identificador del módulo 5															
SMB19	Registro de errores del módulo 5															
SMB20	Identificador del módulo 6															
SMB21	Registro de errores del módulo 6															

M: Módulo presente

0 = presente

1 = no presente

tt: 00 Módulo de ampliación no inteligente

01 Módulo inteligente

10 Reservado

11 Reservado

a: Tipo de E/S

0 = digital

1 = analógica

ii: 00 Sin entradas

01 2 AI u 8 DI

10 4 AI ó 16 DI

11 8 AI ó 32 DI

QQ: 00 Sin salidas

01 2 AQ u 8 DQ

10 4 AQ ó 16 DQ

11 8 AQ ó 32 DQ

C: Error de configuración

b: Fallo de bus o error de paridad

r: Área excedida

P: Error alimentac. usuario

f: Fusible fundido

t: Bloque de terminales suelto

SMW22 a SMW26: Tiempos de ciclo

Como muestra la tabla C-9, las marcas especiales SMW22, SMW24 y SMW26 informan sobre el tiempo de ciclo. Permiten leer el último tiempo de ciclo, así como los tiempos de ciclo mínimo y máximo.

Tabla C-9 Palabras de marcas SMW22 a SMW26

Palabra de marcas	Descripción (sólo lectura)
SMW22	Esta palabra indica el tiempo del último ciclo.
SMW24	Esta palabra indica el tiempo de ciclo mínimo.
SMW26	Esta palabra indica el tiempo de ciclo máximo.

SMB28 y SMB29: Potenciómetros analógicos

Como muestra la tabla C-10, SMB28 almacena el valor digital que representa la posición del potenciómetro analógico 0. SMB29 almacena el valor digital que representa la posición del potenciómetro analógico 1.

Tabla C-10 Bytes de marcas SMB28 y SMB29

Byte de marcas	Descripción (sólo lectura)
SMB28	Este byte almacena el valor leído del potenciómetro analógico 0. El valor se actualiza una vez por ciclo en STOP/RUN.
SMB29	Este byte almacena el valor leído del potenciómetro analógico 1. El valor se actualiza una vez por ciclo en STOP/RUN.

SMB30 y SMB130: Registros de control del modo Freeport

SMB30 y SMB130 controlan la comunicación Freeport en los puertos 0 y 1, respectivamente. SMB30 y SMB130 son marcas de lectura y escritura. Como muestra la tabla C-11, estos bytes configuran la comunicación Freeport en los respectivos puertos y permiten seleccionar si se debe asistir el modo Freeport o el protocolo de sistema.

Tabla C-11 Bytes de marcas SMB30

Puerto 0	Puerto 1	Descripción
Formato de SMB30	Formato de SMB130	MSB 7 LSB 0 p p d b b b m m Byte de control del modo Freeport
SM30.6 y SM30.7	SM130.6 y SM130.7	pp: Selección de paridad 00 = sin paridad 01 = paridad par 10 = sin paridad 11 = paridad impar
SM30.5	SM130.5	d: Bits por carácter 0 = 8 bits por carácter 1 = 7 bits por carácter
SM30.2 a SM30.4	SM130.2 a SM130.4	bbb: Velocidad de transferencia 000 = 38.400 bits/s 001 = 19.200 bit/s 010 = 9.600 bit/s 011 = 4.800 bit/s 100 = 2.400 bit/s 101 = 1.200 bit/s 110 = 600 bit/s 111 = 300 bit/s
SM30.0 y SM30.1	SM130.0 y SM130.1	mm: Selección de protocolo 00 = Protocolo de interface punto a punto (PPI/modo esclavo) 01 = Protocolo Freeport 10 = PPI/modo maestro 11 = Reservado (estándar: PPI/modo esclavo) Nota: Si se selecciona el código mm = 10 (maestro PPI), la CPU pasará a ser una estación maestra en la red, permitiendo que se ejecuten las operaciones NETR y NETW. Los bits 2 a 7 se ignoran en el modo PPI.

SMB31 y SMW32: Control de escritura en la memoria no volátil (EEPROM)

Un valor almacenado en la memoria de variables (memoria V) se puede guardar en la memoria no volátil (EEPROM) mediante el programa. A tal efecto, cargue en SMW32 la dirección que desee guardar. Cargue después SMB31 con el comando de guardar el valor. Una vez cargado el comando, no modifique el valor en la memoria V hasta que la CPU haya desactivado SM31.7, indicando que ha finalizado la memorización.

La CPU comprueba al final de cada ciclo si se debe guardar algún valor en la memoria no volátil. En caso afirmativo, el valor indicado se almacenará allí.

Como muestra la tabla C-12, SMB31 define el tamaño de los datos a guardar en la memoria no volátil, incorporando también el comando que inicia la memorización. SMW32 almacena la dirección inicial en la memoria V de los datos a guardar en la memoria no volátil.

Tabla C-12 Byte de marcas SMB31 y palabra de marcas SMW32

Byte de marcas	Descripción
Formato	SMB31: Comando de software <pre> MSB LSB 7 0 +---+---+---+---+---+---+---+---+ c 0 0 0 0 0 s s +---+---+---+---+---+---+---+ </pre> SMW32: Dirección en la memoria V <pre> MSB LSB 15 0 +-----+ Dirección en la memoria V +-----+ </pre>
SM31.0 y SM31.1	ss: Tamaño del valor a guardar 00 = bit 01 = byte 10 = palabra 11 = palabra doble
SM31.7	c: Guardar en la memoria no volátil (EEPROM) 0 = No hay petición de guardar. 1 = El programa de usuario solicita que la CPU guarde datos en la memoria no volátil. La CPU desactiva este bit después de cada almacenamiento.
SMW32	La dirección en la memoria V (memoria de variables) de los datos a guardar se almacena en SMW32. Este valor se indica como desplazamiento (offset) de V0. Al ejecutarse la memorización, el valor contenido en esta dirección de la memoria V se escribe en la correspondiente dirección V en la memoria no volátil (EEPROM).

SMB34 y SMB35: Duración de las interrupciones temporizadas

Como muestra la tabla C-13, SMB34 y SMB35 especifican la duración de las interrupciones temporizadas 0 y 1, respectivamente. Los valores de esta duración se pueden indicar (en incrementos de 1 ms) de 1 ms a 255 ms. La CPU captará el valor cuando la correspondiente interrupción temporizada sea asociada a una rutina de interrupción. Para cambiar su duración es preciso reasociar la interrupción temporizada a la misma rutina de interrupción, o bien a una diferente. Para terminar el evento de interrupción temporizada hay que desasociarlo.

Tabla C-13 Bytes de marcas SMB34 y SMB35

Byte de marcas	Descripción
SMB34	Este byte indica la duración (en incrementos de 1 ms, de 1 ms a 255 ms) de la interrupción temporizada 0.
SMB35	Este byte indica la duración (en incrementos de 1 ms, de 1 ms a 255 ms) de la interrupción temporizada 1.

SMB36 a SMB65: Bytes de programación de los contadores rápidos HSC0, HSC1 y HSC2

Como muestra la tabla C-14, los bytes de marcas SMB36 a SMB65 se utilizan para supervisar y controlar el funcionamiento de los contadores rápidos HSC0, HSC1 y HSC2.

Tabla C-14 Bytes de marcas SMB36 a SMB65

Byte de marcas	Descripción
SM36.0 a SM36.4	Reservados
SM36.5	Bit de estado del sentido de conteo actual de HSC0: 1 = contar adelante
SM36.6	El valor actual de HSC0 es igual al bit de estado del valor predeterminado: 1 = igual
SM36.7	El valor actual de HSC0 es mayor que el bit de estado del valor predeterminado: 1 = mayor que
SM37.0	Bit de control para nivel de actividad de la entrada de puesta a 0: 0 = actividad alta, 1 = actividad baja
SM37.1	Reservado
SM37.2	Velocidad de conteo de los contadores A/B: 0 = velocidad cuádruple; 1 = velocidad simple
SM37.3	Bit de control del sentido de conteo de HSC0: 1 = adelante
SM37.4	Actualizar el sentido de conteo de HSC0: 1 = actualizar el sentido de conteo
SM37.5	Actualizar el valor predeterminado de HSC0: 1 = escribir nuevo valor predeterminado en HSC0
SM37.6	Actualizar el valor actual de HSC0: 1 = escribir nuevo valor actual en HSC0
SM37.7	Bit de habilitación de HSC0: 1 = habilitar
SMB38 SMB39 SMB40 SMB41	Nuevo valor actual de HSC0. SMB38 es el byte más significativo y SMB41 el byte menos significativo.
SMB42 SMB43 SMB44 SMB45	Nuevo valor predeterminado de HSC0. SMB42 es el byte más significativo y SMB45 el byte menos significativo.
SM46.0 a SM46.4	Reservados
SM46.5	Bit de estado del sentido de conteo actual de HSC1: 1 = contar adelante
SM46.6	El valor actual de HSC1 es igual al bit de estado del valor predeterminado: 1 = igual
SM46.7	El valor actual de HSC1 es mayor que el bit de estado del valor predeterminado: 1 = mayor que
SM47.0	Bit de control de nivel de actividad para puesta a 0 de HSC1: 0 = actividad alta, 1 = actividad baja
SM47.1	Bit de control de nivel de actividad para arranque de HSC1: 0 = actividad alta, 1 = actividad baja
SM47.2	Selección de velocidad de conteo de HSC1: 0 = cuádruple, 1 = simple
SM47.3	Bit de control del sentido de conteo de HSC1: 1 = adelante
SM47.4	Actualizar el sentido de conteo de HSC1: 1 = actualizar el sentido de conteo

Tabla C-14 Bytes de marcas SMB36 a SMB65 (continuación)

Byte de marcas	Descripción
SM47.5	Actualizar el valor predeterminado de HSC1: 1 = escribir nuevo valor predeterminado en HSC1
SM47.6	Actualizar el valor actual de HSC1: 1 = escribir nuevo valor actual en HSC1
SM47.7	Bit de habilitación de HSC1: 1 = habilitar
SMB48 SMB49 SMB50 SMB51	Nuevo valor actual de HSC1. SMB48 es el byte más significativo y SMB51 el byte menos significativo.
SMB52 a SMB55	Nuevo valor predeterminado de HSC1. SMB52 es el byte más significativo y SMB55 el byte menos significativo.
SM56.0 a SM56.4	Reservados
SM56.5	Bit de estado del sentido de conteo actual de HSC2: 1 = contar adelante
SM56.6	El valor actual de HSC2 es igual al bit de estado del valor predeterminado: 1 = igual
SM56.7	El valor actual de HSC2 es mayor que el bit de estado del valor predeterminado: 1 = mayor que
SM57.0	Bit de control de nivel de actividad para puesta a 0 de HSC2: 0 = actividad alta, 1 = actividad baja
SM57.1	Bit de control de nivel de actividad para arranque de HSC2: 0 = actividad alta, 1 = actividad baja
SM57.2	Selección de velocidad de conteo de HSC2: 0 = cuádruple, 1 = simple
SM57.3	Bit de control del sentido de conteo de HSC2: 1 = adelante
SM57.4	Actualizar el sentido de conteo de HSC2: 1 = actualizar el sentido de conteo
SM57.5	Actualizar el valor predeterminado de HSC2: 1 = escribir nuevo valor predeterminado en HSC2
SM57.6	Actualizar el valor actual de HSC2: 1 = escribir nuevo valor actual en HSC2
SM57.7	Bit de habilitación de HSC2: 1 = habilitar
SMB58 SMB59 SMB60 SMB61	Nuevo valor actual de HSC2. SMB58 es el byte más significativo y SMB61 es el byte menos significativo.
SMB62 SMB63 SMB64 SMB65	Nuevo valor predeterminado de HSC2. SMB62 es el byte más significativo y SMB65 es el byte menos significativo.

SMB66 a SMB85: Funciones PTO/PWM

Como muestra la tabla C-15, SMB66 a SMB85 se utilizan para supervisar y controlar las funciones de modulación de salida de impulsos y de ancho de impulsos de las operaciones PTO/PWM. Consulte las informaciones sobre la operación de salida de impulsos en el capítulo 9 para obtener una descripción detallada de estas marcas.

Tabla C-15 Bytes de marcas SMB66 a SMB85

Byte de marcas	Descripción
SM66.0 a SM66.3	Reservados
SM66.4	Interrupción anormal del perfil PTO0: 0 = sin error, 1 = interrupción debida a un error de cálculo delta
SM66.5	Interrupción anormal del perfil PTO0: 0 = no causada por el usuario, 1 = causada por el usuario
SM66.6	Desbordamiento de pipeline PTO0 (el sistema lo pone a 0 al utilizarse perfiles externos; en caso contrario, el usuario lo deberá poner a 0): 0 = sin desbordamiento; 1 = desbordamiento.
SM66.7	Bit de PTO0 en vacío: 0 = PTO en ejecución, 1 = PTO en vacío
SM67.0	Actualizar el tiempo de ciclo PTO0/PWM0: 1 = escribir nuevo valor del tiempo de ciclo
SM67.1	Actualizar el ancho de impulsos de PWM0: 1 = escribir nuevo ancho de impulsos
SM67.2	Actualizar el valor de conteo de impulsos de PTO0: 1 = escribir nuevo valor de conteo de impulsos
SM67.3	Base de tiempo PTO0/PWM0: 0 = 1 μ s/ciclo, 1 = 1 ms/ciclo
SM67.4	Actualización de PWM0: 0 = actualización asíncrona, 1 = actualización síncrona
SM67.5	Función PTO0: 0 = función monosegmento (tiempo de ciclo y conteo de impulsos almacenados en la memoria SM), 1 = función multisegmento (tabla de perfiles almacenada en la memoria V)
SM67.6	Elegir modo PTO0/PWM0 0 = PTO, 1 = PWM
SM67.7	Bit de habilitación de PTO0/PWM0: 1 = habilitar
SMB68 SMB69	Valor del tiempo de ciclo PTO0/PWM0 (2 a 65.535 unidades de la base de tiempo); SMB68 es el byte más significativo y SMB69 el byte menos significativo.
SMB70 SMB71	Valor del ancho de impulsos PWM0 (0 a 65.535 unidades de la base de tiempo); SMB70 es el byte más significativo y SMB71 el byte menos significativo.
SMB72 SMB73 SMB74 SMB75	Valor del conteo de impulsos PTO0 (1 a $2^{32}-1$); SMB72 es el byte más significativo y SMB75 el byte menos significativo.
SM76.0 a SM76.3	Reservados
SM76.4	Interrupción anormal del perfil PTO1: 0 = sin error, 1 = interrupción debida a un error de cálculo delta
SM76.5	Interrupción anormal del perfil PTO1: 0 = no causada por el usuario, 1 = causada por el usuario

Tabla C-15 Bytes de marcas SMB66 a SMB85 (continuación)

Byte de marcas	Descripción
SM76.6	Desbordamiento de pipeline PTO1 (el sistema lo pone a 0 al utilizarse perfiles externos; en caso contrario, el usuario lo deberá poner a 0): 0 = sin desbordamiento; 1 = desbordamiento.
SM76.7	Bit de PTO1 en vacío: 0 = PTO en ejecución, 1 = PTO en vacío
SM77.0	Actualizar el tiempo de ciclo PTO1/PWM1: 1 = escribir nuevo valor del tiempo de ciclo
SM77.1	Actualizar el ancho de impulsos de PWM1: 1 = escribir nuevo ancho de impulsos
SM77.2	Actualizar el valor de conteo de impulsos de PTO1: 1 = escribir nuevo valor de conteo de impulsos
SM77.3	Base de tiempo PTO1/PWM1: 0 = 1 µs/ciclo, 1 = 1 ms/ciclo
SM77.4	Actualización de PWM1: 0 = actualización asíncrona, 1 = actualización síncrona
SM77.5	Función PTO1: 0 = función monosegmento (tiempo de ciclo y conteo de impulsos almacenados en la memoria SM), 1 = función multisegmento (tabla de perfiles almacenada en la memoria V)
SM77.6	Elegir modo PTO1/PWM1 0 = PTO, 1 = PWM
SM77.7	Bit de habilitación de PTO1/PWM1: 1 = habilitar
SMB78 SMB79	Valor de tiempo de ciclo PTO1/PWM1 (2 a 65.535 unidades de la base de tiempo); SMB78 es el byte más significativo y SMB79 el byte menos significativo.
SMB80 SMB81	Valor del ancho de impulsos PWM1 (0 a 65.535 unidades de la base de tiempo); SMB80 es el byte más significativo y SMB81 el byte menos significativo.
SMB82 SMB83 SMB84 SMB85	Valor del conteo de impulsos PTO1 (1 a $2^{32}-1$); SMB82 es el byte más significativo y SMB85 el byte menos significativo.

SMB86 a SMB94 y SMB186 a SMB194: Control de recepción de mensajes

Como muestra la tabla C-16, los bytes de marcas SMB86 a SMB94 y SMB186 a SMB194 se utilizan para controlar y leer el estado de la operación Recibir mensaje.

Tabla C-16 Marcas especiales SMB86 a SMB94 y SMB186 a SMB194

Puerto 0	Puerto 1	Descripción								
SMB86	SMB186	MSB 7 <table><tr><td>n</td><td>r</td><td>e</td><td>0</td><td>0</td><td>t</td><td>c</td><td>p</td></tr></table> LSB 0 Byte de estado de recepción de mensajes n: 1 = El usuario ha inhibido la recepción de mensajes r: 1 = Se ha finalizado la recepción de mensajes: error en parámetros de entrada o falta condición inicial o final. e: 1 = Carácter final recibido. t: 1 = Se ha finalizado la recepción de mensajes: ha transcurrido la temporización. c: 1 = Se ha finalizado la recepción de mensajes: se ha excedido el número máximo de caracteres. p: 1 = Se ha finalizado la recepción de mensajes debido a un error de paridad.	n	r	e	0	0	t	c	p
n	r	e	0	0	t	c	p			

Tabla C-16 Marcas especiales SMB86 a SMB94 y SMB186 a SMB194 (continuación)

Puerto 0	Puerto 1	Descripción																						
SMB87	SMB187	MSB 7 LSB 0 <table><tr><td>en</td><td>sc</td><td>ec</td><td>il</td><td>c/m</td><td>tmr</td><td>bk</td><td>0</td></tr></table> Byte de control de recepción de mensajes en: 0 = Inhibida la función de recibir mensajes. 1 = Habilitada la función de recibir mensajes. El bit para habilitar/inhibir la recepción de mensajes se comprueba cada vez que se ejecuta la operación RCV. sc: 0 = Ignorar SMB88 o SMB188. 1 = Utilizar el valor de SMB88 o de SMB188 para detectar el comienzo del mensaje. ec: 0 = Ignorar SMB89 o SMB189. 1 = Utilizar el valor de SMB89 o de SMB189 para detectar el final del mensaje. il: 0 = Ignorar SMB90 o SMB190. 1 = Utilizar el valor de SMW90 para detectar una condición de inactividad. c/m: 0 = Utilizar el temporizador como temporizador entre caracteres. 1= Utilizar el temporizador como temporizador de mensajes. tmr: 0 = Ignorar SMB92 o SMB192. 1 = Finalizar la recepción si se excede el período de tiempo indicado en SMW92 o SMW192. bk: 0 = Ignorar condiciones BREAK; 1 = Utilizar condición BREAK como comienzo de la detección de mensajes. Las marcas del byte de control de interrupción de mensajes se utilizan para definir los criterios con los que se identifica el mensaje. Se definen los criterios tanto de comienzo como de final del mensaje. Para determinar el comienzo de un mensaje, uno de los dos juegos de criterios de comienzo de mensaje combinados lógicamente mediante Y deberán ser verdaderos y deberán ocurrir en secuencia (inactividad seguida de un comienzo de carácter o condición BREAK seguida de un comienzo de carácter). Para determinar el fin de un mensaje, los criterios habilitados se combinan mediante O. A continuación se indican las ecuaciones de comienzo y de final: Comienzo del mensaje = $il * sc + bk * sc$ Final del mensaje = $ec + tmr + \text{contaje máximo de caracteres alcanzado}$ Programar los criterios de comienzo de mensaje para: <table><tr><td>1. Detección de inactividad:</td><td>$il=1, sc=0, bk=0, SMW90>0$</td></tr><tr><td>2. Detección de carácter inicial:</td><td>$il=0, sc=1, bk=0, SMW90$ no es relevante</td></tr><tr><td>3. Detección BREAK:</td><td>$il=0, sc=1, bk=1, SMW90$ no es relevante</td></tr><tr><td>4. Cualquier respuesta a una petición:</td><td>$il=1, sc=0, bk=0, SMW90=0$ (El temporizador de mensajes se puede utilizar para terminar la recepción si no hay respuesta).</td></tr><tr><td>5. BREAK y carácter inicial:</td><td>$il=0, sc=1, bk=1, SMW90$ no es relevante</td></tr><tr><td>6. Inactividad y carácter inicial:</td><td>$il=1, sc=1, bk=0, SMW90 >0$</td></tr><tr><td>7. Inactividad y carácter inicial (no válido):</td><td>$il=1, sc=1, bk=0, SMW90=0$</td></tr></table> Nota: La recepción se finalizará automáticamente si ocurre un error de desbordamiento o de paridad (si se han habilitado).	en	sc	ec	il	c/m	tmr	bk	0	1. Detección de inactividad:	$il=1, sc=0, bk=0, SMW90>0$	2. Detección de carácter inicial:	$il=0, sc=1, bk=0, SMW90$ no es relevante	3. Detección BREAK:	$il=0, sc=1, bk=1, SMW90$ no es relevante	4. Cualquier respuesta a una petición:	$il=1, sc=0, bk=0, SMW90=0$ (El temporizador de mensajes se puede utilizar para terminar la recepción si no hay respuesta).	5. BREAK y carácter inicial:	$il=0, sc=1, bk=1, SMW90$ no es relevante	6. Inactividad y carácter inicial:	$il=1, sc=1, bk=0, SMW90 >0$	7. Inactividad y carácter inicial (no válido):	$il=1, sc=1, bk=0, SMW90=0$
en	sc	ec	il	c/m	tmr	bk	0																	
1. Detección de inactividad:	$il=1, sc=0, bk=0, SMW90>0$																							
2. Detección de carácter inicial:	$il=0, sc=1, bk=0, SMW90$ no es relevante																							
3. Detección BREAK:	$il=0, sc=1, bk=1, SMW90$ no es relevante																							
4. Cualquier respuesta a una petición:	$il=1, sc=0, bk=0, SMW90=0$ (El temporizador de mensajes se puede utilizar para terminar la recepción si no hay respuesta).																							
5. BREAK y carácter inicial:	$il=0, sc=1, bk=1, SMW90$ no es relevante																							
6. Inactividad y carácter inicial:	$il=1, sc=1, bk=0, SMW90 >0$																							
7. Inactividad y carácter inicial (no válido):	$il=1, sc=1, bk=0, SMW90=0$																							
SMB88	SMB188	Carácter de comienzo del mensaje.																						

Tabla C-16 Marcas especiales SMB86 a SMB94 y SMB186 a SMB194 (continuación)

Puerto 0	Puerto 1	Descripción
SMB89	SMB189	Carácter de fin del mensaje.
SMB90 SMB91	SMB190 SMB191	Tiempo de línea de inactividad en milisegundos. El primer carácter recibido una vez transcurrido el tiempo de línea de inactividad es el comienzo del nuevo mensaje. SM90 (o SM190) es el byte más significativo y SM91 (ó SM191) es el byte menos significativo.
SMB92 SMB93	SMB192 SMB193	Vigilancia del temporizador entre caracteres/de mensajes en milisegundos. Si se excede el tiempo, se finaliza la recepción de mensajes. SM92 (o SM192) es el byte más significativo y SM93 (o SM193) es el byte menos significativo.
SMB94	SMB194	Número máximo de caracteres a recibir (1 a 255 bytes). Nota: Este margen debe ajustarse al tamaño máximo esperado para el búfer, incluso si no se utiliza la finalización de mensajes por el conteo de caracteres.

SMB98 y SMB99

Como muestra la tabla C-17, SMB98 y SMB99 indican el número de errores en el bus de ampliación.

Tabla C-17 Bytes de marcas SMB98 y SMB99

Byte de marcas	Descripción
SMB98 SMB99	Esta dirección se incrementa cada vez que se detecta un error de paridad en el bus de ampliación. Se borra durante el encendido y cuando el usuario la pone a 0. SMB98 es el byte más significativo.

SMB130 a SMB165: Bytes de programación de los contadores rápidos HSC3, HSC4 y HSC5

Como muestra la tabla C-18, los bytes de marcas SMB130 a SMB165 se utilizan para supervisar y controlar el funcionamiento de los contadores rápidos HSC3, HSC4 y HSC5.

Tabla C-18 Bytes de marcas SMB130 a SMB165

Byte de marcas	Descripción
SMB131 a SMB135	Reservados
SM136.0 a SM136.4	Reservados
SM136.5	Bit de estado del sentido de conteo actual de HSC3: 1 = contar adelante
SM136.6	El valor actual de HSC3 es igual al bit de estado del valor predeterminado: 1 = igual
SM136.7	El valor actual de HSC3 es mayor que el bit de estado del valor predeterminado: 1 = mayor que
SM137.0 a SM137.2	Reservados
SM137.3	Bit de control del sentido de conteo de HSC3: 1 = adelante

Tabla C-18 Bytes de marcas SMB130 a SMB165 (continuación)

Byte de marcas	Descripción
SM137.4	Actualizar sentido de conteo de HSC3: 1 = actualizar el sentido de conteo
SM137.5	Actualizar valor predeterminado de HSC3: 1 = escribir nuevo valor predeterminado en HSC3
SM137.6	Actualizar valor actual de HSC3: 1 = escribir nuevo valor actual en HSC3
SM137.7	Bit de habilitación de HSC3: 1 = habilitar
SM138 a SM141	Nuevo valor actual de HSC3. SMB38 es el byte más significativo y SMB41 el byte menos significativo.
SM142 a SM145	Nuevo valor predeterminado de HSC3. SMB142 es el bit más significativo y SMB145 el bit menos significativo.
SM146.0 a SM146.4	Reservados
SM146.5	Bit de estado del sentido de conteo actual de HSC4: 1 = contar adelante
SM146.7	El valor actual de HSC4 es mayor que el bit de estado del valor predeterminado: 1 = mayor que
SM147.0	Bit de control para nivel de actividad de la entrada de puesta a 0: 0 = actividad alta, 1 = actividad baja
SM147.1	Reservado
SM147.2	Velocidad de conteo de los contadores A/B: 0 = velocidad cuádruple; 1 = velocidad simple
SM147.3	Bit de control del sentido de conteo de HSC4: 1 = adelante
SM147.4	Actualizar sentido de conteo de HSC4: 1 = actualizar el sentido de conteo
SM147.5	Actualizar valor predeterminado de HSC4: 1 = escribir nuevo valor predeterminado en HSC4
SM147.6	Actualizar valor actual de HSC4: 1 = escribir nuevo valor actual en HSC4
SM147.7	Bit de habilitación de HSC4: 1 = habilitar
SMB148 a SMB151	Nuevo valor actual de HSC4. SMB148 es el byte más significativo y SMB151 el byte menos significativo.
SMB152 a SMB155	Nuevo valor predeterminado de HSC4. SMB152 es el byte más significativo y SMB155 el byte menos significativo.
SM156.0 a SM156.4	Reservados
SM156.5	Bit de estado del sentido de conteo actual de HSC5: 1 = contar adelante
SM156.6	El valor actual de HSC5 es igual al bit de estado del valor predeterminado: 1 = igual
SM156.7	El valor actual de HSC5 es mayor que el bit de estado del valor predeterminado: 1 = mayor que
SM157.0 a SM157.2	Reservados
SM157.3	Bit de control del sentido de conteo de HSC5: 1 = adelante
SM157.4	Actualizar sentido de conteo de HSC5: 1 = actualizar el sentido de conteo
SM157.5	Actualizar valor predeterminado de HSC5: 1 = escribir nuevo valor predeterminado en HSC5
SM157.6	Actualizar valor actual de HSC5: 1 = escribir nuevo valor actual en HSC5
SM157.7	Bit de habilitación de HSC5: 1 = habilitar

Tabla C-18 Bytes de marcas SMB130 a SMB165 (continuación)

Byte de marcas	Descripción
SMB158 a SMB161	Nuevo valor actual de HSC5. SMB158 es el bit más significativo y SMB161 el bit menos significativo.
SMB162 a SMB165	Nuevo valor predeterminado de HSC5. SMB162 es el bit más significativo y SMB165 el bit menos significativo.

SMB166 a SMB194: Tabla de definición de perfiles PTO, PT1

Como muestra la tabla C-19, los bytes de marcas SMB166 a SMB194 se utilizan para mostrar el número de pasos del perfil activo y la dirección de la tabla de perfiles en la memoria V.

Tabla C-19 Bytes de marcas SMB166 a SMB194

Byte de marcas	Descripción
SMB166	Número actual del paso del perfil activo de PTO0.
SMB167	Reservado
SMB168 a SMB169	Dirección en la memoria V de la tabla del perfil PTO0 indicada como offset desde V0. SMB168 es el byte más significativo del offset de dirección.
SMB170 a SMB175	Reservados
SMB176	Número actual del paso del perfil activo de PTO1.
SMB177	Reservado
SMB178 a SMB179	Dirección en la memoria V de la tabla del perfil PTO1 indicada como offset desde V0. SMB178 es el byte más significativo del offset de dirección.
SMB180 a SMB194	Reservados

SMB200 a SMB299: Estado del módulo inteligente

Las marcas especiales SMB200 a SMB299 están reservadas para la información de estado de los módulos de ampliación inteligentes, como p.ej. el módulo EM 277 PROFIBUS-DP. SMB200 a SMB249 están reservadas para el primer módulo de ampliación inteligente integrado en el sistema (el más próximo a la CPU), en tanto que SMB250 a SMB299 están reservadas para el segundo. En los datos técnicos del Anexo A se indica cómo los módulos utilizan las marcas especiales SMB200 a SMB299.

Eliminar errores

Tabla D-1 Eliminar errores

Problema	Causas posibles	Solución
Las salidas han dejado de funcionar.	- El dispositivo controlado ha causado una sobretensión que ha averiado la salida. - Error en el programa de usuario. - Cableado suelto o incorrecto. - Carga excesiva. - Salidas forzadas.	- Al conectar la CPU a una carga inductiva (p.ej. un motor o un relé) es preciso utilizar un circuito de supresión adecuado (consulte el apartado 2.4). - Corregir el programa de usuario. - Comprobar y corregir el cableado. - Comprobar la carga en la E/S. - Comprobar las E/S forzadas.
El diodo "SF" (System Fault) de la CPU se enciende.	La lista siguiente describe las causas más frecuentes: - Error en el programa de usuario. 0003 Error de tiempo de vigilancia (watchdog). 0011 Direccionamiento indirecto 0012 Valor en coma flotante no válido. - Interferencia eléctrica. 0001 a 0009 - Componente averiado. 0001 a 0010	Lea el código de error fatal y consulte el apartado B.1: - En caso de un error de programación, consulte la descripción de las operaciones FOR, NEXT, JMP, LBL, así como de las operaciones de comparación. - En caso de una interferencia eléctrica: - Consulte las reglas de cableado en el apartado 2.3. Es muy importante que el armario eléctrico esté conectado correctamente a tierra y que el cableado de alta y baja tensión no se conduzcan en paralelo. - Conecte a tierra el terminal M de la alimentación de sensores de DC 24 V.
Fuente de alimentación averiada.	Sobretensión en los cables conectados al equipo.	Conecte un dispositivo para medir la magnitud y la duración de las puntas de sobretensión. Conforme a dichas informaciones, incorpore un dispositivo apropiado de supresión de sobretensiones. Para obtener informaciones más detalladas acerca de la instalación del cableado de campo, consulte el apartado 2.3.
Interferencias eléctricas	- Puesta a tierra incorrecta. - Conducción del cableado en el armario eléctrico. - Filtros de entrada configurados para una velocidad demasiado rápida.	Consulte las reglas de cableado en el apartado 2.3. Es muy importante que el armario eléctrico esté conectado correctamente a tierra y que el cableado de alta y baja tensión no se conduzcan en paralelo. Conecte a tierra el terminal M de la alimentación de sensores de DC 24 V. Incrementar en el bloque de datos el retardo del filtro de entrada (v. apt. 5.2.)

Tabla D-1 Eliminar errores(continued)

Problema	Causas posibles	Solución
Red de comunicación averiada al conectar un dispositivo externo. (Están averiados el puerto del PC, el puerto de la CPU o el cable PC/PPI).	El cable de comunicación puede convertirse en una ruta de corrientes indeseadas si los dispositivos que no tengan separación galvánica (tales como las CPUs, los PCs u otros dispositivos) y que estén conectados al cable, no comparten un mismo hilo de referencia en el circuito. Las corrientes indeseadas pueden causar errores de comunicación o averiar los circuitos.	• Consulte el apartado 2.3 donde se indican las reglas de cableado y el capítulo 7 para obtener más información acerca de la comunicación en redes. • Sustituya el cable PC/PPI. • Utilice un repetidor de RS-485 a RS-485 con separación galvánica al conectar máquinas que no tengan una referencia eléctrica común.
Problemas de comunicación en STEP 7-Micro/WIN 32		Para obtener informaciones más detalladas acerca de la comunicación en redes, consulte el capítulo 7.
Tratamiento de errores		Para obtener informaciones más detalladas acerca de los códigos de error, consulte el Anexo B.

Números de referencia

CPUs	Nº de referencia
CPU 221 DC/DC/DC 6 entradas/4 salidas	6ES7 211-0AA21-0XB0
CPU 221 AC/DC/relé 6 entradas/4 salidas de relé	6ES7 211-0BA21-0XB0
CPU 222 DC/DC/DC 8 entradas/6 salidas	6ES7 212-1AB21-0XB0
CPU 222 AC/DC/relé 8 entradas/6 salidas de relé	6ES7 212-1BB21-0XB0
CPU 224 DC/DC/DC 14 entradas/10 salidas	6ES7 214-1AD21-0XB0
CPU 224 AC/DC/relé 14 entradas/10 salidas de relé	6ES7 214-1BD21-0XB0
CPU 226 DC/DC/DC 24 entradas/16 salidas	6ES7 216-2AD21-0XB0
CPU 226 AC/DC/relé 24 entradas/16 salidas de relé	6ES7 216-2BD21-0XB0

Módulos de ampliación	Nº de referencia
EM221 8 entradas digitales x DC 24 V	6ES7 221-1BF20-0XA0
EM222 8 salidas digitales x DC 24 V	6ES7 222-1BF20-0XA0
EM 222 8 salidas digitales x relé	6ES7 222-1HF20-0XA0
EM223 4 entradas digitales/4 salidas digitales x DC 24 V	6ES7 223-1BF20-0XA0
EM223 4 entradas digitales/4 salidas de relé x DC 24 V	6ES7 223-1HF20-0XA0
EM223 8 entradas digitales/8 salidas digitales x DC 24 V	6ES7 223-1BH20-0XA0
EM223 8 entradas digitales/8 salidas de relé x DC 24 V	6ES7 223-1PH20-0XA0
EM223 16 entradas digitales/16 salidas digitales x DC 24 V	6ES7 223-1BL20-0XA0
EM223 16 entradas digitales/16 salidas de relé x DC 24 V	6ES7 223-1PL20-0XA0
EM 231 4 entradas analógicas x DC 24 V	6ES7 231-0HC20-0XA0
EM 232 2 salidas analógicas x DC 24 V	6ES7 232-0HB20-0XA0
EM 235 4 entradas analógicas/1 salida analógica x DC 24 V	6ES7 235-0KD20-0XA0
EM 231 RTD, 2 entradas analógicas x DC 24 V	6ES7 231-7PB20-0XA0
EM 231 Termopar, 4 entradas analógicas x DC 24 V	6ES7 231-7PD20-0XA0
EM 277 PROFIBUS-DP	6ES7 277-0AA20-0XA0
Procesador de comunicaciones CP 243-2	6ES7 243-2AX00-0XA0

Cartuchos y cables	Nº de referencia
Cartucho de memoria MC 291, 32K x 8 EEPROM	6ES7 291-8GE20-0XA0
Cartucho, reloj/calendario con pila CC 292, CPU 22x	6ES7 297-1AA20-0XA0
Cartucho de pila BC 293, CPU 22x	6ES7 291-8BA20-0XA0
Cable para módulos de ampliación, 0,8 metros, CPU 22x/EM	6ES7 290-6AA20-0XA0
Cable, PC/PPI, aislado, 5 interruptores DIP, 5 metros	6ES7 901-3BF20-0XA0

Software de programación	Nº de referencia
STEP 7-Micro/WIN 32 (V3.1), licencia única (disquete)	6ES7 810-2BA01-0YX0
STEP 7-Micro/WIN 32 (V3.1), licencia de actualización (disquete)	6ES7 810-2BA01-0YX3
STEP 7-Micro/WIN 32 (V3.1), licencia única (CD-ROM)	6ES7 810-2BC01-0YX0
STEP 7-Micro/WIN 32 (V3.1), licencia de actualización (CD-ROM)	6ES7 810-2BC01-0YX3
STEP 7-Micro/WIN 32 Toolbox, licencia única (CD-ROM)	6ES7 810-2PC01-0YX0

Tarjetas de comunicación	Nº de referencia
Tarjeta MPI: Short AT ISA	6ES7793-2AA01-0AA0
CP 5411: Short AT ISA	6GK1 41-1AA00
CP 5511: PCMCIA, tipo II	6GK1 551-1AA00
CP 5611: Tarjeta PCI (versión 3.0 o superior)	6GK1 561-1AA00

Manuales	Nº de referencia
Manual del usuario del visualizador de textos TD 200	6ES7 272-0AA20-8BA0
S7-200 Point-to-Point Interface Communication Manual (inglés/alemán)	6ES7 298-8GA00-8XH0
CP 243-2 Communications Processor Manual (inglés)	6GK7 243-2AX00-8BA0
Sistema de automatización S7-200, Manual del sistema (alemán)	6ES7 298-8FA21-8AH0
Sistema de automatización S7-200, Manual del sistema (inglés)	6ES7 298-8FA21-8BH0
Sistema de automatización S7-200, Manual del sistema (francés)	6ES7 298-8FA21-8CH0
Sistema de automatización S7-200, Manual del sistema (español)	6ES7 298-8FA21-8DH0
Sistema de automatización S7-200, Manual del sistema (italiano)	6ES7 298-8FA21-8EH0

Cables, conectores de bus y repetidores	Nº de referencia
Cable MPI	6ES7 901-0BF00-0AA0
Cable para redes PROFIBUS	6XVI 830-0AH10
Conector de bus (con conector de interface de programación), salida vertical del cable	6ES7 972-0BB11-0XA0
Conector de bus (sin conector de interface de programación), salida vertical del cable	6ES7 972-0BA11-0XA0
Conector de bus RS-485 con salida del cable a 35° (sin conector para el puerto de programación)	6ES7 972-0BA40-0XA0
Conector de bus RS-485 con salida del cable a 35° (con conector para el puerto de programación)	6ES7 972-0BB40-0XA0
Bloque de conectores CPU 22x/EM, 7 terminales, extraíble	6ES7 292-1AD20-0AA0
Bloque de conectores CPU 22x/EM, 12 terminales, extraíble	6ES7 292-1AE20-0AA0

Cables, conectores de bus y repetidores	Nº de referencia
Bloque de conectores CPU 22x/EM, 14 terminales, extraíble	6ES7 292-1AF20-0AA0
Bloque de conectores CPU 22x/EM, 18 terminales, extraíble	6ES7 292-1AG20-0AA0
Repetidor RS-485 IP 20, aislado	6ES7 972-0AA00-0XA0

Visualizadores de textos, paneles de operador y paneles táctiles	Nº de referencia
Visualizador de textos TD 200	6ES7 272-0AA20-0YA0
Panel de operador OP3	6AV3 503-1DB10
Panel de operador OP7	6AV3 607-1JC20-0AX1
Panel de operador OP17	6AV3 617-1JC20-0AX1
Panel táctil TP070	6AV6 545-0AA15-2AX0
Panel táctil TP170A	6AV6 545-0BA15-2AX0

Varios	Nº de referencia
Topes para raíles DIN	6ES5 728-8MA11
Bloque de 12 bornes de conexión para cableado de campo (CPU 221, CPU 222), paquete de 10	6ES7 290-2AA00-0XA0
Kit de tapas de repuesto (contiene 4 de las siguientes tapas): tapas de bloques para 7, 12, 14, 18, 2x12, 2x14 terminales; tapa de acceso a la CPU, tapa de acceso a módulos de ampliación	6ES7 291-3AX20-0XA0
Simulador de 8 entradas	6ES7 274 1XF00-0XA0
Simulador de 14 entradas	6ES7 274 1XF00-0XA0
Simulador de 24 entradas	6ES7 274 1XK00-0XA0

Tiempos de ejecución de las operaciones AWL

F

Impacto de la circulación de la corriente sobre el tiempo de ejecución

El cálculo del tiempo de ejecución básico de una operación AWL (v. tabla F-4) indica el tiempo necesario para ejecutar la lógica o la función de la operación cuando se aplica corriente (estando activado (puesto a "1") ese valor). En algunas operaciones, la ejecución de la función depende de si se aplica corriente o no. La CPU sólo ejecuta la función si se aplica corriente a la operación, estando entonces activado (puesto a "1") el primer valor de la pila. Si no se aplica corriente a la operación (estando desactivado (puesto a "0") el primer valor de la pila), se deberá utilizar un tiempo de ejecución sin circulación de corriente para calcular dicho tiempo. La tabla F-1 muestra el tiempo de ejecución de una operación AWL sin circulación de corriente (estando desactivado (puesto a "0") el primer valor de la pila) para cada una de las CPUs S7-700.

Tabla F-1 Tiempo de ejecución de operaciones sin circulación de corriente

Operación sin circulación de corriente	CPU S7-200
Todas las operaciones AWL	3 µs

Impacto del direccionamiento indirecto sobre el tiempo de ejecución

El cálculo del tiempo de ejecución básico de una operación AWL (v. tabla F-4) indica el tiempo necesario para ejecutar la operación si los operandos o las constantes se direccionan directamente. Si en el programa se usan direcciones indirectas, el tiempo de ejecución de cada operando direccionado indirectamente se incrementa como muestra la tabla F-2.

Tabla F-2 Tiempo adicional para el direccionamiento indirecto

Operación de direccionamiento indirecto	CPU S7-200
Cada operando direccionado indirectamente	22 µs

Tiempos de ejecución

El acceso a ciertas áreas de memoria, tales como AI (entradas analógicas), AQ (salidas analógicas), L (memoria local) y AC (acumuladores), prolonga también el tiempo de ejecución. La tabla F-3 muestra un factor que se debe añadir al tiempo de ejecución básico de cada operando que acceda a dichas áreas.

Tabla F-3 Factor a añadir al tiempo de ejecución para acceder a algunas áreas de memoria

Área de memoria	CPU S7-200
Entradas analógicas (AI)	149 µs
Salidas analógicas (AQ)	73 µs
Memoria local (L)	5,4 µs
Acumuladores (AC)	4,4 µs

Tiempos de ejecución básicos de las operaciones AWL

La tabla F-4 indica los tiempos de ejecución básicos de las operaciones AWL para cada una de las CPUs S7-200.

Tabla F-4 Tiempos de ejecución de las operaciones AWL (en µs)

Operación	Descripción	S7-200 CPU (en µs)
=	Tiempo de ejecución básico: I L SM, T, C, V, S, Q, M	0,37 19,2 1,8
+D	Tiempo de ejecución básico	55
−D	Tiempo de ejecución básico	55
*D	Tiempo de ejecución básico	92
/D	Tiempo de ejecución básico	376
+I	Tiempo de ejecución básico	46
−I	Tiempo de ejecución básico	47
*I	Tiempo de ejecución básico	71
/I	Tiempo de ejecución básico	115
=I	Tiempo de ejecución básico: Salida integrada Salida en un módulo de ampliación	29 39
+R	Tiempo de ejecución básico Tiempo de ejecución máximo	110 163
−R	Tiempo de ejecución básico Tiempo de ejecución máximo	113 166
*R	Tiempo de ejecución básico Tiempo de ejecución máximo	100 130
/R	Tiempo de ejecución básico Tiempo de ejecución máximo	300 360

Tabla F-4 Tiempos de ejecución de las operaciones AWL (en μ s) (continuación)

Operación	Descripción	S7-200 CPU (en μ s)
A	Tiempo de ejecución básico: I L SM, T, C, V, S, Q, M	0,37 10,8 1,1
AB < =, =, >=, >, <, <>	Tiempo de ejecución básico	35
AD < =, =, >=, >, <, <>	Tiempo de ejecución básico	53
AI	Tiempo de ejecución básico: Entrada integrada Entrada en un módulo de ampliación	27 35
ALD	Tiempo de ejecución básico	0,37
AN	Tiempo de ejecución básico: I L SM, T, C, V, S, Q, M	0,37 10,8 1,1
ANDB	Tiempo de ejecución básico	37
ANDD	Tiempo de ejecución básico	55
ANDW	Tiempo de ejecución básico	48
ANI	Tiempo de ejecución básico: Entrada integrada Entrada en un módulo de ampliación	27 35
AR < =, =, >=, >, <, <>	Tiempo de ejecución básico	54
ATCH	Tiempo de ejecución básico	20
ATH	Total = tiempo básico + (longitud * ML) Tiempo de ejecución básico (longitud constante) Tiempo de ejecución básico (longitud variable) Multiplicador de longitud (ML)	41 55 20
ATT	Tiempo de ejecución básico	70
AW < =, =, >=, >, <, <>	Tiempo de ejecución básico	45
BCDI	Tiempo de ejecución básico	66
BIR	Tiempo de ejecución básico: Entradas integradas Entradas en un módulo de ampliación	43 51
BIW	Tiempo de ejecución básico: Entradas integradas Entradas en un módulo de ampliación	42 52
BMB	Total = tiempo básico + (longitud * ML) Tiempo de ejecución básico (longitud constante) Tiempo de ejecución básico (longitud variable) Multiplicador de longitud (ML)	21 51 11
BMD	Total = tiempo básico + (longitud * ML) Tiempo de ejecución básico (longitud constante) Tiempo de ejecución básico (longitud variable) Multiplicador de longitud (ML)	21 51 20
BMW	Total = tiempo básico + (longitud * ML) Tiempo de ejecución básico (longitud constante) Tiempo de ejecución básico (longitud variable) Multiplicador de longitud (ML)	21 51 16

Tabla F-4 Tiempos de ejecución de las operaciones AWL (en μ s) (continuación)

Operación	Descripción	S7-200 CPU (en μ s)
CALL	Sin parámetros: Tiempo de ejecución	15
	Con parámetros: Tiempo total de ejecución = Tiempo básico + Σ (tiempo de gestión de los operandos de entrada)	
	Tiempo de ejecución básico	32
	Tiempo de gestión de los operandos de entrada (bit)	23
	Tiempo de gestión de los operandos de entrada (byte)	21
	Tiempo de gestión de los operandos de entrada (palabra)	24
	Tiempo de gestión de los operandos de entrada (palabra doble)	27
COS	Tiempo de ejecución básico	1525
	Tiempo de ejecución máximo	1800
CRET	Tiempo total de ejecución = Tiempo básico + Σ (tiempo de gestión de los operandos de salida)	
	Tiempo de ejecución básico	13
	Tiempo de gestión de los operandos de salida (bit)	21
	Tiempo de gestión de los operandos de salida (byte)	14
	Tiempo de gestión de los operandos de salida (palabra)	18
	Tiempo de gestión de los operandos de salida (palabra doble)	20
CRETI	Tiempo de ejecución básico	23
CTD	Tiempo de ejecución básico al producirse un flanco en la entrada de conteo	48
	Tiempo de ejecución básico en otro caso	36
CTU	Tiempo de ejecución básico al producirse un flanco en la entrada de conteo	53
	Tiempo de ejecución básico en otro caso	35
CTUD	Tiempo de ejecución básico al producirse un flanco en la entrada de conteo	64
	Tiempo de ejecución básico en otro caso	45
DECB	Tiempo de ejecución básico	30
DECD	Tiempo de ejecución básico	42
DECO	Tiempo de ejecución básico	36
DECW	Tiempo de ejecución básico	37
DISI	Tiempo de ejecución básico	18
DIV	Tiempo de ejecución básico	119
DTCH	Tiempo de ejecución básico	18
DTR	Tiempo de ejecución básico	60
	Tiempo de ejecución máximo	70
ED	Tiempo de ejecución básico	15
ENCO	Tiempo de ejecución mínimo	39
	Tiempo de ejecución máximo	43
FIN	Tiempo de ejecución básico	0,9
ENI	Tiempo de ejecución básico	53
EU	Tiempo de ejecución básico	15
EXP	Tiempo de ejecución básico	1170
	Tiempo de ejecución máximo	1375

Tabla F-4 Tiempos de ejecución de las operaciones AWL (en μ s) (continuación)

Operación	Descripción	S7-200 CPU (en μ s)
FIFO	Total = tiempo básico + (longitud * ML) Tiempo de ejecución básico Multiplicador de longitud (ML)	70 14
FILL	Total = tiempo básico + (longitud * ML) Tiempo de ejecución básico (longitud constante) Tiempo de ejecución básico (longitud variable) Multiplicador de longitud (ML)	29 50 7
FND <, =, >, <>	Total = tiempo básico + (longitud * ML) Tiempo de ejecución básico Multiplicador de longitud (ML)	85 12
FOR	Total = tiempo básico + (número de repeticiones * ML) Tiempo de ejecución básico Multiplicador de lazos (ML)	64 50
GPA	Tiempo de ejecución básico	31
HDEF	Tiempo de ejecución básico	35
HSC	Tiempo de ejecución básico	37
HTA	Total = tiempo básico + (longitud * ML) Tiempo de ejecución básico (longitud constante) Tiempo de ejecución básico (longitud variable) Multiplicador de longitud (ML)	38 48 11
IBCD	Tiempo de ejecución básico	114
INCB	Tiempo de ejecución básico	29
INCD	Tiempo de ejecución básico	42
INCW	Tiempo de ejecución básico	37
INT	Tiempo de ejecución típico con 1 interrupción	47
INVB	Tiempo de ejecución básico	31
INVD	Tiempo de ejecución básico	42
INVW	Tiempo de ejecución básico	38
JMP	Tiempo de ejecución básico	0,9
LBL	Tiempo de ejecución básico	0,37
LD	Tiempo de ejecución básico: I L SM, T, C, V, S, Q, M SM0.0	0,37 10,9 1,1 0,37
LDB <=, =, >=, >, <, <>	Tiempo de ejecución básico	35
LDD <=, =, >=, >, <, <>	Tiempo de ejecución básico	52
LDI	Tiempo de ejecución básico: Entrada integrada Entrada en un módulo de ampliación	26 34
LDN	Tiempo de ejecución básico: I L SM, T, C, V, S, Q, M	0,37 10,9 1,1

Tabla F-4 Tiempos de ejecución de las operaciones AWL (en μ s) (continuación)

Operación	Descripción	S7-200 CPU (en μ s)
LDNI	Tiempo de ejecución básico: Entrada integrada Entrada en un módulo de ampliación	26 34
LDR<=, =, >=, >, <, <>	Tiempo de ejecución básico	55
LDS	Tiempo de ejecución básico	0,37
LDW <=, =, >=, >, <, <>	Tiempo de ejecución básico	42
LIFO	Tiempo de ejecución básico	70
LN	Tiempo de ejecución básico Tiempo de ejecución máximo	1130 1275
LPP	Tiempo de ejecución básico	0,37
LPS	Tiempo de ejecución básico	0,37
LRD	Tiempo de ejecución básico	0,37
LSCR	Tiempo de ejecución básico	12
MEND	Tiempo de ejecución básico	0.5
MOVB	Tiempo de ejecución básico	29
MOVD	Tiempo de ejecución básico	38
MOVR	Tiempo de ejecución básico	38
MOVW	Tiempo de ejecución básico	34
MUL	Tiempo de ejecución básico	70
NEXT	Tiempo de ejecución básico	0
NETR	Tiempo de ejecución básico	179
NETW	Total = tiempo básico + (longitud * ML) Tiempo de ejecución básico Multiplicador de longitud (ML)	175 8
NOP	Tiempo de ejecución básico	0,37
NOT	Tiempo de ejecución básico	0,37
O	Tiempo de ejecución básico: I L SM, T, C, V, S, Q, M	0,37 10,8 1,1
OB <=, =, >=, >, <, <>	Tiempo de ejecución básico	35
OD <=, =, >=, >, <, <>	Tiempo de ejecución básico	53
OI	Tiempo de ejecución básico: Entrada integrada Entrada en un módulo de ampliación	27 35
OLD	Tiempo de ejecución básico	0,37
ON	Tiempo de ejecución básico: I L SM, T, C, V, S, Q, M	0,37 10,8 1,1
ONI	Tiempo de ejecución básico: Entrada integrada Entrada en un módulo de ampliación	27 35
OR<=, =, >=, >, <, <>	Tiempo de ejecución básico	55

Tabla F-4 Tiempos de ejecución de las operaciones AWL (en μs) (continuación)

Operación	Descripción	S7-200 CPU (en μs)
ORB	Tiempo de ejecución básico	37
ORD	Tiempo de ejecución básico	55
ORW	Tiempo de ejecución básico	48
OW < =, =, >=, >, <, <>	Tiempo de ejecución básico	45
PID	Tiempo de ejecución básico Sumador para recalcular ($K_c * T_s/T_i$) y ($K_c * T_d/T_s$) antes del cálculo PID. Se recalcula si K_c , T_s , T_i , o T_s se han modificado desde la anterior ejecución de esta operación o si se ha cambiado a modo automático.	750 1000
PLS	Tiempo de ejecución básico: PWM PTO monosegmento PTO multisegmento	57 67 92
R	Para longitud=1 y especificado como constante (p.ej. R V0.2,1) Tiempo de ejecución del operando = C Tiempo de ejecución del operando = T Tiempo de ejecución de todos los demás operandos En otro caso, Total=tiempo de ejecución básico +(longitud * ML) Tiempo de ejecución básico del operando = C, T Tiempo de ejecución básico de todos los demás operandos Multiplicador de longitud (ML) del operando = C Multiplicador de longitud (ML) del operando = T Multiplicador de longitud (ML) de todos los demás operandos Si la longitud se almacena en una variable en lugar de especificarse en calidad de constante, incrementar el tiempo de ejecución básico sumando:	17 24 5 19 28 8,6 16,5 0,9 29
RCV	Tiempo de ejecución básico	80
RET	Tiempo de ejecución total = tiempo básico + Σ (tiempo de gestión de los operandos de salida) Tiempo de ejecución básico Tiempo de gestión de los operandos de salida (bit) Tiempo de gestión de los operandos de salida (byte) Tiempo de gestión de los operandos de salida (palabra) Tiempo de gestión de los operandos de salida (palabra doble)	13 21 14 18 20
RETI	Tiempo de ejecución básico	23
RI	Total = tiempo básico + (longitud * ML) Tiempo de ejecución básico Multiplicador de longitud (salida integrada) Multiplicador de longitud (salida en módulo de ampliación) Si la longitud se almacena en una variable, en vez de ser una constante, incrementar el tiempo de ejecución básico sumando:	18 22 32 30
RLB	Total = tiempo básico + (longitud * ML) Tiempo de ejecución básico Multiplicador de longitud (ML)	42 0,6

Tabla F-4 Tiempos de ejecución de las operaciones AWL (en μ s) (continuación)

Operación	Descripción	S7-200 CPU (en μ s)
RLD	Total = tiempo básico + (longitud * ML) Tiempo de ejecución básico Multiplicador de longitud (ML)	52 2,5
RLW	Total = tiempo básico + (longitud * ML) Tiempo de ejecución básico Multiplicador de longitud (ML)	49 1,7
RRB	Total = tiempo básico + (longitud * ML) Tiempo de ejecución básico Multiplicador de longitud (ML)	42 0,6
RRD	Total = tiempo básico + (longitud * ML) Tiempo de ejecución básico Multiplicador de longitud (ML)	52 2,5
RRW	Total = tiempo básico + (longitud * ML) Tiempo de ejecución básico Multiplicador de longitud (ML)	49 1,7
S	Para longitud = 1 y especificada como constante (p.ej. S V0.2, 1) Tiempo de ejecución En otro caso, Total=tiempo de ejecución básico+(longitud * ML) Tiempo de ejecución básico de todos los demás operandos Multiplicador de longitud (ML) para todos los demás operandos Si la longitud se almacena en una variable, en vez de ser una constante, incrementar el tiempo de ejecución básico sumando:	5 27 0,9 29
SBR	Tiempo de ejecución básico	0
SCRE	Tiempo de ejecución básico	0,37
SCRT	Tiempo de ejecución básico	17
SEG	Tiempo de ejecución básico	30
SHRB	Total = tiempo básico + (longitud * ML1) + ((longitud/8) * ML2) Tiempo de ejecución básico (longitud constante) Tiempo de ejecución básico (longitud variable) Multiplicador de longitud 1 (ML1) Multiplicador de longitud 2 (ML2)	76 84 1.6 4
SI	Total = tiempo básico + (longitud * ML) Tiempo de ejecución básico Multiplicador de longitud (ML) (salida integrada) Multiplicador de longitud (ML) (salida en módulo de ampliación) Si la longitud se almacena en una variable, en vez de ser una constante, incrementar el tiempo de ejecución básico sumando:	18 22 32 30
SIN	Tiempo de ejecución básico Tiempo de ejecución máximo	1525 1800
SLB	Total = tiempo básico + (longitud * ML) Tiempo de ejecución básico Multiplicador de longitud (ML)	43 0,7

Tabla F-4 Tiempos de ejecución de las operaciones AWL (en μ s) (continuación)

Operación	Descripción	S7-200 CPU (en μ s)
SLD	Total = tiempo básico + (longitud * ML) Tiempo de ejecución básico Multiplicador de longitud (ML)	53 2,6
SLW	Total = tiempo básico + (longitud * ML) Tiempo de ejecución básico Multiplicador de longitud (ML)	51 1,3
SPA	Tiempo de ejecución básico	243
SQRT	Tiempo de ejecución básico Tiempo de ejecución máximo	725 830
SRB	Total = tiempo básico + (longitud * ML) Tiempo de ejecución básico Multiplicador de longitud (ML)	43 0,7
SRD	Total = tiempo básico + (longitud * ML) Tiempo de ejecución básico Multiplicador de longitud (ML)	53 2,6
SRW	Total = tiempo básico + (longitud * ML) Tiempo de ejecución básico Multiplicador de longitud (ML)	51 1,3
STOP	Tiempo de ejecución básico	16
SWAP	Tiempo de ejecución básico	32
TAN	Tiempo de ejecución básico Tiempo de ejecución máximo	1825 2100
TODR	Tiempo de ejecución básico	2400
TODW	Tiempo de ejecución básico	1600
TOF	Tiempo de ejecución básico	64
TON	Tiempo de ejecución básico	64
TONR	Tiempo de ejecución básico	56
TRUNC	Tiempo de ejecución básico Tiempo de ejecución máximo	103 178
WDR	Tiempo de ejecución básico	16
XMT	Tiempo de ejecución básico	78
XORB	Tiempo de ejecución básico	37
XORD	Tiempo de ejecución básico	55
XORW	Tiempo de ejecución básico	48

Breviario del S7-200

Este anexo contiene informaciones sobre los siguientes temas:

- Marcas especiales
- Descripción de los eventos de interrupción
- Resumen de las áreas de memoria y funciones de las CPUs S7-200
- Contadores rápidos HSC0, HSC1, HSC2, HSC3, HSC4, HSC5
- Operaciones S7-200

Tabla G-1 Marcas especiales

Marcas especiales			
SM0.0	Siempre ON	SM1.0	Resultado de la operación = 0
SM0.1	Primer ciclo	SM1.1	Desbordamiento o valor no válido
SM0.2	Datos remanentes perdidos	SM1.2	Resultado negativo
SM0.3	Alimentación	SM1.3	División por 0
SM0.4	30 s OFF / 30 s ON	SM1.4	Desbordamiento tabla
SM0.5	0,5 s OFF / 0,5 s ON	SM1.5	Tabla vacía
SM0.6	OFF 1 ciclo / ON 1 ciclo	SM1.6	Error de conversión de BCD a binario
SM0.7	Selector en posición RUN	SM1.7	Error de conversión de ASCII a hexa-decimal

Tabla G-2 Prioridades de los eventos de interrupción

Nº de evento	Descripción de la interrupción	Grupo de prioridad	Prioridad en el grupo
8	Puerto 0: Recibir carácter	Comunicación (más alta)	0
9	Puerto 0: Transmisión finalizada		0
23	Puerto 0: Recepción de mensajes finalizada		0
24	Puerto 1: Recepción de mensajes finalizada		1
25	Puerto 1: Recibir carácter		1
26	Puerto 1: Transmisión finalizada		1
19	PTO 0 interrupción completa	Digital (media)	0
20	PTO 1 interrupción completa		1
0	Flanco positivo, I0.0		2
2	Flanco positivo, I0.1		3
4	Flanco positivo, I0.2		4
6	Flanco positivo, I0.3		5
1	Flanco negativo, I0.0		6
3	Flanco negativo, I0.1		7
5	Flanco negativo, I0.2		8
7	Flanco negativo, I0.3		9
12	HSC0 CV=PV (valor actual = valor predeterminado)		10
27	HSC0 cambio de sentido		11
28	HSC0, puesto a 0 externamente		12
13	HSC1 CV=PV (valor actual = valor predeterminado)		13
14	HSC1, cambio de sentido		14
15	HSC1 puesto a 0 externamente		15
16	HSC2 CV=PV		16
17	HSC2 cambio de sentido		17
18	HSC2 puesto a 0 externamente		18
32	HSC3 CV=PV (valor actual = valor predeterminado)		19
29	HSC4 CV=PV (valor actual = valor predeterminado)		20
30	HSC4 cambio de sentido		21
31	HSC4, puesto a 0 externamente		22
33	HSC5 CV=PV (valor actual = valor predeterminado)		23
10	Interrupción temporizada 0	Temporizada (más baja)	0
11	Interrupción temporizada 1		1
21	Interrupción temporizador T32 CT = PT		2
22	Interrupción temporizador T96 CT = PT		3

Tabla G-3 Resumen de las áreas de memoria y funciones de las CPUs S7-200

Descripción	Margen				Accesible como ...			
	CPU 221	CPU 222	CPU 224	CPU 226	Bit	Byte	Palabra	Palabra doble
Tamaño del programa de usuario	2K palabras	2K palabras	4K palabras	4K palabras				
Tamaño de los datos de usuario	1K palabras	1K palabras	2,5K palabras	2,5K palabras				
Imagen del proceso de las entradas	I0.0 a I15.7	I0.0 a I15.7	I0.0 a I15.7	I0.0 a I15.7	Ix.y	IBx	IWx	IDx
Imagen del proceso de las salidas	Q0.0 a Q15.7	Q0.0 a Q15.7	Q0.0 a Q15.7	Q0.0 a Q15.7	Qx.y	QBx	QWx	QDx
Entradas analógicas (sólo lectura)	—	AIW0 a AIW30	AIW0 a AIW62	AIW0 a AIW62			AIWx	
Salidas analógicas (sólo escritura)	—	AQW0 a AQW30	AQW0 a AQW62	AQW0 a AQW62			AQWx	
Memoria de variables (V) ¹	VB0.0 a VB2047.7	VB0.0 a VB2047.7	VB0.0 a VB5119.7	VB0.0 a VB5119.7	Vx.y	VBx	VWx	VDx
Memoria local (L) ²	LB0.0 a LB63.7	LB0.0 a LB63.7	LB0.0 a LB63.7	LB0.0 a LB63.7	Lx.y	LBx	LWx	LDx
Área de marcas (M)	M0.0 a M31.7	M0.0 a M31.7	M0.0 a M31.7	M0.0 a M31.7	Mx.y	MBx	MWx	MDx
Marcas especiales (SM) Sólo lectura	SM0.0 a SM179.7 SM0.0 a SM29.7	SM0.0 a SM299.7 SM0.0 a SM29.7	SM0.0 a SM299.7 SM0.0 a SM29.7	SM0.0 a SM299.7 SM0.0 a SM29.7	SMx.y	SMBx	SMWx	SMDx
Temporizadores Ret. a la conexión mem. 1 ms Ret. a la conexión mem. 10 ms Ret. a la conexión mem. 100 ms Retardo a la con./descon. 1 ms Retardo a la con./descon. 10 ms Retardo a la con./descon. 100 ms	256 (T0 a T255) T0, T64 T1 a T4, T65 a T68 T5 a T31, T69 a T95 T32, T96 T33 a T36, T97 a T100 T37 a T63, T101 a T255	256 (T0 a T255) T0, T64 T1 a T4, T65 a T68 T5 a T31, T69 a T95 T32, T96 T33 a T36, T97 a T100 T37 a T63, T101 a T255	256 (T0 a T255) T0, T64 T1 a T4, T65 a T68 T5 a T31, T69 a T95 T32, T96 T33 a T36, T97 a T100 T37 a T63, T101 a T255	256 (T0 a T255) T0, T64 T1 a T4, T65 a T68 T5 a T31, T69 a T95 T32, T96 T33 a T36, T97 a T100 T37 a T63, T101 a T255	Tx		Tx	
Contadores	C0 a C255	C0 a C255	C0 a C255	C0 a C255	Cx		Cx	
Contadores rápidos	HC0, HC3, HC4, HC5	HC0, HC3, HC4, HC5	HC0 a HC5	HC0 a HC5				HCx
Relés de control secuencial (S)	S0.0 a S31.7	S0.0 a S31.7	S0.0 a S31.7	S0.0 a S31.7	Sx.y	SBx	SWx	SDx
Acumuladores	AC0 a AC3	AC0 a AC3	AC0 a AC3	AC0 a AC3		ACx	ACx	ACx
Salto a metas	0 a 255	0 a 255	0 a 255	0 a 255				
Llamadas a subrutinas	0 a 63	0 a 63	0 a 63	0 a 63				
Rutinas de interrupción	0 a 127	0 a 127	0 a 127	0 a 127				
Lazos PID	0 a 7	0 a 7	0 a 7	0 a 7				
Puerto	Puerto 0	Puerto 0	Puerto 0	Puerto 0, Puerto 1				

1

Todo el contenido de la memoria V se puede guardar en la memoria no volátil.

2

STEP 7–Micro/WIN 32 (versión 3.0 o posterior) reserva LB60 a LB63.

Tabla G-4 Contadores rápidos HSC0, HSC3, HSC4 y HSC5

Modo	HSC0			HSC3	HSC4			HSC5
	I0.0	I0.1	I0.2	I0.1	I0.3	I0.4	I0.5	I0.4
0	Reloj			Reloj	Reloj			Reloj
1	Reloj		Puesta a 0		Reloj		Puesta a 0	
2								
3	Reloj	Sentido			Reloj	Sentido		
4	Reloj	Sentido	Puesta a 0		Reloj	Sentido	Puesta a 0	
5								
6	Reloj adelante	Reloj atrás			Reloj adelante	Reloj atrás		
7	Reloj adelante	Reloj atrás	Puesta a 0		Reloj adelante	Reloj atrás	Puesta a 0	
8								
9	Fase A	Fase B			Fase A	Fase B		
10	Fase A	Fase B	Puesta a 0		Fase A	Fase B	Puesta a 0	
11								

Tabla G-5 Contadores rápidos HSC1 y HSC2

Modo	HSC1				HSC2			
	I0.6	I0.7	I1.0	I1.1	I1.2	I1.3	I1.4	I1.5
0	Reloj				Reloj			
1	Reloj		Puesta a 0		Reloj		Puesta a 0	
2	Reloj		Puesta a 0	Arranque	Reloj		Puesta a 0	Arranque
3	Reloj	Sentido			Reloj	Sentido		
4	Reloj	Sentido	Puesta a 0		Reloj	Sentido	Puesta a 0	
5	Reloj	Sentido	Puesta a 0	Arranque	Reloj	Sentido	Puesta a 0	Arranque
6	Reloj adelante	Reloj atrás			Reloj adelante	Reloj atrás		
7	Reloj adelante	Reloj atrás	Puesta a 0		Reloj adelante	Reloj atrás	Puesta a 0	
8	Reloj adelante	Reloj atrás	Puesta a 0	Arranque	Reloj adelante	Reloj atrás	Puesta a 0	Arranque
9	Fase A	Fase B			Fase A	Fase B		
10	Fase A	Fase B	Puesta a 0		Fase A	Fase B	Puesta a 0	
11	Fase A	Fase B	Puesta a 0	Arranque	Fase A	Fase B	Puesta a 0	Arranque

Operaciones booleanas			Operaciones aritméticas, incrementar y decrementar		
LD	N	Cargar	+I	IN1, OUT	Sumar enteros, palabras dobles o números reales
LDI	N	Cargar directamente	+D	IN1, OUT	
LDN	N	Cargar valor negado	+R	IN1, OUT	IN1 + OUT = OUT
LDNI	N	Cargar valor negado directamente	-I	IN1, OUT	Restar enteros, palabras dobles o números reales
A	N	AND	-D	IN1, OUT	
AI	N	Y directa	-R	IN1, OUT	OUT-IN1=OUT
AN	N	Y-NO	MUL	IN1, OUT	Multiplicar enteros (16*16→32) o reales
ANI	N	Y-NO directa	*R	IN1, OUT	Multiplicar enteros o enteros dobles
O	N	OR	*D, *I	IN1, OUT	IN1 * IN2 = OUT
OI	N	O directa	DIV	IN1, OUT	Dividir enteros (16/16→32) o reales
ON	N	O-NO	/R	IN1, OUT	Dividir enteros o enteros dobles
ONI	N	O-NO directa	/D, /I	IN1, OUT	IN1 / OUT = OUT
LDBx	N1, N2	Cargar resultado de la comparación de bytes N1 (x:<, <=, =, >=, >, <>) N2	SQRT	IN, OUT	Raíz cuadrada
ABx	N1, N2	Combinar mediante Y el resultado de la comparación de bytes N1 (x:<, <=, =, >=, >, <>) N2	LN	IN, OUT	Logaritmo natural
OBx	N1, N2	Combinar mediante O el resultado de la comparación de bytes N1 (x:<, <=, =, >=, >, <>) N2	EXP	IN, OUT	Exponencial natural
LDWx	N1, N2	Cargar resultado de la comparación de palabras N1 (x:<, <=, =, >=, >, <>) N2	SIN	IN, OUT	Seno
AWx	N1, N2	Combinar mediante Y el resultado de la comparación de palabras N1 (x:<, <=, =, >=, >, <>) N2	COS	IN, OUT	Coseno
OWx	N1, N2	Combinar mediante O el resultado de la comparación de palabras N1 (x:<, <=, =, >=, >, <>) N2	TAN	IN, OUT	Tangente
LDDx	N1, N2	Cargar resultado de la comparación de palabras dobles N1 (x:<, <=, =, >=, >, <>) N2	INCB	OUT	Incrementar byte, palabra o palabra doble
ADx	N1, N2	Combinar mediante Y el resultado de la comparación de palabras dobles N1 (x:<, <=, =, >=, >, <>) N2	INCW	OUT	
ODx	N1, N2	Combinar mediante O el resultado de la comparación de palabras dobles N1 (x:<, <=, =, >=, >, <>) N2	INCD	OUT	
LDRx	N1, N2	Cargar resultado de la comparación de números reales N1 (x:<, <=, =, >=, >, <>) N2	DECB	OUT	Decrementar byte, palabra o palabra doble
ARx	N1, N2	Combinar mediante Y el resultado de la comparación de números reales N1 (x:<, <=, =, >=, >, <>) N2	DECW	OUT	
ORx	N1, N2	Combinar mediante O el resultado de la comparación de números reales N1 (x:<, <=, =, >=, >, <>) N2	DECD	OUT	
NOT		Negar primer valor de pila	PID	Table, Loop	Regulación PID
EU		Detectar flanco positivo	Operaciones de temporización y conteo		
ED		Detectar flanco negativo	TON	Txxx, PT	Temporizador de retardo a la conexión
=	N	Asignar	TOF	Txxx, PT	Temporizador de retardo a la desconexión
=I	N	Asignar directamente	TONR	Txxx, PT	Temporizador de retardo a la conexión memorizado
R	S_BIT, N	Poner a 1 (activar)	CTU	Cxxx, PV	Contar adelante
SI	S_BIT, N	Poner a 0 (desactivar)	CTD	Cxxx, PV	Contar atrás
RI	S_BIT, N	Poner a 1 directamente	CTUD	Cxxx, PV	Contar adelante/atrás
		Poner a 0 directamente	Operaciones del reloj de tiempo real		
			TODR	T	Leer reloj de tiempo real
			TODW	T	Escribir reloj de tiempo real
			Operaciones de control del programa		
			FIN		Fin condicional del programa
			STOP		Pasar a STOP
			WDR		Borrar temporizador de vigilancia (300 ms)
			JMP	N	Saltar a meta
			LBL	N	Definir meta
			CALL	N [N1,...]	Llamar a subrutina [N1, ... hasta 16 parámetros opcionales]
			CRET		Retorno condicional de subrutina
			FOR	INDX, INIT, FINAL	Bucle FOR/NEXT
			NEXT		
			LSCR	N	Cargar, transición y fin del relé de control secuencial Segmento
			SCRT	N	
			SCRE		

Operaciones de transferencia, desplazamiento, rotación e inicialización	
MOVB IN, OUT MOVWIN, OUT MOVD IN, OUT MOVR IN, OUT BIR IN, OUT BIW IN, OUT	Transferir bytes, palabras, palabras dobles, reales Lectura directa y transferencia de bytes, Escritura directa y transferencia de bytes
BMB IN, OUT, N BMW IN, OUT, N BMD IN, OUT, N	Transferir bytes, palabras o palabras dobles en bloque
SWAP IN	Invertir bytes de una palabra
SHRB DATA, S_BIT, N	Registro de desplazamiento
SRB OUT, N SRW OUT, N SRD OUT, N	Desplazar byte, palabra o palabra doble a la derecha
SLB OUT, N SLW OUT, N SLD OUT, N	Desplazar byte, palabra o palabra doble a la izquierda
RRB OUT, N RRW OUT, N RRD OUT, N	Rotar byte, palabra o palabra doble a la derecha
RLB OUT, N RLW OUT, N RLD OUT, N	Rotar byte, palabra o palabra doble a la izquierda
FILL IN, OUT, N	Inicializar memoria
Operaciones lógicas	
ALD OLD	Combinar primer y segundo valor mediante Y Combinar primer y segundo valor mediante O
LPS LRD LPP LDS	Duplicar primer valor de la pila Copiar segundo valor de la pila Sacar primer valor de la pila Cargar pila
AENO	Y-ENO
ANDB IN1, OUT ANDWIN1, OUT ANDD IN1, OUT	Combinación Y con bytes, palabras o palabras dobles
ORB IN1, OUT ORW IN1, OUT ORD IN1, OUT	Combinación O con bytes, palabras o palabras dobles
XORB IN1, OUT XORWIN1, OUT XORD IN1, OUT	Combinación O-exclusiva con bytes, palabras o palabras dobles
INVW OUT INVD OUT	Invertir byte, palabra o palabra doble (complemento a 1)
Operaciones de tabla, de búsqueda y de conversión	
ATT TABLE, DATA	Registrar valor en tabla
LIFO TABLE, DATA FIFO TABLE, DATA	Obtener datos de la tabla
FND= SRC, PATRN, INDX FND<>SRC, PATRN, INDX FND< SRC, PATRN, INDX FND> SRC, PATRN, INDX	Buscar valor en tabla que concuerde con la comparación
BCDI OUT IBCD OUT	Convertir de BCD a entero Convertir de entero a BCD
BTI IN, OUT ITB IN, OUT ITD IN, OUT DTI IN, OUT	Convertir de byte a entero Convertir de entero a byte Convertir de entero a entero doble Convertir de entero doble a entero
DTR IN, OUT TRUNC IN, OUT ROUND IN, OUT	Convertir de palabra doble a real Convertir de real a palabra doble Convertir de real a entero doble
ATH IN, OUT, LEN HTA IN, OUT, LEN ITA IN, OUT, FMT	Convertir de ASCII a hexadecimal Convertir de hexadecimal a ASCII Convertir de entero a ASCII
DTA IN, OUT, FM RTA IN, OUT, FM	Convertir de entero doble a ASCII Convertir de real a ASCII
DECO IN, OUT ENCO IN, OUT	Decodificar Codificar
SEG IN, OUT	Generar configuración binaria de 7 segmentos
Descripción	
CRETI	Retorno condicional desde rutina de interrupción
ENI DISI	Habilitar todos los eventos de interrupción Inhibir todos los eventos de interrupción
ATCH INT, EVENT DTCH EVENT	Asociar interrupción Desasociar interrupción
Operaciones de comunicación	
XMT TABLE, PORT RCV TABLE, PORT	Transmitir mensaje (Freeport) Recibir mensaje (Freeport)
NETR TABLE, PORT NETW TABLE, PORT	Leer de la red Escribir en la red
GPA ADDR, PORT SPA ADDR, PORT	Leer dirección de puerto Ajustar dirección de puerto
Operaciones con contadores rápidos	
HDEF HSC, Mode	Definir modo para contador rápido
HSC N	Activar contador rápido
PLS X	Salida de impulsos

Índice alfabético

A

Acceso

- áreas de memoria, modificar un puntero, 5-14
- áreas de memoria
 - & y *, 5-13
- direccionamiento indirecto, 5-13–5-15
- áreas de operandos, 8-8
- direccionamiento directo, 5-2

Acceso a bits, 5-2

Acceso a bytes, 5-2

- utilizando punteros, 5-14

Acceso a palabras, 5-2

- utilizando punteros, 5-14

Acceso mediante bits, CPU 221/222/224/226, 8-8

Acceso mediante bytes, CPU 221/222/224, 8-8

Acceso mediante palabras, CPU 221/222/224/226, 8-8

Acceso mediante palabras dobles, CPU 221/222/224/226, 8-8

Accionamiento MicroMaster, conectar, 11-17

Accionamientos

- conectar a la CPU, 11-17
- configurar, 11-18

Accionamientos, tiempo de comunicación, 11-3

Activar contador rápido, 9-27

Actualización síncrona, función PWM, 9-59

Acumuladores, direccionamiento, 5-10

Adaptador de módem nulo, 7-25–7-26, 7-38, 7-41

Ajustar dirección de puerto, 9-196

Ajustar reloj de tiempo real, 9-71

Ajustes en el cuadro de diálogo Interface PG/PC, 7-6

Algoritmo PID, 9-89–9-93

Almacenar el programa en la memoria no volátil, 5-20

AND, 10-25

Área de marcas, direccionamiento, 5-5

Área de memoria de variables, direccionamiento, 5-5

Áreas de funciones, crear, 4-3

Áreas de memoria, G-3

- acceder a los datos, 5-2
- áreas de operandos, 8-8
- bits, 5-2
- bytes, 5-2
- CPU, 5-2

Áreas de operandos, CPU 221/222/224/226, 8-8

Áreas remanentes, definir, 5-19

Armario eléctrico

desmontaje del, 2-8

dimensiones

- CPU 221, 2-4
- CPU 222, 2-4
- CPU 226, 2-5

montaje, 2-6

ASCII

constantes, 5-12

operaciones de conversión

- Convertir de ASCII a hexadecimal, 9-139
- Convertir de entero a ASCII, 9-140
- Convertir de entero doble a ASCII, 9-142
- Convertir de hexadecimal a ASCII, 9-139
- Convertir de real a ASCII, 9-143

Asignación de pines, puerto de comunicación, 7-32

Asignar, 9-6

Asignar directamente, 9-6

Asignar salidas, configurar el estado, 6-8

Asistencia técnica, solicitar, vi

Asistencia, complementaria, vi

Asistente PID, 9-88

Asociar interrupción, 9-169

AWL, estado del programa, 4-35

Ayuda en pantalla, STEP 7-Micro/WIN 32, 3-2

B

Bias (suma integral), algoritmo PID, 9-91

Bits, 5-2

Bits de estado (SMB0), C-1

Bits, marcas especiales, C-1–C-13

Bloque de terminales

- CPU 224, 2-12
- módulo de ampliación, 2-12
- soltar, 2-12

Bloque funcional biestable (posicionar dominante), 10-7

Bloque funcional biestable (rearmar dominante), 10-7

Bobina (operación IEC), 10-4

Borrar primer registro de la tabla, 9-111

Borrar temporizador de vigilancia, 9-146–9-148

- consideraciones, 9-146

Borrar último registro de la tabla, 9-112

Buscar valor en tabla, 9-109

Byte de estado de los contadores rápidos, 9-39

Byte, formato de direcciones, 5-2

Bytes, y margen de enteros, 5-4

Bytes de programación de los contadores rápidos, C-9

C

Cable de ampliación, datos técnicos e instalación, A-89

Cable PC/PPI

ajustar los interruptores DIP, 3-5, 7-39

ajustar los parámetros, 7-10

asignación de pines, A-91

conexión, 3-5, 7-39

datos técnicos, A-91

posición de los interruptores DIP para seleccionar la velocidad de transferencia, 7-36, A-91

utilización con un módem, 7-25–7-26, 7-38, 7-41

utilización en modo Freeport, 7-36–7-37

Cableado

bloque de bornes opcional, 2-11

circuitos de supresión, 2-16–2-17

desmontar módulos, 2-8

entradas de los contadores rápidos, 9-32

reglas, 2-9–2-14

instalación con corriente alterna, 2-13

instalación con corriente continua, 2-14

Cableado (diagrama)

CPU 214 AC/DC/Relé, A-20

CPU 221 AC/DC/Relé, A-10

CPU 221 DC/DC/DC, A-10

CPU 222 AC/DC/Relé, A-15

CPU 222 DC/DC/DC, A-15

CPU 224 DC/DC/DC, A-20

CPU 226 AC/DC/Relé, A-25

CPU 226 DC/DC/DC, A-25

EM 221, 8 entradas digitales x DC 24 V, A-27

EM 222, 8 salidas digitales x DC 24 V, A-29

EM 222, 8 salidas digitales x relé, A-29

EM 223, 16 entradas digitales DC 24 V/
16 salidas digitales de relé, A-38

EM 223, 16 entradas digitales/16 salidas
digitales, A-38

EM 223, 4 entradas digitales/4 salidas de relé,
A-32

EM 223, 4 entradas digitales/4 salidas
digitales, A-32

EM 223, 8 entradas digitales DC 24 V/8
salidas digitales de relé, A-35

EM 223, 8 entradas digitales/8 salidas
digitales, A-35

EM 231 Termopar, A-68

EM 231, 4 salidas analógicas, A-41

EM 232, 2 salidas analógicas, A-41

EM 235, 4 entradas analógicas/1 salida
analógica, A-41

Cableado de campo

conector opcional, 2-11

diámetro, 2-9

instalación, 2-9

Cablear el RTD, al sensor con 4, 3 y 2 hilos, A-81

Cables

desmontar módulos, 2-8

nº de referencia, E-2

PC/PPI, ajustar los parámetros, 7-10

red PROFIBUS, 7-34

Calcular requisitos de alimentación, 2-18–2-20

Calibración de entradas, módulos analógicos,
A-42

Cambiar un puntero, 5-14

Canadian Standards Association (CSA), A-2

Captura de impulsos, 6-5

Cargar en la CPU

en modo RUN, 4-41

modo de operación necesario, 4-25

programa, 5-15

Cargar en la PG, programa, 5-15

Cargar pila, 9-198–9-200

Cartucho de memoria

códigos de error, B-2

copiar en, 5-22

datos técnicos, A-88

dimensiones, A-88

instalar, 5-22

nº de referencia, E-1

restablecer el programa, 5-24

retirar, 5-22

utilización, 5-22

Cartucho de pila, 5-15

datos técnicos, A-88

nº de referencia, E-1

Cartucho de reloj, datos técnicos, A-88

Casilla de verificación Red multimaestro, 7-11

Cerrar, redes, 7-33

Certificado CE, A-2

Ciclo

bits de estado, C-1

interrumpir, 4-24

tareas, 4-22

y función de forzado, 4-37

y tabla de estado/forzado, 4-37

Circuitos de seguridad cableados, diseñar, 4-3

Circuitos de supresión, reglas

salidas AC, 2-17

transistores en DC, 2-16, 2-17

Codificar, 9-135

Combinación O con bytes, 9-114

Combinación O con palabras, 9-115

- Combinación O con palabras dobles, 9-116
- Combinación O-exclusiva con bytes, 9-114
- Combinación O-exclusiva con palabras, 9-115
- Combinación O-exclusiva con palabras dobles, 9-116
- Combinación Y con bytes, 9-114
- Combinación Y con palabras, 9-115
- Combinación Y con palabras dobles, 9-116
- Combinar primer y segundo valor mediante O, 9-197–9-199
- Combinar primer y segundo valor mediante Y, 9-197–9-199
- Comenzar rutina de interrupción, 9-171
- Comparación, CPUs S7-200, 1-3
- Comparar byte, 9-10
- Comparar entero, 9-11
- Comparar palabra doble, 9-12
- Comparar real, 9-13
- Compatibilidad electromagnética, S7-200, A-4
- Compilación, errores
 - reacción del sistema, 4-45
 - violación de reglas, B-4
- Comprobación de datos, 5-12
- Comprobar, programa, 4-30–4-32
- Comunicación
 - cambiar los parámetros de la CPU, 3-10
 - componentes de redes, 7-32
 - conectar el PC, 7-2
 - conectar un cable PC/PPI, 3-5
 - configurar, 7-2–7-19
 - establecer con la CPU, 3-9
 - hardware
 - instalar en Windows NT, 7-8
 - instalar/desinstalar, 3-2–3-4
 - módem, 7-25–7-30
 - modo Freeport, 9-187, C-6
 - MPI, 7-30
 - PPI, 7-2, 7-30
 - procesar peticiones, 4-23
 - protocolo PROFIBUS, 7-31
 - protocolos soportados, 7-29
 - seleccionar la parametrización utilizada, 7-9–7-10
 - utilizando una tarjeta CP, 7-4–7-5
 - utilizando una tarjeta MPI, 7-4–7-5
 - utilizar módems, 7-16
 - velocidades de transferencia, 7-26
 - verificar la configuración, 7-4
- Comunicación DP estándar, A-52
- Comunicación MPI, tarjetas CP, 7-4
- Comunicación punto a punto, 1-3
- Comunidad Europea (CE), certificado, A-2
- Conceptos de programación, 4-5
- Condensador de alto rendimiento, 5-15
- Condiciones ambientales, A-3
- Conectar la alimentación, respaldar datos, 5-17–5-21
- Conectar los accionamientos, 11-17
- Conector de bus, desmontar módulos de ampliación, 2-8
- Conector de puerto del esclavo DP, EM 277 PROFIBUS-DB, A-52
- Conectores
 - de bus, 7-33
 - nº de referencia, E-2
 - puerto de ampliación de bus, retirar la tapa, 2-8
- Configuración
 - áreas remanentes, 5-19
 - comunicación, 7-2–7-19
 - crear planos, 4-4
 - de un PC con tarjeta CP y unidad de programación, 7-12
 - de un PC con tarjeta MPI y unidad de programación, 7-12
 - EM 231, módulo analógico, A-43
 - EM 235, módulo analógico, A-43
 - EM 277 PROFIBUS-DP, A-54
 - estado de las salidas, 6-8
 - hardware de comunicación, 3-2, 7-3
 - módulos analógicos, A-42
- Configuraciones soportadas, CPU S7-200 y módulo EM 277 PROFIBUS-DP, 7-4
- Configurar, parámetros de comunicación, 7-4
- Congelar salidas, 6-8
- Consideraciones
 - al utilizar la operación Borrar temporizador de vigilancia, 9-146
 - entornos con vibraciones fuertes, 2-7
 - montaje del hardware, 2-2–2-4
 - montaje vertical, 2-7
 - utilización de topes, 2-7
- Constantes, 5-12
- Constantes ASCII, 5-12
- Constantes de números en coma flotante, 5-12
- Constantes de números reales, 5-12
- Constantes decimales, 5-4
- Constantes hexadecimales, 5-4, 5-12
- Contactos de comparación, ejemplo, 9-14
- Contactos directos, operaciones, 9-3
- Contactos estándar, operaciones, 9-2
- Contador ascendente, 10-15
- Contador ascendente-descendente, 10-16
- Contador descendente, 10-16
- Contador rápido, 6-10, 9-39
 - cambio de sentido, 9-45
 - ejemplos, 9-47

- Contadores
 - CPU 221/222/224/226, 8-7
 - direccionamiento, 5-8
 - tipos, 5-8
 - variables, 5-8
- Contadores rápidos, 9-27–9-46
 - ajustar los valores actuales y predeterminados, 9-38
 - área de memoria, direccionamiento, 5-11
 - byte de control, 9-38
 - byte de estado, 9-39
 - bytes de programación (SMB36-SMB65), C-9
 - cableado de las entradas, 9-32
 - cargar nuevo valor actual/predeterminado, 9-45
 - cronogramas, 9-28–9-31
 - direccionamiento, 9-36
 - ejemplos, 9-28–9-31
 - elegir el nivel de actividad, 9-37
 - inhibir, 9-46
 - interrupciones, 9-39
 - modos, G-4
 - modos de inicialización, 9-41–9-44
 - modos de operación, 9-33
 - utilización, 9-28
- Contraseña
 - borrar, 4-29
 - CPU, 4-27
 - configurar, 4-28
 - nivel de protección, 4-27
 - olvidada, 4-29
 - restringir el acceso, 4-27
- Control de escritura, C-7
- Control de interrupción de caracteres, 9-192
- Control de lazos PID
 - adelante/atrás, 9-94
 - modos, 9-96
- Control directo de las entradas y salidas, 4-24
- Convenciones, programación con Micro/WIN 32, 8-2
- Conversión, operaciones de, 4-17
- Convertir
 - enteros de 16 bits a números reales, 9-93
 - entradas del lazo, 9-93
 - números reales a valores normalizados, 9-93
- Convertir de ASCII a hexadecimal, 9-139
- Convertir de BCD a entero, 9-130, 10-30
- Convertir de byte a entero, 9-133, 10-32
- Convertir de entero a ASCII, 9-140
- Convertir de entero a BCD, 9-130, 10-30
- Convertir de entero a byte, 9-133, 10-33
- Convertir de entero a entero doble, 9-132
- Convertir de entero a entero doble, 10-32
- Convertir de entero doble a ASCII, 9-142
- Convertir de entero doble a entero, 9-132
- Convertir de entero doble a entero, 10-31
- Convertir de entero doble a real, 9-130, 9-132
- Convertir de entero doble a real, 10-31
- Convertir de hexadecimal a ASCII, 9-139
- Convertir de real a ASCII, 9-143
- Convertir de real a entero doble, 10-31
- Copiar segundo valor, 9-197–9-199
- Corriente necesaria, calcular, 2-20
- Coseno, 9-86, 10-22
- CP (procesador de comunicaciones), 7-4
 - configuración con PC, 7-12
- CP 5411, nº de referencia, E-2
- CP 5511
 - ajustar los parámetros de la tarjeta MPI (PPI), 7-14
 - nº de referencia, E-2
- CP 5611
 - ajustar los parámetros de la tarjeta MPI (PPI), 7-14
 - nº de referencia, E-2
- CPU
 - ajustar el modo de operación, 4-25
 - áreas de memoria, 5-2, G-3
 - áreas de operandos, 8-8
 - borrar memoria, 4-29
 - cambiar los parámetros de comunicación, 3-10
 - ciclo, 4-22
 - conexión de módem, 7-25–7-30
 - contraseña, 4-27
 - datos técnicos generales, A-3
 - dimensiones
 - CPU 224, 2-4
 - CPU 226, 2-5
 - módulos de ampliación, 2-5
 - tornillos para el montaje, 2-4–2-5
 - eliminación de errores, 4-43
 - errores fatales, B-2
 - establecer la comunicación con, 3-9
 - funcionamiento básico, 4-5
 - hardware soportado para la comunicación en redes, 7-3
 - identificador (SMB6), C-4
 - módulos, 1-6
 - montaje, armario eléctrico, 2-6
 - procedimiento, desmontaje, 2-8
 - requisitos de alimentación, 2-18
 - simuladores de entradas, datos técnicos, A-93
 - tornillos para el montaje, 2-4–2-5
- CPU 212, ejemplo de numeración de E/S, 6-3

- CPU 221
- áreas de operandos, 8-8
 - backup, 1-3
 - E/S, 1-3
 - filtros de entrada, 1-3
 - funciones, 8-7
 - interrupciones, número máximo, 9-176
 - interrupciones soportadas, 1-3
 - márgenes de memoria, 8-7
 - memoria, 1-3
 - módulos de ampliación, 1-3
 - operaciones soportadas, 1-3
 - protocolos soportados, 1-3
 - puertos de comunicación, 1-3
 - resumen, 1-3
- CPU 221 AC/DC/Relé, nº de referencia, E-1
- CPU 221 AC/DC/Relé
- datos técnicos, A-6
 - identificación de terminales de conexión, A-10
- CPU 221 DC/DC/DC
- datos técnicos, A-6
 - identificación de terminales de conexión, A-10
 - nº de referencia, E-1
- CPU 222
- áreas de operandos, 8-8
 - copias de seguridad, 1-3
 - E/S, 1-3
 - filtros de entrada, 1-3
 - funciones, 8-7
 - interrupciones, número máximo, 9-176
 - interrupciones soportadas, 1-3
 - márgenes de memoria, 8-7
 - memoria, 1-3
 - módulos de ampliación, 1-3
 - operaciones soportadas, 1-3
 - protocolos soportados, 1-3
 - puertos de comunicación, 1-3
 - resumen, 1-3
- CPU 222 AC/DC/Relé, nº de referencia, E-1
- CPU 222 AC/DC/Relé
- datos técnicos, A-11
 - identificación de terminales de conexión, A-15
- CPU 222 DC/DC/DC
- datos técnicos, A-11
 - identificación de terminales de conexión, A-15
- CPU 222 DC/DC/DC, nº de referencia, E-1
- CPU 224
- áreas de operandos, 8-8
 - bloque de terminales, 2-12
 - copias de seguridad, 1-3
 - E/S, 1-3
 - ejemplo de numeración de E/S, 6-3
 - filtros de entrada, 1-3
 - funciones, 8-7
 - interrupciones, número máximo, 9-176
 - interrupciones soportadas, 1-3
 - márgenes de memoria, 8-7
 - memoria, 1-3
 - módulos de ampliación, 1-3
 - operaciones soportadas, 1-3
 - protocolos soportados, 1-3
 - puertos de comunicación, 1-3
 - resumen, 1-3
- CPU 224 AC/DC/Relé, nº de referencia, E-1
- CPU 224 AC/DC/Relé
- datos técnicos, A-16
 - identificación de terminales de conexión, A-20
- CPU 224 DC/DC/DC
- datos técnicos, A-16
 - identificación de terminales de conexión, A-20
 - nº de referencia, E-1
- CPU 226
- áreas de operandos, 8-8
 - copias de seguridad, 1-3
 - E/S, 1-3
 - filtros de entrada, 1-3
 - funciones, 8-7
 - interrupciones, número máximo, 9-176
 - interrupciones soportadas, 1-3
 - márgenes de memoria, 8-7
 - memoria, 1-3
 - módulos de ampliación, 1-3
 - operaciones soportadas, 1-3
 - protocolos soportados, 1-3
 - puertos de comunicación, 1-3
 - resumen, 1-3
- CPU 226 AC/DC/Relé
- datos técnicos, A-21
 - identificación de terminales de conexión, A-25
- CPU 226 DC/DC/DC
- datos técnicos, A-21
 - identificación de terminales de conexión, A-25

CPU S7-200, configuraciones soportadas, 7-4
 Crear un programa de ejemplo: ajustar una interrupción temporizada, 4-18
 Cronogramas, contadores rápidos, 9-28
 Cuadro de diálogo Instalar/desinstalar interfaces, 7-7
 Cuadro de diálogo Interface PG/PC, 7-6
 Cuadro Registro de desplazamiento (SHRB), 9-128

D

Datos técnicos
 gama S7-200, A-3
 simulador de entradas, A-93
 Decodificar, 9-135
 Decrementar, 10-20
 ejemplo, 9-81, 10-20
 Decrementar byte, 9-79
 Decrementar palabra, 9-79
 Decrementar palabra doble, 9-80
 Definir meta, 9-148
 Definir modo para contador rápido, 9-27
 modos de los contadores, 9-36
 Desasociar interrupción, 9-169
 Desbordamiento de la cola de espera (SMB4), C-3
 Desigualdad (NE), 10-8
 Desmontaje
 CPU, 2-8
 dimensiones
 CPU 221, 2-4
 CPU 222, 2-4
 CPU 224, 2-4
 CPU 226, 2-5
 módulos de ampliación, 2-5
 tornillos para el montaje, 2-4–2-5
 espacio necesario para el montaje, 2-2
 módulo de ampliación, 2-8
 procedimiento, 2-8
 tornillos para el montaje, 2-4–2-5
 Desplazar a la derecha, 10-27
 Desplazar a la izquierda, 10-27
 Desplazar byte a la derecha, 9-120
 Desplazar byte a la izquierda, 9-120
 Desplazar palabra a la derecha, 9-121
 Desplazar palabra a la izquierda, 9-121
 Desplazar palabra doble a la derecha, 9-122
 Desplazar palabra doble a la izquierda, 9-122
 Detectar flanco negativo, 9-4

Detectar flanco positivo, 9-4
 Diagrama de funciones, elementos básicos, 4-6
 Dimensiones
 cartucho de memoria, A-88
 CPU 221, 2-4
 CPU 222, 2-4
 CPU 224, 2-4
 CPU 226, 2-5
 módulos de ampliación, 2-5
 tornillos para el montaje, 2-4–2-5
 Diodos supresores, 2-16
 Dirección de estación más alta, 7-42
 Direccionamiento
 acumuladores, 5-10
 área de marcas, 5-5
 áreas de memoria, 5-2
 byte.bit, 5-2
 contadores, 5-8
 contadores rápidos, 5-11, 9-36
 E/S de los módulos de ampliación, 6-2
 E/S integradas, 6-2
 entradas analógicas, 5-9
 equipos en una red, 7-29
 imagen del proceso de las entradas, 5-4
 imagen del proceso de las salidas, 5-4
 indirecto (punteros), 5-13–5-15
 & y *, 5-13
 modificar un puntero, 5-14
 marcas especiales, 5-5
 memoria de variables, 5-5
 relés de control secuencial, 5-5
 salidas analógicas, 5-9
 temporizadores, 5-7
 Direccionamiento directo, 5-2
 operaciones sobrecargadas, 4-16
 Direccionamiento indirecto, 5-13–5-15
 & y *, 5-13
 modificar un puntero, 5-14
 Diseñar una solución de automatización con un Micro-PLC, 4-2
 Diseño de sistemas, Micro-PLC, 4-2
 Dividir, 10-18
 Dividir enteros de 16 bits, 9-75
 Dividir enteros de 16 bits a enteros de 32 bits, 9-77
 Dividir enteros de 32 bits, 9-76
 Dividir reales, 9-83
 Documentación adicional, iv
 DRV_CTRL, 11-7
 Duplicar primer valor, 9-197–9-199

E

E/S integradas, direccionamiento, 6-2

Editores

AWL (Lista de instrucciones), 4-6

FUP (Diagrama de funciones), 4-9

KOP (Esquema de contactos), 4-8

EEPROM, 5-15, 5-17

códigos de error, B-2

copiar memoria de variables en, 5-20

guardar de la memoria V, C-7

Ejemplos

ajustar una interrupción temporizada, 4-18

AND, OR y XOR, 10-26–10-28

aritmética, 9-78, 9-84

Borrar primer registro de la tabla, 9-111

Borrar último registro de la tabla, 9-112

Buscar valor en tabla, 9-110

calcular requisitos de alimentación, 2-18

contador rápido, 9-47

funcionamiento con puesta a 0 y arranque, 9-29

funcionamiento con puesta a 0 y sin arranque, 9-28

funcionamiento de HSC0 en modo 0 y de HSC1 en modos 0, 1 ó 2, 9-29

funcionamiento de HSC1 o HSC2 en modo 3, 4 ó 5, 9-30

funcionamiento de HSC1 o HSC2 en modo 6, 7 u 8, 9-30

funcionamiento de HSC1 o HSC2 en modo 9, 10 u 11, 9-31

funcionamiento de HSC1 ó HSC2 en modo 9, 10 u 11, 9-31

conversión de números reales, 10-33

Convertir de ASCII a hexadecimal, 9-139

Convertir números reales, 9-134

Convertir y truncar, 9-134, 10-33

operación lógica de pila, 9-200–9-202

operación de conteo, 9-25

Decodificar/Codificar, 9-136

decrementar, 9-81, 10-20

desplazamiento y rotación, 9-126–9-128, 10-29–10-31

FOR/NEXT, 9-156–9-158

incrementar, 9-81, 10-20

Inicializar memoria, 9-113–9-115

Invertir, 9-119–9-121

Leer de la red/Escribir en la red, 9-182–9-184

Llamar subrutina, 9-153–9-155

Modulación del ancho de impulsos, 9-63

numeración de E/S, 6-2, 6-3

operación de conteo, 10-17

Registro de desplazamiento, 9-129–9-131

operaciones aritméticas, 10-19

operaciones con contactos, 9-5, 10-4

operaciones con rutinas de interrupción, 9-178

operaciones con salidas, 9-9, 10-6

operaciones de comparación, 9-14

operaciones de transmisión, 9-193

potenciómetros analógicos, 6-13

programa del protocolo USS, 11-20

Registrar valor en tabla, 9-108

regulación PID, 9-98–9-100

relé de control secuencial, 9-159–9-164

convergir cadenas secuenciales, 9-163–9-166

dividir cadenas secuenciales, 9-161

transiciones condicionales, 9-166

Saltar a meta, 9-148–9-150

Segmento, 9-138

STOP, END y Borrar temporizador de

vigilancia, 9-147–9-149

tarjeta MPI con esclavo/maestro, 7-4

TDs 200 integrados en una red, 7-12

temporizador de retardo a la conexión, 9-20, 9-21

temporizador de retardo a la desconexión, 9-22

temporizador de retardo al conectar, 10-13

temporizador de retardo al desconectar, 10-13

temporizador por impulsos, 10-14

transferencia de bloques, 9-104–9-106

transferir e invertir, 9-105–9-107, 10-24–10-26

Tren de impulsos, 9-65, 9-68

Truncar, 9-134, 10-33

Y, O y O-exclusiva, 9-117–9-119

Eliminación

errores de compilación, B-4

errores de programación del tiempo de ejecución, B-3

errores fatales, B-2

Eliminación de errores, 4-43

contraseña olvidada, 4-29

fatales, 4-43–4-44

lazo PID, 9-97

leer de la red/escribir en la red, 9-180

no fatales, 4-45

rearrancar la CPU después de un error fatal, 4-44

responder a errores, 4-43

S7-200, D-1

Eliminar fallos, instalación de Micro/WIN 32, 3-4

EM 221, 8 entradas digitales x DC 24 V,

identificación de terminales de conexión, A-27

nº de referencia, E-1

- EM 222, 8 salidas de relé, nº de referencia, E-1
- EM 222, 8 salidas digitales x DC 24 V, nº de referencia, E-1
- EM 222, 8 salidas digitales x relé, identificación de terminales de conexión, A-29
- EM 222, 8 salidas digitales x DC 24 V, identificación de terminales de conexión, A-29
- EM 222, salidas DC 24 V y salidas de relé, datos técnicos, A-28
- EM 223, 16 entradas digitales DC 24 V/16 salidas digitales DC 24 V, identificación de terminales de conexión, A-38
- EM 223, 16 entradas digitales DC 24 V/16 salidas digitales de relé, identificación de terminales de conexión, A-38
- EM 223, 16 entradas x DC 24 V/16 salidas de relé, datos técnicos, A-36
- EM 223, 16 entradas/16 salidas x DC 24 V, datos técnicos, A-36
- EM 223, 4 entradas DC 24 V/4 salidas de relé, identificación de terminales de conexión, A-32
- EM 223, 4 entradas x DC 24 V/4 salidas de relé, datos técnicos, A-30
- EM 223, 4 entradas/4 salidas x DC 24 V datos técnicos, A-30
- identificación de terminales de conexión, A-32
- EM 223, 8 entradas digitales DC 24 V/8 salidas digitales de relé, identificación de terminales de conexión, A-35
- EM 223, 8 entradas digitales/8 salidas de relé x DC 24 V, nº de referencia, E-1
- EM 223, 8 entradas x DC 24 V/8 salidas de relé, datos técnicos, A-33
- EM 223, 8 entradas digitales/8 salidas digitales x DC 24 V, nº de referencia, E-1
- EM 223, 8 entradas/8 salidas x DC 24 V datos técnicos, A-33
- identificación de terminales de conexión, A-35
- EM 231 Termopar
 - compatibilidad, A-68
 - identificación de terminales de conexión, A-68
- EM 231, 2 salidas analógicas, datos técnicos, A-39
- EM 231, 4 entradas analógicas/1 salida analógica, datos técnicos, A-39
- EM 231, 4 salidas analógicas
 - datos técnicos, A-39
 - identificación de terminales de conexión, A-41
- EM 232, 2 salidas analógicas, identificación de terminales de conexión, A-41
- EM 235, 4 entradas analógicas/1 salida analógica, identificación de terminales de conexión, A-41
- EM 277 PROFIBUS-DP, 7-28
 - coherencia de datos, A-57
 - compatibilidad con las CPUs S7-200, A-51
 - conectar una CPU en calidad de estación esclava, A-53
 - conector de puerto del esclavo DP, A-52
 - configuración, A-54
 - consideraciones relativas al programa, A-58
 - datos técnicos, A-50
 - interruptores de direccionamiento y LEDs, A-51
 - LEDs de estado, A-60
- EN/ENO, reglas, 4-18
- END, 9-145
- Energía necesaria, tabla de cálculo, 2-20
- Enlaces, MPI lógicos, 7-30
- Enlaces lógicos, MPI, 7-30
- ENO, 9-168
- Enteros de 16 bits, convertir a números reales, 9-93
- Entornos con vibraciones fuertes, utilización de topes, 2-7
- Entradas, funcionamiento básico, 4-5
- Entradas analógicas
 - acceder a, 4-22
 - direccionamiento, 5-9
 - leer el valor con una rutina de interrupción, 9-179
- Entradas digitales
 - leer, 4-22
 - captura de impulsos, 6-5
- Entradas y salidas rápidas, 6-10
- Equipos necesarios, S7-200, 1-2
- Errores
 - de programación del tiempo de ejecución, B-3
 - fatales, B-2
 - lazo PID, 9-97
 - Leer de la red/Escribir en la red, 9-180
 - no fatales, B-3, B-4
 - operaciones del protocolo USS, 11-16
 - SMB1, errores de ejecución, C-2
 - violación de reglas de compilación, B-4
- Errores de programación del tiempo de ejecución, B-3
- Errores durante el tiempo de ejecución, reacción del sistema, 4-45
- Errores fatales, B-2
 - y el funcionamiento de la CPU, 4-44
- Errores no fatales
 - eliminación de, 4-45
 - reacción del sistema, 4-45
 - y el funcionamiento de la CPU, 4-45

Escalar la salida del lazo, 9-94
 Escribir en la red, 9-180
 ejemplo, 9-182–9-184
 errores, 9-180
 Escritura directa y transferencia de bytes, 9-106
 Espacio necesario para el montaje, 2-2
 Esquema de circuitos
 módulo RTD, A-80
 módulo termopar, A-72
 Esquema de contactos, elementos básicos, 4-6
 Esquemas de conexiones de las entradas
 EM 231 y EM 235, A-45
 EM 232 y EM 235, A-46
 Estaciones de operador, definir, 4-4
 Estado de las entradas y salidas (SMB5), C-3
 Estado del módulo inteligente,
 SMB200 a SMB299, C-16
 Eventos de interrupción, descripción, G-2
 Exponencial natural, 9-86, 10-22

F

Factor de actualización GAP, 7-42
 Factory Mutual Research, A-2
 Fecha, ajustar, 9-71
 Filtrar entradas analógicas, 6-9
 Filtro de entrada, supresión de ruidos, 6-4
 Filtros de entrada, y captura de impulsos, 6-5
 FOR, 9-154
 Formato de la palabra de datos de entrada,
 EM 235, A-44
 Formato de la palabra de datos de salida,
 EM 232 y EM 235, A-46
 Forzar valores, 4-37
 Freeport
 definición del modo, 9-173
 protocolos definidos por el usuario, 7-31
 Freeport, modo, utilización del cable PC/PPI,
 7-36–7-37
 Funciones PTO/PWM
 inicialización, 9-58
 registro de control, 9-56
 registros de control, SMB66 a SMB85, C-11
 tabla de referencia hexadecimal, 9-56
 Funciones PTO/PWM (SMB66 a SMB85), C-11
 FUP, estado del programa, 4-33

G

Ganancia del lazo, convertir, 9-93
 GAP. Véase Factor de actualización GAP
 Guardar
 el programa en la memoria no volátil, 5-20
 valor en EEPROM, C-7

H

Habilitar todos los eventos de interrupción, 9-173
 Hardware
 desinstalar en Micro/WIN 32, 7-7
 instalar en Micro/WIN 32, 7-7
 Hardware y software necesarios
 STEP 7-Micro/WIN 32, 3-2
 STEP 7-Micro/WIN 32 Toolbox, 3-2
 Homologaciones, iv, A-2
 HSC3, HSC4, HSC5, SMB130 a SMB165, C-14

I

Identificación de terminales de conexión
 CPU 214 AC/DC/Relé, A-20
 CPU 221 AC/DC/Relé, A-10
 CPU 221 DC/DC/DC, A-10
 CPU 222 AC/DC/Relé, A-15
 CPU 222 DC/DC/DC, A-15
 CPU 224 DC/DC/DC, A-20
 CPU 226 AC/DC/Relé, A-25
 CPU 226 DC/DC/DC, A-25
 EM 221, 8 entradas digitales x DC 24 V, A-27
 EM 222, 8 salidas digitales x DC 24 V, A-29
 EM 222, 8 salidas digitales x relé, A-29
 EM 223, 16 entradas digitales DC 24 V/
 16 salidas digitales de relé, A-38
 EM 223, 16 entradas digitales/16 salidas
 digitales, A-38
 EM 223, 4 entradas digitales/4 salidas de relé,
 A-32
 EM 223, 4 entradas digitales/4 salidas
 digitales, A-32
 EM 223, 8 entradas digitales DC 24 V/
 8 salidas digitales de relé, A-35
 EM 223, 8 entradas digitales/8 salidas
 digitales, A-35
 EM 231 Termopar, A-68
 EM 231, 4 salidas analógicas, A-41
 EM 232, 2 salidas analógicas, A-41
 EM 235, 4 entradas analógicas/1 salida
 analógica, A-41

IEC 1131-3, tipificación de variables, 4-11
 juego de operaciones, 4-10
Igualdad (EQ), 10-8
Imagen del proceso de las entradas, 4-24
 direccionamiento, 5-4
 funcionamiento, 4-22
Imagen del proceso de las salidas, 4-23, 4-24
 direccionamiento, 5-4
Incrementar, 10-20
Incrementar byte, 9-79
Incrementar palabra, 9-79
Incrementar palabra doble, 9-80
Incrementar un puntero, 5-14
Inhibir todos los eventos de interrupción, 9-173
Iniciación
 contadores rápidos, 9-41–9-44
 función PWM, 9-59
 modo Freeport, 9-188
 PTO/PWM, 9-58
Iniciar, Tren de impulsos (PTO), 9-60
Iniciar memoria, 9-113
Instalación
 cable de E/S de ampliación, A-89
 cartucho de memoria, 5-22
 dimensiones
 CPU 221, 2-4
 CPU 222, 2-4
 CPU 224, 2-4
 CPU 226, 2-5
 módulos de ampliación, 2-5
 raíl estándar, 2-3
 tornillos para el montaje, 2-4–2-5
 en entornos con vibraciones fuertes,
 utilizando topes, 2-7
 espacio necesario para el montaje, 2-2
 hardware de comunicación, 3-2–3-4
 Micro/WIN 32, 3-3
 procedimiento
 armario eléctrico, 2-6
 módulo de expansión, 2-6–2-8
 raíl DIN, 2-7
 simulador de entradas DC, A-93
 tornillos para el montaje, 2-4–2-5
Instalación con corriente alterna, reglas, 2-13
Instalación con corriente continua, reglas, 2-14
Instalar, hardware de comunicación, instrucciones
 especiales para usuarios de Windows NT, 7-8

Interface, verificar los parámetros estándar, 3-6
Interface PPI, nº de referencia del manual, E-2
Internet, dirección de Siemens, vi
Interrupción temporizada, ejemplo, 4-18, 9-179
Interrupciones
 ciclo de la CPU, 4-24
 colas de espera, 9-176
 configurar, 9-169
 contador rápido, 9-39
 contadores rápidos, 9-39
 CPU 221/222/224/226, 8-7
 datos compartidos con el programa principal,
 9-172
 E/S, 9-173
 flancos positivo y negativo, 9-173
 habilitar e inhibir, 9-173
 marcas para el desbordamiento de colas de
 espera, 9-176
 prioridades, 9-176
 restricciones, 9-171
 rutinas, 9-171
 soporte del sistema, 9-171
 temporizadas, 9-175, C-8
 configurar para leer entradas analógicas,
 9-179
 tipos y números de eventos
 CPUs 221/222/224/226, 9-170
 prioridades, 9-177
Interrupciones temporizadas, 9-175
 SMB34, SMB35, C-8
Interruptores DIP, cable PC/PPI, 3-5, 7-39
Invertir byte, 9-118
Invertir bytes de una palabra, 9-105
Invertir palabra, 9-118
Invertir palabra doble, 9-118

J

Juegos de operaciones
 IEC-1131, 4-10
 SIMATIC, 4-10

K

KOP, estado del programa, 4-32

L**Lazos PID**

- acción positiva/negativa, 9-94
- ajustar la suma integral (bias), 9-95
- convertir salidas, 9-94
- elegir el tipo de regulación, 9-92
- errores, 9-97
- márgenes, variables, 9-94
- modos, 9-96
- programa de ejemplo, 9-98–9-100
- tabla del lazo, 9-97

Lectura directa y transferencia de bytes, 9-106**Leer de la red, 9-180**

- ejemplo, 9-182–9-184
- errores, 9-180

Leer dirección de puerto, 9-196**Leer reloj de tiempo real, 9-71****Lenguajes de programación, descripción, 4-6****Lista de instrucciones, 4-6****Llamar subrutina, con parámetros, 9-150****Logaritmo natural, 9-85, 10-21****M****Maestros**

- módem, 7-25
- protocolo MPI, 7-4, 7-30
- protocolo PPI, 7-30
- protocolo PROFIBUS, 7-31

Manuales, nº de referencia, E-2**Marcas de control, contadores rápidos, 9-37****Marcas especiales, C-1–C-13**

- direccionamiento, 5-5
- SMB0: bits de estado, C-1
- SMB1: bits de estado, C-2
- SMB166 a SMB194: Tabla de definición de perfiles PTO, PT1, C-16
- SMB186 a SMB194: control de recepción de mensajes, C-12
- SMB2: búfer de recepción de caracteres en modo Freeport, C-2
- SMB200 a SMB299: estado del módulo inteligente, C-16
- SMB28 y SMB29: potenciómetros analógicos, C-6
- SMB3: error de paridad en modo Freeport, C-2
- SMB30 a SMB165: Registro HSC, C-14
- SMB30, SMB130 (control del modo Freeport), 9-188
- SMB30, SMB130: registros de control del modo Freeport, C-6
- SMB31: control de escritura en la memoria no volátil (EEPROM), C-7
- SMB34/SMB35: duración de las interrupciones temporizadas, C-8
- SMB36 a SMB65: bytes de programación de los contadores rápidos, C-9
- SMB4: desbordamiento de la cola de espera, C-3
- SMB5: estado de las entradas y salidas, C-3
- SMB6: identificador de la CPU, C-4
- SMB66 a SMB85: funciones PTO/PWM, C-11
- SMB7: reservado, C-4
- SMB8 a SMB21: registro de errores e identificadores de los módulos de ampliación, C-5
- SMB86 a SMB94: control de recepción de mensajes, C-12
- SMB98 y SMB99, C-14
- SMW222 a SMW26: tiempos de ciclo, C-6
- SMW32: Control de escritura en la memoria no volátil (EEPROM), C-7
- Márgenes de memoria, CPU 221/222/224/226, 8-7
- Márgenes válidos para las CPUs, 8-7
- Memoria, borrar, 4-29
- Memoria de byte, 5-2
- Memoria V, copiar en la EEPROM, 5-20
- Mensajes, red con “token passing”, 7-44

- Micro/WIN 32
 - convenciones de programación, 8-2
 - instalación, 3-3
 - eliminar fallos, 3-4
- Micro/WIN 32 Toolbox, hardware y software necesarios, 3-2
- Módem
 - 10 bits, 7-23
 - 11 bits, 7-25
 - adaptador de módem nulo, 7-38, 7-41
 - cables necesarios, 7-25
 - comunicación en la red, 7-25–7-30
 - conexión PC/PG a CPU, 7-25–7-26
 - utilización con el cable PC/PPI, 7-38, 7-41
- Módem nulo, adaptador de, 7-38
- Módems, configurar la comunicación, 7-16
- Modificar un puntero (direccionamiento indirecto), 5-14
- Modo Freeport
 - control de interrupción de caracteres, 9-192
 - funcionamiento, 9-187
 - habilitar, 9-187
 - inicializar, 9-188
 - SMB2, búfer de recepción de caracteres, C-2
 - SMB3, error de paridad, C-2
 - SMB30, SMB130 (control del modo Freeport), 9-188
 - SMB30, SMB130: registros de control del modo Freeport, C-6
 - y modos de operación, 9-187
- Modo RUN
 - cargar el programa en la CPU, 4-41
 - editar programas, 4-39
- Modos de operación
 - bits de estado, C-1
 - cambiar, 4-25, 4-26
 - comunicación Freeport, 9-187
 - contadores rápidos, 9-33
 - y la función de forzado, 4-37
- Modos, control en lazos PID, 9-96
- Modulación del ancho de impulsos (PWM), 6-12, 9-49, 9-50
 - cambiar el ancho de impulsos, 9-59
 - ejemplo, 9-63
 - función, 9-50
 - inicializar, 9-59
- Módulo de ampliación, bloque de terminales, 2-12
- Módulo de ampliación analógico, direccionamiento, 6-2
- Módulo de ampliación digital, direccionamiento, 6-2
- Módulo EM 277 PROFIBUS-DP, configuraciones soportadas, 7-4
- Módulos analógicos, precisión y repetibilidad, A-48
- Módulos de ampliación, 1-5, 1-6
 - desmontaje, 2-8
 - dimensiones
 - CPU 221, 2-4
 - CPU 222, 2-4
 - módulos de 8 y 16 E/S, 2-5
 - tornillos para el montaje, 2-4–2-5
 - direccionamiento de E/S, 6-2
 - identificadores y registro de errores (SMB8 a SMB21), C-5
 - montaje
 - armario eléctrico, 2-6
 - raíl DIN, 2-7
 - nº de referencia, E-1
 - requisitos de alimentación, 2-18
 - tornillos para el montaje, 2-4–2-5
- Montaje
 - configuración, 2-2
 - dimensiones
 - CPU 221, 2-4
 - CPU 222, 2-4
 - CPU 224, 2-4
 - CPU 226, 2-5
 - módulos de ampliación, 2-5
 - raíl estándar, 2-3
 - tornillos para el montaje, 2-4–2-5
 - en entornos con vibraciones fuertes, utilización de topes, 2-7
 - en sentido vertical, utilizar topes, 2-7
 - espacio necesario para el montaje, 2-2
 - procedimiento
 - armario eléctrico, 2-6
 - módulo de expansión, 2-6–2-8
 - raíl DIN, 2-7
 - tornillos para el montaje, 2-4–2-5
- MPI (interface multipunto), protocolo, 7-30
 - velocidad de transferencia, 7-4
- MPI, comunicación, 7-30
- MPI, enlaces lógicos, 7-30
- MPI, tarjeta
 - ajustar los parámetros de la tarjeta MPI (PPI), 7-14
 - configuración con PC, 7-12
 - parámetros PPI, 7-14
- Multimaestro, red, 7-4
- Multiplicar, 10-18
- Multiplicar enteros de 16 bits, 9-75
- Multiplicar enteros de 16 bits a enteros de 32 bits, 9-77
- Multiplicar enteros de 32 bits, 9-76
- Multiplicar reales, 9-83

N

NEXT, 9-154

Nombres simbólicos, definir, 4-4

Normas, nacionales e internacionales, A-2

Normas de emisión electromagnética, A-2

Normas de inmunidad electromagnética, A-2

NOT, 10-26

Números

representación de, 5-4

utilizar valores de constantes, 5-12

O

Observar

estado del programa, 4-32, 4-33, 4-35

programa, 4-30–4-32

Opciones de visualización

estado AWL, 4-35

estado FUP, 4-33

estado KOP, 4-32

Operación NOT, 9-4

Operación nula, 9-8

Operaciones

Activar contador rápido, 9-27

Ajustar dirección de puerto, 9-196

Ajustar reloj de tiempo real, 9-71

AND, 10-25

Asignar, 9-6

Asignar directamente, 9-6

Asociar interrupción, 9-169

Bloque funcional biestable
(posicionar dominante), 10-7Bloque funcional biestable
(rearman dominante), 10-7

Bobina, 10-4

Borrar primer registro de la tabla, 9-111

Borrar temporizador de vigilancia, 9-146–9-148

Borrar último registro de la tabla, 9-112

Buscar valor en tabla, 9-109

Cargar pila, 9-198–9-200

Codificar, 9-135

Combinación O con bytes, 9-114

Combinación O con palabras, 9-115

Combinación O con palabras dobles, 9-116

Combinación O-exclusiva con bytes, 9-114

Combinación O-exclusiva con palabras, 9-115

Combinación O-exclusiva con palabras dobles,
9-116

Combinación Y con bytes, 9-114

Combinación Y con palabras, 9-115

Combinación Y con palabras dobles, 9-116

Combinar primer y segundo valor mediante O,
9-197–9-199Combinar primer y segundo valor mediante Y,
9-197–9-199

Comenzar rutina de interrupción, 9-171

Comparar byte, 9-10

Comparar entero, 9-11

Comparar palabra doble, 9-12

Comparar real, 9-13

con contadores, 9-24

contactos directos, 9-3

contactos estándar, 9-2, 10-2

Contador ascendente, 10-15

Contador ascendente-descendente, 10-16

Contador descendente, 10-16

contadores rápidos, 9-27–9-70

conversión, 4-17–4-19

Convertir de ASCII a hexadecimal, 9-139

Convertir de BCD a entero, 9-130, 10-30

Convertir de byte a entero, 9-133, 10-32

Convertir de entero a ASCII, 9-140

Convertir de entero a BCD, 9-130, 10-30

Convertir de entero a byte, 9-133, 10-33

Convertir de entero a entero doble, 9-132,
10-32

Convertir de entero a real, 9-132

Convertir de entero doble a ASCII, 9-142

Convertir de entero doble a entero, 9-132,
10-31

Convertir de entero doble a real, 9-130, 10-31

Convertir de hexadecimal a ASCII, 9-139

Convertir de real a ASCII, 9-143

Convertir de real a entero doble, 10-31

Copiar segundo valor, 9-197–9-199

Coseno, 9-86, 10-22

de búsqueda, 9-107–9-113

de tabla, 9-107–9-113

Decodificar, 9-135

Decrementar, 10-20

Decrementar byte, 9-79

Decrementar palabra, 9-79

Decrementar palabra doble, 9-80

Definir modo para contador rápido, 9-27

Desasociar interrupción, 9-169

Desigualdad (NE), 10-8

Desplazar a la derecha, 10-27

Desplazar a la izquierda, 10-27

Desplazar byte a la derecha, 9-120

Desplazar byte a la izquierda, 9-120

Desplazar palabra a la derecha, 9-121

Desplazar palabra a la izquierda, 9-121

Desplazar palabra doble a la derecha, 9-122

Desplazar palabra doble a la izquierda, 9-122

Detectar flanco negativo, 9-4

Detectar flanco positivos, 9-4

- Dividir enteros de 16 bits, 9-75
- Dividir enteros de 16 bits a enteros de 32 bits, 9-77
- Dividir enteros de 32 bits, 9-76
- Dividir reales, 9-83
- DRV_CTRL, 11-7
- Duplicar primer valor, 9-197–9-199
- END, 9-145
- ENO, 9-168
- Escribir en la red, 9-180
- Escritura directa y transferencia de bytes, 9-106
- Exponencial natural, 9-86, 10-22
- FOR, 9-154
- Habilitar todos los eventos de interrupción, 9-173
- Igualdad (EQ), 10-8
- Incrementar byte, 9-79
- Incrementar palabra, 9-79
- Incrementar palabra doble, 9-80
- incrementar un puntero, 5-14
- Inhibir todos los eventos de interrupción, 9-173
- Inicializar memoria, 9-113
- Invertir byte, 9-118
- Invertir bytes de una palabra, 9-105
- Invertir palabra, 9-118
- Invertir palabra doble, 9-118
- Lectura directa y transferencia de bytes, 9-106
- Leer de la red, 9-180
- Leer dirección de puerto, 9-196
- Leer reloj de tiempo real, 9-71
- Logaritmo natural, 9-85, 10-21
- modificar un puntero, 5-14
- Multiplicar, 10-18
- Multiplicar enteros de 16 bits, 9-75
- Multiplicar enteros de 16 bits a enteros de 32 bits, 9-77
- Multiplicar enteros de 32 bits, 9-76
- Multiplicar reales, 9-83
- NEXT, 9-154
- NOT, 9-4, 10-26
- Operación nula, 9-8
- OR, 10-25
- PID, 9-87–9-101
- Poner a 0, 9-7
- Poner a 0 directamente, 9-8
- Protocolo USS, 11-2
- Raíz cuadrada, 9-85, 10-21
- READ_PM, 11-11
- Recibir mensaje, 9-186
- Redondear a entero doble, 9-131
- Registrar valor en tabla, 9-107
- Registro de desplazamiento, 9-127
- Registro de desplazamiento (SHRB), 9-128
- Regulación PID, 9-87–9-101
- Relé de control secuencial, 9-157
- reloj de tiempo real, 9-71
- Restar, 10-18
- Restar enteros de 32 bits, 9-74
- Restar reales, 9-82
- Retorno de subrutina, 9-149
- Retorno desde rutina de interrupción, 9-171
- Rotar a la derecha, 10-28
- Rotar byte a la derecha, 9-123
- Rotar byte a la izquierda, 9-123
- Rotar palabra a la derecha, 9-124
- Rotar palabra a la izquierda, 9-124
- Rotar palabra doble a la derecha, 9-125
- Rotar palabra doble a la izquierda, 9-125
- Sacar primer valor, 9-198–9-200
- Salida de impulsos, 6-12, 9-49
- Salida de impulsos (PLS), 6-12
- Saltar a meta, 9-148
- Secuencia ascendente (LT), 10-9
- Secuencia decreciente (GT), 10-10
- Secuencia monótona (GE), 10-10
- Secuencia monótona (LE), 10-9
- Segmento, 9-137
- Seno, 9-86, 10-22
- SET, 10-5
- sobrecargadas, 4-15
- STOP, 9-145
- Sumar, 10-18
- Sumar enteros de 16 bits, 9-73
- Sumar enteros de 32 bits, 9-74
- Sumar reales, 9-82
- Tangente, 9-86, 10-22
- Temporizador con retardo al conectar, 10-11
- Temporizador con retardo al desconectar, 10-11
- Temporizador de retardo a la conexión, 9-15
- Temporizador de retardo a la conexión memorizado, 9-15
- Temporizador de retardo a la desconexión, 9-15
- Temporizador por impulsos, 10-12
- tiempo de ejecución, F-1
- Transferir byte, 9-102
- Transferir bytes en bloque, 9-103
- Transferir en bloque, 10-24
- Transferir palabra, 9-102

- Transferir palabra doble, 9-102
- Transferir palabras dobles en bloque, 9-103
- Transferir palabras en bloque, 9-103
- Transferir real, 9-102
- Transferir y asignar valores, 10-23
- Transición negativa, 10-3
- Transición positiva, 10-3
- Transmitir mensaje, 9-186
- Truncar, 9-131, 10-30
- USS_INIT, 11-5
- WRITE_PM, 11-13
- XOR, 10-25
- Operaciones aritméticas
 - Coseno, 9-86, 10-22
 - Decrementar, 10-20
 - Dividir, 10-18
 - Dividir enteros de 16 bits, 9-75
 - Dividir enteros de 16 bits a enteros de 32 bits, 9-77
 - Dividir enteros de 32 bits, 9-76
 - Dividir reales, 9-83
 - ejemplo, 9-78, 9-84, 10-19
 - Exponencial natural, 9-86, 10-22
 - Incrementar, 10-20
 - Logaritmo natural, 9-85, 10-21
 - Multiplicar, 10-18
 - Multiplicar enteros de 16 bits, 9-75
 - Multiplicar enteros de 16 bits a enteros de 32 bits, 9-77
 - Multiplicar enteros de 32 bits, 9-76
 - Multiplicar reales, 9-83
 - Raíz cuadrada, 9-85, 10-21
 - Restar, 10-18
 - Restar enteros de 16 bits, 9-73
 - Restar enteros de 32 bits, 9-74
 - Restar reales, 9-82
 - Seno, 9-86, 10-22
 - Sumar, 10-18
 - Sumar enteros de 16 bits, 9-73
 - Sumar enteros de 32 bits, 9-74
 - Sumar reales, 9-82
 - Tangente, 9-86, 10-22
- Operaciones AWL
 - breviario, G-5
 - tiempo de ejecución, F-1
- Operaciones booleanas
 - AND, 10-25
 - ejemplo, AND, OR y XOR, 10-26–10-28
 - NOT, 10-26
 - OR, 10-25
 - XOR, 10-25
- Operaciones con bobinas
 - Bobina, 10-4
 - RESET, 10-5
 - SET, 10-5
- Operaciones con contactos
 - Bloque funcional biestable (posicionar dominante), 10-7
 - Bloque funcional biestable (rearman dominante), 10-7
 - contactos estándar, 10-2
 - ejemplo, 9-5, 10-4
 - NOT, 9-4
 - Transición negativa, 10-3
 - Transición positiva, 10-3
- Operaciones con contactos estándar, 10-2
- Operaciones con contadores
 - Contador ascendente, 10-15
 - Contador ascendente-descendente, 10-16
 - Contador descendente, 10-16
 - funcionamiento, 10-15, 10-16
- Operaciones con contadores rápidos, 9-27–9-70
- Operaciones con salidas
 - Asignar, 9-6
 - Asignar directamente, 9-6
 - ejemplo, 9-9, 10-6
 - Operación nula, 9-8
 - Poner a 0, 9-7
 - Poner a 0 directamente, 9-8
- Operaciones de búsqueda, 9-107–9-113
 - Borrar primer registro de la tabla, 9-111
 - Borrar último registro de la tabla, 9-112
 - Buscar valor en tabla, 9-109
 - Registrar valor en tabla, 9-107
- Operaciones de comparación
 - Comparar byte, 9-10
 - Comparar entero, 9-11
 - Comparar palabra doble, 9-12
 - Comparar real, 9-13
 - Desigualdad (NE), 10-8
 - ejemplo, 9-14
 - Igualdad (EQ), 10-8
 - Secuencia ascendente (LT), 10-9
 - Secuencia decreciente (GT), 10-10
 - Secuencia monótona (GE), 10-10
 - Secuencia monótona (LE), 10-9

- Operaciones de comunicación
 - Ajustar dirección de puerto, 9-196
 - Escribir en la red, 9-180
 - Leer de la red, 9-180
 - Leer dirección de puerto, 9-196
 - Recibir mensaje, 9-186
 - Transmitir mensaje, 9-186
- Operaciones de contaje, 9-24
 - ejemplo, 9-25, 10-17
- Operaciones de control del programa
 - Borrar temporizador de vigilancia, 9-146–9-148
 - ejemplo, 9-147–9-149
 - END, 9-145
 - ejemplo, 9-147–9-149
 - ENO, 9-168
 - FOR, 9-154
 - FOR/NEXT, ejemplo, 9-156–9-158
 - Llamar subrutina, ejemplo, 9-153–9-155
 - NEXT, 9-154
 - Relé de control secuencial, 9-157
 - Retorno de subrutina, 9-149
 - Saltar a meta, 9-148
 - ejemplo, 9-148–9-150
 - STOP, 9-145
 - ejemplo, 9-147–9-149
- Operaciones de conversión
 - Codificar, 9-135
 - Convertir de ASCII a hexadecimal, 9-139
 - Convertir de BCD a entero, 9-130, 10-30
 - Convertir de byte a entero, 9-133, 10-32
 - Convertir de entero a ASCII, 9-140
 - Convertir de entero a BCD, 9-130, 10-30
 - Convertir de entero a byte, 9-133, 10-33
 - Convertir de entero a entero doble, 9-132, 10-32
 - Convertir de entero a real, 9-132
 - Convertir de entero doble a ASCII, 9-142
 - Convertir de entero doble a entero, 9-132, 10-31
 - Convertir de entero doble a real, 9-130, 10-31
 - Convertir de hexadecimal a ASCII, 9-139
 - Convertir de real a ASCII, 9-143
 - Convertir de real a entero doble, 10-31
 - Decodificar, 9-135
 - Redondear a entero doble, 9-131
 - Segmento, 9-137
 - Truncar, 9-131, 10-30
- Operaciones de decrementar
 - Decrementar byte, 9-79
 - Decrementar palabra, 9-79
 - Decrementar palabra doble, 9-80
 - Restar enteros de 16 bits, 9-73
 - Restar enteros de 32 bits, 9-74
- Operaciones de desplazamiento
 - Desplazar a la derecha, 10-27
 - Desplazar a la izquierda, 10-27
 - Desplazar byte a la derecha, 9-120
 - Desplazar byte a la izquierda, 9-120
 - Desplazar palabra a la derecha, 9-121
 - Desplazar palabra a la izquierda, 9-121
 - Desplazar palabra doble a la derecha, 9-122
 - Desplazar palabra doble a la izquierda, 9-122
 - ejemplo de desplazamiento y rotación, 9-126–9-128, 10-29–10-31
 - ejemplo de registro de desplazamiento, 9-129–9-131
 - Registro de desplazamiento, 9-127
- Operaciones de incrementar
 - Incrementar byte, 9-79
 - Incrementar palabra, 9-79
 - Incrementar palabra doble, 9-80
 - Sumar enteros de 16 bits, 9-73
 - Sumar enteros de 32 bits, 9-74
- Operaciones de inicialización, ejemplo, 9-113–9-115
- Operaciones de interrupción
 - Asociar interrupción, 9-169
 - Comenzar rutina de interrupción, 9-171
 - Desasociar interrupción, 9-169
 - descripción, 9-169
 - ejemplo, 9-178
 - Habilitar todos los eventos de interrupción, 9-173
 - Inhibir todos los eventos de interrupción, 9-173
 - Retorno desde rutina de interrupción, 9-171

- Operaciones de rotación
 - ejemplo de desplazamiento y rotación, 9-126–9-128, 10-29–10-31
 - Rotar a la derecha, 10-28
 - Rotar a la izquierda, 10-28
 - Rotar byte a la derecha, 9-123
 - Rotar byte a la izquierda, 9-123
 - Rotar palabra a la derecha, 9-124
 - Rotar palabra a la izquierda, 9-124
 - Rotar palabra doble a la izquierda, 9-125
- Operaciones de segmentación (operaciones SCR), 9-158
- Operaciones de tabla, 9-107–9-113
 - Borrar primer registro de la tabla, 9-111
 - Borrar último registro de la tabla, 9-112
 - Buscar valor en tabla, 9-109
 - Inicializar memoria, 9-113
 - Registrar valor en tabla, 9-107
- Operaciones de temporización
 - ejemplo de un temporizador de retardo a la conexión, 9-20, 9-21
 - ejemplo de un temporizador de retardo a la desconexión, 9-22
 - ejemplo de un temporizador por impulsos, 10-14
 - Temporizador con retardo al conectar, 10-11
 - Temporizador con retardo al desconectar, 10-11
 - Temporizador de retardo a la conexión, 9-15
 - Temporizador de retardo a la conexión memorizado, 9-15
 - Temporizador de retardo a la desconexión, 9-15
 - Temporizador por impulsos, 10-12
- Operaciones de transferencia
 - ejemplo de transferir e invertir, 9-105–9-107, 10-24–10-26
 - ejemplo de una operación de transferencia de bloques, 9-104–9-106
 - Escritura directa y transferencia de bytes, 9-106
 - Invertir bytes de una palabra, 9-105
 - Lectura directa y transferencia de bytes, 9-106
 - Transferir, 10-23
 - Transferir byte, 9-102
 - Transferir bytes en bloque, 9-103
 - Transferir en bloque, 10-24
 - Transferir palabra, 9-102
 - Transferir palabra doble, 9-102
 - Transferir palabras dobles en bloque, 9-103
 - Transferir palabras en bloque, 9-103
 - Transferir real, 9-102
- Operaciones del protocolo USS, 11-2
 - DRV_CTRL, 11-7
 - errores de ejecución, 11-16
 - READ_PM, 11-11
 - restricciones, 11-3
 - secuencia de programación, 11-4
 - USS_INIT, 11-5
 - WRITE_PM, 11-13
- Operaciones del relé de control secuencial, ejemplos, 9-159–9-163
- Operaciones del reloj de tiempo real, 9-71
 - Ajustar reloj de tiempo real, 9-71
 - Leer reloj de tiempo real, 9-71
- Operaciones incrementar, ejemplo, 9-81, 10-20
- Operaciones lógicas
 - Combinación O con bytes, 9-114
 - Combinación O con palabras, 9-115
 - Combinación O con palabras dobles, 9-116
 - Combinación O-exclusiva con bytes, 9-114
 - Combinación O-exclusiva con palabras, 9-115
 - Combinación O-exclusiva con palabras dobles, 9-116
 - Combinación Y con bytes, 9-114
 - Combinación Y con palabras, 9-115
 - Combinación Y con palabras dobles, 9-116
 - ejemplo
 - Invertir, 9-119–9-121
 - Y, O y O-exclusiva, 9-117–9-119
 - Invertir byte, 9-118
 - Invertir palabra, 9-118
 - Invertir palabra doble, 9-118
- Operaciones lógicas con contactos, ejemplo, 9-5, 10-4
- Operaciones lógicas de pilas
 - Cargar pila, 9-198–9-200
 - Combinar primer y segundo valor mediante O, 9-197–9-199
 - Combinar primer y segundo valor mediante Y, 9-197–9-199
 - Copiar segundo valor, 9-197–9-199
 - Duplicar primer valor, 9-197–9-199
 - ejemplo, 9-200–9-202
 - funcionamiento, 9-198
 - Sacar primer valor, 9-198–9-200
- Operaciones PID, ejemplo, 9-98–9-100
- OR, 10-25

P

Palabras, margen de enteros, 5-4
 Palabras dobles, margen de enteros, 5-4
 Panel de operador, nº de referencia, E-3
 Panel de operador OP17, nº de referencia, E-3
 Panel de operador OP3, nº de referencia, E-3
 Panel de operador OP7, nº de referencia, E-3
 Panel táctil TP070, nº de referencia, E-3
 Parametrización, cable PC/PPI (PPI), 7-10–7-11
 Parametrización utilizada
 seleccionar, 7-9–7-10
 tarjeta MPI (PPI), 7-14
 Parámetros
 cable PC/PPI (PPI), 7-10–7-11
 tarjeta MPI (PPI), 7-14
 PID, lazos, CPU 221/222/224/226, 8-7
 PID, operaciones, 9-87–9-101
 Pila lógica, relés de control secuencial (SCRs), 9-157
 Polarizar, redes, 7-33
 Poner a 0, 9-7
 Poner a 0 directamente, 9-8
 Potenciómetros, SMB28 y SMB29, 6-13
 Potenciómetros analógicos, 6-13
 SMB28, SMB29, C-6
 Potenciómetros analógicos: SMB28 y SMB29, 6-13, C-6
 PPI (interface punto a punto)
 comunicación, 7-2, 7-30
 protocolo, 7-30
 Precisión y repetibilidad, módulos analógicos, A-48
 Procesador de comunicaciones (CP), nº de referencia, E-2
 Procesador de comunicaciones CP 243-2
 datos técnicos, A-85
 nº de referencia del manual, E-2
 resumen breve, A-86
 Process Field Bus, iv
 PROFIBUS
 asignación de pines, 7-32
 comunicación, 7-31
 datos técnicos del cable, 7-34
 protocolo, 7-31
 repetidores, 7-35

Programa

 almacenamiento, 5-15–5-18, 5-22
 cargar en la CPU, 5-15
 en modo RUN, 4-41
 cargar en la PG, 5-15
 comprobar, 4-30–4-32
 ejecutar, 4-23
 elementos básicos, 4-18
 entradas analógicas, 4-22
 entradas/salidas (E/S), 4-5
 estructura, 4-18
 guardar datos en la memoria no volátil, 5-20
 observar, 4-30–4-32
 observar el estado, 4-32, 4-33, 4-35
 restablecer con el cartucho de memoria, 5-24
 utilizar subrutinas, 9-149
 utilizar una tabla de estado/forzado, 4-31
 Protocolo USS, programa de ejemplo, 11-20
 Protocolos definidos por el usuario, Freeport, 7-31
 PTO/PWM
 ancho y conteo de impulsos, 9-57
 marca especial de estado, 9-57
 marcas especiales de control, 9-57
 tiempo de ciclo, 9-57
 PTO/PWM, funciones
 calcular los valores de la tabla de perfiles, 9-54
 registros de control, 9-56
 PTO/PWM, tabla de referencia hexadecimal, 9-56
 Puerto de comunicación
 asignación de pines, 7-32
 interrupciones, 9-173
 Puesta a tierra para circuitos, reglas de cableado, 2-10
 Punteros, 5-13–5-15
 & y *, 5-13
 modificar un puntero, 5-14

R

Raíl DIN

 desmontaje, 2-8
 dimensiones, 2-3
 entornos con vibraciones fuertes, 2-7
 espacio necesario para el montaje, 2-2–2-4
 instalación, 2-7
 montaje vertical, 2-7
 nº de referencia, E-3
 utilización de topes, 2-7

- Raíl estándar
 - dimensiones, 2-3
 - entornos con vibraciones fuertes, 2-7
 - espacio necesario para el montaje, 2-2–2-4
 - instalación, 2-7
 - montaje vertical, 2-7
 - utilización de topes, 2-7
- Raíz cuadrada, 9-85, 10-21
- READ_PM, 11-11
- Reales, números, representación de, 5-4
- Rearrancar la CPU, después de un error fatal, 4-44
- Recibir mensaje, 9-186, 9-189
 - SMB86 a SMB94, SMB186 a SMB194, C-12
- Recursos para las operaciones del protocolo USS, 11-2
- Red
 - componentes, 7-32
 - maestros, 7-27
 - multimaestro, 7-4, 7-28
 - panorámica, 7-27
 - puerto de comunicación, 7-32
- Red multimaestro, 7-28
- Redes
 - cerrar, 7-33
 - conectores, 7-33
 - configurar la comunicación, 7-2–7-19
 - datos técnicos del cable, 7-34
 - dirección de estación más alta, 7-42
 - enviar mensajes, 7-44
 - factor de actualización GAP, 7-42
 - instalar el hardware de comunicación, 3-2–3-4
 - optimizar el rendimiento, 7-42
 - polarizar, 7-33
 - repetidores, 7-35
 - segmentos, 7-29
 - seleccionar la parametrización utilizada, 7-9
 - tiempo de rotación del testigo (token), 7-44–7-47
- Redes de resistencia/condensador, aplicaciones de relé, 2-17
- Redondear a entero doble, 9-131
- Registrar valor en tabla, 9-107
- Registro de desplazamiento, 9-127, 9-128
- Registro de desplazamiento (SHRB), 9-128
- Registro de errores e identificadores de los módulos de ampliación (SMB8 a SMB21), C-5
- Reglas
 - cableado, 2-9
 - aislamiento, 2-10
 - circuitos de supresión, 2-16
 - salidas AC, 2-17
 - transistores en DC, 2-17
 - diseñar una solución de automatización, 4-2–4-4
 - entornos con vibraciones fuertes, 2-7
 - instalación con corriente alterna, 2-13
 - instalación con corriente continua, 2-14
 - modificar un puntero para el direccionamiento indirecto, 5-14
 - montaje vertical, 2-7
 - puesta a tierra para circuitos, 2-10
 - utilización de topes, 2-7
- Reglas de cableado para fases unipolares, 2-13
- Reglas para instalaciones aisladas con corriente continua, 2-14
- Regulación del lazo, convertir entradas, 9-93
- Regulación PID, 9-87–9-101
 - acción positiva/negativa, 9-94
 - ajustar suma integral (bias), 9-95
 - bits de historial, 9-96
 - convertir entradas, 9-93
 - convertir salidas, 9-94
 - ejemplo, 9-98–9-100
 - elegir el tipo, 9-92
 - errores, 9-97
 - márgenes/variables, 9-94
 - modos, 9-96
 - programa de ejemplo, 9-98–9-100
 - tabla del lazo, 9-97
- Relé corriente continua, 2-17
- Relé de control secuencial, 9-157
- Relés, redes de resistencia/condensador, 2-17
- Relés de control secuencial
 - CPU 221/222/224/226, 8-7
 - direccionamiento, 5-5
- Reloj, bits de estado, C-1
- Reloj, tiempo real, 9-71
- Repetidores
 - nº de referencia, E-2
 - red PROFIBUS, 7-35
- Requisitos de alimentación
 - calcular, 2-18
 - CPU, 2-18
 - ejemplo, 2-19
 - módulos de ampliación, 2-18

- Requisitos de las operaciones del protocolo USS, 11-2
- RESET, 10-5
- Respaldar datos, 5-15–5-20
 - áreas remanentes, 5-19
 - cartucho de pila (opcional), 5-15
 - condensador de alto rendimiento, 5-15
 - conectar la alimentación, 5-17–5-21
 - EEPROM, 5-15, 5-17, 5-20
- Respaldar la memoria, 5-15–5-20
- Restar, 10-18
- Restar enteros de 16 bits, 9-73
- Restar enteros de 32 bits, 9-74
- Restar reales, 9-82
- Restringir el acceso a la CPU. Véase Contraseña
- Resumen de las CPUs S7-200
 - áreas de operandos, 8-8
 - márgenes de memoria, 8-7
- Resumen de las CPUs S7-200, funciones, 1-3
- Retirar, cartucho de memoria, 5-22
- Retorno de subrutina, 9-149
- Retorno desde rutina de interrupción, 9-171
- Rotación del testigo, rendimiento de la red, 7-43
- Rotar a la derecha, 10-28
- Rotar a la izquierda, 10-28
- Rotar byte a la izquierda, 9-123
- Rotar palabra a la derecha, 9-124
- Rotar palabra a la izquierda, 9-124
- Rotar palabra doble a la derecha, 9-125
- Rotar palabra doble a la izquierda, 9-125
- RTD
 - cableado, A-79
 - configurar, A-77
 - datos técnicos, A-67
 - indicadores de estado, A-81
 - márgenes de temperatura/precisión, A-82
- RUN, 4-25
- Rutinas de interrupción, reglas, 4-18
- S**
 - S7-200
 - compatibilidad electromagnética, A-4
 - componentes, 1-5
 - Componentes de sistema, 1-2
 - condiciones ambientales, A-3
 - CPU, desmontaje, 2-8
 - datos técnicos, A-3
 - dimensiones
 - CPU 221, 2-4
 - CPU 222, 2-4
 - CPU 224, 2-4
 - CPU 226, 2-5
 - módulos de ampliación, 2-5
 - tornillos para el montaje, 2-4–2-5
 - módulos de ampliación, 1-5
 - desmontaje, 2-8
 - montaje, armario eléctrico, 2-6
 - resumen de las CPUs, 1-3
 - tornillos para el montaje, 2-4–2-5
- Sacar primer valor, 9-198–9-200
- Salida de impulsos, 6-10
 - cambiar el ancho de impulsos, 6-12
 - funciones PTO/PWM, SMB66 a SMB85:
 - marcas especiales, C-11
 - operación, 9-49
- Salida de impulsos (PLS), 6-12
- Salidas
 - congelar, 6-8
 - de impulsos rápidos, 6-12
 - funcionamiento básico, 4-5
- Salidas AC, 2-17
- Salidas analógicas
 - acceder a, 4-23
 - direccionamiento, 5-9
- Salidas de impulsos, 6-12
- Salidas digitales, escribir, 4-23
- Saltar a meta, 9-148
- Secuencia ascendente (LT), 10-9
- Secuencia de programación,
 - operaciones del protocolo USS, 11-4
- Secuencia decreciente (GT), 10-10
- Secuencia monótona (GE), 10-10
- Secuencia monótona (LE), 10-9
- Segmento
 - direcciones de los equipos, 7-29
 - unidades esclavas, 7-29
 - unidades maestras, 7-29

Segmento (operación de conversión), 9-137
 Segmentos, red, 7-29
 Selector, de modos de operación, 4-25
 Seno, 9-86, 10-22
 SET, 10-5
 SIMATIC, juego de operaciones, 4-10
 Simulador de entradas, A-93
 SM0.2 (marca Datos remanentes perdidos), 5-18
 SMB0: bits de estado, C-1
 SMB1: bits de estado, C-2
 SMB166 a SMB194: Tabla de definición de perfiles PTO, PT1, C-16
 SMB186 a SMB194: control de recepción de mensajes, C-12
 SMB2 (recepción de caracteres en modo Freeport), control de interrupción de caracteres, 9-192
 SMB2: búfer de recepción de caracteres en modo Freeport, C-2
 SMB200 a SMB299: estado del módulo inteligente, C-16
 SMB3 (error de paridad en modo Freeport), control de interrupción de caracteres, 9-192
 SMB3: error de paridad en modo Freeport, C-2
 SMB30 a SMB165: Registro HSC, C-14
 SMB30, SMB130 (control del modo Freeport), 9-188
 SMB30, SMB130: registros de control del modo Freeport, C-6
 SMB34/SMB35: duración de las interrupciones temporizadas, C-8
 SMB36 a SMB65: bytes de programación de los contadores rápidos, C-9
 SMB5: estado de las entradas y salidas, C-3
 SMB6: identificador de la CPU, C-4
 SMB7 (reservado), C-4
 SMB86 a SMB94: control de recepción de mensajes, C-12
 SMB98 y SMB99, C-14
 SMW22 a SMW26: tiempos de ciclo, C-6
 Sobrecarga, 4-15
 Software de programación, nº de referencia, E-2
 Soltar, bloque de terminales, 2-12
 STEP 7-Micro/WIN 32, iv
 Ayuda en pantalla, 3-2
 comunicación con módems, 7-25–7-30
 configurar la comunicación, 7-5
 hardware para la comunicación en redes, 3-2, 7-3
 hardware y software necesarios, 3-2
 instalar el hardware de comunicación, 3-2–3-4
 nº de referencia, E-2
 nº de referencia, actualización, E-2

STEP 7-Micro/WIN 32 Toolbox, iv
 hardware y software necesarios, 3-2
 nº de referencia, E-2
 operaciones del protocolo USS, 11-1
 STOP, 4-25, 9-145
 Subrutina
 con parámetros, 9-150
 ejemplo, 4-18
 Subrutinas
 añadir al programa, 9-149
 reglas, 4-18
 Suma integral (bias), ajustar, regulación PID, 9-95
 Sumar, 10-18
 Sumar enteros de 16 bits, 9-73
 Sumar enteros de 32 bits, 9-74
 Sumar reales, 9-82
 Supresión de ruidos, filtro de entrada, 6-4

T

Tabla de definición de perfiles PTO, PT1, SMB166 a SMB194, C-16
 Tabla de estado/forzado
 ciclo de la CPU, 4-37
 modificar el programa, 4-31
 Tabla de referencia hexadecimal PTO/PWM, 9-56
 Tabla de símbolos, protocolo USS, 11-2
 Tabla de símbolos globales, 11-2
 Tabla del lazo, 9-97
 Tabla del lazo PID, 9-97
 Tamaño de los módulos
 CPU 221, 2-4
 CPU 222, 2-4
 CPU 224, 2-4
 CPU 226, 2-5
 módulos de ampliación, 2-5
 tornillos para el montaje, 2-4–2-5
 Tamaño físico
 CPU 221, 2-4
 CPU 222, 2-4
 CPU 224, 2-4
 CPU 226, 2-5
 módulos de ampliación, 2-5
 tornillos para el montaje, 2-4–2-5
 Tangente, 9-86, 10-22
 Tarjeta MPI, 7-4
 nº de referencia, E-2
 TD 200
 nº de referencia, E-3
 nº de referencia del manual, E-2
 Temporizador con retardo al conectar, 10-11
 Temporizador con retardo al desconectar, 10-11

Temporizador de retardo a la conexión, 9-15
Temporizador de retardo a la conexión memorizado, 9-15
Temporizador de retardo a la desconexión, 9-15
Temporizador por impulsos, 10-12
Temporizadores
 CPU 221/222/224/226, 8-7
 direccionamiento, 5-7
 funcionamiento, 10-11, 10-12
 número, 10-11, 10-12
 resolución, 10-11, 10-12
Temporizadores T32/T96, interrupciones, 9-175
TERM, 4-25
Término diferencial, algoritmo PID, 9-92
Término integral, algoritmo PID, 9-91
Término proporcional, algoritmo PID, 9-90
Termopar
 cableado, A-71
 configurar, A-68
 datos técnicos, A-67
 indicadores de estado, A-73
 precisión y márgenes de temperatura, A-74
Test de aislamiento de alto potencial, A-4
Tiempo de ciclo, función Tren de impulsos (PTO), 9-60
Tiempo de ciclo, (SMW22 a SMW26), C-6
Tiempo de ejecución, operaciones AWL, F-1
Tiempo de rotación del testigo (token), 7-44–7-47
Tiempo de rotación del testigo, número de estaciones, 7-45
Tiempo, ajustar, 9-71
Tipificación de datos, 5-12
Tipos de datos
 complejos, 4-12
 elementales, 4-11
 verificación, 4-12–4-16
 ventajas, 4-14
Tornillos (para el montaje), 2-4–2-5
Transferir byte, 9-102
Transferir bytes en bloque, 9-103
Transferir en bloque, 10-24
Transferir palabra doble, 9-102
Transferir palabras dobles en bloque, 9-103
Transferir palabras en bloque, 9-103
Transferir real, 9-102
Transferir y asignar valores, 10-23
Transición negativa, 10-3
Transición positiva, 10-3
transistores en DC, proteger, 2-16
Transmitir mensaje, 9-186, 9-188
 ejemplo, 9-193
Tratamiento de errores, fatales, 4-44

Tren de impulsos (PTO), 6-12, 9-51
 cambiar el conteo de impulsos, 9-61
 cambiar el tiempo de ciclo, 9-60
 cambiar el tiempo de ciclo y el conteo de impulsos, 9-61
 ejemplo, 9-65, 9-68
 función, 9-51
 inicializar, 9-60
Tren de impulsos rápidos (PTO), 9-49
Truncar, 9-131, 10-30

U

USS, protocolo, requisitos de las operaciones, 11-2
USS_INIT, 11-5
Utilizar punteros, 5-13
 & y *, 5-13
 modificar un puntero, 5-14
Utilizar subrutinas, 9-149

V

Valores actuales de los temporizadores, actualizar, 9-19
Valores en coma flotante
 regulación PID, 9-93
 representación de, 5-4
Variable del proceso, convertir, 9-93
Variables, forzar, 4-37
VDE 0160, A-2
Velocidad de transferencia
 ajustar los interruptores del cable PC/PPI, 3-5, 7-39
 posición de los interruptores DIP en el cable PC/PPI, 7-36, A-91
Vibraciones fuertes, utilizar topes, 2-7
Vida útil de los relé, A-5
Visualizador de textos TD 200, manual del usuario, iv

W

Windows NT, instalar hardware, 7-8
WRITE_PM, 11-13

X

XOR, 10-25

Descripción	CPU 221	CPU 222	CPU 224	CPU 226
Tamaño del programa de usuario	2K palabras	2K palabras	4K palabras	4K palabras
Tamaño de los datos de usuario	1K palabras	1K palabras	2,5K palabras	2,5K palabras
Imagen del proceso de las entradas	I0.0 a I15.7	I0.0 a I15.7	I0.0 a I15.7	I0.0 a I15.7
Imagen del proceso de las salidas	Q0.0 a Q15.7	Q0.0 a Q15.7	Q0.0 a Q15.7	Q0.0 a Q15.7
Entradas analógicas (sólo lectura)	—	AIW0 a AIW30	AIW0 a AIW62	AIW0 a AIW62
Salidas analógicas (sólo escritura)	—	AQW0 a AQW30	AQW0 a AQW62	AQW0 a AQW62
Memoria de variables (V) ¹	VB0.0 a VB2047.7	VB0.0 a VB2047.7	VB0.0 a VB5119.7	VB0.0 a VB5119.7
Memoria local (L) ²	LB0.0 a LB63.7	LB0.0 a LB63.7	LB0.0 a LB63.7	LB0.0 a LB63.7
Área de marcas (M)	M0.0 a M31.7	M0.0 a M31.7	M0.0 a M31.7	M0.0 a M31.7
Marcas especiales (SM) Sólo lectura	SM0.0 a SM179.7 SM0.0 a SM29.7	SM0.0 a SM299.7 SM0.0 a SM29.7	SM0.0 a SM299.7 SM0.0 a SM29.7	SM0.0 a SM299.7 SM0.0 a SM29.7
Temporizadores	256 (T0 a T255)	256 (T0 a T255)	256 (T0 a T255)	256 (T0 a T255)
Retardo a la conexión memorizado 1 ms	T0, T64	T0, T64	T0, T64	T0, T64
Retardo a la conexión memorizado 10 ms	T1 a T4, T65 a T68	T1 a T4, T65 a T68	T1 a T4, T65 a T68	T1 a T4, T65 a T68
Retardo a la conexión memorizado 100 ms	T5 a T31, T69 a T95	T5 a T31, T69 a T95	T5 a T31, T69 a T95	T5 a T31, T69 a T95
Retardo a la con./descon. 1 ms	T32, T96	T32, T96	T32, T96	T32, T96
Retardo a la con./descon. 10 ms	T33 a T36, T97 a T100	T33 a T36, T97 a T100	T33 a T36, T97 a T100	T33 a T36, T97 a T100
Retardo a la con./descon. 100 ms	T37 a T63, T101 a T255	T37 a T63, T101 a T255	T37 a T63, T101 a T255	T37 a T63, T101 a T255
Contadores	C0 a C255	C0 a C255	C0 a C255	C0 a C255
Contadores rápidos	HC0, HC3, HC4, HC5	HC0, HC3, HC4, HC5	HC0 a HC5	HC0 a HC5
Relés de control secuencial (S)	S0.0 a S31.7	S0.0 a S31.7	S0.0 a S31.7	S0.0 a S31.7
Acumuladores	AC0 a AC3	AC0 a AC3	AC0 a AC3	AC0 a AC3
Salto a metas	0 a 255	0 a 255	0 a 255	0 a 255
Llamadas a subrutinas	0 a 63	0 a 63	0 a 63	0 a 63
Rutinas de interrupción	0 a 127	0 a 127	0 a 127	0 a 127
Lazos PID	0 a 7	0 a 7	0 a 7	0 a 7
Puerto	Puerto 0	Puerto 0	Puerto 0	Puerto 0, puerto 1
¹ Todo el contenido de la memoria V se puede guardar en la memoria no volátil.				
² STEP 7-Micro/WIN 32 (versión 3.0 o posterior) reserva LB60 a LB63.				

AWL	Página	AWL	Página	AWL	Página	AWL	Página	AWL	Página
=	9-6	AW <=	9-11	HTA	9-139	LPP	9-198	OW <>	9-11
+D	9-74	AW =	9-11	IBCD	9-130	LPS	9-197	PID	9-87
−D	9-74	AW >	9-11	INCB	9-79	LRD	9-197	PLS	9-49
*D	9-76	AW >=	9-11	INCD	9-80	LSCR	9-157	R	9-7
/D	9-76	AW <>	9-11	INCW	9-79	MOVB	9-102	RCV	9-186
+I	9-73	BCDI	9-130	INVB	9-118	MOVD	9-102	RI	9-8
−I	8-2	BIR	9-106	INVD	9-118	MOVR	9-102	RLB	9-123
=I	9-6	BIW	9-106	INVW	9-118	MOVW	9-102	RLD	9-125
*I	9-75	BMB	9-103	ITA	9-140	MUL	9-77	RLW	9-124
/I	9-75	BMD	9-103	ITB	9-133	NEXT	9-154	ROUND	9-131
+R	9-82	BMW	9-103	ITD	9-132	NETR	9-180	RRB	9-123
−R	9-82	BTI	9-133	JMP	9-148	NETW	9-180	RRD	9-125
*R	9-83	CALL	9-149	LBL	9-148	NOP	9-8	RRW	9-124
/R	9-83	COS	9-86	LD	9-2	NOT	9-4	RTA	9-143
A	9-2	CRET	9-149	LD >	9-12	O	9-2	S	9-7
AB <=	9-10	CRETI	9-171	LDB <=	9-10	OB =	9-10	SCRE	9-157
AB =	9-10	CTD	9-23	LDB =	9-10	OB >=	9-10	SCRT	9-157
AB >	9-10	CTU	9-23	LDB >=	9-10	OB >	9-10	SEG	9-137
AB <	9-10	CTUD	9-23	LDB >	9-10	OB <	9-10	SHRB	9-127
AB >=	9-10	DECB	9-79	LDB <	9-10	OB <=	9-10	SI	9-8
AB <>	9-10	DECD	9-80	LDB <>	9-10	OB <>	9-10	SIN	9-86
AD <	9-12	DECO	9-135	LDD >=	9-12	OD <	9-12	SLB	9-120
AD <=	9-12	DECW	9-79	LDD <	9-12	OD <=	9-12	SLD	9-122
AD =	9-12	DISI	9-173	LDD <=	9-12	OD =	9-12	SLW	9-121
AD >	9-12	DIV	9-77	LDD =	9-12	OD >	9-12	SPA	9-196
AD >=	9-12	DTA	9-142	LDD >	9-11	OD >=	9-12	SQRT	9-85
AD <>	9-12	DTCH	9-169	LDD <>	9-12	OD <>	9-12	SRB	9-120
AENO	9-168	DTI	9-132	LDI	9-3	OI	9-3	SRD	9-122
AI	9-3	DTR	9-130	LDN	9-2	OLD	9-197	SRW	9-121
ALD	9-197	ED	9-4	LDNI	9-3	ON	9-2	STOP	9-145
AN	9-2	ENCO	9-135	LDR =	9-13	ONI	9-3	SWAP	9-105
ANDB	9-114	FIN	9-145	LDR <	9-13	OR =	9-13	TAN	9-86
ANDD	9-116	ENI	9-173	LDR <=	9-13	OR <	9-13	TODR	9-71
ANDW	9-115	EU	9-4	LDR >	9-13	OR <=	9-13	TODW	9-71
ANI	9-3	EXP	9-86	LDR >=	9-13	OR >	9-13	TOF	9-15
AR =	9-13	FIFO	9-111	LDR <>	9-13	OR >=	9-13	TON	9-15
AR <	9-13	FILL	9-113	LDS	9-198	OR <>	9-13	TONR	9-15
AR <=	9-13	FND <	9-109	LDW <=	9-11	ORB	9-114	TRUNC	9-131
AR >	9-13	FND <>	9-109	LDW <	9-11	ORD	9-116	WDR	9-146
AR >=	9-13	FND =	9-109	LDW =	9-11	ORW	9-115	XMT	9-186
AR <>	9-13	FND >	9-109	LDW >	9-11	OW <	9-11	XORB	9-114
ATCH	9-169	FOR	9-154	LDW >=	9-11	OW <=	9-11	XORD	9-116
ATH	9-139	GPA	9-196	LDW <>	9-11	OW =	9-11	XORW	9-115
ATT	9-107	HDEF	9-27	LIFO	9-112	OW >	9-11		
AW <	9-11	HSC	9-27	LN	9-85	OW >=	9-11		

Siemens AG
A&D AS E 81

Oestliche Rheinbrueckenstr. 50
D-76181 Karlsruhe
R.F.A.

Remitente:

Nombre: _ _ _ _ _
Cargo: _ _ _ _ _
Empresa: _ _ _ _ _
Calle: _ _ _ _ _
Código postal: _ _ _ _ _
Población: _ _ _ _ _
País: _ _ _ _ _
Teléfono: _ _ _ _ _

Indique el ramo de la industria al que pertenece:

☐ Industria del automóvil	☐ Industria farmacéutica
☐ Industria química	☐ Industria del plástico
☐ Industria eléctrica	☐ Industria papelera
☐ Industria alimentaria	☐ Industria textil
☐ Control e instrumentación	☐ Transportes
☐ Industria mecánica	☐ Otros _ _ _ _ _
☐ Industria petroquímica	



Observaciones/sugerencias

Sus observaciones y sugerencias nos permiten mejorar la calidad y utilidad de nuestra documentación. Por ello le rogamos que rellene el presente formulario y lo envíe a Siemens.

Responda, por favor, a las siguientes preguntas dando una puntuación comprendida entre 1 = muy bien y 5 = muy mal:

1.	¿Corresponde el contenido del manual a sus exigencias ?	☐
2.	¿Resulta fácil localizar las informaciones requeridas ?	☐
3.	¿Es comprensible el texto ?	☐
4.	¿Corresponde el nivel de los detalles técnicos a sus exigencias ?	☐
5.	¿Qué opina de la calidad de las ilustraciones y tablas ?	☐

En las líneas siguientes puede exponer los problemas concretos que se le hayan planteado al manejar el manual:
