

## SIMATIC

### Sistema de automatización S7-200

#### Manual del sistema

Referencia del manual:

**6ES7298-8FA01-8DH0**

Prólogo, Índice	
Introducción a los Micro-PLCs S7-200	<b>1</b>
Instalar un Micro-PLC S7-200	<b>2</b>
Instalar y utilizar el software de STEP 7-Micro/WIN,	<b>3</b>
Introducir un programa de ejemplo	<b>4</b>
Funciones adicionales de STEP 7-Micro/WIN	<b>5</b>
Nociones básicas para programar una CPU S7-200	<b>6</b>
Memoria de la CPU: tipos de datos y direccionamiento	<b>7</b>
Entradas y salidas	<b>8</b>
Comunicación en redes con CPUs S7-200	<b>9</b>
Juego de operaciones	<b>10</b>
<b>Anexos</b>	
Datos técnicos	<b>A</b>
Tabla de consumo de corriente	<b>B</b>
Códigos de error	<b>C</b>
Marcas especiales (SM)	<b>D</b>
Utilizar STEP 7-Micro/WIN con STEP 7 y STEP 7-Micro/DOS	<b>E</b>
Tiempos de ejecución básicos de las operaciones AWL	<b>F</b>
Números de referencia	<b>G</b>
Eliminar errores	<b>H</b>
Índice alfabético	

## Consignas de seguridad para el usuario



### 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, lesiones corporales graves o daños materiales considerables.



### Cuidado

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

## 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 o 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.

## Marca registrada

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

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

### Copyright © Siemens AG 1998 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 Nuernberg

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 1998  
Sujeto a cambios sin previo aviso.

6ES7298-8FA01-8DH0

# 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 numerosas aplicaciones pequeñas de control. La gran variedad de tamaños y fuentes de alimentación de las CPUs, así como las múltiples opciones de programación proporcionan la flexibilidad necesaria para solucionar las tareas de automatización.

El presente manual contiene informaciones acerca de cómo instalar y programar los Micro-PLCs S7-200, incluyendo los siguientes temas:

- Montar y cablear la CPU S7-200 y los módulos de ampliación e instalar el software de STEP 7-Micro/WIN.
- Estructurar y crear programas.
- Comprender la funcionalidad de la CPU, los tipos de datos y los modos de direccionamiento, el ciclo de la CPU, la protección con contraseña y la comunicación en redes.

El manual incluye también descripciones y ejemplos de las operaciones de programación, los tiempos de ejecución típicos de las operaciones, así como las hojas de datos técnicos de los equipos S7-200.

## 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 212 versión 1.01, CPU 214 versión 1.01, CPU 215 versión 1.02 y CPU 216 versión 1.02
- Paquetes del software de programación STEP 7-Micro/WIN (versión 2.1):
  - STEP 7-Micro/WIN 16 para Windows 3.1x (16 bits)
  - STEP 7-Micro/WIN 32 para Windows 95 y Windows NT (32 bits)

## Homologaciones

La gama SIMATIC S7-200 cumple las normas y prescripciones de las organizaciones que se nombran a continuación:

- 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, homologado (Industrial Control Equipment)
- Canadian Standards Association: CSA C22.2 n° 142, certificado (Process Control Equipment)
- Factory Mutual Research: FM clase I, categoría 2, grupo A, B, C, y D Hazardous Locations, T4A
- VDE 0160: equipos electrónicos de uso en instalaciones de corriente eléctrica

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

### Informaciones adicionales

Los siguientes manuales contienen informaciones más detalladas acerca de temas especiales:

- *Sistemas de periferia descentralizada ET 200*: describe cómo instalar y utilizar los productos ET 200 para la periferia descentralizada.
- 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. Si ya dispone de los conocimientos necesarios, consulte el índice para encontrar la información que necesite.

El manual está estructurado 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 directrices básicas para instalar las CPUs y los módulos de ampliación.
- El capítulo 3 ("Instalar y utilizar el software STEP 7-Micro/WIN") describe cómo instalar el software de programación y las funciones básicas del mismo.
- El capítulo 4 ("Introducir un programa de ejemplo") indica cómo crear un programa de ejemplo utilizando el software STEP 7-Micro/WIN.
- El capítulo 5 ("Funciones adicionales de STEP 7-Micro/WIN") describe cómo utilizar el Asistente TD 200 y el Asistente de operaciones S7-200, así como las demás funciones nuevas de STEP 7-Micro/WIN.
- Los capítulos 6 ("Nociones básicas para programar una CPU S7-200"), 7 ("Memoria de la CPU: tipos de datos y direccionamiento") y 8 ("Entradas y salidas") proporcionan información acerca de cómo la CPU S7-200 procesa los datos y ejecuta el programa de usuario.
- El capítulo 9 ("Comunicación en redes con CPUs S7-200") describe cómo conectar las CPUs S7-200 a diferentes tipos de redes.
- El capítulo 10 ("Juego de operaciones") contiene explicaciones y ejemplos de las operaciones de programación utilizadas por las CPUs S7-200.

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

### 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 la siguiente dirección 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>

# Indice

<b>1</b>	<b>Introducción a los Micro-PLCs S7-200</b>	<b>1-1</b>
1.1	Funciones de los diversos Micro-PLCs S7-200 .....	1-2
1.2	Principales componentes de un Micro-PLC S7-200 .....	1-4
<b>2</b>	<b>Instalar un Micro-PLC S7-200</b>	<b>2-1</b>
2.1	Preparar el montaje .....	2-2
2.2	Montar y desmontar un Micro-PLC S7-200 .....	2-5
2.3	Instalar el cableado de campo .....	2-8
2.4	Utilizar circuitos de supresión .....	2-13
2.5	Alimentación de corriente .....	2-15
<b>3</b>	<b>Instalar y utilizar el software STEP 7-Micro/WIN</b>	<b>3-1</b>
3.1	Instalar el software de STEP 7-Micro/WIN .....	3-2
3.2	Utilizar STEP 7-Micro/WIN para configurar la comunicación .....	3-4
3.3	Establecer la comunicación con una CPU S7-200 .....	3-7
3.4	Configurar las preferencias para STEP 7-Micro/WIN .....	3-25
3.5	Crear y guardar un proyecto .....	3-26
3.6	Crear un programa .....	3-27
3.7	Crear un bloque de datos .....	3-32
3.8	Utilizar la tabla de estado .....	3-34
3.9	Utilizar el direccionamiento simbólico .....	3-36
<b>4</b>	<b>Introducir un programa de ejemplo</b>	<b>4-1</b>
4.1	Crear un programa para una aplicación de ejemplo .....	4-2
4.2	Tarea: Crear un proyecto .....	4-6
4.3	Tarea: Crear una tabla de símbolos .....	4-8
4.4	Tarea: Introducir el programa en KOP .....	4-10
4.5	Tarea: Crear una tabla de estado .....	4-14
4.6	Tarea: Cargar y observar el programa de ejemplo .....	4-15
<b>5</b>	<b>Funciones adicionales de STEP 7-Micro/WIN</b>	<b>5-1</b>
5.1	Utilizar el Asistente TD 200 para configurar el visualizador de textos TD 200 .....	5-2
5.2	Utilizar el Asistente de operaciones S7-200 .....	5-12
5.3	Utilizar el Asistente para filtrar entradas analógicas .....	5-14

5.4	Utilizar las referencias cruzadas .....	5-17
5.5	Usar la lista de elementos utilizados .....	5-18
5.6	Utilizar la función para buscar y reemplazar .....	5-19
5.7	Documentar el programa .....	5-21
5.8	Imprimir el programa .....	5-23
<b>6</b>	<b>Nociones básicas para programar una CPU S7-200</b>	<b>6-1</b>
6.1	Crear una solución de automatización con un Micro-PLC .....	6-2
6.2	Programas S7-200 .....	6-4
6.3	Lenguajes de programación para las CPUs S7-200 .....	6-5
6.4	Elementos básicos para estructurar un programa .....	6-8
6.5	El ciclo de la CPU .....	6-10
6.6	Ajustar el modo de operación de la CPU .....	6-13
6.7	Determinar una contraseña para la CPU .....	6-14
6.8	Comprobar y observar el programa .....	6-16
6.9	Eliminar errores de las CPUs S7-200 .....	6-19
<b>7</b>	<b>Memoria de la CPU: tipos de datos y direccionamiento</b>	<b>7-1</b>
7.1	Direccionamiento directo de las áreas de memoria de la CPU .....	7-2
7.2	Direccionamiento indirecto de las áreas de memoria de la CPU .....	7-9
7.3	Respaldar datos en la CPU S7-200 .....	7-11
7.4	Utilizar el programa para guardar datos en la memoria no volátil .....	7-16
7.5	Guardar el programa en un cartucho de memoria .....	7-17
<b>8</b>	<b>Entradas y salidas</b>	<b>8-1</b>
8.1	Entradas y salidas integradas y adicionales .....	8-2
8.2	Utilizar filtros de entrada para suprimir interferencias .....	8-5
8.3	Configurar los estados de señal para las salidas .....	8-6
8.4	Entradas y salidas rápidas .....	8-7
8.5	Potenciómetros analógicos .....	8-8
<b>9</b>	<b>Comunicación en redes con CPUs S7-200</b>	<b>9-1</b>
9.1	Protocolos de comunicación de las CPUs S7-200 .....	9-2
9.2	Componentes para la comunicación en redes .....	9-6
9.3	Utilizar un cable PC/PPI para la comunicación .....	9-9
9.4	Utilizar una tarjeta MPI o CP para la comunicación .....	9-13
9.5	Comunicación en redes de periferia descentralizada (DP) .....	9-15
9.6	Rendimiento de la red .....	9-28
<b>10</b>	<b>Juego de operaciones</b>	<b>10-1</b>
10.1	Márgenes válidos para las CPUs S7-200 .....	10-2
10.2	Operaciones con contactos .....	10-4
10.3	Operaciones de comparación .....	10-7

10.4	Operaciones con salidas .....	10-10
10.5	Operaciones con temporizadores, contadores, contadores rápidos, reloj de tiempo real y salida de impulsos .....	10-13
10.6	Operaciones aritméticas y de regulación PID .....	10-50
10.7	Operaciones para incrementar y decrementar .....	10-66
10.8	Operaciones de transferencia, inicializar memoria y tabla .....	10-68
10.9	Operaciones de desplazamiento y rotación .....	10-78
10.10	Operaciones de control del programa .....	10-84
10.11	Operaciones lógicas de pilas .....	10-99
10.12	Operaciones lógicas .....	10-102
10.13	Operaciones de conversión .....	10-108
10.14	Operaciones de interrupción y comunicación .....	10-114
<b>A</b>	<b>Datos técnicos</b>	<b>A-1</b>
A.1	Datos técnicos generales .....	A-3
A.2	CPU 212 alimentación DC, entradas DC, salidas DC .....	A-6
A.3	CPU 212 alimentación AC, entradas DC, salidas de relé .....	A-8
A.4	CPU 212 alimentación AC 24 V, entradas DC, salidas de relé .....	A-10
A.5	CPU 212 alimentación AC, entradas AC, salidas AC .....	A-12
A.6	CPU 212 alimentación AC, entradas DC tipo fuente, salidas de relé ....	A-14
A.7	CPU 212 alimentación AC, entradas AC 24 V, salidas AC .....	A-16
A.8	CPU 212 alimentación AC, entradas AC, salidas de relé .....	A-18
A.9	CPU 214 alimentación DC, entradas DC, salidas DC .....	A-20
A.10	CPU 214 alimentación AC, entradas DC, salidas de relé .....	A-22
A.11	CPU 214 alimentación AC, entradas AC, salidas AC .....	A-24
A.12	CPU 214 alimentación AC, entradas tipo fuente DC, salidas de relé ....	A-26
A.13	CPU 214 alimentación AC, entradas AC 24 V, salidas AC .....	A-28
A.14	CPU 214 alimentación AC, entradas AC, salidas de relé .....	A-30
A.15	CPU 215 alimentación DC, entradas DC, salidas DC .....	A-32
A.16	CPU 215 alimentación AC, entradas DC, salidas de relé .....	A-34
A.17	CPU 216 alimentación DC, entradas DC, salidas DC .....	A-36
A.18	CPU 216 alimentación AC, entradas DC, salidas de relé .....	A-38
A.19	Módulo de ampliación EM 221, 8 entradas digitales DC 24 V .....	A-40
A.20	Módulo de ampliación EM 221, 8 entradas digitales AC 120 V .....	A-41
A.21	Módulo de ampliación EM 221, 8 entradas digitales tipo fuente DC 24 V	A-42
A.22	Módulo de ampliación EM221, 8 entradas digitales AC 24 V .....	A-43
A.23	Módulo de ampliación EM222, 8 salidas digitales DC 24 V .....	A-44
A.24	Módulo de ampliación EM 222, 8 salidas digitales de relé .....	A-45
A.25	Módulo de ampliación EM222, 8 salidas digitales AC 120/230 V .....	A-46

A.26	Módulo de ampliación EM 223, 4 entradas digitales DC 24 V / 4 salidas digitales DC 24 V .....	A-48
A.27	Módulo de ampliación EM 223, 8 entradas digitales DC 24 V / 8 salidas digitales DC 24 V .....	A-50
A.28	Módulo de ampliación EM 223, 16 entradas digitales DC 24 V / 16 salidas digitales DC 24 V .....	A-52
A.29	Módulo de ampliación EM 223, 4 entradas digitales 4 DC V / 4 salidas digitales de relé .....	A-54
A.30	Módulo de ampliación EM 223, 4 entradas digitales AC 120 V / 4 salidas digitales AC 120/230 V .....	A-55
A.31	Módulo de ampliación EM 223, 8 entradas digitales DC 24 V / 8 salidas digitales de relé .....	A-56
A.32	Módulo de ampliación EM 223, 16 entradas digitales DC 24 V / 16 salidas digitales de relé .....	A-58
A.33	Módulo de ampliación EM 231, 3 entradas analógicas AI de 12 bits ....	A-60
A.34	Módulo de ampliación EM 232, 2 salidas analógicas AQ de 12 bits ....	A-66
A.35	Módulo de ampliación EM 235, 3 entradas analógicas AI / 1 salida analógica AQ de 12 bits .....	A-69
A.36	Cartucho de memoria 8K x 8 .....	A-78
A.37	Cartucho de memoria 16K x 8 .....	A-79
A.38	Cartucho de pila .....	A-80
A.39	Cable de E/S de ampliación .....	A-81
A.40	Cable PC/PPI .....	A-82
A.41	Simulador de entradas DC para la CPU 212 .....	A-84
A.42	Simulador de entradas DC para la CPU 214 .....	A-85
A.43	Simulador de entradas DC para las CPUs 215/216 .....	A-86
<b>B</b>	<b>Tabla de consumo de corriente</b>	<b>B-1</b>
<b>C</b>	<b>Códigos de error</b>	<b>C-1</b>
C.1	Códigos de errores fatales y mensajes .....	C-2
C.2	Errores de programación del tiempo de ejecución .....	C-3
C.3	Violación de reglas de compilación .....	C-4
<b>D</b>	<b>Marcas especiales (SM)</b>	<b>D-1</b>
<b>E</b>	<b>Utilizar STEP 7-Micro/WIN con STEP 7 y STEP 7-Micro/DOS</b>	<b>E-1</b>
E.1	Utilizar STEP 7-Micro/WIN con STEP 7 .....	E-2
E.2	Importar archivos de STEP 7-Micro/DOS .....	E-4
<b>F</b>	<b>Tiempos de ejecución básicos de las operaciones AWL</b>	<b>F-1</b>
<b>G</b>	<b>Números de referencia</b>	<b>G-1</b>
<b>H</b>	<b>Eliminar errores</b>	<b>H-1</b>
	<b>Indice alfabético</b>	<b>Indice-1</b>

## 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 se adecúan para numerosas aplicaciones pequeñas de control. Además, los diversos tamaños y fuentes de alimentación de las CPUs ofrecen la flexibilidad necesaria para solucionar las tareas de automatización.

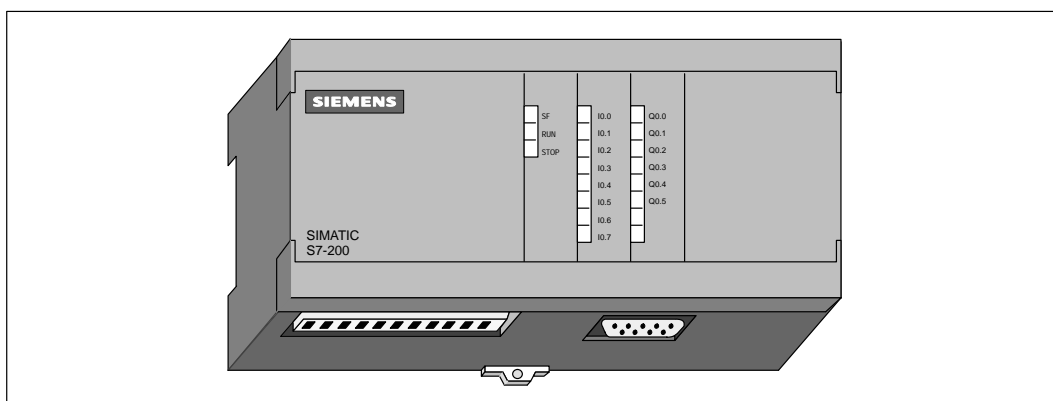


Figura 1-1 Micro-PLC S7-200

### Índice del capítulo

Apartado	Descripción	Página
1.1	Funciones de los diversos Micro-PLCs S7-200	1-2
1.2	Principales componentes de un Micro-PLC S7-200	1-4

## 1.1 Funciones de los diversos Micro-PLCs S7-200

### Equipos necesarios

La figura 1-2 muestra la estructura básica de un Micro-PLC S7-200 que incluye una CPU S7-200, un PC, el software de programación STEP 7-Micro/WIN 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.

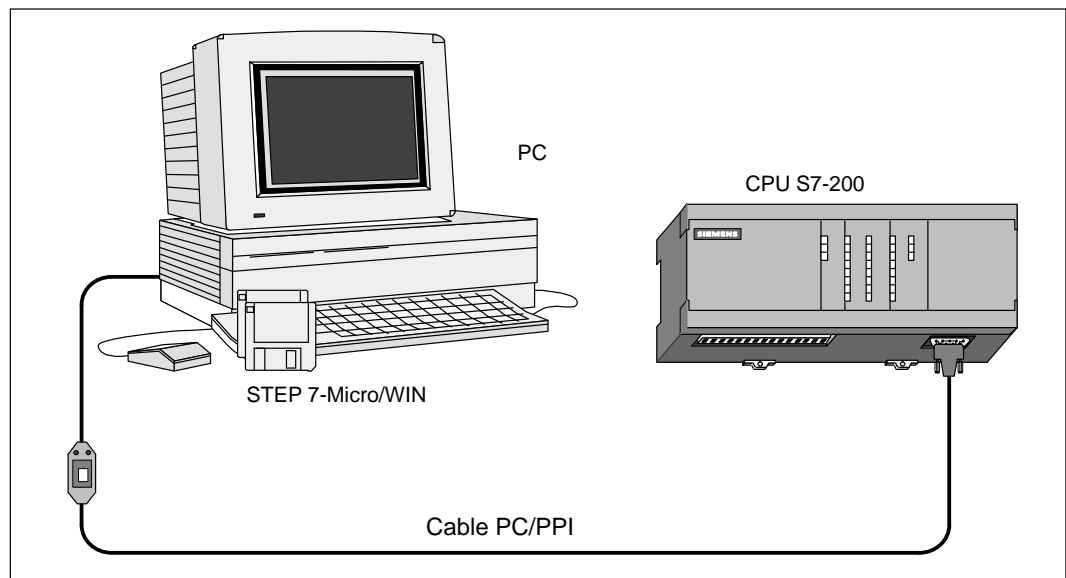


Figura 1-2 Componentes de un Micro-PLC S7-200

### Capacidad de las CPUs S7-200

La serie S7-200 comprende diversas CPUs. Por lo tanto, se dispone de una amplia gama de funciones que permiten diseñar soluciones de automatización a un precio razonable. La tabla 1-1 resume las principales funciones de cada CPU.

Tabla 1-1 Resumen de las CPUs S7-200

Función	CPU 212	CPU 214	CPU 215	CPU 216
<b>Tamaño físico</b>	160 mm x 80 mm x 62 mm	197 mm x 80 mm x 62 mm	218 mm x 80 mm x 62 mm	218 mm x 80 mm x 62 mm
<b>Memoria</b>				
Programa (EEPROM)	512 palabras	2K palabras	4K palabras	4K palabras
Datos de usuario	512 palabras	2K palabras	2,5K palabras	2,5K palabras
Marcas internas	128	256	256	256
Cartucho de memoria	No	Sí (EEPROM)	Sí (EEPROM)	Sí (EEPROM)
Cartucho de pila opcional	No	200 días (típ.)	200 días (típ.)	200 días (típ.)
Respaldo (condensador de alto rendimiento)	50 horas (típ.)	190 horas (típ.)	190 horas (típ.)	190 horas (típ.)
<b>Entradas/salidas (E/S)</b>				
E/S integradas	8 DI / 6 DQ	14 DI / 10 DQ	14 DI / 10 DQ	24 DI / 16 DQ
Módulos de ampliación (máx.)	2 módulos	7 módulos	7 módulos	7 módulos
Imagen del proceso de E/S	64 DI / 64 DQ	64 DI / 64 DQ	64 DI / 64 DQ	64 DI / 64 DQ
E/S analógicas (ampliación)	16 AI / 16 AQ	16 AI / 16 AQ	16 AI / 16 AQ	16 AI / 16 AQ
Filtros de entrada	No	Sí	Sí	Sí
<b>Operaciones</b>				
Velocidad de ejecución booleana	1,2 µs/operación	0,8 µs/operación	0,8 µs/operación	0,8 µs/operación
Contadores / temporizadores	64/64	128/128	256/256	256/256
Bucles FOR/NEXT	No	Sí	Sí	Sí
Aritmética en coma fija	Sí	Sí	Sí	Sí
Aritmética en coma flotante	No	Sí	Sí	Sí
PID	No	No	Sí	Sí
<b>Funciones adicionales</b>				
Contadores rápidos	1 S/W	1 S/W, 2 H/W	1 S/W, 2 H/W	1 S/W, 2 H/W
Potenciómetros analógicos	1	2	2	2
Salidas de impulsos	No	2	2	2
Interrupciones de comunicación	1 emisor / 1 receptor	1 emisor / 1 receptor	1 emisor / 2 receptores	2 emisores / 4 receptores
Interrupciones temporizadas	1	2	2	2
Entradas de interrupción de hardware	1	4	4	4
Reloj de tiempo real	No	Sí	Sí	Sí
<b>Comunicación</b>				
Interfaces	1 (RS-485)	1 (RS-485)	2 (RS-485)	2 (RS-485)
Protocolos asistidos	Interface 0:	PPI, Freeport	PPI, Freeport	PPI, Freeport, MPI
	Interface 1:	N/A	N/A	DP, MPI
Punto a punto	Sólo esclavo	Sí	Sí	Sí

## 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 aparato autónomo compacto que incorpora una unidad central de procesamiento (CPU), la 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.
- La fuente de alimentación proporciona corriente a la unidad central y a los módulos de ampliación conectados.
- Las entradas y salidas controlan el sistema de automatización. Las entradas vigilan las señales de los aparatos de campo (p.ej. sensores e interruptores) y las salidas vigilan las bombas, motores u otros dispositivos del proceso.
- El interface de comunicación permite conectar la CPU a una unidad de programación o a otros dispositivos. Algunas CPUs S7-200 disponen de dos interfaces de comunicación.
- Los diodos luminosos indican el modo de operación de la CPU (RUN o STOP), el estado de las entradas y salidas integradas, así como los posibles fallos del sistema que se hayan detectado.

Las figuras 1-3, 1-4 y 1-5 muestran las diferentes CPUs.

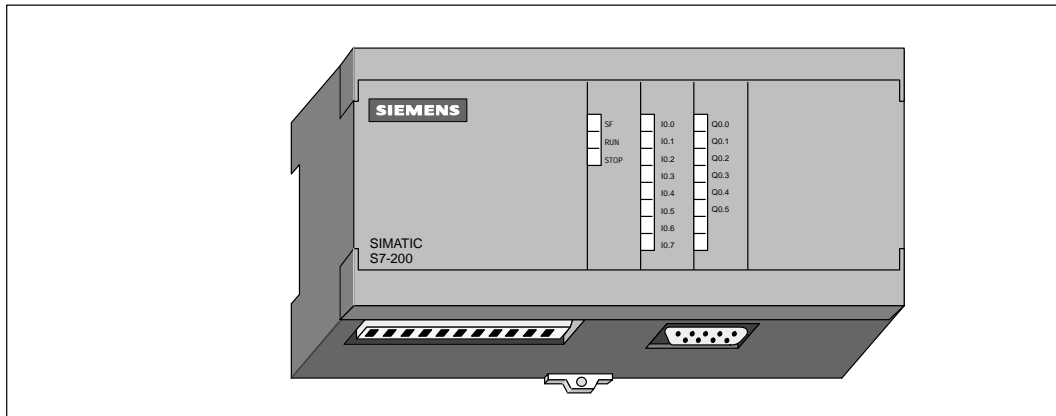


Figura 1-3 CPU 212

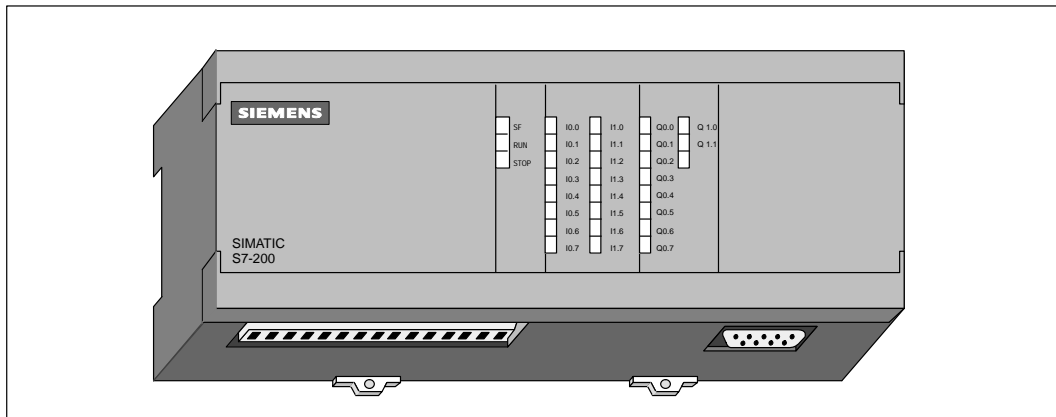


Figura 1-4 CPU 214

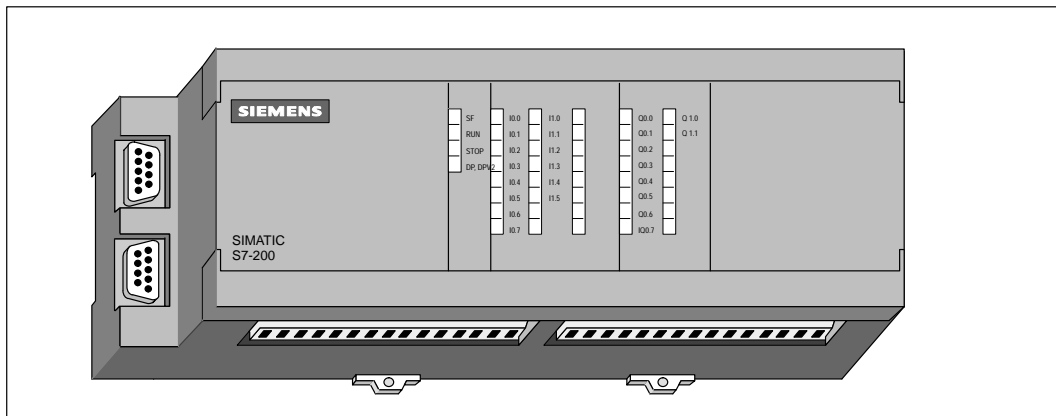


Figura 1-5 CPUs 215 y 216

### Módulos de ampliación

Los módulos de ampliación para las CPU S7-200 ofrecen un número determinado de entradas y salidas integradas. Si se conecta un módulo de ampliación se dispondrá de más entradas y salidas. Como muestra la figura 1-6, los módulos de ampliación disponen de un conector de bus para su unión al aparato central.

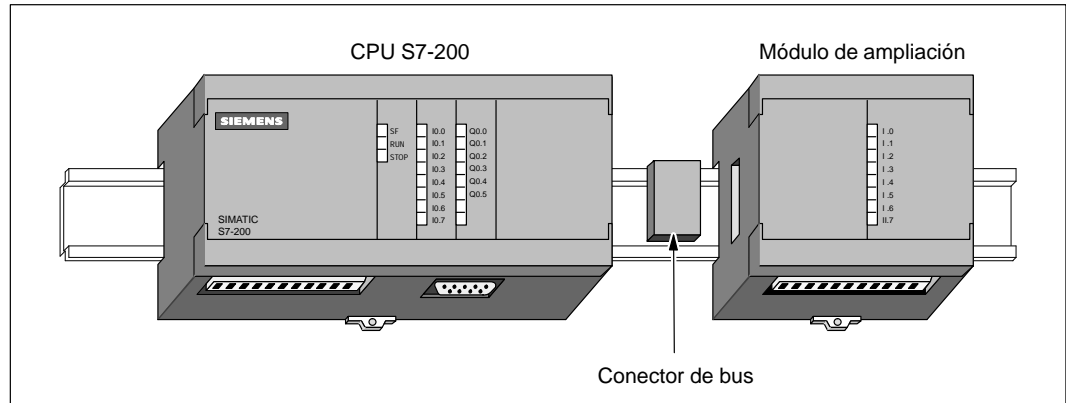


Figura 1-6 CPU con un módulo de ampliación

## 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 a tal efecto, o bien en un perfil soporte mediante los correspondientes ganchos de retención. Sus reducidas dimensiones permiten ahorrar espacio.

Este capítulo indica 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	2-5
2.3	Instalar el cableado de campo	2-8
2.4	Utilizar circuitos de supresión	2-13
2.5	Alimentación de corriente	2-15

## 2.1 Preparar el montaje

### Disposición

Los sistemas de automatización S7-200 se pueden disponer en un armario eléctrico o en un perfil soporte. Es posible montarlos de forma horizontal o vertical. Con objeto de flexibilizar aún más el montaje, se ofrecen también cables de conexión para los módulos de ampliación (módulos E/S). La figura 2-1 muestra dos ejemplos típicos de disposición.

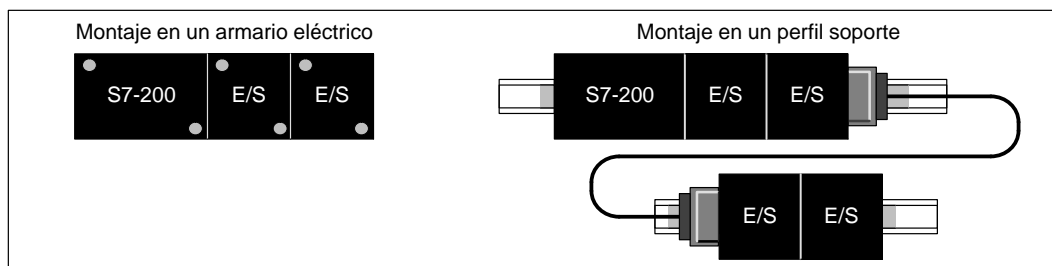


Figura 2-1 Disposición para el montaje

### Espacio necesario para montar una CPU S7-200

Al configurar 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 puede ser necesario reducir la carga de salida debido a las contracciones térmicas. Consulte en el Anexo A la hoja de datos técnicos de su CPU. Si monta la CPU y los módulos de ampliación en un perfil soporte, es recomendable asegurarlos con frenos.
- Para el montaje horizontal o vertical en un armario eléctrico, el espesor mínimo de éste último deberá ser de 75 mm (v. fig. 2-2).
- Si tiene pensado montar módulos de ampliación en posición horizontal o vertical, deberá dejar un margen mínimo de 25 mm de ambos lados de la unidad para poder montar y desmontarlos. El espacio adicional se requiere para acoplar y desacoplar el conector de bus.
- Al configurar 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.

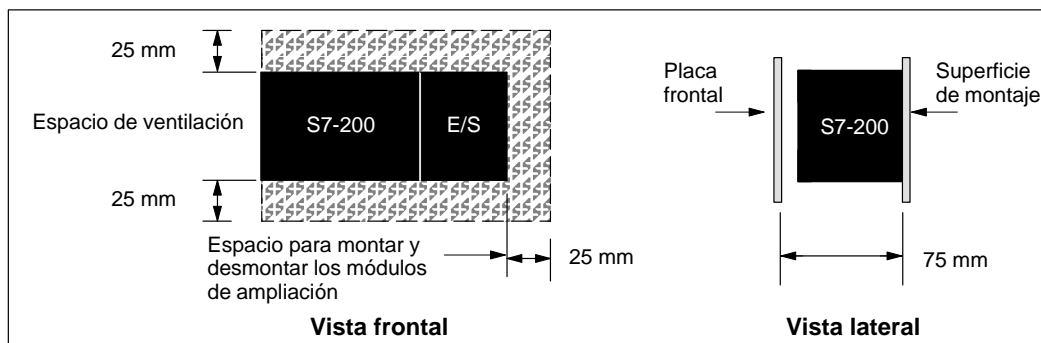


Figura 2-2 Espacio necesario para montar una CPU S7-200 de forma horizontal o vertical

### Montaje en un perfil soporte

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

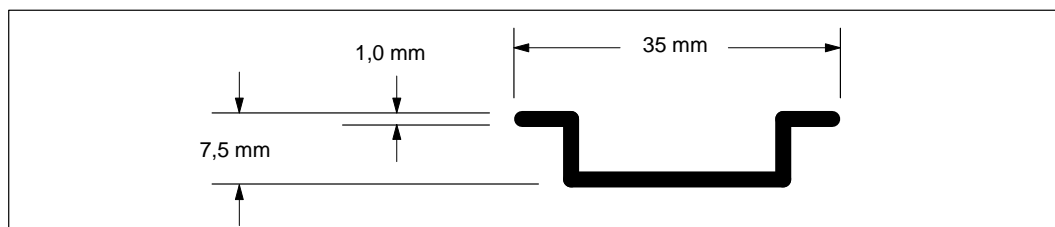


Figura 2-3 Dimensiones del perfil soporte

### 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 2-8 se indican las dimensiones necesarias para montar los diversos módulos S7-200.

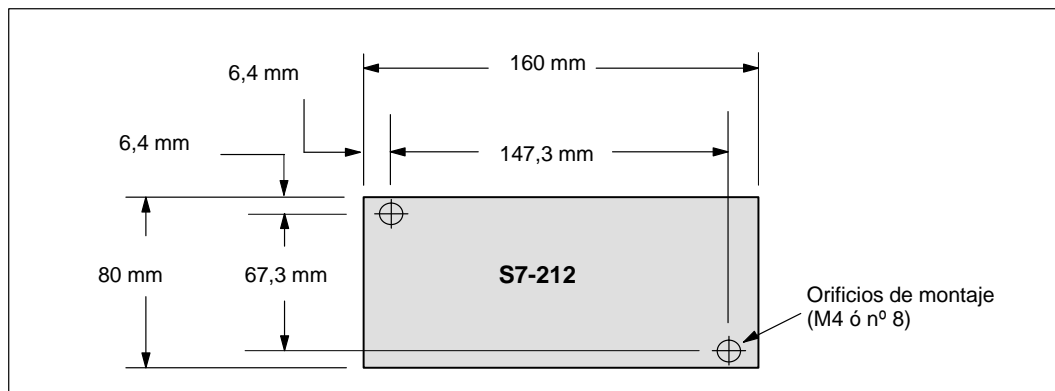


Figura 2-4 Dimensiones de montaje para una CPU S7-212

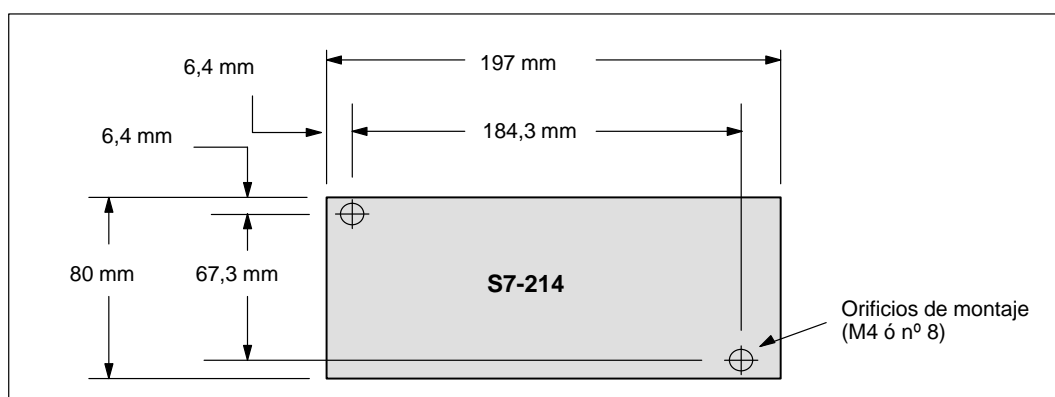


Figura 2-5 Dimensiones de montaje para una CPU S7-214

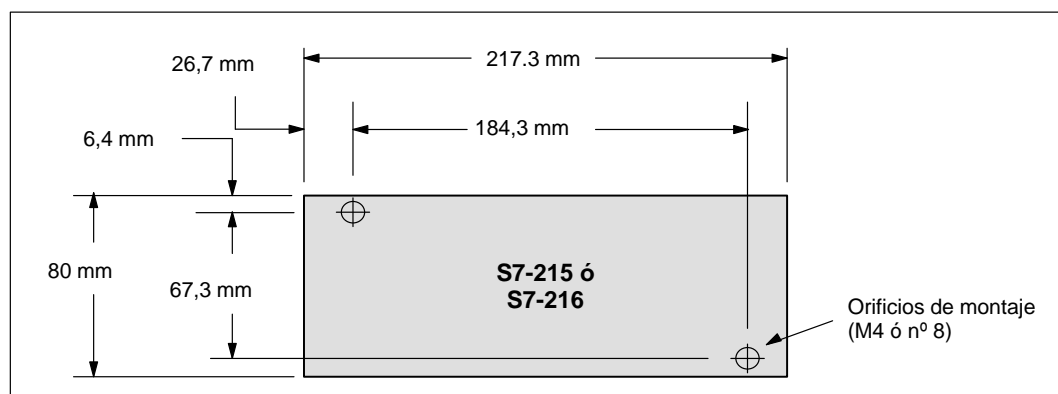


Figura 2-6 Dimensiones de montaje para una CPU S7-215 ó S7-216

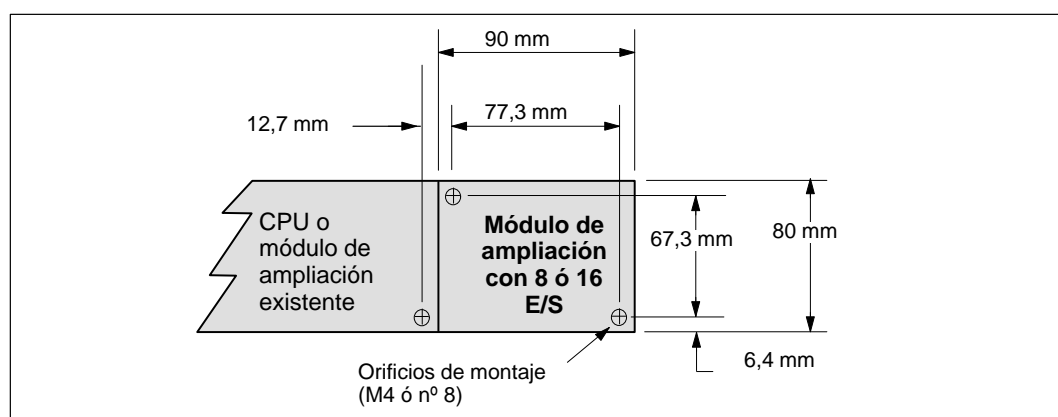


Figura 2-7 Dimensiones de montaje para un módulo de ampliación con 8 ó 16 E/S

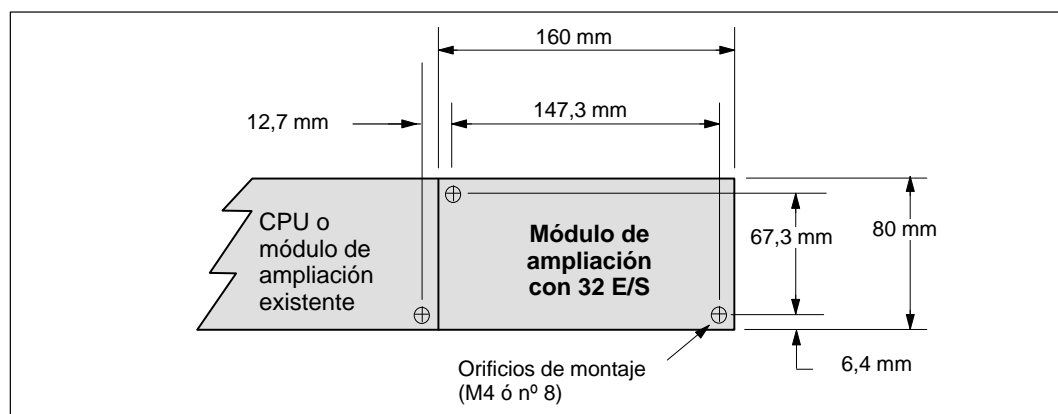


Figura 2-8 Dimensiones de montaje para un módulo de ampliación con 32 E/S

## 2.2 Montar y desmontar un Micro-PLC S7-200

### Montar una CPU S7-200 en un armario eléctrico



#### Precaución

Si se intenta montar o desmontar los sistemas de automatización y/o los equipos conectados durante el arranque, puede producirse un choque eléctrico.

Si antes del montaje o desmontaje no se ha desconectado por completo la alimentación 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 asegúrese de que la alimentación de los módulos S7-200 se haya desconectado antes de proceder a su montaje.

Para montar una CPU S7-200:

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:

1. Retire la tapa del interface de ampliación de bus de la carcasa del módulo existente. A tal efecto, inserte un destornillador entre la tapa y la carcasa, haciendo palanca con suavidad. Retire todos los restos de plástico y tenga cuidado de no deteriorar el módulo. La figura 2-9 muestra la utilización apropiada del destornillador.
2. Inserte el conector de bus en el interface de ampliación del módulo existente y asegúrese de que se enclave correctamente.
3. Verifique que el módulo esté orientado correctamente respecto a la CPU. Si utiliza un cable de ampliación, insértelo en el módulo de manera que el lado superior del cable señale hacia adelante.
4. Enchufe el módulo de ampliación al conector de bus, desplazando el módulo hacia el conector hasta que se enclave correctamente.

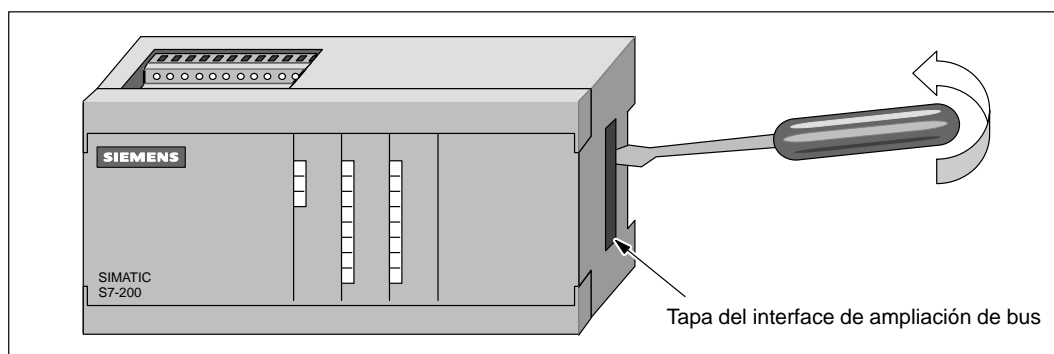


Figura 2-9 Retirar la tapa del interface de ampliación de bus de un módulo S7-200

## Montar una CPU S7-200 en un perfil soporte



---

### Precaución

Si se intenta montar o desmontar los sistemas de automatización y/o los equipos conectados durante el arranque, puede producirse un choque eléctrico.

Si antes del montaje o desmontaje no se ha desconectado por completo la alimentación 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 asegúrese de que la alimentación de los módulos S7-200 se haya desconectado antes de proceder a su montaje.

---

Para montar una CPU S7-200:

1. Fije el perfil soporte en el armario eléctrico utilizando tornillos cada 75 mm.
2. Abra el gancho de retención (ubicado en el lado inferior de la CPU) y enganche la parte posterior del módulo en el perfil soporte.
3. Cierre el gancho de retención y asegúrese de que el módulo haya enganchado correctamente en el perfil.

---

### Nota

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

---

Para montar un módulo de ampliación:

1. Retire la tapa del interface de ampliación de bus de la carcasa del módulo existente. A tal efecto, inserte un destornillador entre la tapa y la carcasa, haciendo palanca con suavidad. Retire todos los restos de plástico y tenga cuidado de no deteriorar el módulo. La figura 2-9 muestra la utilización apropiada del destornillador.
2. Inserte el conector de bus en el interface de ampliación del módulo existente y asegúrese de que se enclave correctamente.
3. Verifique que el módulo esté orientado correctamente respecto a la CPU. Si utiliza un cable de ampliación, insértelo en el módulo de manera que el lado superior del cable señale hacia adelante.
4. Abra el gancho de retención y enganche la parte posterior del módulo en el perfil soporte. Desplace el módulo hacia el conector hasta que se enclave correctamente.
5. Cierre el gancho de retención para fijar el módulo de ampliación al perfil soporte. Asegúrese de que el módulo se haya enganchado correctamente en el perfil.

## Desmontar los módulos S7-200



### Precaución

Si se intenta montar o desmontar los sistemas de automatización y/o los equipos conectados durante el arranque, puede producirse un choque eléctrico.

Si antes del montaje o desmontaje no se ha desconectado por completo la alimentación 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 asegúrese de que la alimentación de la CPU y de los módulos de ampliación S7-200 se haya desconectado antes de proceder a su montaje o desmontaje.

Para desmontar una CPU o un módulo de ampliación S7-200:

1. Desconecte todos los cables conectados al módulo que desee desmontar. Si el módulo está en medio de una estructura, las unidades situadas a la derecha o a la izquierda del mismo se deberán desplazar 25 mm como mínimo para poder desacoplar el conector de bus (v. fig. 2-10).
2. Afloje los tornillos de sujeción o abra el gancho de retención y desplace el módulo 25 mm para poder desacoplar el conector de bus. Éste se deberá desacoplar en ambos lados del módulo.
3. Retire el módulo del armario eléctrico o del perfil y monte otra unidad.



### Precaución

Si instala un módulo incorrecto, es posible que el programa contenido en el PLC 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. Si utiliza un cable de ampliación, insértelo en el módulo de manera que el lado superior del cable señale hacia adelante.

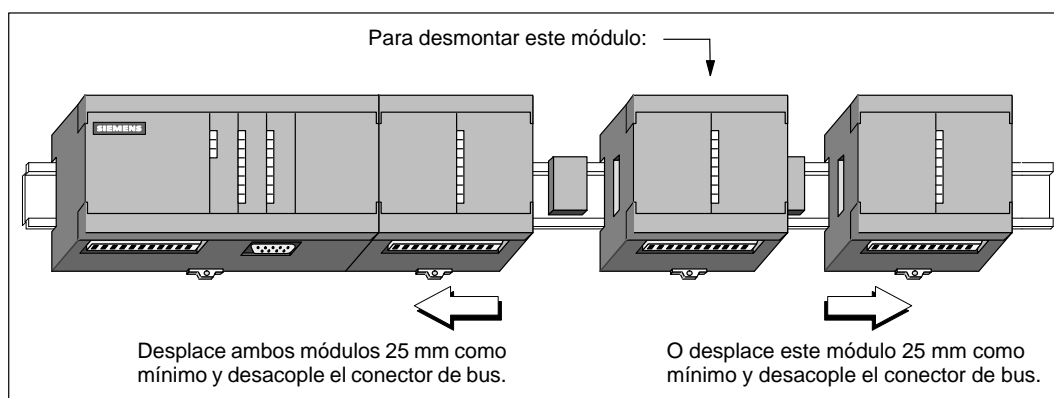


Figura 2-10 Desmontar un módulo de ampliación

## 2.3 Instalar el cableado de campo



---

### Precaución

Si se intenta montar o desmontar los sistemas de automatización y/o los equipos conectados durante el arranque, puede producirse un choque eléctrico.

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

Antes de instalar el cableado de campo es necesario tomar siempre las precauciones de seguridad adecuadas y verificar que estén desconectadas las fuentes de alimentación de los módulos del 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 el Micro-PLC S7-200 es necesario respetar todos los reglamentos, códigos y normas eléctricas aplicables. 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 un diámetro adecuado para la intensidad. Los módulos del S7-200 aceptan cables con sección de 1,50 mm<sup>2</sup> a 0,50 mm<sup>2</sup> (14 AWG a 22 AWG).
- Asegúrese de que los tornillos de los bornes no pasen 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 apareado con un cable activo.
- 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 módulos S7-200; de ser necesario, prevea 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 por ellos controlados.

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.
- 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 no 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.

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. La hoja de datos técnicos en el Anexo A incluye las especificaciones relativas a puntos de aislamiento y 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 la CPU 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 la CPU 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 o interfaces de comunicación de la CPU tienen el mismo potencial de referencia que la lógica de la CPU (excepto los interfaces DP).
- Las entradas y salidas (E/S) analógicas no están aisladas respecto a la lógica de la CPU. Las entradas analógicas son de tipo diferencial, es decir tienen una baja razón de rechazo en modo común.
- La lógica de la CPU está aislada de la tierra hasta DC 100 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 entre sí hasta AC 500 V.
- Las salidas de relé, las salidas AC y las entradas AC están aisladas de la lógica de la CPU hasta AC 1.500 V.
- Los grupos de salida AC y por relés están aislados entre sí hasta AC 1.500 V.
- La fase y el neutro de la alimentación en alterna están aislados de tierra, la lógica de la CPU y todas las E/S hasta AC 1.500 V.

### Uso del bloque de bornes opcional

El bloque de bornes opcional para el cableado de campo (v. fig. 2-11) permite independizar el cableado de campo de la CPU S7-200, facilitando en gran medida el desmontaje y remontaje de la misma. En el Anexo G se indica el número de referencia de dicho bloque.

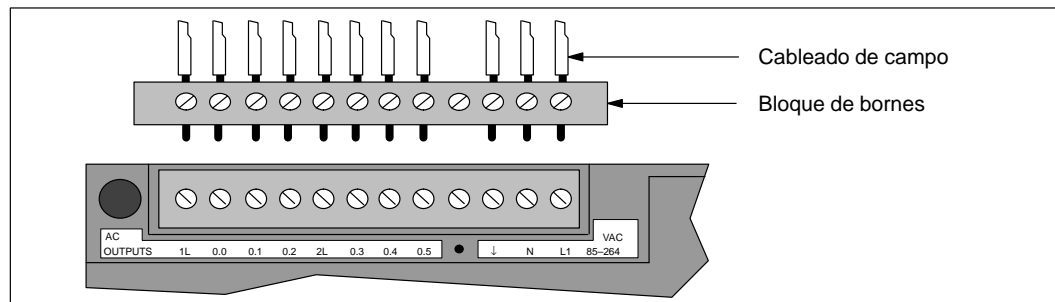


Figura 2-11 Bloque de bornes opcional para el cableado de campo

### Reglas para la instalación con corriente alterna

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

- Instale un interruptor unipolar (1) para cortar la alimentación de la CPU, todos circuitos de entrada y todos circuitos de salida (la carga).
- Prevea dispositivos de sobrecorriente (2) para proteger la alimentación de la CPU, las salidas y 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 24 VDC para sensores (3) integrada en el Micro-PLC. Esta fuente para sensores está protegida contra cortocircuitos.
- Conecte todos los terminales de tierra del S7-200 por el camino más corto a tierra (4) 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 un sección de 14 AWG ó 1,5 mm<sup>2</sup>.
- La fuente de alimentación DC para sensores integrada en el módulo base puede usarse también para alimentar las entradas de dicho componente (5), las entradas DC de ampliación (6) y las bobinas de los relés del módulo de ampliación (7). Esta fuente para sensores está protegida contra cortocircuitos.

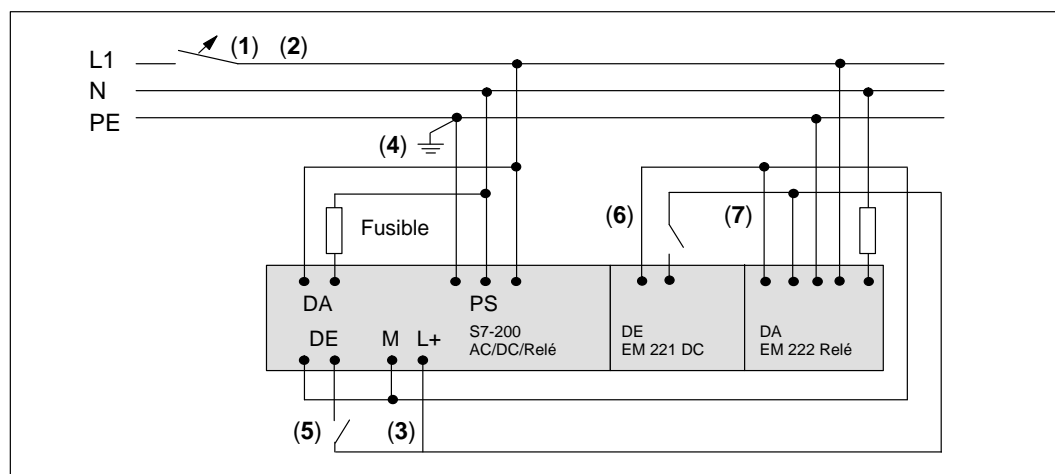


Figura 2-12 AC 120/230 V Uso de un interruptor de sobrecorriente único para proteger la CPU y la carga

## Reglas para la instalación con corriente continua

Seguidamente se indican reglas de carácter general para instalaciones con corriente continua aisladas (v. fig. 2-13).

- Instale un interruptor unipolar (1) para cortar la alimentación de la CPU, todos los circuitos de entrada y todos los circuitos de salida (la carga).
- Prevea dispositivos de sobrecorriente para proteger la alimentación de la CPU (2), las salidas (3) y las entradas (4). 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 24 VDC para sensores integrada en el Micro-PLC. Esta última fuente de alimentación dispone de una función de limitación interna de corriente.
- 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 (5) externos adecuados.
- Equipe las fuentes de alimentación DC no puestas a tierra con una resistencia y un condensador en paralelo (6) 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. También es posible crear un sistema DC puesto a tierra, conectando la fuente de alimentación DC con tierra (7).
- Conecte todos los terminales de tierra del S7-200 por el camino más corto con tierra (8) 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 mm<sup>2</sup>.
- 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

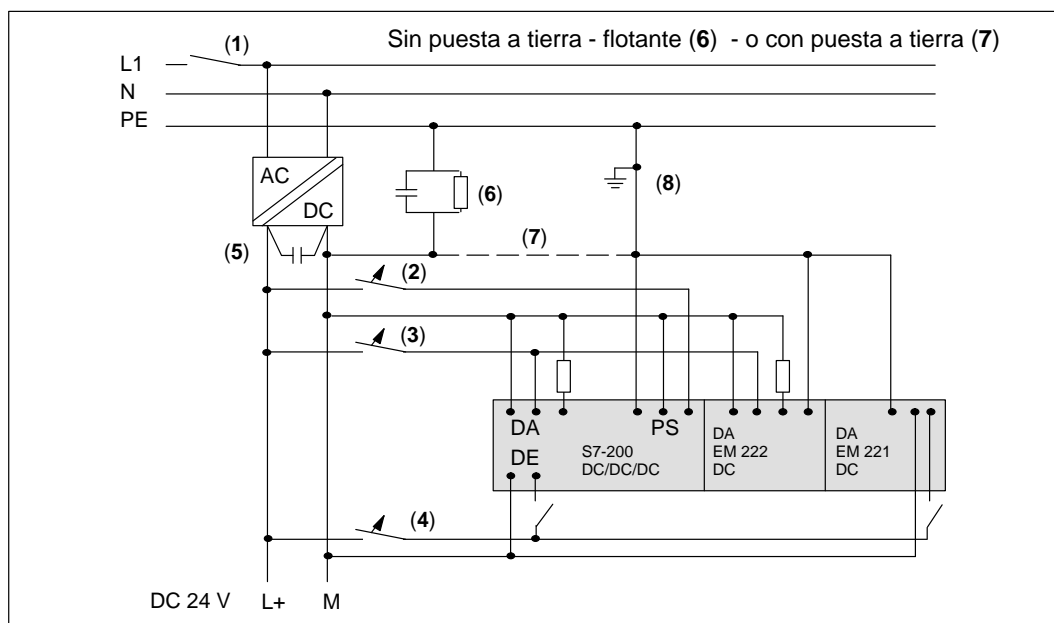


Figura 2-13 Instalación en una red de corriente continua aislada

## Reglas para la instalación en Norteamérica

A continuación se indican reglas de carácter general para el cableado de instalaciones en Norteamérica en aplicaciones que presenten diferentes tensiones en corriente alterna. La figura 2-14 se utilizará como referencia.

- Instale un interruptor unipolar (1) para cortar la alimentación de la CPU, todos los circuitos de entrada y todos los circuitos de salida (la carga).
- Prevea dispositivos de sobrecorriente para proteger la alimentación de la CPU (2), las salidas (3) y las entradas (4). Para mayor protección es posible instalar un fusible en cada salida.
- Efectúe las conexiones de alimentación de la red de alterna a la CPU, cargas de salida alimentadas en alterna, cargas conmutadas por relé, bien sea entre neutro y fase (5) o entre fase y fase (6).
- Conecte todos los terminales de tierra del S7-200 por el camino más corto con la tierra (7) 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 mm<sup>2</sup>.



### Cuidado

Las tensiones entre fase en redes con AC 230 V de tensión nominal entre fase y neutro superan la tensión nominal de alimentación y de las entradas y salidas del S7-200.

Debido a ello puede fallar el S7-200 y los equipos conectados al mismo.

Por consiguiente, la alimentación no se debe conectar entre fases si la tensión entre fases supera la tensión nominal del módulo S7-200 en cuestión.

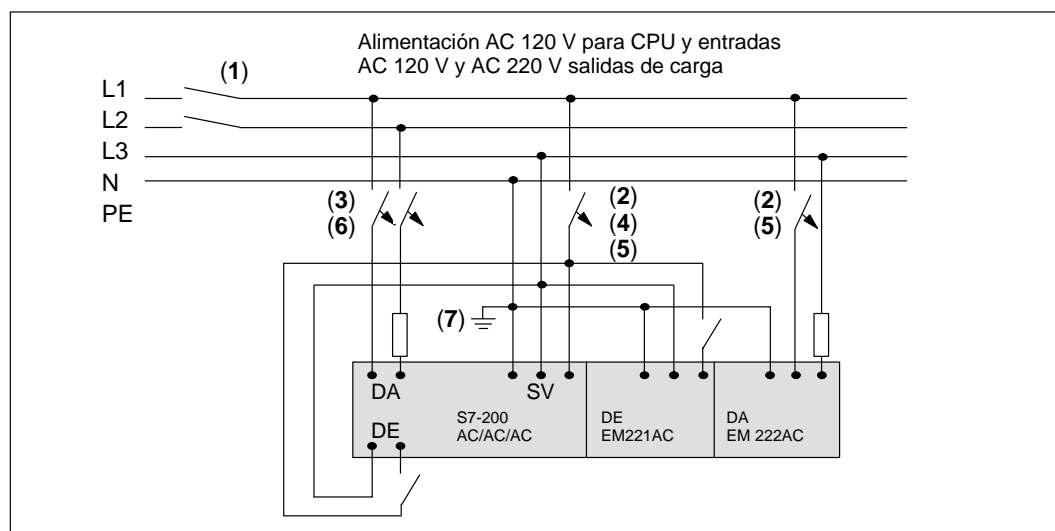


Figura 2-14 Instalación en una red de corriente alterna

## 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. Asegúrese de 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 contienen diodos zener adecuados para múltiples casos de aplicación. Para prevenir la sobrecarga de los diodos internos, utilice diodos supresores externos en caso de cargas inductivas elevadas o que se conmuten con frecuencia. Las figuras 2-15 y 2-16 muestran aplicaciones típicas para salidas DC a transistor.

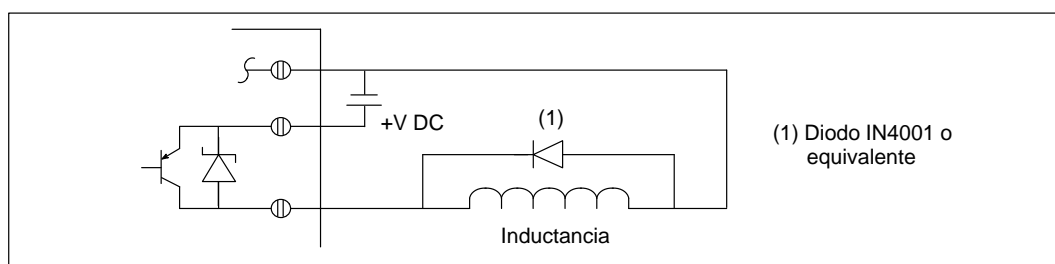


Figura 2-15 Protección por diodo supresor

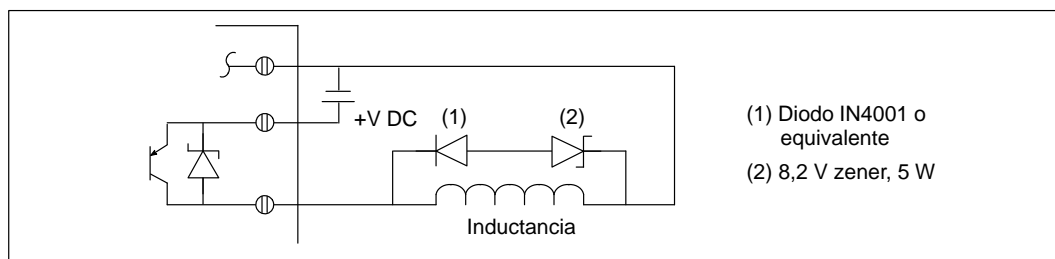


Figura 2-16 Protección por diodo zener

### Proteger relés que controlan corriente continua

La figura 2-17 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.

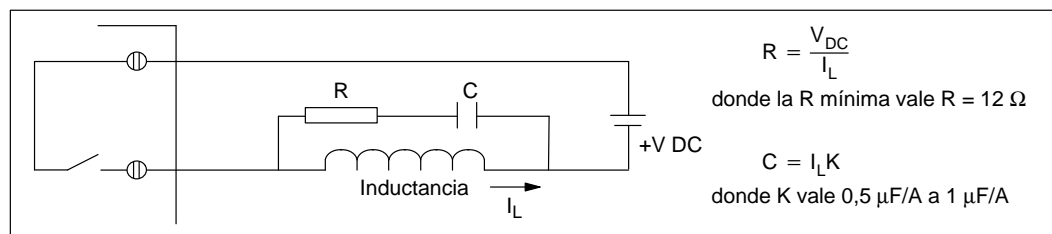


Figura 2-17 Red resistencia/condensador conectada a una carga DC controlada por relés

Para las aplicaciones con relé en corriente continua se puede utilizar también la protección por diodo supresor como muestran las figuras 2-15 y 2-16. Se permite una tensión de umbral de hasta 36 V si se utiliza un diodo zener conectado de forma inversa.

### Proteger relés y salidas AC que controlan corriente alterna

Si se utiliza un relé o una salida AC para conmutar cargas con 115 V/AC 230 V, se deben conectar redes resistencia/condensador entre los contactos del relé o las salidas AC como muestra la figura 2-18. También pueden utilizarse varistores de óxido metálico (MOV) para limitar la tensión de pico. Asegúrese de que la tensión de trabajo del varistor MOV sea como mínimo un 20% superior a la tensión nominal de fase.

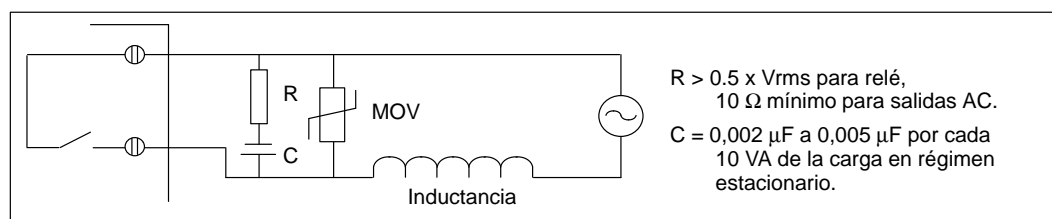


Figura 2-18 Carga AC con red conectada entre contactos del relé o salidas AC

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

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  $183 VA / 115 V = 1,59 A$ , 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  $68 \Omega$  porque es el valor estándar. Dimensionamiento del condensador =  $(17 VA / 10) \times 0,005 = 0,0085 \mu F$ ; elegir  $0,01 \mu F$  porque es el valor estándar.

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

## 2.5 Alimentación de corriente

Los módulos base del S7-200 tienen integrada una fuente de alimentación capaz de abastecer el módulo base, los módulos de ampliación y otras cargas que precisen 24 VDC. Utilice la información siguiente como guía al determinar cuánta energía (o corriente) puede suministrar el módulo central a la configuración en cuestión.

### Requisitos de alimentación

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

- Cada CPU dispone de una fuente de alimentación para sensores de 24 VDC que puede suministrar esta tensión para las entradas locales o para las bobinas de relés en los módulos de ampliación. Si el consumo de 24 VDC supera la corriente que es capaz de aportar el módulo CPU, entonces puede añadirse una fuente de alimentación externa de 24 VDC para abastecer con 24 VDC los módulos de ampliación.
- La CPU alimenta también con 5 VDC los módulos de ampliación cuando se conectan al módulo base. Si el consumo de 5 VDC 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.



#### Precaución

Si se conecta una fuente de alimentación externa de 24 VDC 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.

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.

### 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 214 DC/DC/DC
- Tres módulos de ampliación EM 221, 8 entradas digitales DC 24 V
- Dos módulos de ampliación EM 222, 8 salidas digitales de relé

La CPU de este ejemplo suministra suficiente corriente de 5 VDC para los módulos de ampliación; sin embargo se precisa una fuente de alimentación adicional para que aporte la corriente necesaria de 24 VDC. (Las E/S requieren 448 mA a 24 VDC, pero la CPU sólo puede suministrar 280 mA). En el Anexo B figura una tabla en blanco para calcular los requisitos de alimentación.

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

Corriente máx. CPU	DC 5 V	DC 24 V
<i>CPU 214 DC/DC/DC</i>	<i>660 mA</i>	<i>280 mA</i>
<b>menos</b>		
Consumo del sistema	DC 5 V	DC 24 V
<i>CPU 214 DC/DC/DC</i>	<b>UNIDAD CENTRAL</b>	<i>14 entradas x 7 mA = 98 mA</i>
<i>Tres módulos de ampliación EM 221</i>	<i>3 x 60 mA = 180 mA</i>	<i>3 x 60 mA = 180 mA</i>
<i>Dos módulos de ampliación EM 222</i>	<i>2 x 80 mA = 160 mA</i>	<i>2 x 85 mA = 170 mA</i>
<b>Consumo total</b>	<i>340 mA</i>	<i>448 mA</i>
<b>igual a</b>		
Balance de corriente	DC 5 V	DC 24 V
<b>Balance total de corriente</b>	<i>320 mA</i>	<i>[168 mA]</i>

# Instalar y utilizar el software STEP 7-Micro/WIN

# 3

En el presente manual se describe la versión 2.1 de STEP 7-Micro/WIN. Las versiones anteriores del software ofrecían otras funciones.

STEP 7-Micro/WIN es una aplicación que se puede ejecutar tanto en el entorno Windows 3.1 de 16 bits (STEP 7-Micro/WIN 16) como en los entornos Windows 95 y Windows NT de 32 bits (STEP 7-Micro/WIN 32). Para poder utilizar STEP 7-Micro/WIN se requieren los siguientes equipos:

- Recomendable: PC con procesador 80586 o superior y 16 MB de RAM o una unidad de programación Siemens (p.ej. PG 740); requisito mínimo: procesador 80486 con 8 MB de RAM
- Uno de los equipos siguientes:
  - Un cable PC/PPI conectado al puerto de comunicación del PC o la PG (COM1 ó COM2).
  - Un procesador de comunicaciones (CP) y un cable de interface multipunto (MPI).
  - Una tarjeta de interface multipunto (MPI) (junto con la tarjeta MPI se suministra un cable de comunicación).
- Una pantalla VGA, o una pantalla asistida por Microsoft Windows
- 50 MB libres en el disco duro (como mínimo)
- Microsoft Windows 3.1, Windows para Trabajo en Grupo 3.11, Windows 95 o Windows NT 4.0o superior
- Opcional pero recomendable: un ratón asistido por Microsoft Windows

STEP 7-Micro/WIN incorpora una Ayuda online detallada. Para obtener más información sobre cualquier pregunta, elija uno de los comandos del menú **Ayuda** o pulse el botón F1 .

## Índice del capítulo

Apartado	Descripción	Página
3.1	Instalar el software de STEP 7-Micro/WIN	3-2
3.2	Utilizar STEP 7-Micro/WIN para configurar la comunicación	3-4
3.3	Establecer la comunicación con una CPU S7-200	3-7
3.4	Configurar las preferencias para STEP 7-Micro/WIN	3-25
3.5	Crear y guardar un proyecto	3-26
3.6	Crear un programa	3-27
3.7	Crear un bloque de datos	3-32
3.8	Utilizar la tabla de estado	3-34
3.9	Utilizar el direccionamiento simbólico	3-36

## 3.1 Instalar el software de STEP 7-Micro/WIN

### 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, haga 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.

Es posible que deba arrancar de nuevo su PC antes de proceder a la instalación.

### Instalación en Windows 3.1

Para instalar el software STEP 7-Micro/WIN 16 en un PC con Windows 3.1 (Windows para Trabajo en Grupo 3.11):

1. Inserte el disquete 1 en la correspondiente unidad del PC (que es, por lo general, la unidad de disquete A: o B:).
2. En el Administrador de programas, elija el comando de menú **Archivo ► Ejecutar...**
3. En el cuadro de diálogo "Ejecutar", introduzca **a:\setup** y haga clic en "Aceptar" o pulse la tecla ENTER. Así se inicia la instalación.
4. Siga las instrucciones que vayan apareciendo en pantalla hasta finalizar la instalación.

### Instalación en Windows 95 o Windows NT 4.0

Para instalar el software STEP 7-Micro/WIN 32 en un PC con Windows 95 o Windows NT 4.0:

1. Inserte el disquete 1 en la correspondiente unidad del PC (que es, por lo general, la unidad de disquete A: o B:).
2. Haga clic en el botón "Inicio" para abrir el menú de arranque de Windows 95.
3. Haga clic en el comando **Ejecutar**.
4. En el cuadro de diálogo "Ejecutar", introduzca **a:\setup** y haga clic en "Aceptar" o pulse la tecla ENTER. Así se inicia la instalación.
5. Siga las instrucciones que vayan apareciendo en pantalla hasta finalizar la instalación.
6. Al finalizar la instalación aparece automáticamente el cuadro de diálogo donde se pueden instalar o desinstalar las tarjetas.(v. fig. 3-1). El hardware de comunicación se puede instalar ahora mismo (v. apt. 3.2) o posteriormente (v. apt. 3.3).



Figura 3-1 Cuadro de diálogo “Instalar/desinstalar tarjetas”

### Fallos durante la instalación

Las siguientes situaciones pueden causar fallos durante la instalación:

- Memoria insuficiente: se requieren como mínimo 50 MB libres en el disco duro.
- Disquete defectuoso: asegúrese de que el disquete está defectuoso y diríjase a su representante de Siemens.
- Error de operador: reinicie la instalación y lea las instrucciones detenidamente.
- No se han cerrado todas las demás aplicaciones, incluyendo la barra de herramientas de Microsoft Office.

Lea el archivo LEAMEx.TXT incluido en los disquetes de instalación. Dicho archivo contiene la información más actual acerca de STEP 7-Micro/WIN. (En lugar de x, aparecerá la letra A = alemán, B = inglés, C = francés, D = español, E = italiano).

## 3.2 Utilizar STEP 7-Micro/WIN para configurar la comunicación

### Información general sobre cómo instalar o desinstalar el hardware de comunicación

Si utiliza Windows 95 o Windows NT 4.0, el cuadro de diálogo "Instalar/desinstalar tarjetas" aparecerá automáticamente cuando termine de instalar el software (v. fig. 3-1). En Windows 3.1, proceda de la siguiente forma:

1. Elija el comando de menú **Instalar ► Comunicación...** para acceder al cuadro de diálogo "Comunicación".
2. Haga clic en el botón "Interface PG/PC..." para acceder al cuadro de diálogo "Ajustar interface PG/PC".
3. Haga clic en el botón "Instalar..." para acceder al cuadro de diálogo "Instalar/desinstalar tarjetas" (v. fig. 3-1).

La instalación del hardware de comunicación depende de los criterios siguientes:

- El sistema operativo utilizado (Windows 3.1, Windows 95 o Windows NT 4.0).
- El tipo de hardware utilizado:
  - PC con cable PC/PPI
  - PC o unidad de programación SIMATIC con tarjeta de interface multipunto (MPI) o con procesador de comunicaciones (CP)
  - CPU 212, CPU 214, CPU 215, CPU 216
  - Módem
- La velocidad de transferencia utilizada.

La tabla 3-1 muestra las configuraciones del hardware y las velocidades de transferencia que asiste STEP 7-Micro/WIN, dependiendo del tipo de CPU utilizado. Para obtener informaciones más detalladas sobre cómo instalar la comunicación, consulte el apartado 3.3.

Tabla 3-1 Configuraciones de hardware asistidas por STEP 7-Micro/WIN

Tipo de CPU	Versión de STEP 7-Micro/WIN	Hardware asistido	Velocidades de transferencia asistidas	Sistema operativo	Parametrización
CPU 212, CPU 214, CPU 216 CPU 215 interface 0	Micro/WIN 16	Cable PC/PPI, tarjeta MPI-ISA	9,6 ó 19,2 kbit/s	Windows 3.1	PPI, PPI multimaestro
				Windows 95 o Windows NT	PPI
	Micro/WIN 32	Cable PC/PPI, tarjeta MPI-ISA, tarjeta MPI-ISA integrada, CP 5411, CP 5511, CP 5611	9,6 ó 19,2 kbit/s	Windows 95 o Windows NT	PPI, PPI multimaestro
CPU 215 interface 1 (interface DP)	Micro/WIN 16	No asistido	No asistidas	Windows 3.1 Windows 95 o Windows NT	No asistido
	Micro/WIN 32	Tarjeta MPI-ISA, tarjeta MPI-ISA integrada, CP 5411, CP 5511, CP 5611	9,6 kbit/s a 12 Mbit/s	Windows 95 o Windows NT	MPI

---

**Nota**

STEP 7-Micro/WIN 16 no asiste la parametrización multimaestro bajo Windows 95 o Windows NT 4.0.

---

Se asisten las siguientes configuraciones:

- CPU 212, CPU 214, CPU 216, CPU 215 (interface 0)
  - Cable PC/PPI (PPI), 9,6 kbit/s ó 19,2 kbit/s
  - Tarjeta MPI (PPI), 9,6 kbit/s ó 19,2 kbit/s
- CPU 215 (interface 1, es decir, el interface DP)
  - Tarjeta MPI (MPI), 9,6 kbit/s a 12 Mbit/s

---

**Nota**

STEP 7-Micro/WIN 16 no asiste la comunicación en el interface 1 de la CPU 215.

---

Las selecciones para la tarjeta MPI en STEP 7-Micro/WIN 16 son diferentes a las de STEP 7-Micro/WIN 32.

Del lado izquierdo del cuadro de diálogo "Instalar/desinstalar tarjetas" aparece una lista de componentes de hardware que no se han instalado todavía (v. fig. 3-1). Del lado derecho aparece una lista de los componentes instalados actualmente. Al utilizar el sistema operativo Windows NT 4.0, el botón "Recursos" se visualiza debajo de ésta última lista.

Para instalar componentes de hardware:

1. En el cuadro de lista "Selección", elija el componente de hardware disponible. En la ventana inferior se visualiza entonces una descripción de su selección.
2. Haga clic en el botón "Instalar -->".

Para desinstalar componentes de hardware:

1. En el cuadro de lista de componentes instalados (que se visualiza del lado derecho), seleccione el componente en cuestión.
2. Haga clic en el botón "<-- Desinstalar".

Tras instalar o desinstalar un componente de hardware, haga clic en el botón "Cerrar". Así se retorna al cuadro de diálogo "Ajustar interface PG/PC". Las selecciones efectuadas aparecen entonces en el cuadro de lista que contiene la parametrización utilizada (v. fig. 3-7).

Para obtener informaciones más detalladas sobre cómo instalar la comunicación, consulte el apartado 3.3.

### Informaciones especiales relativas a la instalación de hardware bajo 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. Dichos 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.

Una vez instalado un componente de hardware, selecciónelo en el cuadro de lista "Instalados" y haga clic en el botón "Recursos" para acceder al correspondiente cuadro de diálogo (v. fig. 3-2). Allí se pueden modificar los ajustes del sistema del componente de hardware que se ha instalado. Si dicho 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 lista, 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.

Para obtener informaciones más detalladas sobre cómo instalar la comunicación, consulte el apartado 3.3.

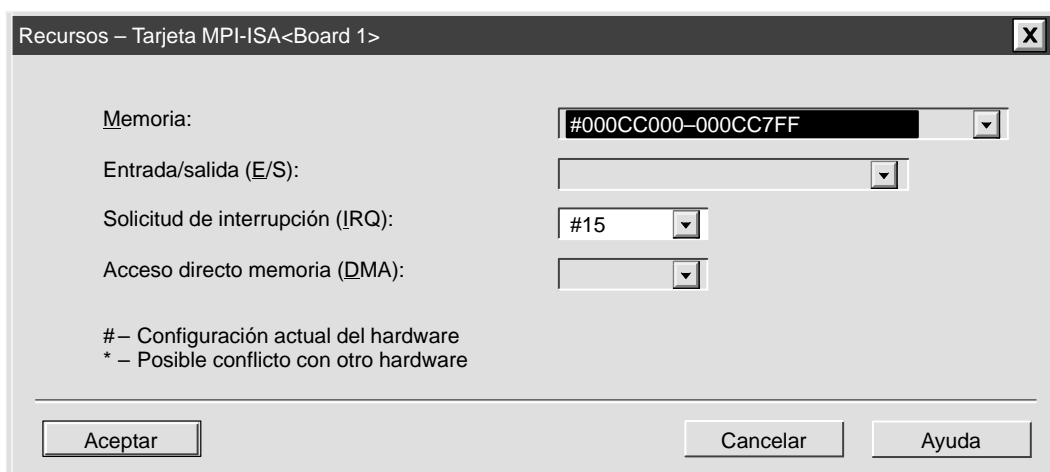


Figura 3-2 Cuadro de diálogo "Recursos" para Windows NT

### 3.3 Establecer la comunicación con una CPU S7-200

Las CPUs S7-200 se pueden disponer en diversas configuraciones para asistir la comunicación en redes. El software STEP 7-Micro/WIN se puede instalar en un PC con Windows 3.1x, Windows 95 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:

- Un sólo maestro se conecta a uno o varios esclavos (v. fig. 3-3).
- Un sólo maestro se conecta a uno o varios esclavos y maestros (v. figuras 3-4 y 3-5).
- La CPU 215 funciona como un módulo de ampliación esclavo de un sistema de automatización S7-300 ó S7-400, o bien de otro maestro PROFIBUS (v. fig. 3-13).
- Un sólo maestro se conecta a uno o varios esclavos. Este maestro se conecta a través de módems de 11 bits bien sea a una CPU S7-200 que actúe de esclava, o bien a una red de CPUs S7-200 también esclavas (v. fig. 3-14).

#### Conectar el PC a la CPU S7-200 utilizando el cable PC/PPI

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

1. Ajuste los interruptores DIP del cable PC/PPI para determinar la velocidad de transferencia deseada.
2. Conecte el extremo RS-232 ("PC") del cable PC/PPI al puerto de comunicación de su PC (COM1 o COM2) y apriete los tornillos de conexión.
3. Conecte el otro extremo (RS-485) del cable PC/PPI al interface de comunicación de la CPU y apriete los tornillos de conexión.

Para obtener información acerca de los datos técnicos del cable PC/PPI, consulte el apartado A.40. Su referencia se indica en el Anexo G.

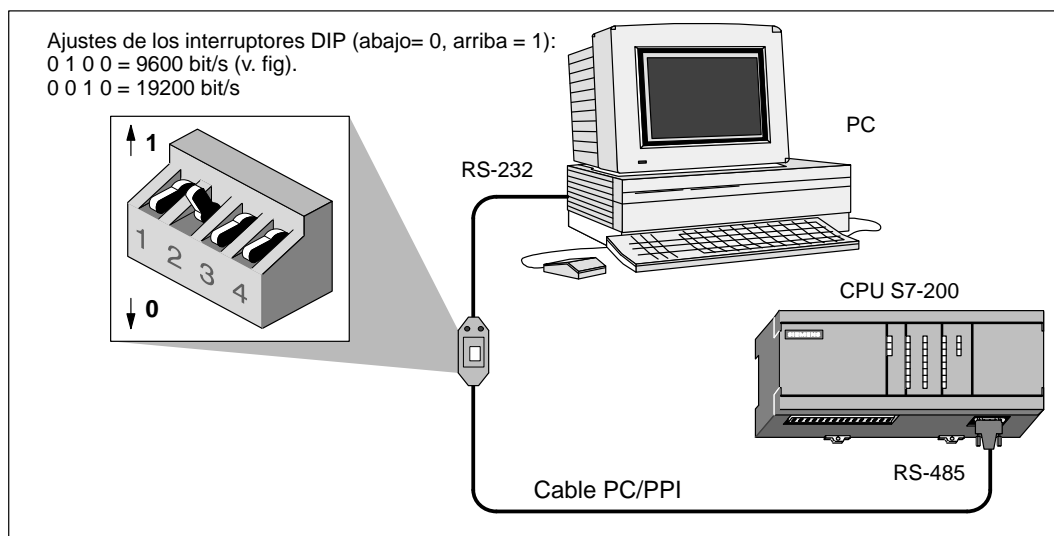


Figura 3-3 Comunicación con una CPU en modo PPI

La figura 3-4 muestra una configuración compuesta por un PC conectado a varias CPUs S7-200. STEP 7-Micro/WIN se ha diseñado para que se comunique con una CPU S7-200 a la vez. No obstante, se puede acceder a cualquier CPU incorporada en la red. Las CPUs que muestra la figura 3-4 podrían ser tanto esclavas como maestras. El TD 200 es una unidad maestra. Para obtener informaciones más detalladas acerca de la comunicación en redes, consulte el capítulo 9.

#### Nota

Sólo en STEP 7-Micro/WIN 16 (bajo Windows 3.1) o en STEP 7-Micro/WIN 32 se asisten varios maestros a través del cable PC/PPI cable, lo que no es el caso en STEP 7-Micro/DOS.

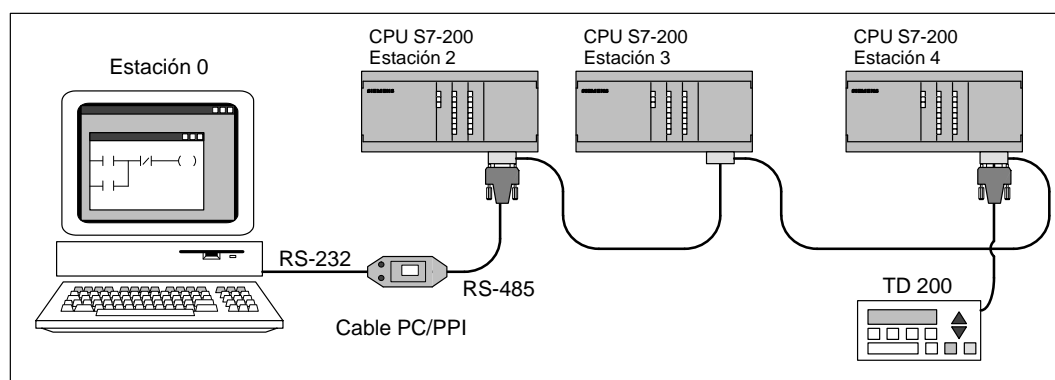


Figura 3-4 Utilizar un cable PC/PPI para la comunicación con varias CPUs S7-200

#### Conectar el PC a la CPU S7-200 mediante una tarjeta MPI o un CP

STEP 7-Micro/WIN se puede utilizar con una tarjeta de interface multipunto (MPI) o con un procesador de comunicaciones (CP). Ambos componentes disponen de un puerto RS-485 sencillo para la conexión a la red mediante un cable MPI. STEP 7-Micro/WIN 32 (la versión de 32 bits) asiste la parametrización MPI para una red MPI, lo que no es el caso en STEP 7-Micro/WIN 16 (la versión de 16 bits). Tras establecer la comunicación MPI, STEP 7-Micro/WIN se puede conectar a una red que contenga otros maestros. Cada uno de ellos debe tener una dirección unívoca. La figura 3-5 muestra una red de ejemplo compuesta por unidades maestras y esclavas. Para obtener informaciones más detalladas acerca de la comunicación en redes, consulte el capítulo 9. Para obtener informaciones más detalladas sobre la tarjeta MPI y los diversos procesadores de comunicaciones (CPs) disponibles, consulte el apartado 9.4. En el Anexo G se indican las respectivas referencias.

#### Nota

Si se utiliza la parametrización PPI, STEP 7-Micro/WIN no asiste dos aplicaciones diferentes que se ejecuten simultáneamente en una misma tarjeta MPI o CP. Cierre las demás aplicaciones antes de conectar STEP 7-Micro/WIN a la red mediante la tarjeta MPI o CP.

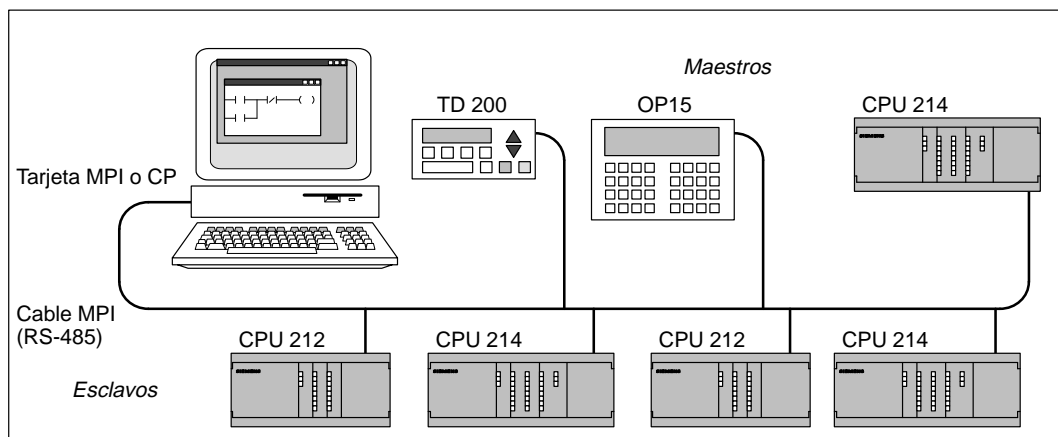


Figura 3-5 Ejemplo de una red de maestros y esclavos con una tarjeta MPI o CP

### ¿Dónde se ajustan los parámetros de comunicación?

Dependiendo del sistema operativo utilizado, puede configurar la comunicación desde uno de los entornos siguientes:

- Bajo Windows 3.1  
Sólo en STEP 7-Micro/WIN 16
- Bajo Windows 95 o Windows NT 4.0
  - En la fase final de la instalación (v. apt. 3.1)
  - A partir del icono "Ajustar interface PG/PC" que se encuentra en el Panel de control de Windows
  - En STEP 7-Micro/WIN 32

### Configurar los parámetros de comunicación en STEP 7-Micro/WIN

STEP 7-Micro/WIN incorpora el cuadro de diálogo "Comunicación" donde se pueden configurar los ajustes de comunicación (v. fig. 3-6). Para acceder a dicho cuadro puede optar por uno de los siguientes métodos:

- Elija el comando de menú **Instalar ► Comunicación....**
- Cree un nuevo proyecto y haga clic en el botón "Comunicación..." en el cuadro de diálogo "Tipo de CPU".
- Si tiene un proyecto abierto, elija el comando de menú **CPU ► Tipo...** y haga clic en el botón "Comunicación..." del cuadro de diálogo "Tipo de CPU".

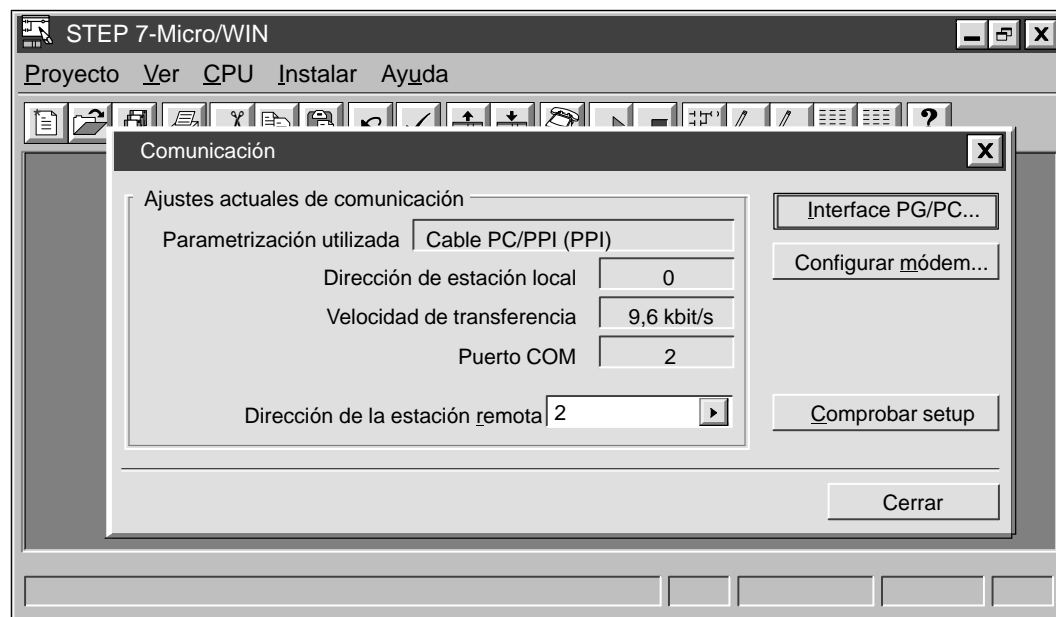


Figura 3-6 Configurar los parámetros de comunicación entre la PG o el PC y la CPU

En el cuadro de diálogo "Comunicación", haga clic en el botón "Interface PG/PC..." para acceder al cuadro de diálogo "Ajustar interface PG/PC" (v. fig. 3-7).

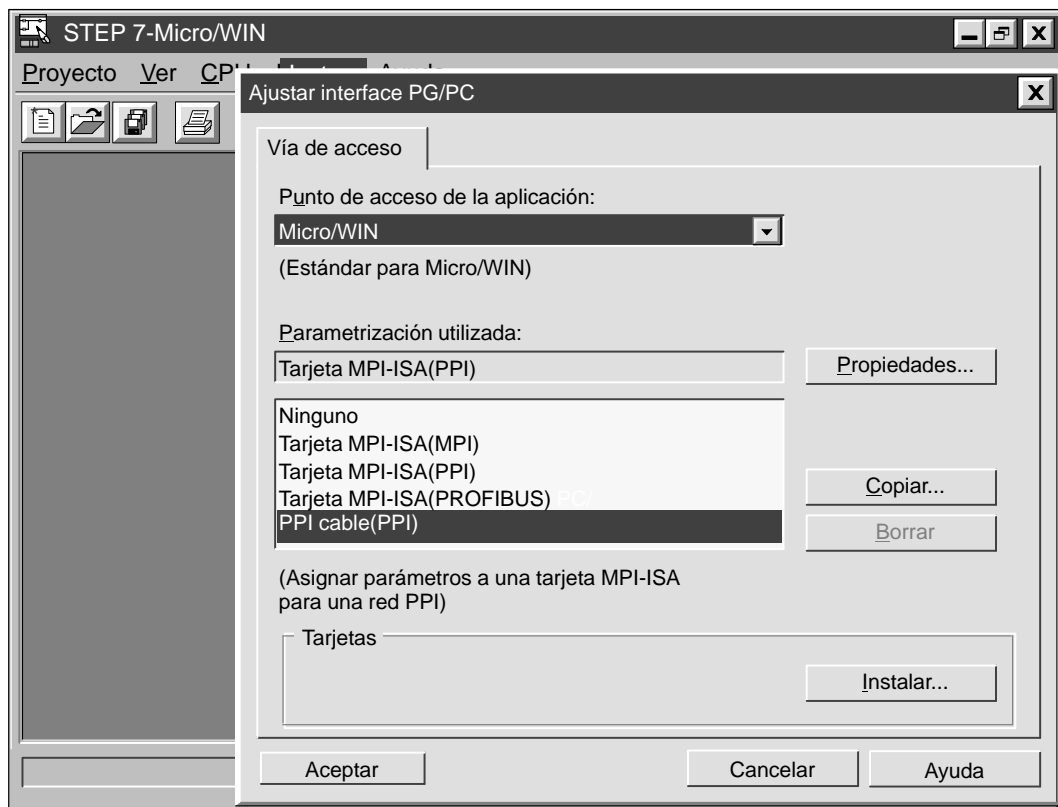


Figura 3-7 Ajustes en el cuadro de diálogo "Interface PG/PC"

### Configurar la comunicación en el Panel de control de Windows

En Windows 95 o Windows NT 4.0 puede configurar la comunicación a través del Panel de control. En el Panel de control, haga doble clic en el icono "Ajustar interface PG/PC" (v. fig. 3-8).

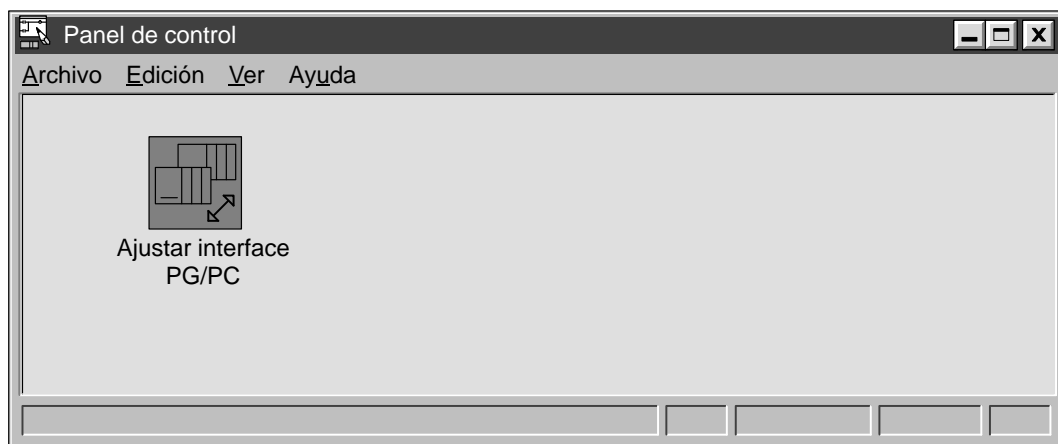


Figura 3-8 Panel de control con el icono "Ajustar interface PG/PC"

### Configurar la comunicación durante la instalación

Si utiliza Windows 95 o Windows NT 4.0, el cuadro de diálogo "Comunicación" aparecerá automáticamente al final de la instalación de STEP 7-Micro/WIN. La comunicación se puede configurar en ese momento o posteriormente.

### Seleccionar y configurar la parametrización correcta

Tras acceder al cuadro de diálogo "Ajustar interface PG/PC" (v. fig. 3-7), es preciso seleccionar "Micro/WIN" en el cuadro de lista "Punto de acceso de la aplicación" de la ficha "Vía de acceso". Este cuadro de diálogo es compartido por diversas aplicaciones (p.ej. STEP 7 y WinCC). Por este motivo es preciso indicarle al programa para qué aplicación se desean 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. Consulte la tabla 3-1 o el capítulo 9 para obtener más información acerca del hardware que asiste su CPU y los requisitos de configuración. En la mayoría de los casos se utilizará el protocolo PPI para todas las CPUs, con excepción del interface DP de la CPU 215. Éste último puerto utiliza el protocolo MPI.

Tras elegir el protocolo a utilizar, puede seleccionar la configuración correcta en el cuadro de lista "Parametrización utilizada" del cuadro de diálogo "Ajustar interface PG/PC". Allí se indican las tarjetas instaladas, junto con el tipo de protocolo (entre paréntesis). Por ejemplo, una configuración sencilla puede exigir que se utilice un cable PC/PPI para comunicarse con una CPU 214. En este caso, deberá elegir "Cable PC/PPI (PPI)." Otro ejemplo es una configuración que exija la comunicación con una CPU 215 a través de su interface rápido (interface DP) mediante una tarjeta MPI-ISA sencilla que se encuentre instalada en el PC. En este caso deberá seleccionar "Tarjeta MPI-ISA (MPI)."

Una vez seleccionada la parametrización correcta, es preciso ajustar los parámetros individuales para la configuración actual. En el cuadro de diálogo "Ajustar interface PG/PC", haga clic en el botón "Propiedades...". Dependiendo de la parametrización seleccionada, se accede entonces a uno de varios cuadros de diálogo posibles. En los apartados siguientes se describe detalladamente cada uno de ellos.

Para seleccionar una parametrización:

1. Estando en el cuadro de diálogo "Ajustar interface PG/PC" (v. fig. 3-7), elija "Micro/WIN" en el cuadro de lista "Punto de acceso de la aplicación" de la ficha "Vía de acceso".
2. Asegúrese de que su componente de hardware esté instalado (v. apt. 3.2).
3. Determine el protocolo a utilizar.
4. Elija la configuración correcta en el cuadro de lista "Parametrización utilizada".
5. Haga clic en el botón "Propiedades...".

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 de los siguientes sistemas operativos y componentes de hardware:

- Windows 3.1: Cable PC/PPI
- Windows 95 o Windows NT 4.0: 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. 3-9).

Para ajustar las propiedades:

1. En la ficha "Red PPI", seleccione un número en el cuadro "Dirección de la estación local". Dicho número corresponde a la dirección de STEP 7-Micro/WIN en la red.
2. Elija un valor en el cuadro "Timeout". Éste representa el tiempo durante el que los drivers de comunicación pueden intentar establecer enlaces. El valor predeterminado debería ser suficiente.
3. Determine si desea que STEP 7-Micro/WIN participe en una red donde existan varios maestros. Para obtener más información al respecto, consulte el capítulo 9. Puede dejar marcada la casilla "Red multimaestro", a menos que utilice un módem. En éste último caso, la casilla no se podrá marcar, puesto que STEP 7-Micro/WIN no asiste esa función.
4. Ajuste la velocidad de transferencia que desee utilizar para la comunicación de STEP 7-Micro/WIN en la red. En el capítulo 9, tabla 9-1 se indican las velocidades de transferencia asistidas por cada CPU.
5. Elija la dirección de estación más alta. A partir de esta dirección, STEP 7-Micro/WIN no busca más maestros en la red.

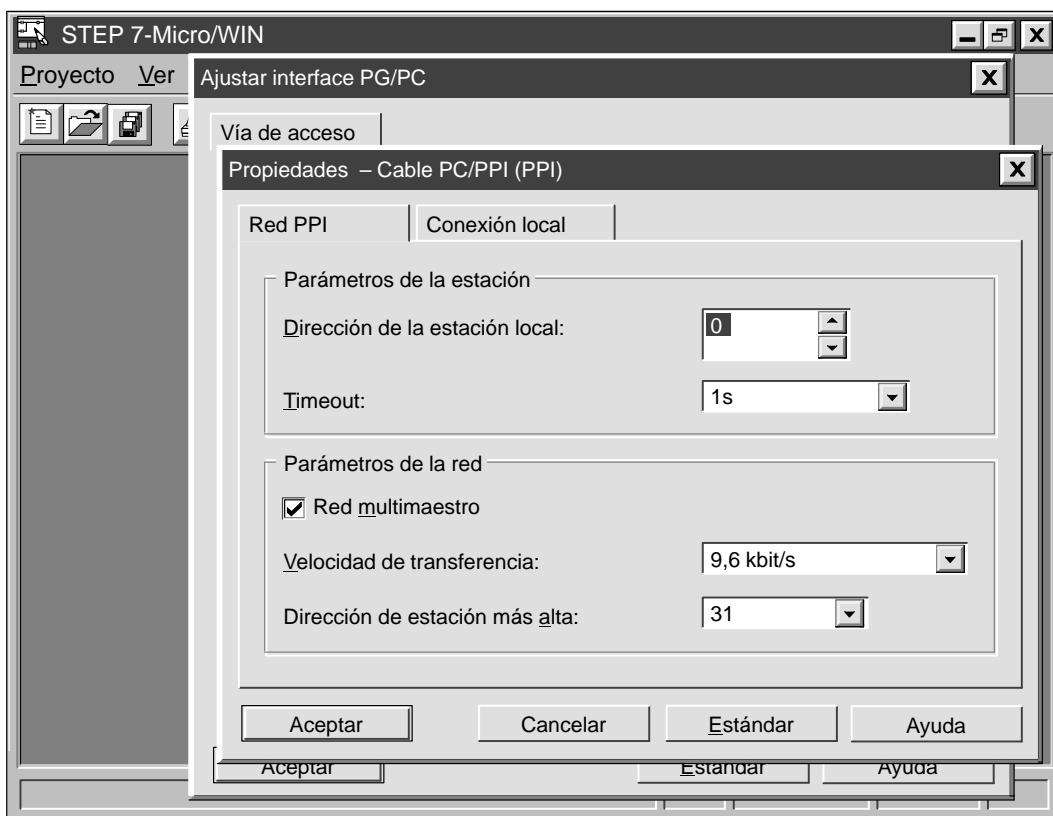


Figura 3-9 Propiedades del cable PC/PPI (PPI), ficha "Red PPI"

6. Haga clic en la ficha "Conexión local" (v. fig. 3-10).
7. En la ficha "Conexión local", seleccione el puerto COM al que esté 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 cerrar el cuadro de diálogo "Ajustar interface PG/PC".

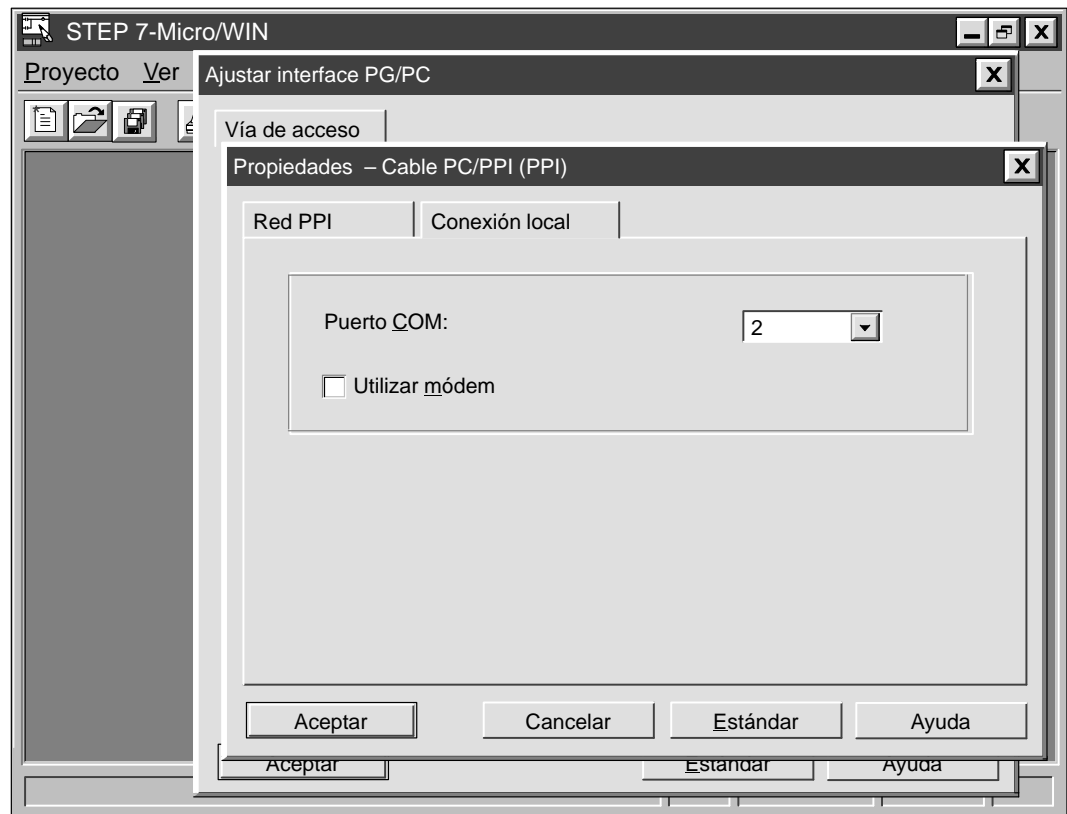


Figura 3-10 Propiedades del cable PC/PPI (PPI), ficha "Conexión local"

### Ajustar los parámetros de la tarjeta MPI (PPI)

Aquí se explica cómo ajustar los parámetros PPI de los siguientes sistemas operativos y componentes de hardware:

- Windows 3.1: Tarjeta MPI-ISA (incluyendo las que se encuentren en las unidades de programación SIMATIC)
- Windows 95 o Windows NT 4.0:
  - Tarjeta MPI-ISA
  - Tarjeta MPI-ISA integrada (tarjetas MPI para las unidades de programación SIMATIC)
  - CP 5411
  - CP 5511
  - CP 5611

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...", aparece 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. 3-11).

Para ajustar las propiedades:

1. En la ficha "Red PPI", seleccione un número en el cuadro "Dirección de la estación local". Dicho número corresponde a la dirección de STEP 7-Micro/WIN en la red.
2. Elija un valor en el cuadro "Timeout". Éste representa el tiempo durante el que los drivers de comunicación pueden intentar establecer enlaces. El valor predeterminado debería ser suficiente.
3. Determine si desea que STEP 7-Micro/WIN participe en una red donde existan varios maestros. Para obtener más información al respecto, consulte el capítulo 9. Puede dejar marcada la casilla "Red multimaestro".
4. Ajuste la velocidad de transferencia que desee utilizar para la comunicación de STEP 7-Micro/WIN en la red. En el capítulo 9, tabla 9-1 se indican las velocidades de transferencia asistidas por cada CPU.
5. Elija la dirección de estación más alta. A partir de esta dirección, STEP 7-Micro/WIN no busca más maestros en la red.
6. Haga clic en el botón "Aceptar" para cerrar el cuadro de diálogo "Ajustar interface PG/PC".

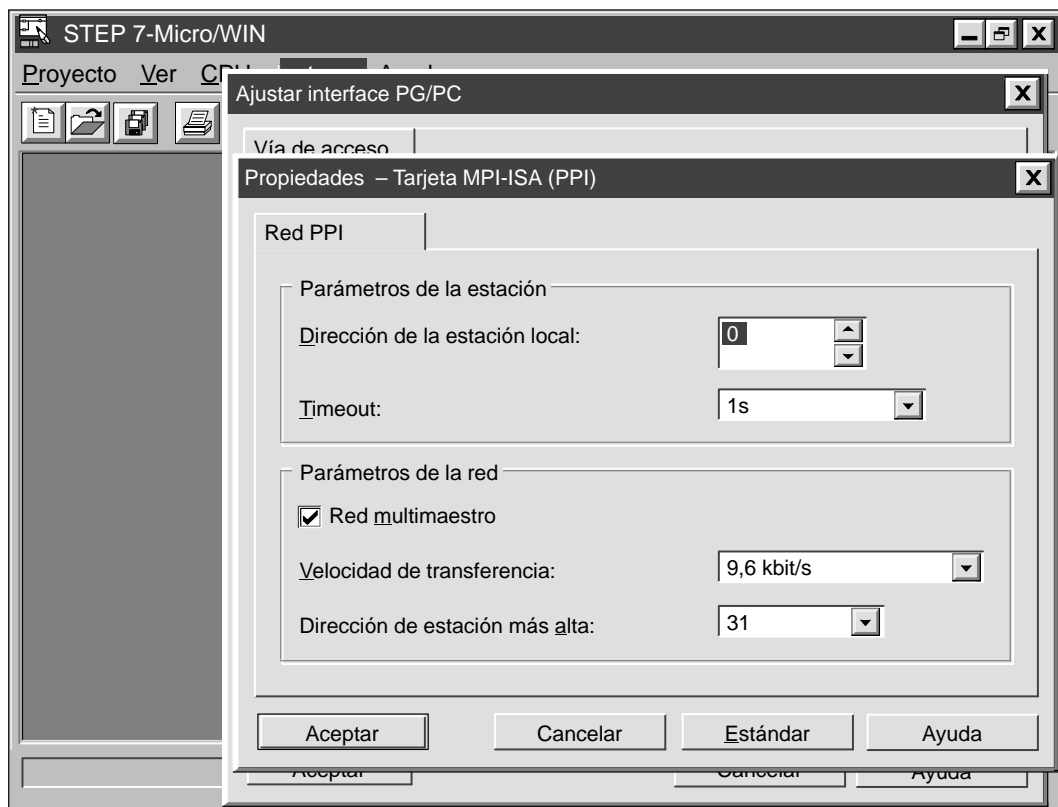


Figura 3-11 Propiedades de la tarjeta MPI-ISA (PPI)

### Ajustar los parámetros de la tarjeta MPI (MPI)

Aquí se explica cómo ajustar los parámetros MPI de los siguientes sistemas operativos y componentes de hardware:

- Windows 3.1: Tarjeta MPI-ISA (incluyendo las que se encuentren en las unidades de programación SIMATIC)
- Windows 95 o Windows NT 4.0:
  - Tarjeta MPI-ISA
  - Tarjeta MPI-ISA integrada (tarjetas MPI para las unidades de programación SIMATIC)
  - CP 5411
  - CP 5511
  - CP 5611

Partiendo del cuadro de diálogo "Ajustar interface PG/PC", si alguna de las tarjetas MPI o CP mencionadas se utiliza junto con el protocolo MPI y se hace clic en el botón "Propiedades...", aparece la ficha de propiedades de la tarjeta XXX (MPI), donde "XXX" es el tipo de tarjeta que se ha instalado (p.ej. MPI-ISA) (v. fig. 3-12).

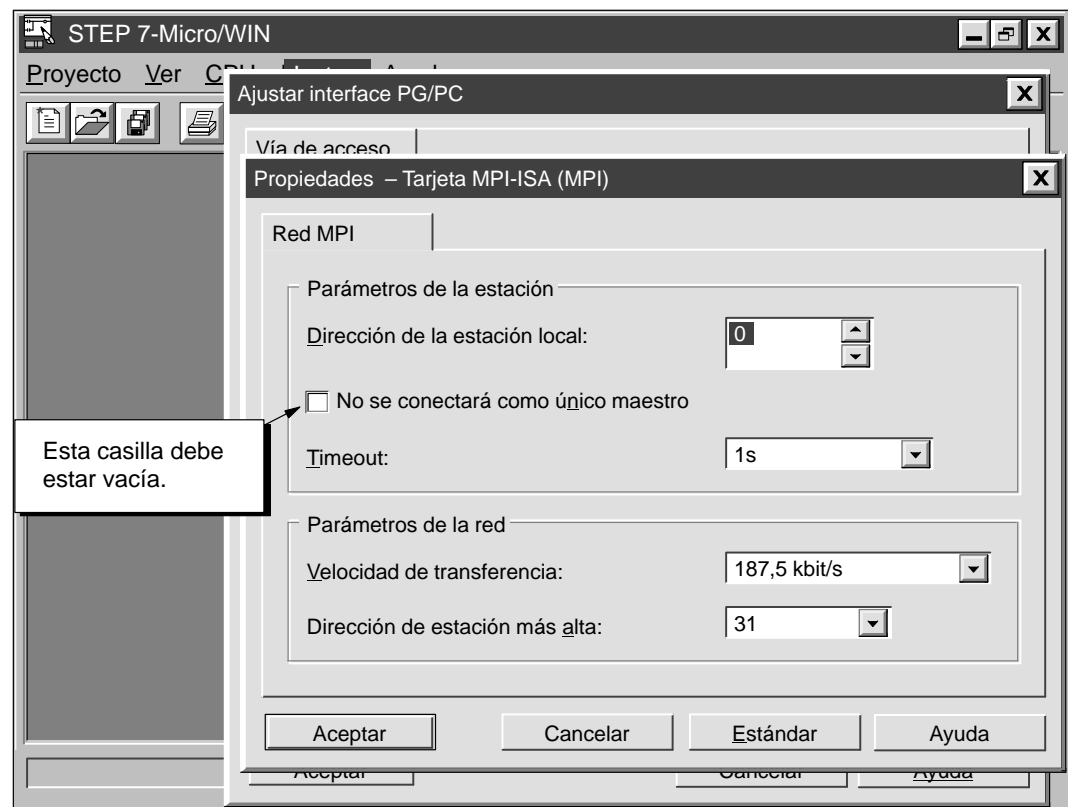


Figura 3-12 Propiedades de la tarjeta MPI-ISA (MPI)

Para ajustar las propiedades:

1. En la ficha "Red MPI", seleccione un número en el cuadro "Dirección de la estación local". Dicho número corresponde a la dirección de STEP 7-Micro/WIN en la red.
2. Verifique que la casilla "No se conectará como único maestro" no esté activada, independientemente de la cantidad de maestros que conformen la red. Si dicha casilla contiene una marca de verificación, haga clic allí para desactivarla. Asegúrese de conectar el cable de comunicación entre la unidad de programación (PG) y la CPU antes de iniciar la comunicación. Si la comunicación se establece antes de conectar la PG a una red de CPUs que incorpore uno o más maestros, se interrumpirá mientras se reinicializa la red.
3. Elija un valor en el cuadro "Timeout". Éste representa el tiempo durante el que los drivers de comunicación pueden intentar establecer enlaces. El valor predeterminado debería ser suficiente.
4. Ajuste la velocidad de transferencia que desee utilizar para la comunicación de STEP 7-Micro/WIN en la red. Puesto que es probable que se esté utilizando el interface DP de una CPU 215, se puede seleccionar cualquier velocidad de transferencia disponible (hasta 12 Mbit/s). En el capítulo 9, tabla 9-1 se indican las velocidades de transferencia asistidas por cada CPU.
5. Elija la dirección de estación más alta. A partir de esta dirección, STEP 7-Micro/WIN no busca más maestros en la red.
6. Haga clic en el botón "Aceptar" para cerrar el cuadro de diálogo "Ajustar interface PG/PC".

#### Eliminar errores de comunicación MPI en aplicaciones de 16 bits

La opción "Tarjeta MPI" activa los drivers MPI contenidos en el archivo de configuración S7DPMPLI.INI que fue copiado en el directorio de Windows durante la instalación de STEP 7-Micro/WIN.

Si se presenta un error de interrupción, es preciso configurar una línea de petición de interrupción (IRQ) libre para la tarjeta MPI. La línea de interrupción predeterminada es IRQ 5. En el campo "IRQ" se visualiza el número de la interrupción utilizada por la tarjeta MPI. Si se presenta un error de interrupción significa que IRQ 5 ya se está utilizando. Para cambiar de línea IRQ:

1. Elija el comando de menú **Instalar ► Comunicación...** para llamar al cuadro de diálogo "Comunicación. Introduzca un valor diferente para la interrupción.
2. Confirme sus ajustes haciendo clic en "Aceptar" o pulsando la tecla ENTER. El software modifica automáticamente el archivo S7DPMPLI.INI e indica si es preciso salir de la aplicación.
3. Reinicie STEP 7-Micro/WIN y seleccione de nuevo la opción MPI.

#### Nota

A continuación se indican las direcciones estándar de las CPUs S7-200 que tienen más de un interface de comunicación:

- CPU 215            Interface 0: 2  
                         Interface 1: 126
- CPU 216            Interface 0: 2  
                         Interface 1: 2

### Eliminar errores de comunicación MPI en Windows NT 4.0

En Windows NT 4.0 es un poco más difícil configurar correctamente la tarjeta MPI. Si se presentan problemas (estando instalada la tarjeta MPI en las pantallas donde se configura la comunicación), proceda de la siguiente manera:

1. Asegúrese de que la tarjeta MPI funciona bien. Hay varios métodos de hacerlo: puede comprobarlo mediante un PC con Windows 95 o bien con la versión 2.0 de STEP 7-Micro/WIN.
2. Verifique la posición de los interruptores DIP de la tarjeta MPI para averiguar cuánta memoria se debe reservar para la misma (v. tabla 3-2).
3. Compruebe qué recursos ha reservado Windows NT, con objeto de garantizar que concuerden con la configuración de los interruptores. Proceda de la siguiente manera:
  - a. Abra el cuadro de diálogo "Ajustar interface PG/PC".
  - b. Haga clic en el botón "Instalar...".
  - c. En la lista "Instalados", elija "Tarjeta MPI".
  - d. Haga clic en el botón "Recursos". Este botón sólo está disponible en Windows NT.
4. Si aunque los ajustes sean correctos la tarjeta no funciona todavía, intente cambiar la línea de petición de interrupción (IRQ) configurada para la tarjeta. Es posible que haya un conflicto con otro componente de hardware. Dicho cambio se puede efectuar en el cuadro de diálogo "Recursos".
5. Si ha ensayado todas las interrupciones y la tarjeta no funciona aún, es preciso cambiar los ajustes de los interruptores DIP de la tarjeta a una dirección diferente. Repita los pasos 3 y 4.
6. Si ha intentado realizar todos los pasos indicados y la tarjeta no funciona todavía, es posible que otros componentes de hardware estén utilizando todos los recursos disponibles. Puede intentar retirar o desinstalar algunos de dichos componentes (p.ej. tarjetas de sonido) para poder disponer de algunos recursos. Comience entonces nuevamente con el 2º paso.
7. Si ésto tampoco resulta, utilice un driver de comunicación diferente.

Consulte la documentación de la tarjeta MPI para obtener informaciones más detalladas acerca de los conflictos de hardware que se pueden presentar.

Tabla 3-2 Memoria necesaria para una tarjeta MPI

Interruptor 1	Interruptor 2	Interruptor 3	Memoria
ON	ON	ON	#000C8000-000C87FF
ON	ON	OFF	#000C9000-000C97FF
ON	OFF	ON	#000CC000-000CC7FF
ON	OFF	OFF	#000D0000-000D07FF
OFF	ON	ON	#000D1000-000D17FF
OFF	ON	OFF	#000DC000-000DC7FF
OFF	OFF	ON	#000E1000-000E17FF

### Conectar una CPU 215 en calidad de esclava

La CPU 215 se puede conectar a una red PROFIBUS, actuando de esclava de un sistema de automatización S7-300 ó S7-400, o bien de otro maestro PROFIBUS (v. fig. 3-13).

La CPU 215 dispone de un interface DP. Éste se utiliza para conectar la CPU 215 en calidad de esclava a una red PROFIBUS.

La dirección de estación del interface DP es el único parámetro que se debe ajustar en la CPU 215 para poder utilizarla como esclava PROFIBUS. Dicha dirección debe coincidir con la indicada en la configuración del maestro ya que éste configura a la CPU 215. Para obtener más información acerca de la comunicación DP estándar, consulte el apartado 9.5.

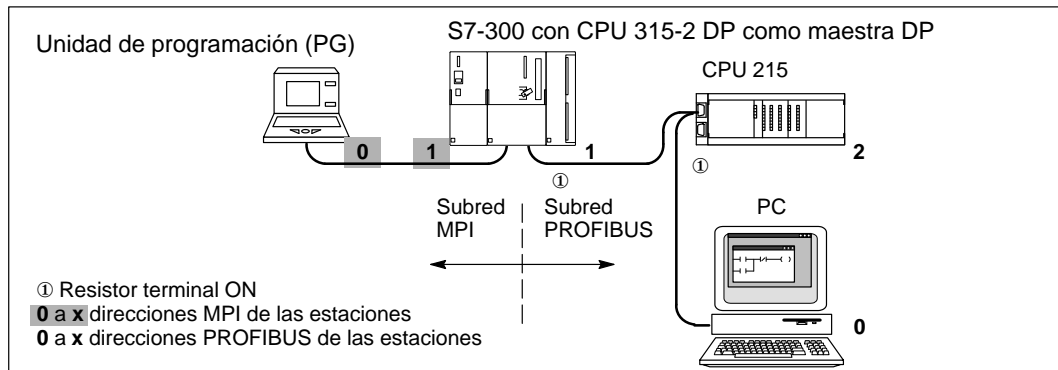


Figura 3-13 CPU 215 en una subred PROFIBUS con subred MPI

### Utilizar módems para conectar una CPU S7-200 a un maestro STEP 7-Micro/WIN

Al utilizar STEP 7-Micro/WIN en un PC con Windows 3.1x, Windows 95 o Windows NT, o bien en una unidad de programación SIMATIC (p.ej. PG 740) como maestro único, se pueden establecer conexiones vía módem con los siguientes dispositivos S7-200:

- Una sola CPU S7-200 como esclava
- Varias CPUs S7-200 como esclavas en una red

Dependiendo de si se desea conectar sólo una o varias CPUs S7-200, se necesitan los siguientes cables y adaptadores (v. fig. 3-14):

- Un cable apto para RS-232 en ambos extremos con objeto de conectar el PC o la unidad de programación SIMATIC a un módem dúplex de 11 bits en uno de los extremos del cable de teléfono.
- Un adaptador de módem nulo para conectar al cable PC/PPI el módem que se encuentra del otro extremo del cable de teléfono.
- Un cable PC/PPI para conectar el adaptador de módem nulo a uno de los interfaces siguientes:
  - Puerto de comunicación de la CPU S7-200 (v. fig. 3-14)
  - Conector de interface de programación Siemens en una red PROFIBUS (v. fig. 9-3)

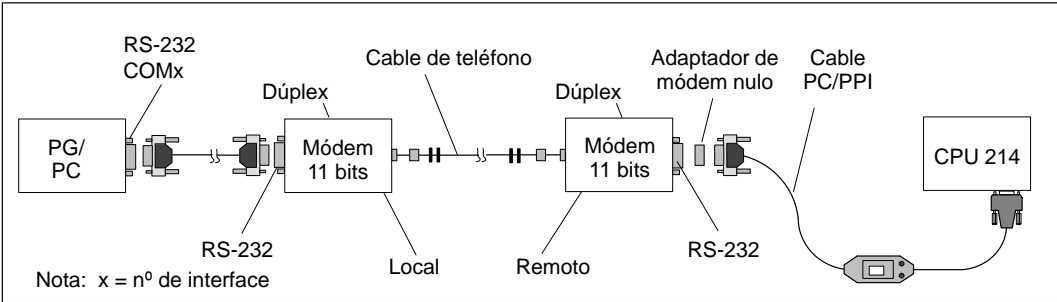


Figura 3-14 Comunicación S7-200 vía módems de 11 bits

Puesto que en estas configuraciones se puede utilizar sólo un maestro, no se efectúa un "token passing" (paso de testigo). Estas configuraciones sólo asisten el protocolo PPI. Para poder comunicarse a través del interface PPI, el sistema de automatización S7-200 exige que el módem utilice una cadena de datos de 11 bits. Para el S7-200 se necesitan un bit de arranque, ocho bits de datos, un bit de paridad par y un bit de parada, comunicación asíncrona y una velocidad de transferencia de 9600 bit/s para PPI. Numerosos módems no asisten dicho formato de datos. Los parámetros necesarios para el módem figuran en la tabla 3-3.

La figura 3-15 muestra la asignación de pines de un adaptador de módem nulo. Para obtener más información sobre la comunicación en redes con un cable PC/PPI, consulte el capítulo 9.

Tabla 3-3 Parámetros necesarios para el módem

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

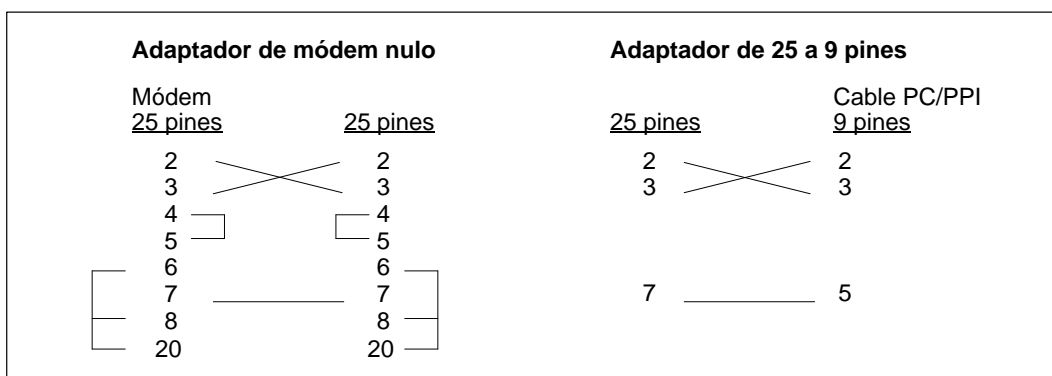


Figura 3-15 Asignación de pines de un adaptador de módem nulo

### 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 podrá activar la función "Configurar módem". Verifique que dicha función esté habilitada y ajuste los parámetros de configuración como se indica a continuación:

#### Nota

La configuración que se indica aquí es aplicable al Multi Tech MultiModemZDX MT1932ZDX. Si no utiliza este tipo de módem, en el cuadro de diálogo "Configurar módem" deberá elegir la opción "Definido por el usuario". Su módem debe ser de 11 bits y asistir una velocidad de transferencia de 9600 bit/s. Consulte el manual de su módem para determinar los parámetros a introducir en las fichas del cuadro de diálogo "Configurar módem".

#### 1. Elija el comando de menú **Instalar ► Comunicación....**

Si en el cuadro de diálogo "Comunicación" se visualiza "Cable PC/PPI (PPI)" como parametrización utilizada, haga clic en el botón "Interface PG/PC..." y siga con el paso 3.

Si "Cable PC/PPI (PPI)" no es la parametrización utilizada, haga clic en el botón "Interface PG/PC..." y continúe con el paso 2.

- En el cuadro de lista "Parametrización utilizada" de la ficha "Vía de acceso", seleccione "Cable PC/PPC (PPI)". Si dicha selección no figura en el cuadro de lista, será preciso instalarla (v. apt. 3.1).
- Haga clic en el botón "Propiedades" para visualizar las propiedades del cable PC/PPI (PPI).
- En el cuadro de propiedades del cable PC/PPI (PPI), haga clic en la ficha "Conexión local".
- En el área "Puerto COM", asegúrese de que en la casilla "Utilizar módem" aparezca una marca de verificación. Si la casilla está vacía, haga clic allí para insertar dicha marca.
- Haga clic en el botón "Aceptar" para visualizar nuevamente la ficha "Vía de acceso".
- Haga clic en el botón "Aceptar" para retornar al cuadro de diálogo "Comunicación".

8. Haga clic en el botón "Configurar módem..." para llamar el correspondiente cuadro de diálogo. (Al botón "Configurar módem..." se puede acceder también eligiendo el comando de menú **Instalar ► Conectar módem....** El botón aparece entonces en el cuadro de diálogo "Conectar").

En la ficha "Información general" del cuadro de diálogo "Configurar módem" se indican los requisitos de los módems con cadenas de datos de 11 bits y los componentes de hardware necesarios para la comunicación. La figura 3-14 muestra esos mismos componentes de hardware.

9. Haga clic en la ficha "Configurar módem local" (v. fig. 3-16).

10. En el cuadro "Módem seleccionado" de la ficha "Configurar módem local", elija Multi Tech MultiModemZDX MT1932ZDX .

Los demás campos editables en esa ficha son "Conectar nº teléfono" y "Timeout". El timeout representa el tiempo durante el que 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.

11. Si desea comprobar la configuración del módem local, haga clic en el botón "Comprobar módem". A tal efecto, el módem debe estar conectado a la unidad de programación (PG) o al PC.
12. Desconecte el módem local y conecte el módem remoto a su PG o PC.

Configurar módem

Configurar módem local | Configurar módem remoto | Información general

Módem seleccionado: Multi Tech MultiModemZDX MT1932ZDX

Conectar nº teléfono: 5538

Opciones de marcación

Inicializar: AT&F0%E5=1&E12MOX3

Prefijo: ATDT Sufijo: ^ M

Timeout: 30 segundos

Desconectar

☐ Utilizar DTR

☒ Utilizar comando

Comando: ATH0

Cadenas de comando

Modo 11 bits: \$EB11

Veloc. transf.: \$SB

Control de flujo

Emisor: Ninguno

Receptor: Ninguno

Estado:

Programar módem Comprobar módem

Aceptar Cancelar

Figura 3-16 Ficha "Configurar módem local" del cuadro de diálogo "Configurar módem"

13. Haga clic en la ficha "Configurar módem remoto" (v. fig. 3-17).
14. En el cuadro "Módem seleccionado" de la ficha "Configurar módem remoto", elija Multi Tech MultiModemZDX MT1932ZDX.
15. Haga clic en el botón "Programar módem..." para transferir los parámetros a un chip de memoria del módem remoto.
16. Si desea verificar si el módem remoto se ha programado correctamente, haga clic en el botón "Comprobar módem".
17. Haga clic en el botón "Aceptar" para retornar al cuadro de diálogo "Comunicación".

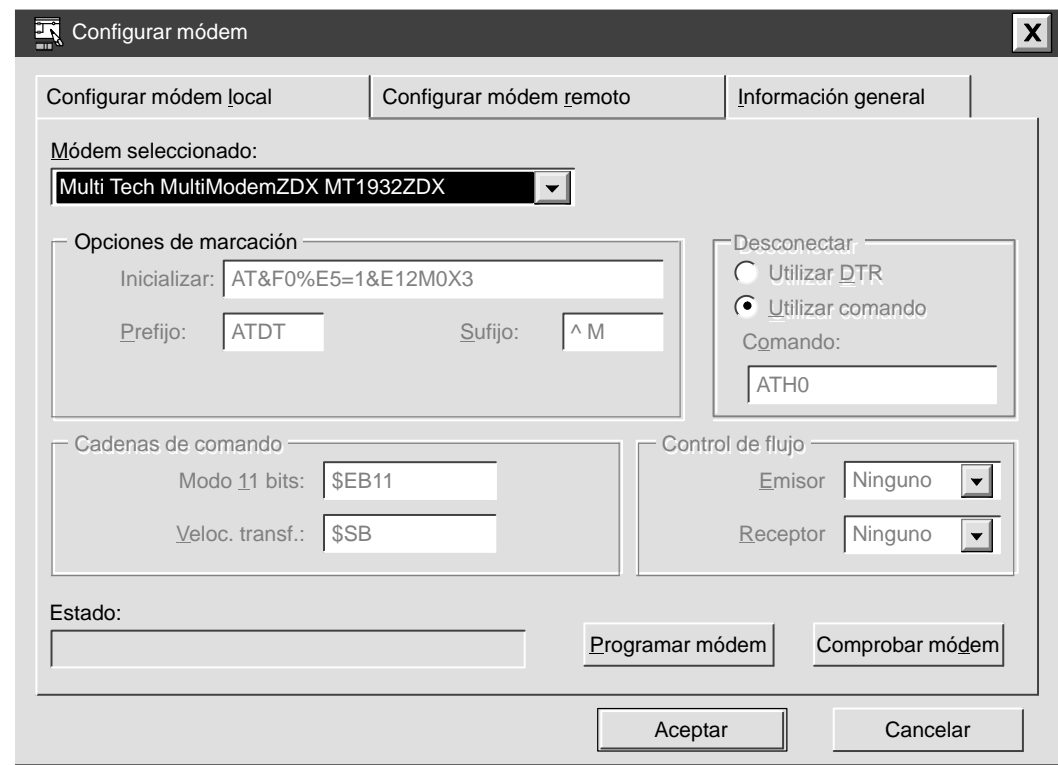


Figura 3-17 Ficha "Configurar módem remoto" del cuadro de diálogo "Configurar módem"

18. Desconecte el módem remoto de su unidad de programación (PG) o PC.
19. Conecte el módem remoto al sistema de automatización S7-200.
20. Conecte el módem local a su unidad de programación (PG) o PC.
21. Verifique que la configuración coincida con la indicada en la ficha "Información general" del cuadro de diálogo "Configurar módem" (v. también la figura 3-14).
22. Tras finalizar la configuración, haga clic en el botón "Aceptar" para salir del cuadro de diálogo "Comunicación".
23. Para conectar el módem, elija el comando de menú **Instalar ► Conectar módem...** con objeto de acceder al cuadro de diálogo "Conectar" (v. fig. 3-18).
24. Introduzca el número de teléfono en el cuadro "Nº de teléfono" si no lo ha hecho aún en el cuadro de diálogo "Configurar módem local" o si desea modificar el número.
25. Haga clic en el botón "Conectar". Así se finaliza la configuración del módem.



Figura 3-18 Cuadro de diálogo "Conectar"

### 3.4 Configurar las preferencias para STEP 7-Micro/WIN

Antes de crear un proyecto, indique sus preferencias para el entorno de programación. Para seleccionar las preferencias:

1. Elija el comando de menú **Instalar ► Preferencias...** como muestra la figura 3-19.
2. Elija sus preferencias de programación en el cuadro de diálogo que aparece a continuación.
3. Confirme sus ajustes pulsando la tecla ENTER o haciendo clic en el botón "Aceptar".

#### Nota

Si ha elegido un idioma diferente deberá salir de STEP 7-Micro/WIN y reiniciar la aplicación para que el cambio tenga efecto.

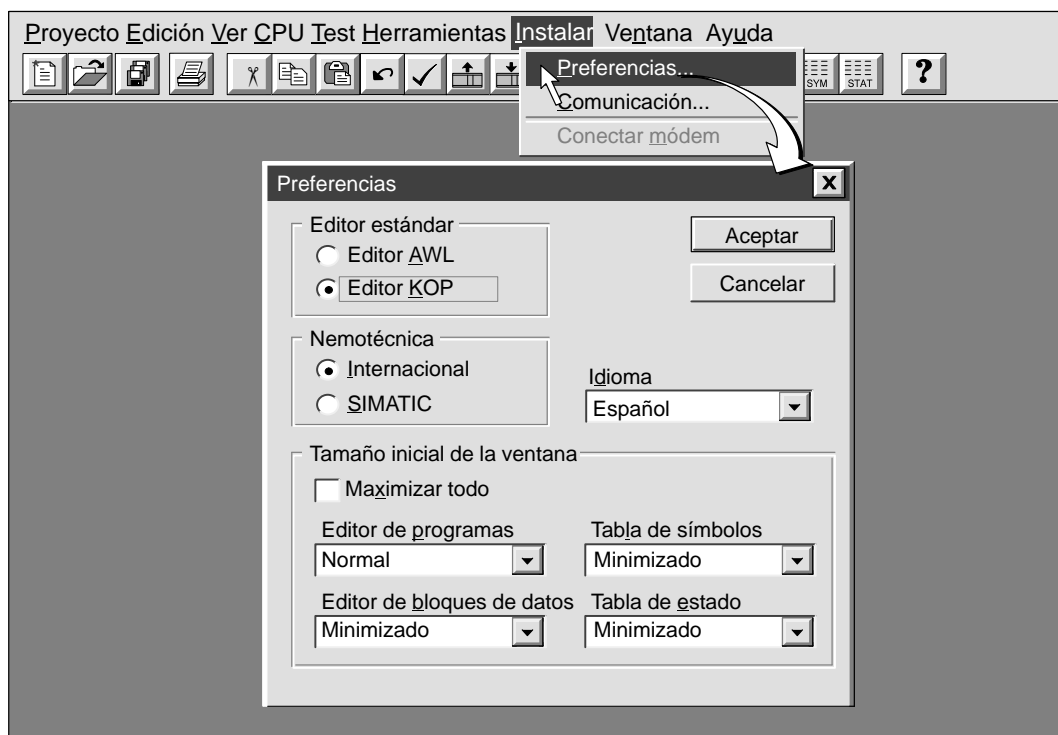


Figura 3-19 Seleccionar las preferencias de programación

### 3.5 Crear y guardar un proyecto

Antes de poder generar un programa es necesario crear o abrir un proyecto. Al crear un proyecto, STEP 7-Micro/WIN abre los siguientes editores:

- Editor KOP o AWL (dependiendo del editor estándar elegido)
- Editor de bloque de datos
- Tabla de estado
- Tabla de símbolos

#### Crear un nuevo proyecto

Mediante el menú Proyecto se puede crear un nuevo proyecto, como muestra la figura 3-20. Elija el comando de menú **Proyecto ► Nuevo....** Aparecerá el cuadro de diálogo "Tipo de CPU". Al elegir una CPU determinada en la lista desplegable, el software visualizará sólo las opciones disponibles para la misma. Si elige "Ninguno", el programa no se restringirá por lo que respecta a la CPU. Al cargar el programa en la CPU, ésta verifica si se utilizan funciones que no están disponibles. Por ejemplo, si su programa contiene una operación no asistida por la CPU correspondiente, el programa se rechazará.


#### Nota

STEP 7-Micro/WIN no comprueba el margen de los parámetros. Se puede indicar, p.ej., VB9999 como parámetro de una operación KOP, aunque dicho parámetro no sea válido.



Figura 3-20 Crear un nuevo proyecto

#### Guardar un proyecto

Para guardar todos los componentes de su proyecto, elija el comando de menú **Proyecto ► Guardar todo** o haga clic en el botón correspondiente: 

Para guardar una copia del proyecto actual bajo otro nombre o en otro directorio, elija el comando de menú **Proyecto ► Guardar como....**

### 3.6 Crear un programa

STEP 7-Micro/WIN permite crear el programa de usuario (OB1) con el editor KOP o el editor AWL.

#### Introducir programas en KOP

La ventana del editor KOP que muestra la figura 3-21 permite escribir programas utilizando símbolos gráficos. La barra de herramientas incluye algunos de los elementos KOP más usuales para introducir los programas. El primer cuadro de lista desplegable (a la izquierda) contiene los grupos de operaciones. Para acceder a dichos grupos, haga clic allí o pulse la tecla F2. Una vez seleccionado un grupo, las operaciones correspondientes al mismo aparecerán en la segunda lista desplegable (a la derecha). Para visualizar una lista de todas las operaciones en orden alfabético, pulse la tecla F9 o elija "Todas las categorías". Alternativamente puede elegir el comando de menú **Ver ► Barra de operaciones** para visualizar la barra de operaciones KOP.

En cada segmento se pueden introducir dos tipos de comentarios:

- Los comentarios de segmento de una sola línea siempre son visibles en el editor KOP. Para acceder a ellos haga clic en cualquier parte del título del segmento.
- Para acceder a los comentarios del segmento de más de una línea haga doble clic en el número del segmento. Dichos comentarios sólo se pueden visualizar en un cuadro de diálogo, pero se imprimen siempre por completo.

Para comenzar a introducir su programa:

1. Para introducir el título del programa, elija el comando de menú **Edición ► Título....** Introduzca el título y haga clic en el botón "Aceptar".
2. Para introducir elementos KOP, elija el tipo de elemento deseado haciendo clic en el botón correspondiente. Alternativamente, puede seleccionarlo de la lista de operaciones.
3. Introduzca la dirección o el parámetro en cada cuadro de texto y pulse la tecla ENTER.

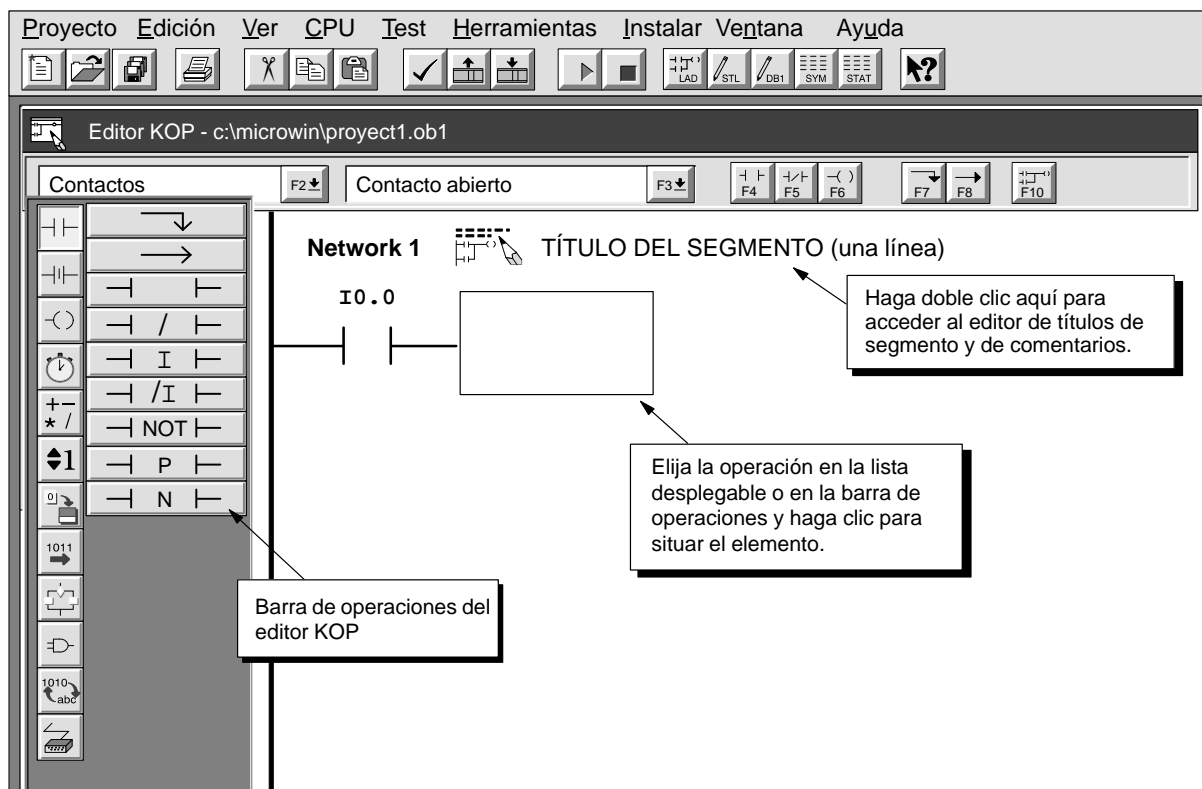


Figura 3-21 Ventana del editor KOP

## Introducir programas en AWL

El editor AWL es un editor de textos que brinda cierta flexibilidad en cuanto a la forma de introducir las operaciones del programa. La figura 3-22 muestra un ejemplo de un programa AWL.

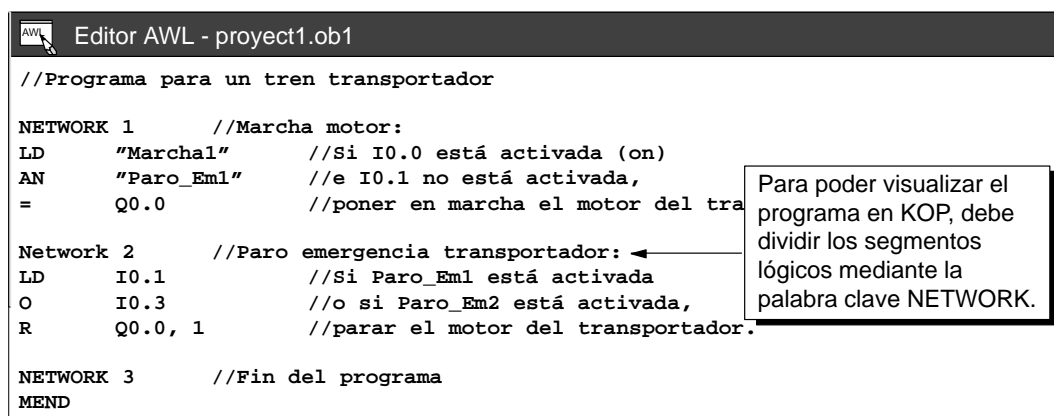


Figura 3-22 Ventana del editor AWL con un programa de ejemplo

Tenga en cuenta las siguientes reglas al introducir programas en AWL:

- Para poder visualizar un programa AWL en KOP debe dividir los segmentos lógicos en segmentos independientes, introduciendo a tal efecto la palabra clave NETWORK (segmento). (Los números de los segmentos se generan automáticamente al compilar o cargar el programa). Para que el programa se pueda visualizar en KOP, entre las palabras clave NETWORK se deberá prever una distancia apropiada.
- Cada comentario debe ir precedido de dos barras inclinadas (/). Cada línea adicional de comentario debe comenzar asimismo con dos barras inclinadas.
- Finalice cada línea pulsando la tecla ENTER.
- Separe cada operación de su dirección o parámetro con un espacio en blanco o con un tabulador.
- No introduzca espacios entre el área de memoria y la dirección (p.ej., introduzca I0.0, y no I 0.0).
- Dentro de una operación, separe cada operando mediante una coma, un espacio en blanco o un tabulador.
- Utilice comillas al introducir los nombres simbólicos. Por ejemplo, si su tabla de símbolos contiene el nombre simbólico Marcha1 para la dirección I0.0, introduzca la operación de la siguiente forma:


```
LD "Marcha1"
```

## Compilar el programa

Una vez completado un segmento o una serie de segmentos puede comprobar la sintaxis lógica eligiendo el comando de menú **CPU ► Compilar** o haciendo clic en el botón correspondiente:



### Cargar el programa en la CPU

Una vez finalizado el programa, el proyecto se puede cargar en la CPU. Elija el comando de menú **Proyecto ► Cargar en CPU** o haga clic en el botón correspondiente de la ventana principal: 

El cuadro de diálogo "Cargar en CPU" permite indicar los componentes del proyecto que se desean cargar (v. fig. 3-23).

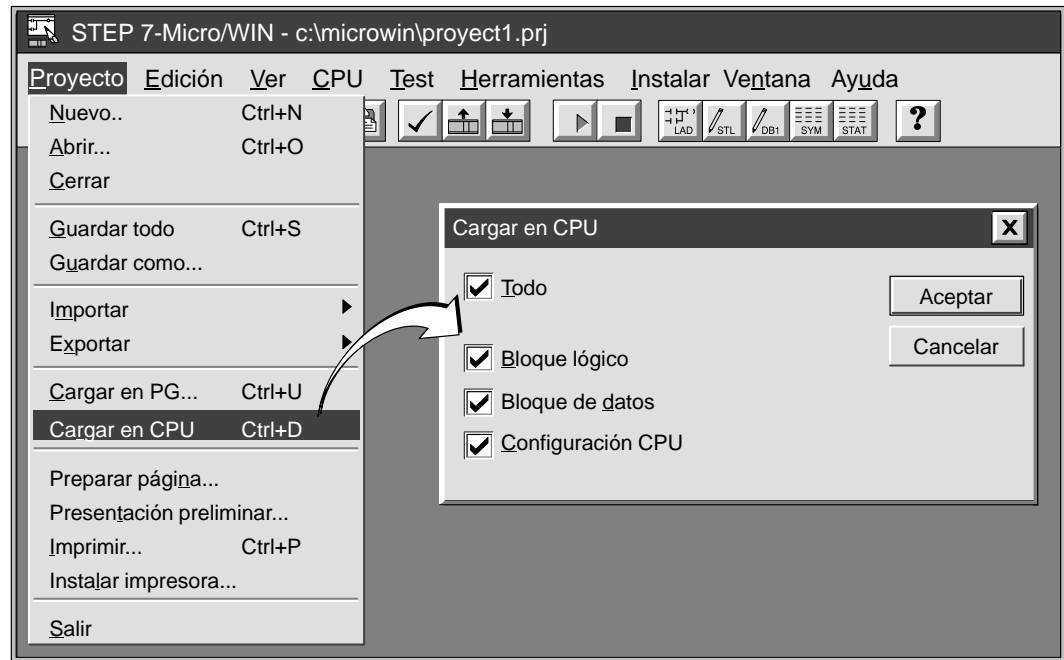


Figura 3-23 Cargar los componentes del proyecto en la CPU

- El bloque lógico (OB1) contiene el programa a ejecutar por la CPU.
- El bloque de datos (DB1) contiene los valores de inicialización utilizados en el programa de usuario.
- La configuración de la CPU (CFG) contiene la información de instalación del sistema, incluyendo los parámetros de comunicación, las áreas remanentes, los ajustes de los filtros de entrada, las contraseñas y los ajustes de las salidas.

Para confirmar los ajustes y cargarlos en la CPU haga clic en el botón "Aceptar" o pulse la tecla ENTER.

### Visualizar un programa en KOP o AWL

Para visualizar un programa en AWL o KOP, elija el comando de menú **Ver ► AWL** o **Ver ► KOP**, como muestra la figura 3-24.

Si cambia de la vista AWL a KOP y luego nuevamente a AWL, es posible que note algunos cambios en la representación del programa AWL tales como:

- Las operaciones y direcciones ya no aparecen en minúsculas sino en mayúsculas.
- Los espacios en blanco entre las operaciones y las direcciones se reemplazan con tabuladores.

Para obtener esa misma representación de las operaciones AWL, puede elegir el comando de menú **CPU ► Compilar** estando activo el editor AWL.

#### Nota

Algunas combinaciones de operaciones AWL no se pueden representar en KOP. En esos casos, el mensaje "Segmento no válido" señala la parte lógica que no se puede representar en KOP.

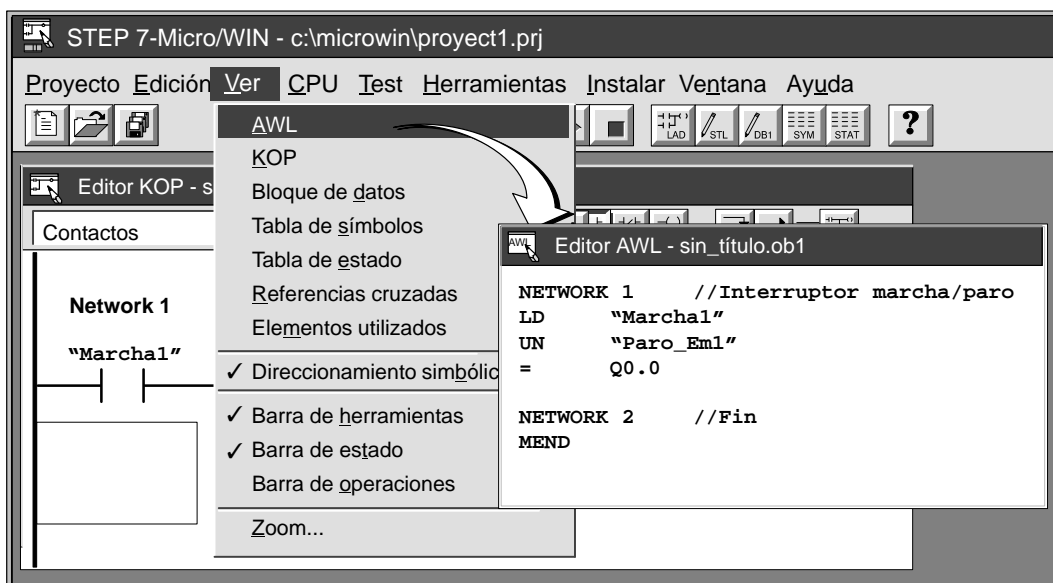


Figura 3-24 Cambiar la visualización del programa de KOP a AWL

### 3.7 Crear un bloque de datos

Con el editor de bloques de datos puede predefinir e inicializar las variables utilizadas en el programa. El uso del bloque de datos es opcional.

El editor de bloques de datos aparece minimizado al tamaño de un icono en la parte inferior de la pantalla principal (si se ha seleccionado así con el comando de menú **Instalar ► Preferencias...**). Para acceder al editor de bloques de datos, haga doble clic en el icono o un solo clic en el botón Restaurar o Maximizar (en Windows 95).

#### Introducir valores de datos

El editor de bloque de datos es un editor de textos que brinda cierta flexibilidad a la hora de elegir el formato para introducir valores de datos.

Tenga en cuenta las siguientes reglas al crear bloques de datos:

- Utilice la primera columna de cada línea para indicar el tamaño de los datos y la dirección inicial de cada valor a almacenar en la memoria V.
- La dirección inicial y los valores de datos se deberán separar con un espacio en blanco o un tabulador.

La figura 3-25 muestra un ejemplo de un bloque de datos con comentarios que describen cada elemento.

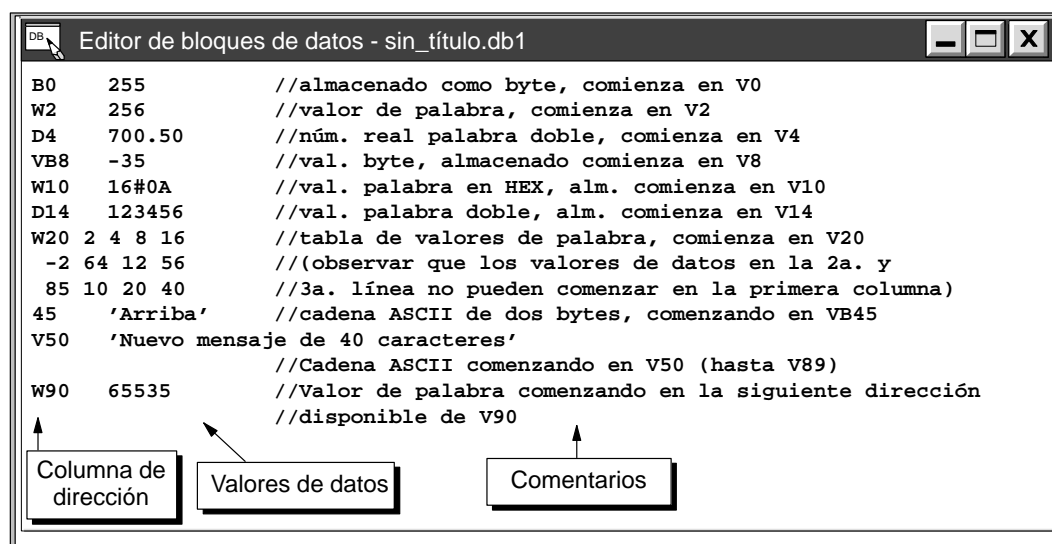


Figura 3-25 Ejemplo de un bloque de datos



### Precaución

STEP 7-Micro/WIN utiliza la primera columna de cada línea del editor de bloques de datos para determinar la dirección inicial de los valores a almacenar en el bloque de datos. Si se introduce un número en la primera columna, éste se interpreta como la dirección inicial en la memoria V para todos los datos siguientes. Si el número contenido en la primera columna es un valor de datos y no una dirección, es posible que los datos introducidos en el bloque de datos se sobrescriban de forma inadvertida con los nuevos datos.

Si se hace referencia a datos incorrectos, el sistema de automatización podría funcionar de forma imprevisible al cargarse el bloque de datos en la CPU. Un funcionamiento imprevisible puede causar la muerte o lesiones personales graves y/o daños a los equipos.

Para asegurar que los datos se almacenen en la dirección correcta de la memoria V, es preciso indicar siempre un tamaño y una dirección, como p.ej. VB100. Asimismo, las entradas efectuadas en la primera columna se deberán leer detenidamente y corregirse en caso necesario, verificando que no se haya introducido allí ningún valor de datos.

En la tabla 3-4 figuran ejemplos de la notación a utilizar al introducir valores para un bloque de datos.

Tabla 3-4 Notación para introducir valores en un bloque de datos

Tipo de datos	Ejemplo
Hexadecimal	16#AB
Entero (decimal)	10    ó    20
Entero con signo (decimal)	-10    ó    +50
Real (en coma flotante): utilice un punto (".") y no una coma (";",")	10.57
Texto (ASCII): texto de la cadena, entre apóstrofos (Nota: "\$" es un comodín para indicar que el carácter siguiente es un apóstrofo o un signo de dólar dentro de una cadena).	'Siemens' 'That's it' 'Sólo \$\$25'

La tabla 3-5 muestra los identificadores válidos para introducir el tamaño de los datos y la dirección inicial.

Tabla 3-5 Identificadores válidos de tamaños

Tamaño de datos	Ejemplo	Descripción
Byte	B0 VB10	Almacena los valores siguientes como bytes de datos, comenzando en la dirección indicada.
Palabra	W0 VW22	Almacena los valores siguientes como palabras de datos, comenzando en la dirección indicada.
Palabra doble	VD100	Almacena los valores siguientes como palabras dobles de datos, comenzando en la dirección indicada.
Autotamaño	10 V10	Almacena los datos en el tamaño mínimo (byte, palabra o palabra doble) requerido para almacenar los valores. Los valores introducidos en esta línea se almacenan comenzando en la dirección de la memoria V que se haya indicado.
Conservar el tamaño anterior	(Columna de dirección vacía)	Almacena los datos en bytes, palabras o palabras dobles, dependiendo del tamaño indicado en la línea anterior.





### 3.8 Utilizar la tabla de estado

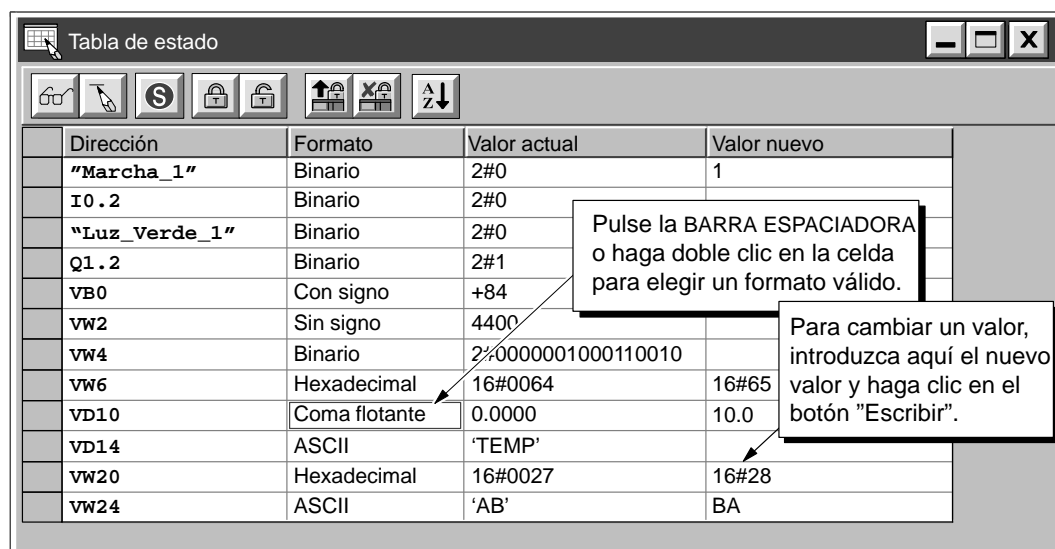
Con la tabla de estado es posible leer, escribir o forzar variables en el programa de usuario.

La tabla de estado aparece minimizada al tamaño de un icono en la parte inferior de la pantalla principal (si se ha seleccionado así con el comando de menú **Instalar ► Preferencias...**). Para acceder a la tabla de estado, haga doble clic en el icono o un clic en el botón Restaurar o Maximizar (en Windows 95).

#### Leer y escribir variables con la tabla de estado

La figura 3-26 muestra un ejemplo de una tabla de estado. Para leer o escribir variables utilizando la tabla de estado:

1. En la primera celda de la columna "Dirección", introduzca la dirección o el nombre simbólico del elemento de su programa cuyo valor desee leer o escribir y pulse la tecla ENTER. Repita este paso para todos los elementos adicionales que desee insertar en la tabla.
2. Si el elemento es un bit (I, Q o M, por ejemplo), se ajusta en la segunda columna el formato binario. Si el elemento es un byte, una palabra o una palabra doble, puede seleccionar la celda en la columna "Formato" y hacer doble clic o pulsar la BARRA ESPACIADORA para hojear los formatos válidos.
3. Para visualizar el valor actual de los elementos de la tabla, haga clic en el botón "Lectura sencilla"  o en el botón "Lectura permanente"  de la tabla de estado.
4. Para detener la actualización del estado, haga clic en el botón "Lectura permanente". 
5. Para cambiar un valor, introduzca el nuevo valor en la columna "Valor nuevo" y haga clic en el botón  para escribir el valor en la CPU.





Dirección	Formato	Valor actual	Valor nuevo
"Marcha_1"	Binario	2#0	1
I0.2	Binario	2#0	
"Luz_Verde_1"	Binario	2#0	
Q1.2	Binario	2#1	
VB0	Con signo	+84	
VW2	Sin signo	4400	
VW4	Binario	2#0000001000110010	
VW6	Hexadecimal	16#0064	16#65
VD10	Coma flotante	0.0000	10.0
VD14	ASCII	'TEMP'	
VW20	Hexadecimal	16#0027	16#28
VW24	ASCII	'AB'	BA




Figura 3-26 Ejemplo de una tabla de estado

## Forzar las variables utilizando la tabla de estado

Para forzar una variable a un valor determinado en la tabla de estado:

1. En la primera celda de la columna "Dirección", introduzca la dirección o el nombre simbólico de la variable que desea forzar.
2. Si el elemento es un bit (p.ej. I, Q o M), se ajusta en la segunda columna el formato binario. Dicho formato no se puede cambiar. Si el elemento es un byte, una palabra o una palabra doble, puede seleccionar la celda en la columna "Formato" y hacer doble clic o pulsar la BARRA ESPACIADORA para hojear los formatos válidos.
3. Para forzar la variable con el valor actual lea primero los valores actuales en el sistema de automatización eligiendo el comando de menú **Test ► Lectura sencilla** o haciendo clic en el botón "Lectura sencilla" .

Haga clic o navegue hasta la celda que contenga el valor actual que desee forzar. Haga clic en el botón "Forzar"  estando el cursor sobre el valor actual que desea utilizar para forzar la variable.

4. Para forzar una variable con un nuevo valor, introduzca éste último en la columna "Nuevo valor" y haga clic en el botón "Forzar". 
5. Para visualizar todos los valores actuales forzados, haga clic en el botón "Leer valores forzados" 
6. Para desforzar en la CPU todas las variables actuales, haga clic en el botón "Desforzar todo" 

## Editar direcciones

Para editar una celda de dirección, utilice las teclas con flecha o el ratón con objeto de seleccionar la celda deseada.

- Al comenzar a teclear el texto, el campo se borrará y se introducirán los nuevos caracteres.
- Si hace doble clic con el ratón o pulsa la tecla F2, el campo se destacará, pudiendo utilizar entonces las teclas con flecha para desplazar el cursor a la posición que desee editar.

### 3.9 Utilizar el direccionamiento simbólico

La tabla de símbolos permite adjudicar nombres simbólicos a las entradas, salidas y marcas internas (v. fig. 3-27). Los símbolos que se hayan asignado a dichas direcciones se pueden utilizar en los editores KOP y AWL, así como en la tabla de estado de STEP 7-Micro/WIN. El editor de bloques de datos no asiste los nombres simbólicos.

#### Reglas para introducir direcciones simbólicas

La primera columna de la tabla de símbolos se utiliza para destacar una fila. Las demás columnas son para el nombre simbólico, la dirección y el comentario. En cada fila se asigna un nombre simbólico a la dirección absoluta de una entrada o salida digitales, de una dirección en la memoria, de una marca especial o de otro elemento. Opcionalmente se puede agregar un comentario a cada símbolo asignado. Tenga en cuenta las siguientes reglas al crear una tabla de símbolos:

- Puede introducir los nombres simbólicos y las direcciones absolutas en cualquier orden.
- Puede utilizar hasta 23 caracteres en el campo del nombre simbólico.
- Puede definir hasta un total de 1.000 símbolos.
- En la tabla de símbolos se distingue entre mayúsculas y minúsculas. Ejemplo: "Bomba1" se considera como un símbolo diferente a "bomba1".
- El editor de la tabla de símbolos elimina todos los espacios en blanco de los nombres simbólicos. Todos los espacios en blanco internos se convierten en caracteres de subrayado sencillo. Por ejemplo, "Motor marcha 2" se convertirá en "Motor\_marcha\_2".
- Los nombres y/o las direcciones simbólicas dobles se marcarán en cursiva de color azul, no se compilarán y no se reconocerán fuera de la tabla de símbolos. Las direcciones que se solapan no se señalarán como si estuvieran repetidas. Por ejemplo, VB0 y VW1 se solapan en la memoria, pero no se identifican como dobles.

#### Llamar al editor de la tabla de símbolos

El editor de la tabla de símbolos aparece minimizado al tamaño de un icono en el borde inferior de la ventana principal. Para acceder a la tabla de símbolos, haga doble clic en este botón o un solo clic en el botón Restaurar o Maximizar (en Windows 95).

	Nombre simbólico	Dirección	Comentario
	<i>Marcha1</i>	I0.0	
	<i>Paro_Em1</i>	I0.1	
	<i>Luz_Verde1</i>	Q1.0	
	<i>MotorMarcha1</i>	Q1.1	<i>Motor línea ensamble 1</i>
	<i>Mezc1_Tempor</i>	T0	
	<i>Mezc2_Tempor</i>	T37	
	<i>Línea1_Contad</i>	C1	
	<i>Relé_1</i>	M0.0	
	<i>Relé_1</i>	M0.1	

Figura 3-27 Ejemplo de una tabla de símbolos

### Funciones de edición en la tabla de símbolos

La tabla de símbolos ofrece las siguientes funciones de edición:

- **Edición ► Cortar / Copiar / Pegar** dentro de una celda o de una celda a otra.
- **Edición ► Cortar / Copiar / Pegar** una o varias filas adyacentes.
- **Edición ► Insertar fila** arriba de la fila en la que esté situado el cursor. Para esta función se puede utilizar también la tecla INSERTAR o INS.
- **Edición ► Borrar fila** para borrar una o varias filas adyacentes que estén destacadas. Para esta función se puede utilizar también la tecla SUPR o DEL.
- Para editar cualquier celda que contenga datos, utilice las teclas con flecha o el ratón con objeto de seleccionarla. Al comenzar a editar se borran los datos existentes y se introducen los nuevos caracteres. Si hace doble clic con el ratón o pulsa la tecla F2, el campo se destacará, pudiendo utilizar entonces las teclas con flecha para desplazar el cursor a la posición que se desee editar.

### Ordenar las entradas de la tabla

Una vez introducidos los nombres simbólicos y sus direcciones absolutas, puede ordenar la tabla de símbolos alfabéticamente por nombres simbólicos o numéricamente por direcciones. Para ordenar las entradas de la tabla:

- Elija el comando de menú **Ver ► Ordenar por nombres** para ordenar los nombres simbólicos alfabéticamente.
- Elija el comando de menú **Ver ► Ordenar por direcciones** para ordenar numéricamente las direcciones absolutas según los tipos de memoria.



## Introducir un programa de ejemplo

Los ejemplos y descripciones que se incluyen en el presente manual asisten la versión 2.1 del software de programación STEP 7-Micro/WIN. Es posible que las versiones antiguas del software funcionen de forma diferente.

Este capítulo describe el uso del software STEP 7-Micro/WIN para realizar las siguientes tareas:

- Introducir un programa de ejemplo para un mezclador con dos bombas alimentadoras
- Crear una tabla de símbolos, una tabla de estado y un bloque de datos
- Observar el programa de ejemplo

STEP 7-Micro/WIN incluye una Ayuda online detallada. Para obtener información más detallada sobre cualquier tema, elija uno de los comandos del menú **Ayuda** o pulse el botón F1.

### Índice del capítulo

Apartado	Descripción	Página
4.1	Crear un programa para una aplicación de ejemplo	4-2
4.2	Tarea: Crear un proyecto	4-6
4.3	Tarea: Crear una tabla de símbolos	4-8
4.4	Tarea: Introducir el programa en KOP	4-10
4.5	Tarea: Crear una tabla de estado	4-14
4.6	Tarea: Cargar y observar el programa de ejemplo	4-15

## 4.1 Crear un programa para una aplicación de ejemplo

### Requisitos para el programa de ejemplo

Después de crear y cargar el programa de ejemplo en la CPU S7-200, ésta podrá procesarlo. La figura 4-1 muestra los componentes necesarios para procesar y observar el programa:

- Cable PC/PPI o tarjeta MPI instalada en el PC y cable RS-485 para conectar a la CPU S7-200
- CPU S7-200
- Simulador de entradas
- Cable y fuente de alimentación
- STEP 7-Micro/WIN 32 versión 2.1 para Windows 95 y Windows NT (32 bits) o STEP 7-Micro/WIN 16 versión 2.1 para Windows 3.1x (16 bits)

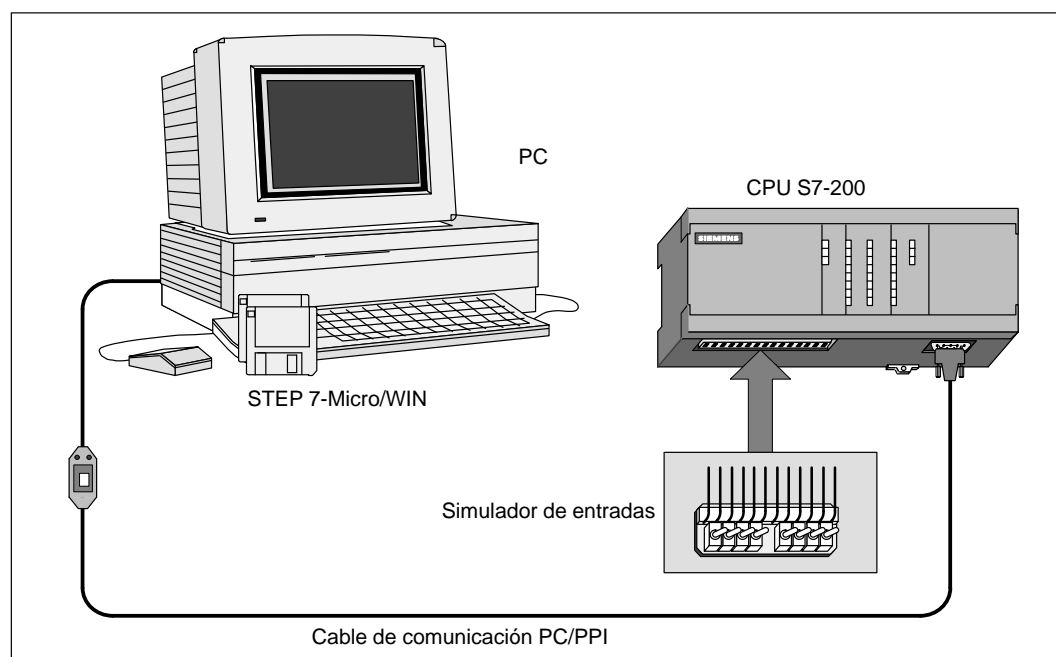


Figura 4-1 Requisitos para ejecutar el programa de ejemplo

**Ejemplo: Mezclador de pintura**

La figura 4-2 muestra un mezclador de pintura. En la parte superior del mezclador hay dos tuberías por donde se suministran dos componentes diferentes. En la parte inferior se aprecia una sola tubería que transporta la mezcla de pintura preparada. Por una tubería en la parte inferior del tanque se conduce la mezcla ya lista. El programa de ejemplo controla la operación de llenado, supervisa el nivel del tanque y controla un ciclo de mezcla y de calefacción como se describe a continuación:

- Paso 1: Llenar el mezclador con el componente 1.
- Paso 2: Llenar el mezclador con el componente 2.
- Paso 3: Supervisar el nivel del mezclador para cerrar el interruptor de nivel superior.
- Paso 4: Mantener el estado de la bomba si se abre el interruptor de arranque.
- Paso 5: Arrancar el ciclo de mezcla y calefacción.
- Paso 6: Poner en marcha el motor del mezclador y la válvula de vapor.
- Paso 7: Vaciar el mezclador.
- Paso 8: Contar cada ciclo.

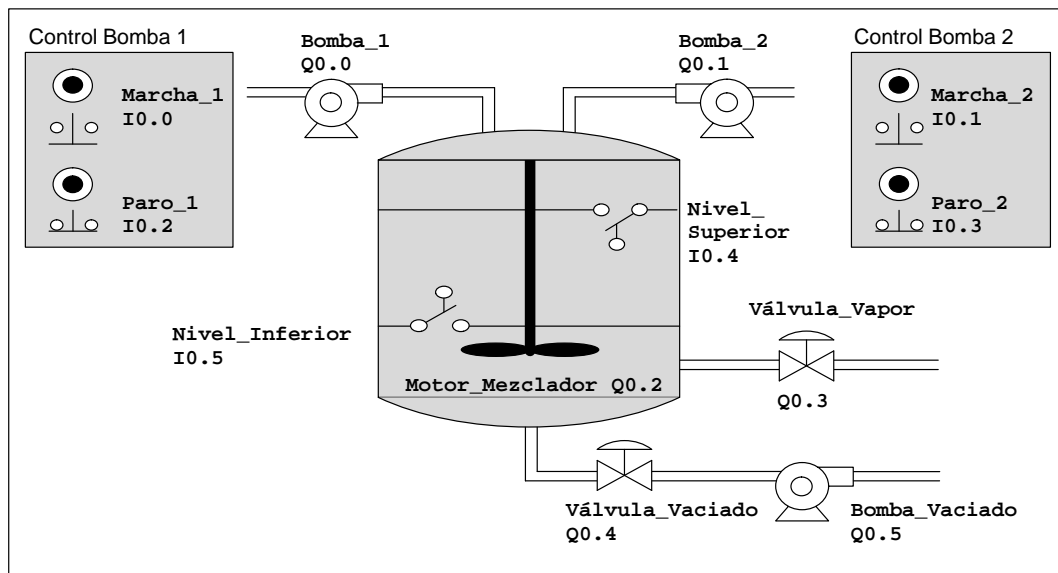


Figura 4-2 Programa de ejemplo: Mezclador de pintura

## Programa de ejemplo en AWL y en KOP

El programa de ejemplo se puede representar en AWL (Lista de instrucciones) o en KOP (Esquema de contactos). En la tabla 4-1 se representa dicho programa en el lenguaje AWL y la figura 4-3 lo muestra en KOP. En los apartados 4.2 a 4.4 se describen las tareas requeridas para representar el programa en AWL o KOP.

Tabla 4-1 Programa de ejemplo en AWL

AWL	Descripción
NETWORK 1	//Llenar el mezclador con el componente 1.
LD "Marcha_1"	
O "Bomba_1"	
A "Paro_1"	
AN "Nivel_Superior"	
= "Bomba_1"	
NETWORK 2	//Llenar el mezclador con el componente 2.
LD "Marcha_2"	
O "Bomba_2"	
A "Paro_2"	
AN "Nivel_Superior"	
= "Bomba_2"	
NETWORK 3	//Activar marca si se alcanza el nivel superior.
LD "Nivel_Superior"	
S "Niv_Sup_Alcanz", 1	
NETWORK 4	//Arrancar temporizador si se alcanza el nivel superior.
LD "Niv_Sup_Alcanz"	
TON "Temporiz_Mezcla"	
NETWORK 5	//Poner en marcha el motor mezclador.
LDN "Temporiz_Mezcla"	
A "Niv_Sup_Alcanz"	
= "Motor_Mezclador"	
= "Válvula_Vapor"	
NETWORK 6	//Vaciar el mezclador.
LD "Temporiz_Mezcla"	
AN "Nivel_Inferior"	
= "Válvula_Vaciado"	
= "Bomba_Vaciado"	
NETWORK 7	//Contar cada ciclo.
LD "Nivel_Inferior"	
A "Temporiz_Mezcla"	
LD "Desactivar"	
CTU "Contador_Ciclos",	
+12	
NETWORK 8	//Desactivar marca si se alcanza el nivel inferior
LD "Nivel_Inferior"	//y si se detiene el temporizador.
A "Temporiz_Mezcla"	
R "Niv_Sup_Alcanz", 1	
NETWORK 9	//Fin del programa principal.
MEND	

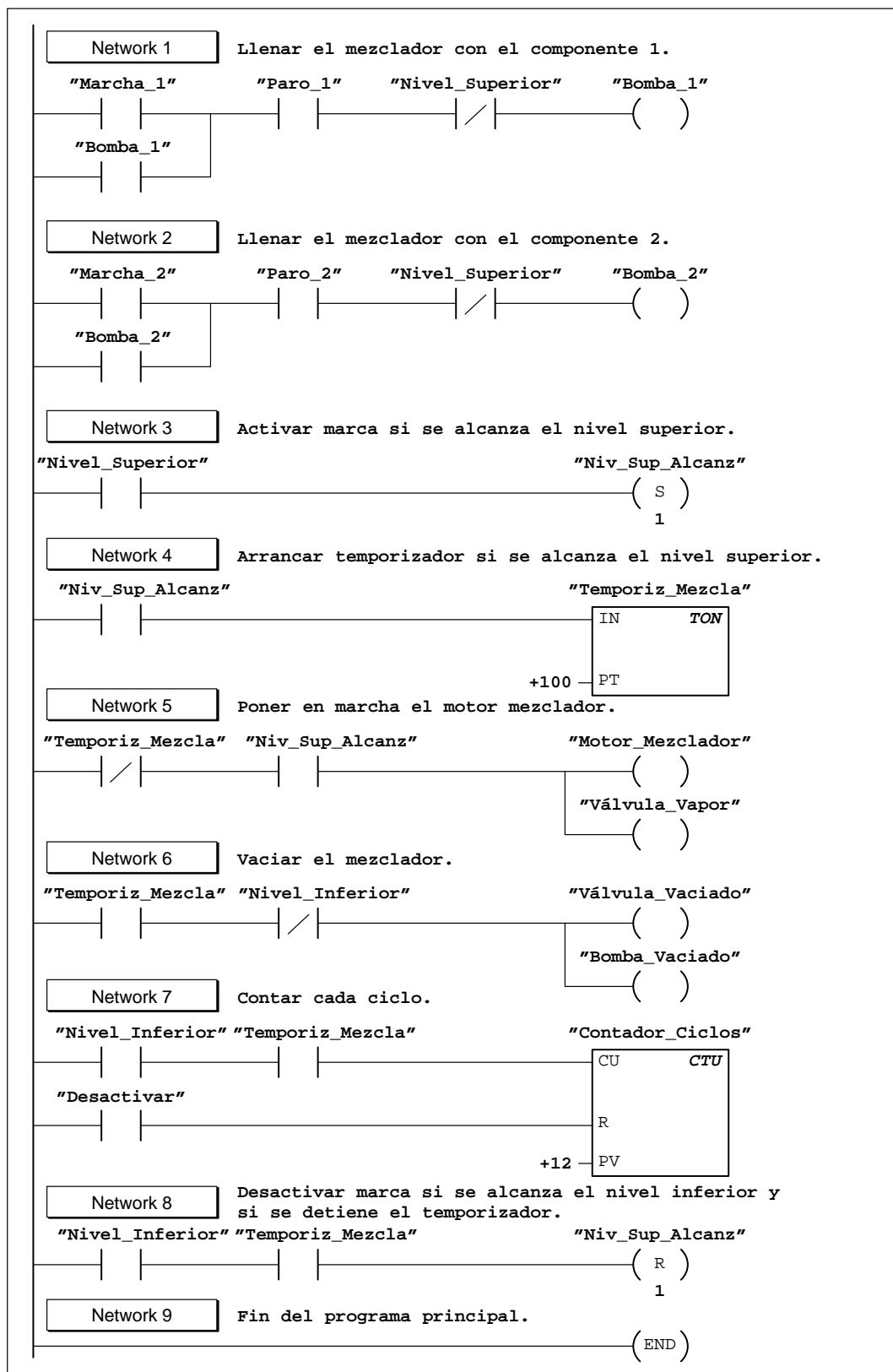


Figura 4-3 Programa de ejemplo en KOP

## 4.2 Tarea: Crear un proyecto

### Crear un nuevo proyecto

Al crear o abrir un proyecto, STEP 7-Micro/WIN inicia el editor KOP o AWL (OB1) y, dependiendo de las preferencias seleccionadas, el editor de bloques de datos (DB1), la tabla de estado y la tabla de símbolos.

Para crear un proyecto, elija el comando de menú **Proyecto ► Nuevo...** como muestra la figura 4-4 o haga clic en el correspondiente botón de la barra de herramientas.



Entonces se abrirá el cuadro de diálogo "CPU". En la lista desplegable, seleccione el tipo de CPU.



Figura 4-4 Crear un nuevo proyecto y seleccionar la CPU

**Adjudicar un nombre al nuevo proyecto**

En cualquier momento puede adjudicar un nombre a su proyecto. Para el presente ejemplo, consulte la figura 4-5 y siga los pasos siguientes para adjudicar un nombre al proyecto:

1. Elija el comando de menú **Proyecto ► Guardar como...**
2. En el campo "Nombre de archivo", introduzca: `project1.prj`.
3. Haga clic en el botón "Guardar".



Figura 4-5 Adjudicar un nombre al nuevo proyecto

### 4.3 Tarea: Crear una tabla de símbolos

#### Abrir el editor de tablas de símbolos

Abra el editor de tablas de símbolos con objeto de definir los nombres simbólicos utilizados para representar las direcciones absolutas en el programa de ejemplo. Haga doble clic en el icono o un solo clic en el botón Restaurar o Maximizar (en Windows 95). Alternativamente, puede elegir el comando de menú **Ver ► Tabla de símbolos**.

#### Introducir los nombres simbólicos

La figura 4-6 muestra una lista de las direcciones absolutas y los correspondientes nombres simbólicos para el programa de ejemplo. Para introducir los nombres simbólicos:

1. Seleccione la primera celda de la columna "Nombre simbólico" e introduzca **Marcha\_1**
2. Pulse la tecla ENTER para desplazarse a la primera celda de la columna "Dirección". Introduzca la dirección **I0.0** y pulse nuevamente la tecla ENTER. Así se pasa a la celda de la columna "Comentario". (Los comentarios son opcionales, pero pueden contribuir a documentar los elementos del programa).
3. Pulse la tecla ENTER para pasar a la siguiente línea y repita los pasos descritos para cada uno de los nombres simbólicos y las direcciones restantes.
4. Utilice el comando de menú **Proyecto ► Guardar todo** para guardar la tabla de símbolos.

Tabla de símbolos - c:\microwin\proyect1.sym			
	Nombre simbólico	Dirección	Comentario
	Marcha_1	I0.0	Interruptor de marcha para componente 1
	Marcha_2	I0.1	Interruptor de marcha para componente 2
	Paro_1	I0.2	Interruptor de paro para componente 1
	Paro_2	I0.3	Interruptor de paro para componente 2
	Nivel_Superior	I0.4	Interruptor de límite para nivel máximo en mezclador
	Nivel_Inferior	I0.5	Interruptor de límite para nivel mínimo en mezclador
	Puesta a 0	I 0.7	Control de desactivación de contador
	Bomba_1	Q0.0	Bomba para componente 1
	Bomba_2	Q0.1	Bomba para componente 2
	Motor_Mezclador	Q0.2	Motor para el mezclador
	Válvula_Vapor	Q0.3	Vapor para calentar la mezcla
	Válvula_Vaciado	Q0.4	Válvula para permitir el vaciado de la mezcla
	Bomba_Vaciado	Q0.5	Bomba para vaciar la mezcla
	Niv_Sup_Alcanz	M0.1	Marca
	Temporiz_Mezcla	T37	Temp. para controlar la mezcla y la calefacción
	Contador_Ciclos	C30	Total de ciclos de mezcla y calefacción completados

Figura 4-6 Tabla de símbolos para el programa de ejemplo

### Programar con direcciones simbólicas

Antes de comenzar a introducir el programa, verifique que haya ajustado el direccionamiento simbólico. Elija el comando de menú **Ver ► Direccionamiento simbólico** y compruebe si junto a dicho comando aparece una marca de verificación para indicar que está habilitado el direccionamiento simbólico.

---

#### **Nota**

En el caso de los nombres simbólicos se distingue entre mayúsculas y minúsculas. El nombre que introduzca deberá coincidir exactamente con el indicado en la tabla de símbolos. En caso contrario, el cursor permanecerá en el elemento y en la barra de estado (en el borde inferior de la ventana principal) aparecerá el mensaje "Parámetro no válido".

---

## 4.4 Tarea: Introducir el programa en KOP

### Abrir el editor KOP

Para acceder al editor KOP, haga doble clic en el botón correspondiente que aparece en el borde inferior de la pantalla principal. La figura 4-7 muestra algunas de las herramientas básicas de dicho editor.

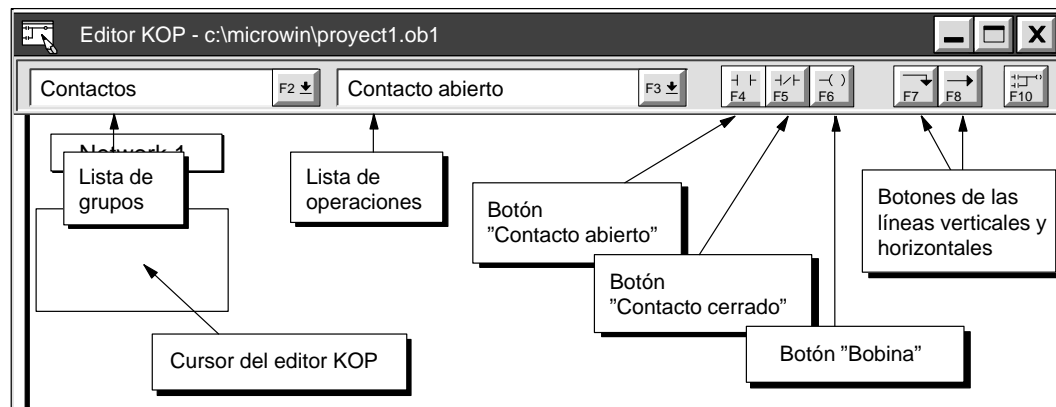


Figura 4-7 Algunas herramientas básicas del editor KOP

### Barra de operaciones del editor KOP

También puede elegir el comando de menú **Ver ► Barra de operaciones** para visualizar la barra de operaciones KOP (v. fig. 4-8).

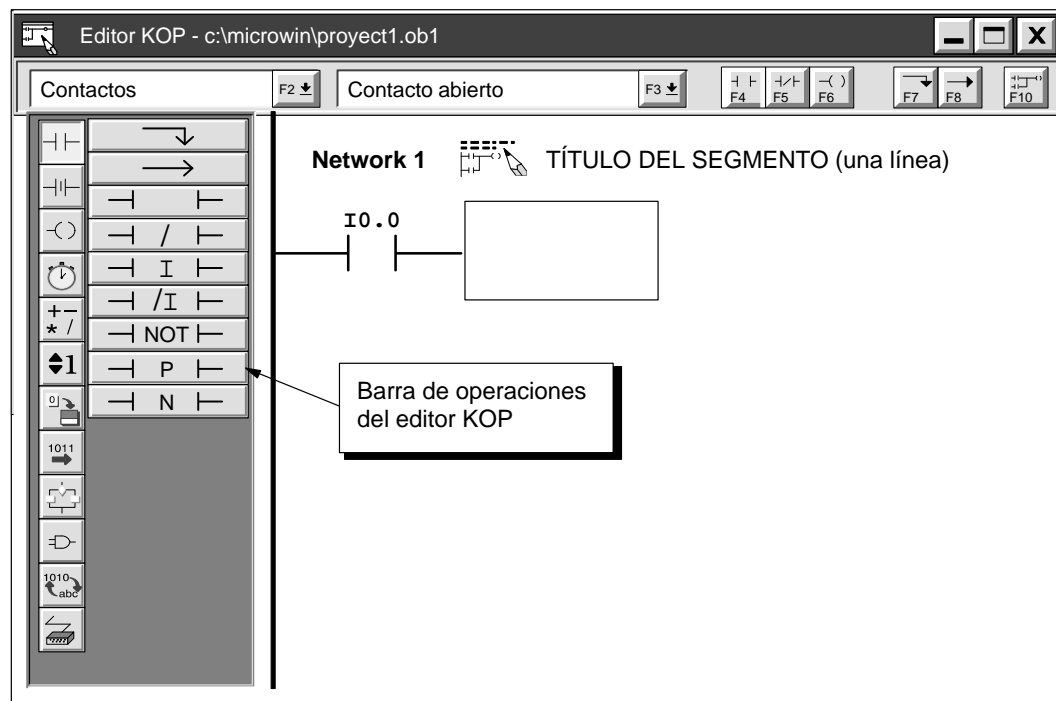


Figura 4-8 Algunas herramientas básicas del editor KOP

### Introducir el primer elemento del segmento

Para introducir el primer segmento del programa de ejemplo:

1. Haga doble clic en el cuadro "Network" numerado o cerca del mismo para acceder al campo "Título" del editor de comentarios. Introduzca el comentario que muestra la figura 4-9 y haga clic en "Aceptar."
  2. Pulse la tecla con flecha abajo. El cursor del editor KOP se desplazará hacia abajo y se situará en la primera posición de la columna izquierda.
  3. Seleccione un contacto abierto eligiendo "Contactos" en la lista desplegable izquierda y luego "Contacto abierto" en la lista que aparece a la derecha.
  4. Pulse la tecla ENTER. Aparecerá un contacto abierto y, arriba del mismo, se visualizará el nombre ("Marcha\_1") destacado.
- (Cada vez que introduzca un contacto, el software visualizará la dirección predeterminada de I0.0 que se ha definido en el presente ejemplo como "Marcha\_1" en la tabla de símbolos).
5. "Marcha\_1" es el primer elemento requerido para el segmento (Network) 1. Pulse la tecla ENTER para confirmar el primer elemento y su nombre simbólico. El cursor del editor KOP se desplazará a la segunda posición de la columna.

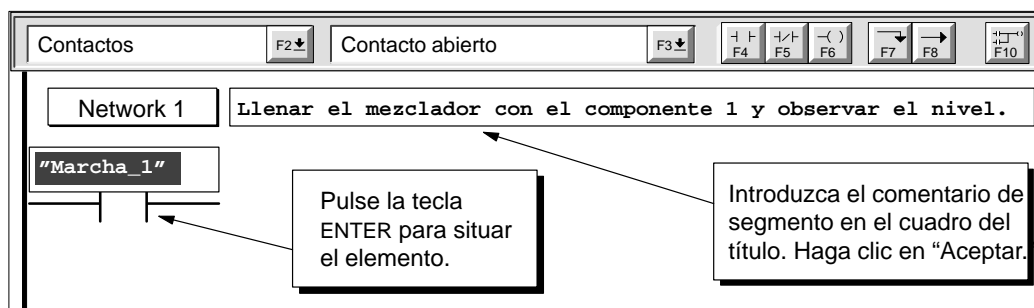


Figura 4-9 Introducir el comentario de segmento y el primer elemento KOP

Para introducir los demás elementos del primer segmento:

1. Pulse la tecla ENTER para introducir el segundo elemento. Aparecerá un contacto abierto y, arriba del mismo, se visualizará destacado el nombre simbólico predeterminado ("Marcha\_1").
2. Introduzca **Paro\_1** y pulse la tecla ENTER. El cursor se desplazará a la siguiente columna.
3. Haga clic en el botón "Contacto cerrado" ("F5"). Aparecerá un contacto cerrado con el nombre simbólico predeterminado ("Marcha\_1") destacado.
4. Introduzca **Nivel\_superior** y pulse la tecla ENTER.

El segmento KOP debería ser entonces similar al que muestra la figura 4-10.

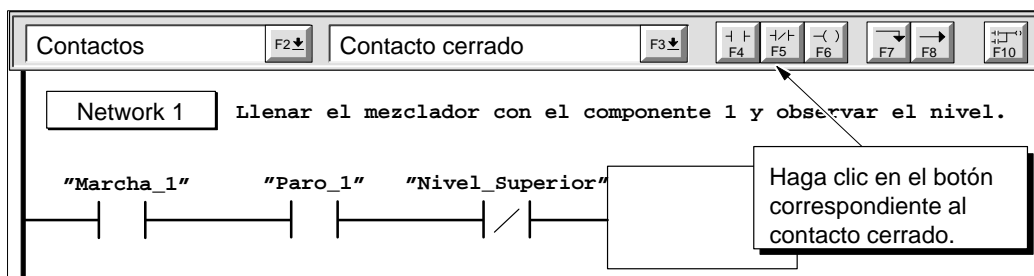


Figura 4-10 Introducir el siguiente elemento de KOP

El cursor del editor KOP está situado ahora a la derecha de la entrada "Nivel\_Superior" representada como contacto cerrado. Consulte la figura 4-11 y siga los siguientes pasos para completar el primer segmento:

1. Haga clic en el botón "Bobina" ("F6"), desplace el puntero del ratón al interior del cursor KOP y haga clic. Aparecerá una bobina y, arriba de la misma, el nombre simbólico "Bomba\_1" destacado. (A cada bobina que introduzca se le asigna la dirección predeterminada Q0.0 que se ha definido en este caso como "Bomba\_1" en la tabla de símbolos).
2. Pulse la tecla ENTER para confirmar la bobina y su nombre simbólico.
3. Utilice el ratón o pulse la tecla con flecha izquierda para retornar el cursor al primer elemento del segmento actual.
4. Haga clic en el botón "Línea vertical" ("F7") para unir con una línea vertical el primer contacto con el segundo.
5. Haga clic en el botón "Contacto abierto" ("F4") de la barra de herramientas KOP y pulse la tecla ENTER. Aparecerá un contacto llamado "Marcha\_1".
6. Introduzca **Bomba\_1** y pulse la tecla ENTER.

Así se completa el primer segmento.

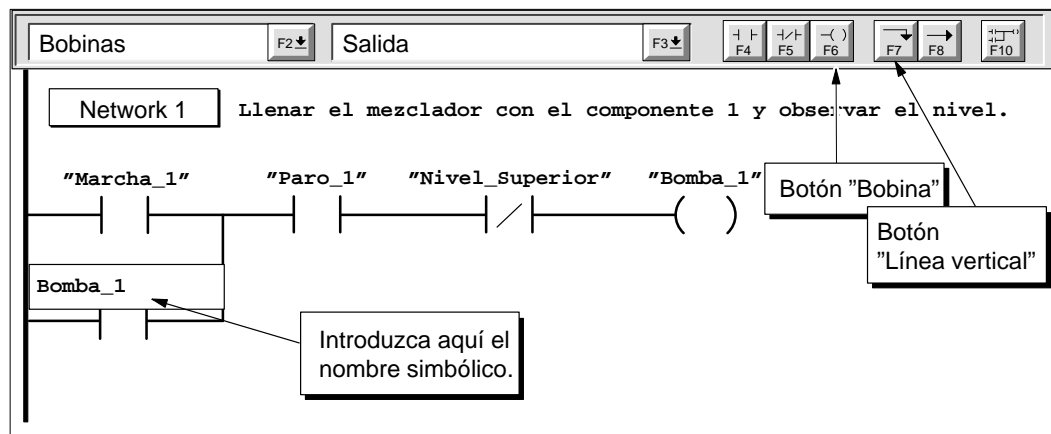


Figura 4-11 Completar el primer segmento

### Introducir el segundo segmento

Para introducir el segundo segmento del programa de ejemplo:

1. Utilice el ratón o pulse la tecla con flecha abajo para desplazar el cursor al segmento (Network) 2.
2. En el campo de comentario del segmento, introduzca el comentario que muestra la figura 4-12. (Puesto que el comentario del segmento 2 es prácticamente idéntico al comentario del segmento 1, puede seleccionar y copiar el texto del segmento 1 y pegarlo en el campo de comentario del segmento 2, cambiando luego el número del componente de 1 a 2).
3. Repita los pasos realizados para introducir los elementos del segmento 1, utilizando los nombres simbólicos que muestra la figura 4-12.
4. Una vez finalizado el segmento 2, desplace el cursor hacia abajo hasta el segmento 3.

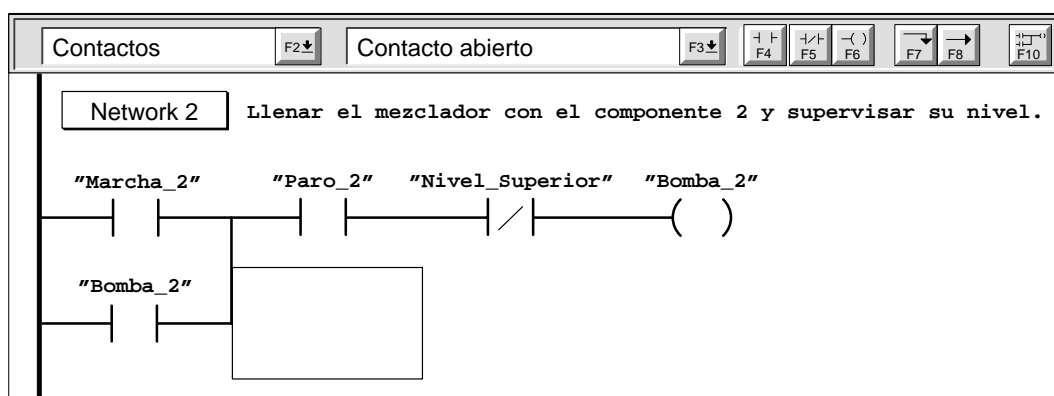



Figura 4-12 Introducir el segundo segmento

### Introducir los demás segmentos


Para introducir los demás segmentos puede seguir el mismo procedimiento utilizado hasta ahora. La figura 4-3 muestra los demás segmentos.

### Compilar el programa

Una vez completado el programa de ejemplo, compruebe la sintaxis eligiendo el comando de menú **CPU ► Compilar** o haciendo clic en el botón "Compilar": 

Si ha introducido correctamente todos los segmentos como muestra el programa de ejemplo, obtendrá el mensaje "Compilación finalizada" que incluye también información sobre el número de segmentos y la cantidad de memoria utilizada por el programa. En caso contrario, el mensaje de compilación indicará qué segmento(s) contiene(n) errores.

### Guardar el programa de ejemplo

Para guardar su proyecto puede elegir el comando de menú **Proyecto ► Guardar todo** o hacer clic en el correspondiente botón.  Así se almacenarán también los demás componentes de su programa de ejemplo.

## 4.5 Tarea: Crear una tabla de estado

### Crear la tabla de estado del programa de ejemplo


Para observar el estado de los elementos seleccionados en el programa de ejemplo es preciso crear una tabla de estado que contenga los elementos que desee observar mientras se ejecuta el programa. Para acceder al editor de la tabla de estado, haga doble clic en el icono correspondiente que aparece en el borde inferior de la pantalla principal. Para crear la tabla de estado del programa de ejemplo:

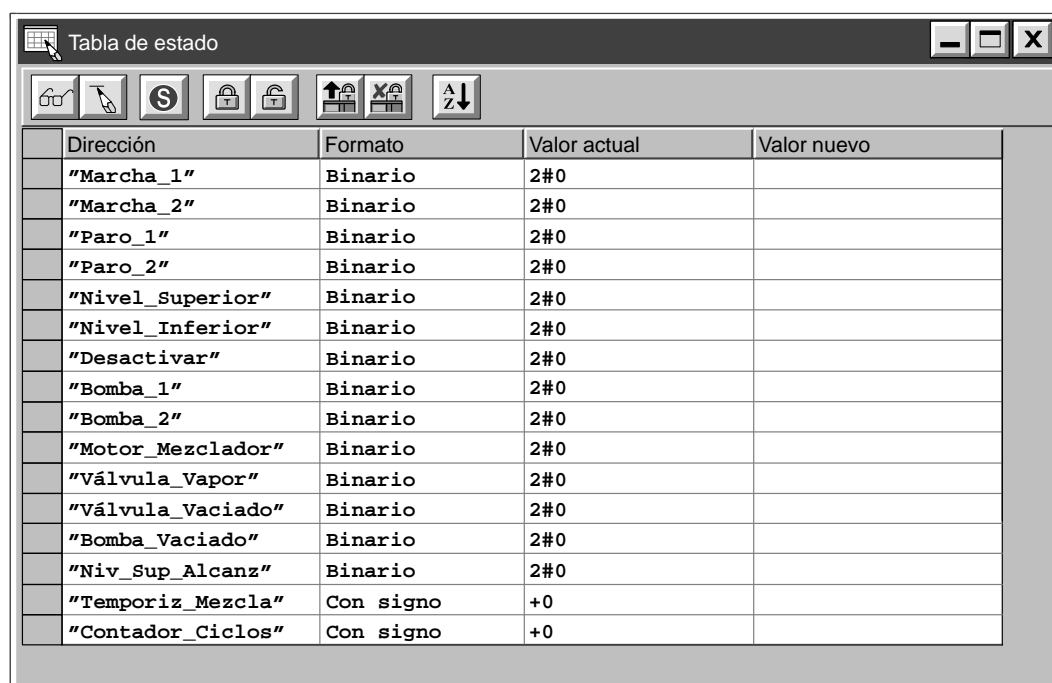
1. Seleccione la primera celda de la columna "Dirección" y teclee **Marcha\_1**
2. Pulse la tecla ENTER para confirmar su entrada. Este tipo de elemento sólo se puede visualizar en formato binario (1 ó 0), por lo que no será posible cambiar el tipo de formato.
3. Seleccione la siguiente fila y repita los pasos descritos arriba para cada uno de los elementos restantes como muestra la figura 4-13.

Si está destacada una celda donde figure una dirección y la fila debajo de la misma está vacía, las direcciones contenidas en las filas siguientes se incrementarán automáticamente al pulsar la tecla ENTER. Para obtener más información acerca de la tabla de estado, consulte la Ayuda online.

Puede utilizar el comando de menú **Edición ► Insertar fila** (o bien la tecla INSERTAR o INS) para insertar una línea en blanco encima de la línea donde está situado el cursor.

4. El temporizador T37 y el contador C30 se pueden visualizar en otros formatos. Estando destacada la celda de la columna "Formato", pulse la BARRA ESPACIADORA con objeto de desplazarse por los formatos válidos para dichos elementos. En el presente ejemplo elija "Con signo" para el temporizador y el contador.

Para guardar la tabla de estado, elija el comando de menú **Proyecto ► Guardar todo** o haga clic en el correspondiente botón. 



Dirección	Formato	Valor actual	Valor nuevo
"Marcha_1"	Binario	2#0	
"Marcha_2"	Binario	2#0	
"Paro_1"	Binario	2#0	
"Paro_2"	Binario	2#0	
"Nivel_Superior"	Binario	2#0	
"Nivel_Inferior"	Binario	2#0	
"Desactivar"	Binario	2#0	
"Bomba_1"	Binario	2#0	
"Bomba_2"	Binario	2#0	
"Motor_Mezclador"	Binario	2#0	
"Válvula_Vapor"	Binario	2#0	
"Válvula_Vaciado"	Binario	2#0	
"Bomba_Vaciado"	Binario	2#0	
"Niv_Sup_Alcanz"	Binario	2#0	
"Temporiz_Mezcla"	Con signo	+0	
"Contador_Ciclos"	Con signo	+0	



Figura 4-13 Tabla de estado para el programa de ejemplo

## 4.6 Tarea: Cargar y observar el programa de ejemplo

Ahora debe cargar el programa en la CPU y cambiar la CPU a modo RUN. De este modo puede observar o comprobar el programa.

### Cargar el proyecto en la CPU

Antes de cargar el programa en la CPU, asegúrese de que ésta se encuentre en modo STOP. Para cargar el programa en la CPU:

1. Coloque el selector de modos de operación de la CPU (ubicado bajo la tapa de acceso de la misma) en la posición TERM o STOP.
2. Elija el comando de menú **CPU ► STOP** o haga clic en el botón correspondiente  en la ventana principal.
3. Elija "Sí" para confirmar esta acción.
4. Elija el comando de menú **Proyecto ► Cargar en CPU...** o haga clic en el botón "Cargar en CPU" en la ventana principal: .
5. El cuadro de diálogo "Cargar en CPU" permite indicar los componentes del proyecto que desee cargar. Pulse la tecla ENTER o haga clic en "Aceptar."

Un mensaje le informa si la operación de carga se realizó satisfactoriamente.

---

#### Nota


STEP 7-Micro/WIN no comprueba si su programa utiliza operandos o direcciones de entradas y salidas válidas para la CPU en cuestión. Si intenta cargar en la CPU un programa que utilice operandos o direcciones no comprendidas en el margen de la CPU u operaciones no asistidas por la misma, la CPU rechazará el intento de cargar el programa y visualizará un mensaje de error.

Verifique que todas las direcciones de la memoria, así como las direcciones de E/S y las operaciones utilizadas en su programa, sean válidas para la CPU en cuestión.

---

### Cambiar la CPU a modo RUN

Si la operación de carga se efectuó con éxito, podrá cambiar la CPU a modo RUN:

1. Elija el comando de menú **CPU ► RUN** o haga clic en el botón correspondiente  en la ventana principal.
2. Elija "Sí" para confirmar esta acción.

## Observar el estado KOP

Si ha activado el estado KOP, se visualizará el estado actual de los eventos en su programa. Abra la ventana del editor KOP y elija el comando de menú

### Test ► Activar estado KOP.

Si tiene un simulador de entradas conectado a los terminales de su CPU puede colocar los interruptores en posición "ON" para observar la circulación de la corriente y la ejecución de la lógica. Por ejemplo, si activa los interruptores I0.0 y I0.2, estando desactivado el interruptor de I0.4 ("Nivel\_Superior"), se completará el flujo de corriente del segmento 1. El aspecto del segmento será como el que muestra la figura 4-14.

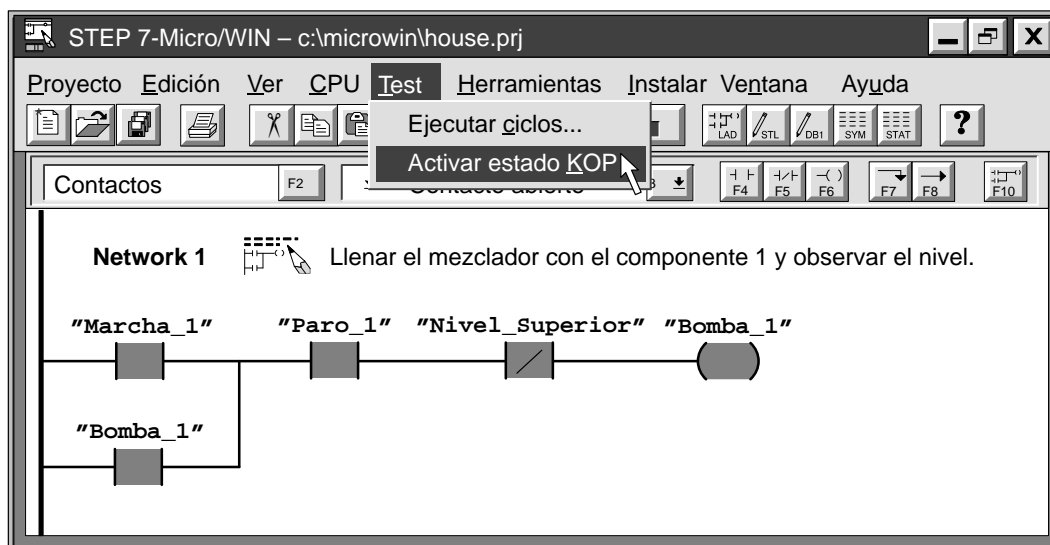


Figura 4-14 Observar el estado del primer segmento

Si el programa de STEP 7-Micro/WIN no es igual al de la CPU, aparecerá el mensaje de advertencia que muestra la figura 4-15. Allí tiene la opción de comparar el programa con la CPU o bien, continuar la acción o cancelar.

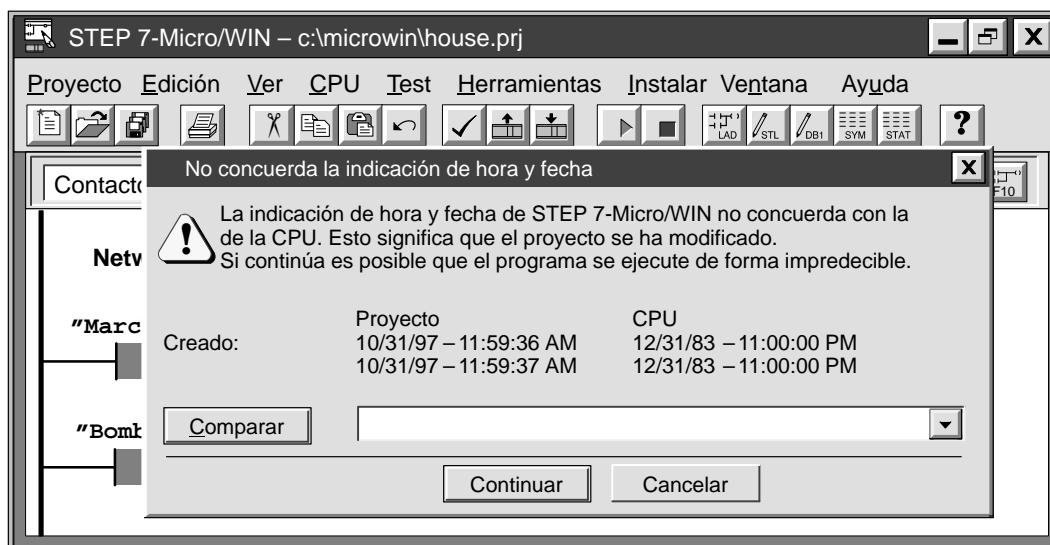





Figura 4-15 Advertencia "No concuerda la indicación de hora y fecha"

### Ver el estado actual de los elementos del programa

La tabla de estado se puede utilizar para observar o modificar los valores actuales de las entradas y salidas, así como de las direcciones de la memoria. Abra nuevamente la tabla de estado y elija el comando de menú **Test ► Lectura permanente** como muestra la figura 4-16. Al activar o desactivar los interruptores estando la CPU en modo RUN, la tabla indica el estado actual de cada elemento.

- Para visualizar el valor actual de los elementos del programa, haga clic en el botón "Lectura sencilla"  o en el botón "Lectura permanente"  en la ventana de la tabla de estado.
- Para detener la lectura del estado, haga clic en el botón "Lectura permanente"  en la ventana de la tabla de estado.

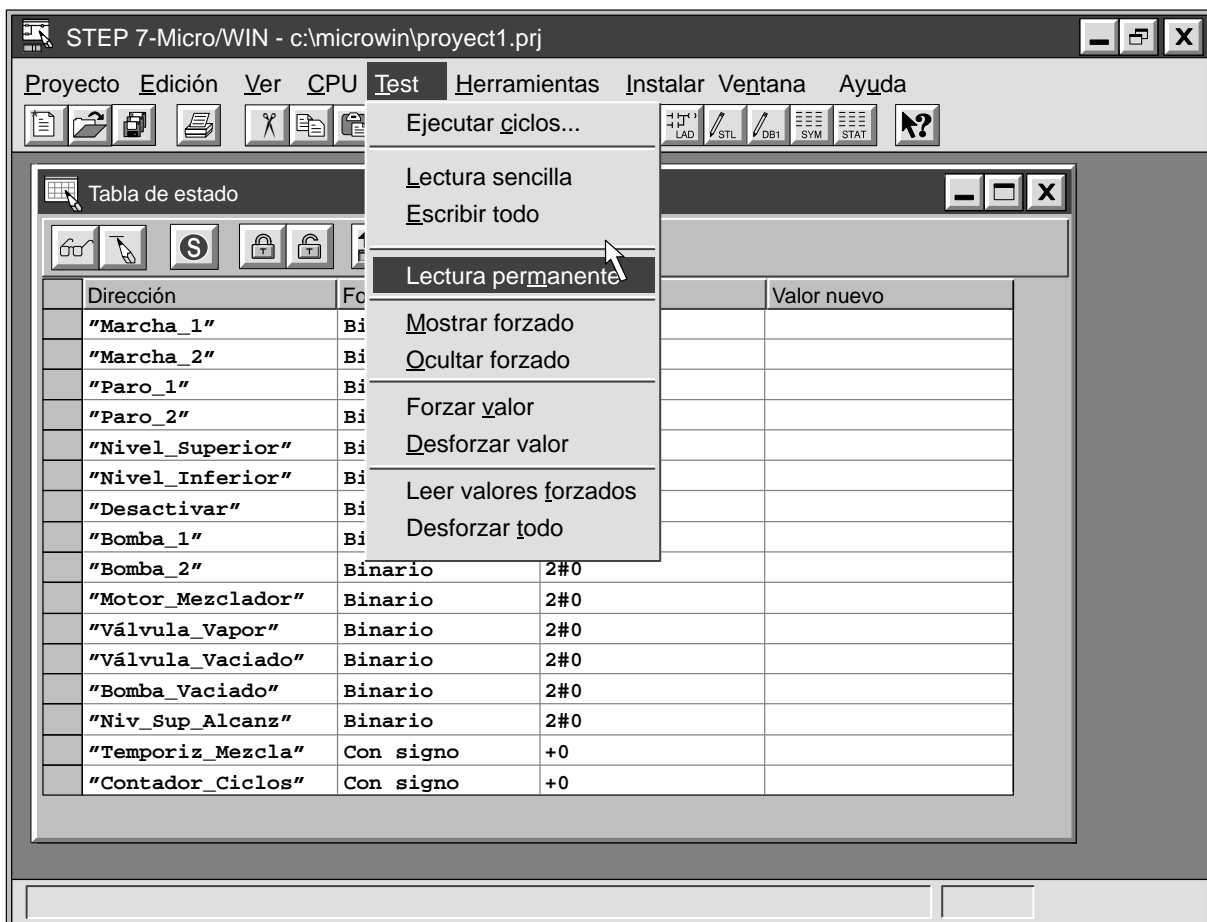


Figura 4-16 Observar el estado del programa de ejemplo



## Funciones adicionales de STEP 7-Micro/WIN

# 5

En el presente capítulo se indica cómo utilizar el Asistente TD 200 para configurar el visualizador de textos TD 200. También se explica cómo emplear el Asistente de operaciones S7-200 para configurar funciones complejas. Asimismo, se describen las demás funciones nuevas de la versión 2.1 de STEP 7-Micro/WIN.

### Índice del capítulo

Apartado	Descripción	Página
5.1	Utilizar el Asistente TD 200 para configurar el visualizador de textos TD 200	5-2
5.2	Utilizar el Asistente de operaciones S7-200	5-12
5.3	Utilizar el Asistente para filtrar entradas analógicas	5-14
5.4	Utilizar las referencias cruzadas	5-17
5.5	Usar la lista de elementos utilizados	5-18
5.6	Utilizar la función para buscar y reemplazar	5-19
5.7	Documentar el programa	5-21
5.8	Imprimir el programa	5-23

### 5.1 Utilizar el Asistente TD 200 para configurar el visualizador de textos TD 200

El TD 200 es un visualizador de textos para ver los mensajes habilitados por la CPU S7-200 (v. fig. 5-1). No es necesario configurar ni programar el TD 200. Los únicos parámetros de operación almacenados en el visualizador de textos son las direcciones del TD 200 y de la CPU, así como la velocidad de transferencia y la dirección del bloque de parámetros. La configuración del TD 200 se encuentra almacenada en un bloque de parámetros depositado en la memoria V (memoria de variables) de la CPU. Los parámetros de operación del TD 200, tales como el idioma, la frecuencia de actualización, los mensajes y los bits de habilitación de mensajes están almacenados en un programa en la CPU.

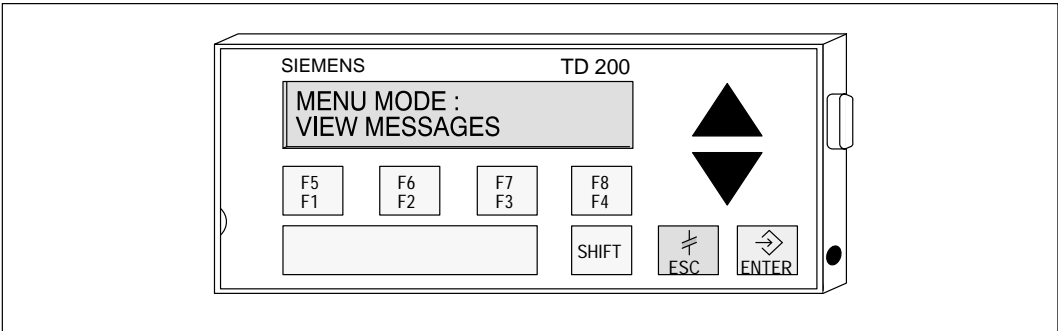


Figura 5-1 Visualizador de textos SIMATIC TD 200

#### Definir el bloque de parámetros del TD 200

El bloque de parámetros comprende 10 ó 12 bytes de la memoria que definen los modos de operación y señalan a la dirección en la memoria de la CPU donde se almacenan los mensajes actuales, como muestra la figura 5-2. Cuando se conecta la alimentación del TD 200, éste busca un identificador del bloque de parámetros en la CPU (en el offset configurado en el TD 200). Dicho identificador lo componen los caracteres ASCII "TD", o bien un offset de la dirección del bloque de parámetros. El TD 200 lee luego los datos contenidos en dicho bloque.

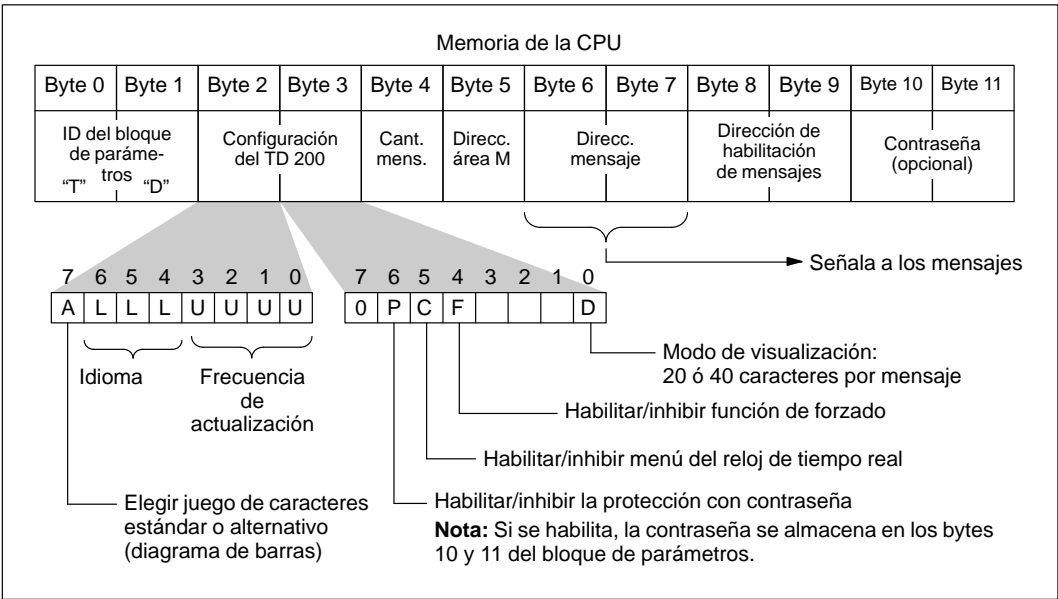


Figura 5-2 Bloque de parámetros del TD 200

### Utilizar el Asistente de configuración del TD 200

STEP 7-Micro/WIN incorpora un Asistente que permite configurar fácilmente el bloque de parámetros y los mensajes en el área de datos de la memoria de la CPU S7-200. El Asistente de configuración del TD 200 escribe automáticamente el bloque de parámetros y los textos de los mensajes en el editor de bloques de datos, una vez que se hayan elegido las opciones y creado los mensajes. Dicho bloque de datos se puede cargar entonces en la CPU. Para obtener informaciones más detalladas acerca del TD 200, consulte el *Manual del usuario Visualizador de textos SIMATIC TD 200*.

Para crear el bloque de parámetros y los mensajes del TD 200:

1. Elija el comando de menú **Herramientas ► Asistente TD 200...** como muestra la figura 5-3.
2. Haga clic en el botón "Siguiente >" o, en la lista desplegable, elija un bloque de parámetros existente y siga los pasos para crear o editar el bloque de parámetros del TD 200 en la memoria V.

En cualquier momento puede hacer clic en el botón "< Anterior" para retornar a un cuadro de diálogo anterior, en caso de que desee cambiar o consultar alguno de los parámetros ajustados.

3. En el cuadro de diálogo final, haga clic en el botón "Cerrar" para confirmar sus ajustes y guardar el bloque de parámetros. El bloque de parámetros configurado se puede visualizar abriendo el editor de bloques de datos.

Tras cargar todos los bloques en la CPU S7-200, el bloque de datos que contiene el bloque de parámetros del TD 200 se depositará en la memoria de la CPU. El TD 200 puede leerlo entonces de allí.

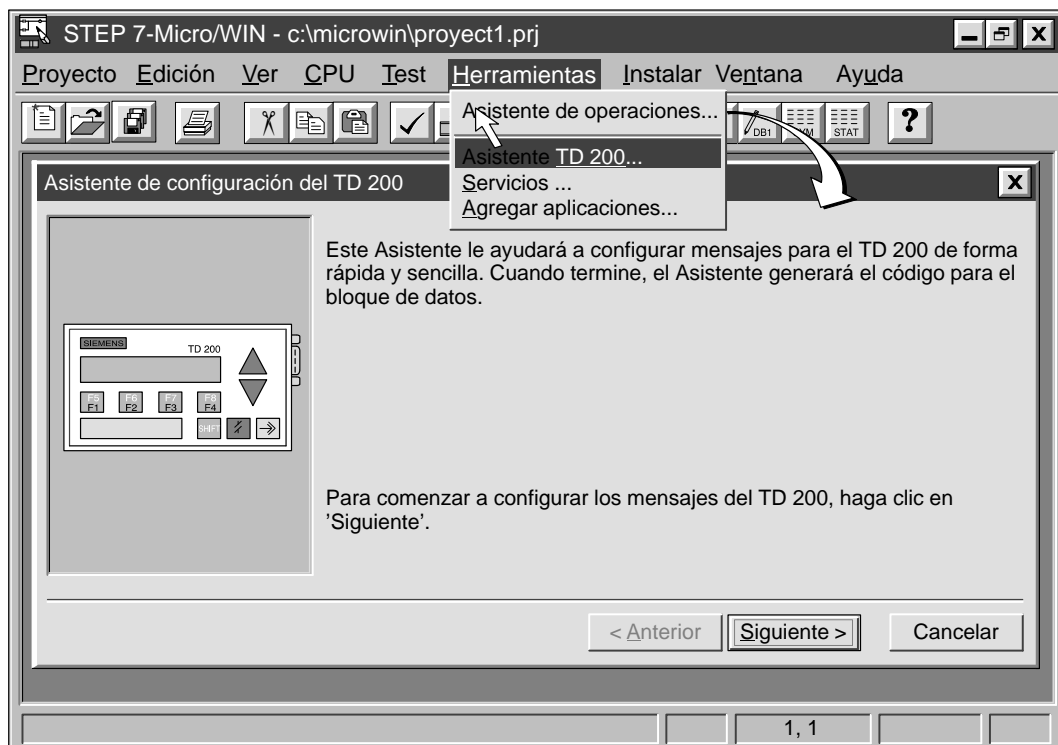


Figura 5-3 Acceder al Asistente de configuración del TD 200

### Elegir el idioma y el juego alternativo de caracteres

En el primer cuadro de diálogo del Asistente de configuración del TD 200 puede elegir el idioma de los menús y el juego de caracteres deseado. Para elegir el idioma en el que desee visualizar los menús del TD 200, utilice el cuadro de lista desplegable que muestra la figura 5-4. Indique si desea activar el juego alternativo de caracteres que permite visualizar diagramas de barras en el TD 200. El Asistente TD 200 activa entonces los bits correspondientes en el byte 2 del bloque de parámetros.

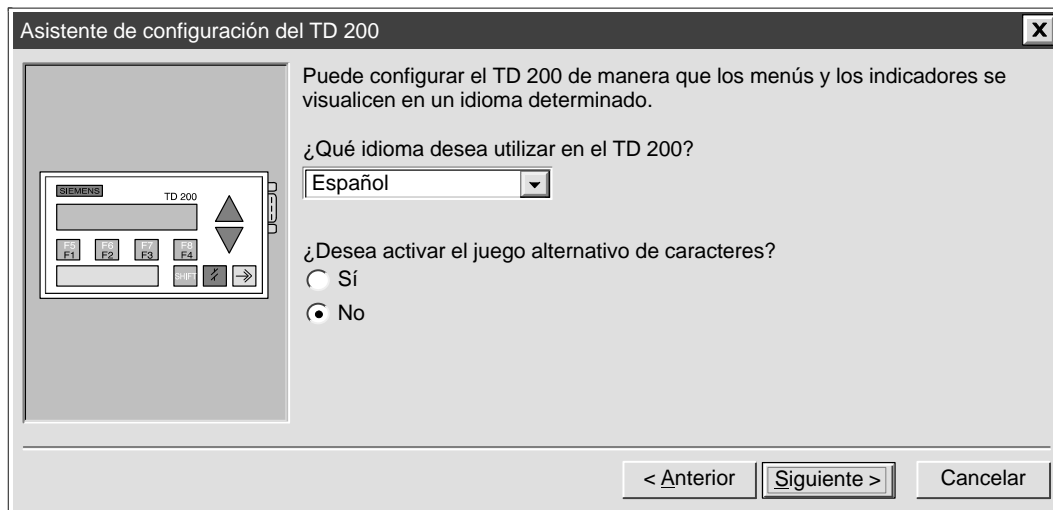


Figura 5-4 Idioma y juego de caracteres del TD 200

### Habilitar el reloj de tiempo real, el forzado de E/S y la protección con contraseña

Utilice los botones de opción para seleccionar las preferencias que muestra la figura 5-5. Si se habilita la protección con contraseña, aparecerá un campo donde es preciso asignar ésta última. Para obtener informaciones más detalladas acerca de estas opciones, consulte el *Manual del usuario Visualizador de textos SIMATIC TD 200*. El Asistente TD 200 activa entonces los bits correspondientes en el byte 3 del bloque de parámetros.

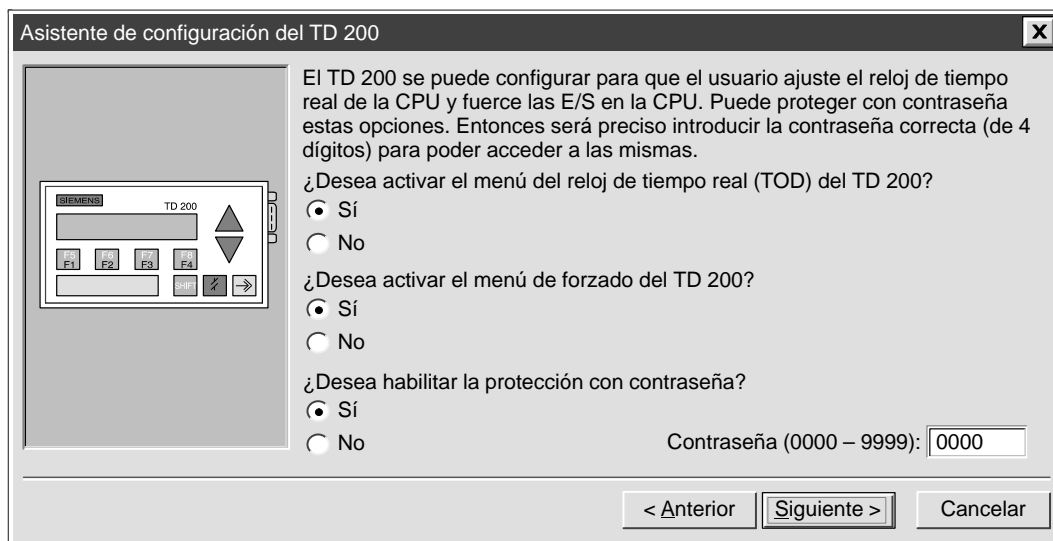


Figura 5-5 Reloj de tiempo real, forzado de E/S y protección con contraseña del TD 200

### Definir las teclas de función para las marcas y visualizar la frecuencia de actualización

Es preciso indicar una dirección de byte en la memoria M para reservar ocho marcas (bits) que correspondan a las teclas de función en el TD 200. Las direcciones válidas están comprendidas entre 0 y 15 (para la CPU 212) o entre 0 y 31 (para las CPUs 214, 215 y 216). El Asistente TD 200 escribe el valor en el byte 5 del bloque de parámetros. Para elegir la frecuencia de actualización, utilice el cuadro de lista desplegable que muestra la figura 5-6. El Asistente TD 200 activa entonces los bits correspondientes en el byte 2 del bloque de parámetros.

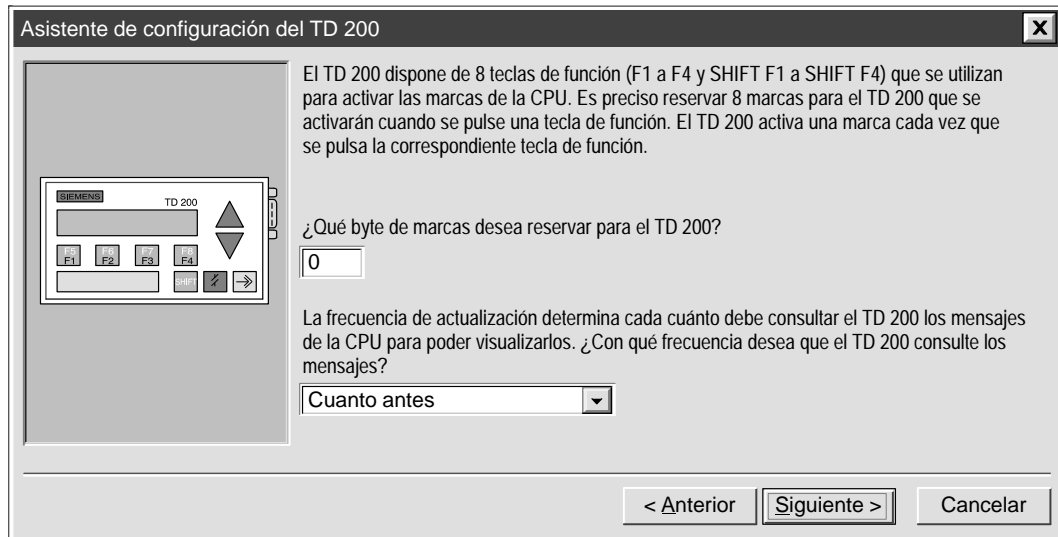


Figura 5-6 Marcas de las teclas de función y frecuencia de actualización del TD 200



#### Precaución

El TD 200 activará una marca cada vez que se pulsa una tecla de función. Si no se tiene previsto utilizar las teclas de función y, por tanto, no se les asigna una dirección en el área de marcas, el TD 200 utiliza el byte M0 como ajuste predeterminado para dichas teclas. Si el programa usa marcas en M0 y un usuario pulsa alguna tecla de función, el TD 200 activa la marca correspondiente en M0, sobrescribiendo el valor asignado a esa marca en el programa de usuario.

Si se efectúan cambios accidentales en las marcas, puede suceder que el programa se ejecute de forma inesperada. El funcionamiento impredecible del sistema de automatización puede causar la muerte o heridas graves al personal y/o daños materiales.

Siempre es necesario reservar una dirección en el área de marcas, aun cuando el programa de usuario no utilice las teclas de función.

### Seleccionar el tamaño y la cantidad de mensajes

Utilice los botones de opción para seleccionar el tamaño de los mensajes (bit 0 del byte 3 en el bloque de parámetros). Introduzca en el cuadro de texto un número comprendido entre 1 y 80 para indicar la cantidad de mensajes que desea crear. El valor correspondiente se escribe en el byte 4 del bloque de parámetros (v. fig. 5-7).

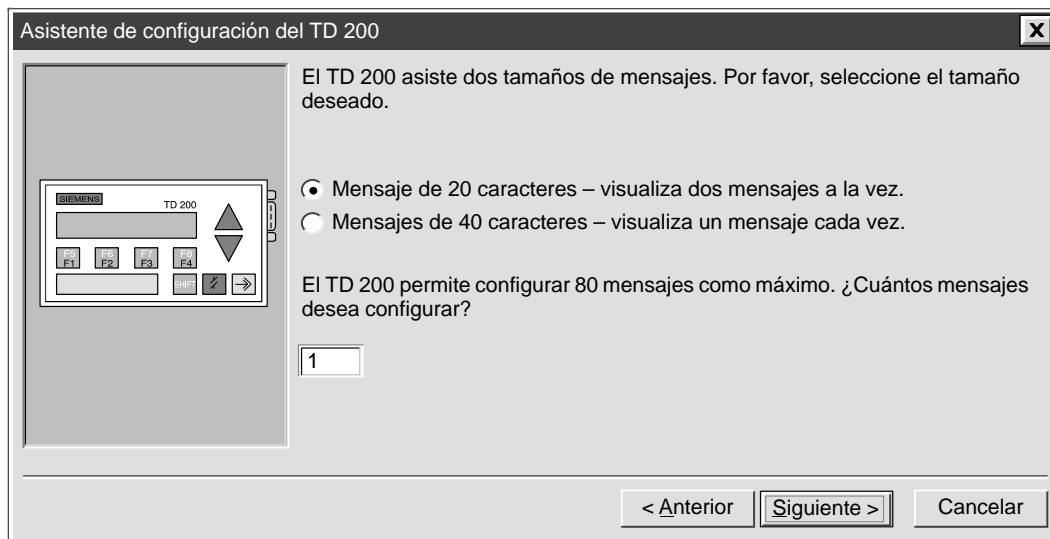


Figura 5-7 Tamaño y cantidad de mensajes del TD 200

### Indicar las direcciones del bloque de parámetros, de las marcas de habilitación de mensajes y de las informaciones sobre los mensajes

En el cuadro de diálogo que muestra la figura 5-8 se pueden indicar las direcciones del bloque de parámetros, de las marcas de habilitación de mensajes y de las informaciones sobre los mensajes.

- El TD 200 busca siempre un identificador del bloque de parámetros en el offset configurado en la CPU. Utilice el primer cuadro de texto para indicar la dirección del bloque de parámetros (sólo si desea cambiar la dirección estándar). El valor (TD) se escribe en los bytes 0 y 1 del bloque de parámetros.
- Indique luego una dirección para las marcas de habilitación de mensajes en la memoria V. Este valor se escribe en los bytes 8 y 9 del bloque de parámetros.
- Por último, indique una dirección inicial en la memoria V donde se deban depositar los mensajes en bytes consecutivos. (32 es sólo un valor estándar). La dirección indicada se escribirá en los bytes 6 y 7 del bloque de parámetros. En el cuadro de diálogo que muestra la figura 5-8 se indican los bytes necesarios conforme a la cantidad de mensajes determinada antes. Recuerde que cada mensaje de 20 caracteres utiliza 20 bytes consecutivos de memoria V, en tanto que cada mensaje de 40 caracteres usa 40 bytes consecutivos.

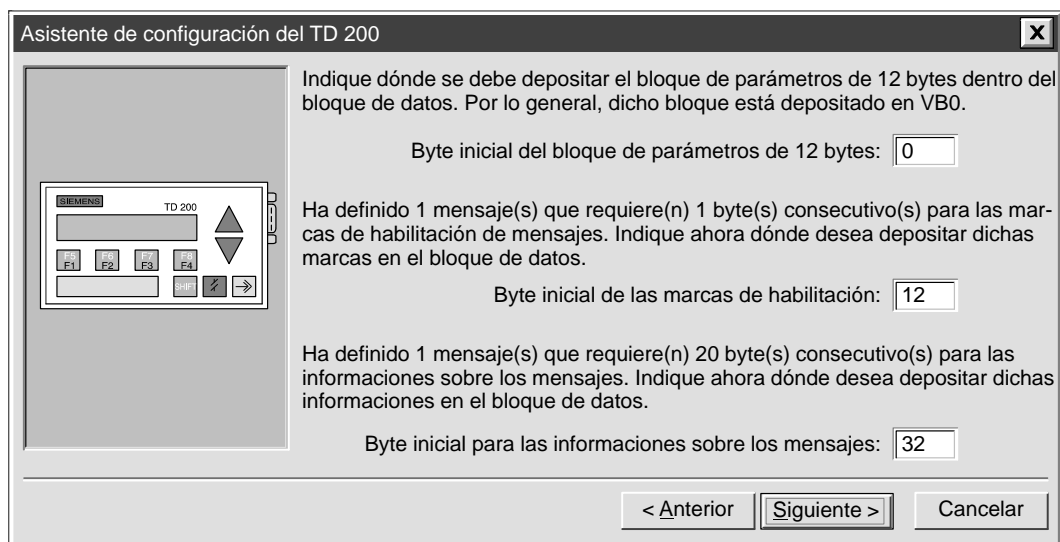


Figura 5-8 Direcciones del bloque de parámetros, de las marcas de habilitación y de las informaciones sobre los mensajes del TD 200

### Crear mensajes para el TD 200

El cuadro de diálogo que muestra la figura 5-9 permite crear los mensajes de 20 ó 40 caracteres indicados en la figura 5-8. Como muestra la figura 5-8, los mensajes se almacenan en la memoria V comenzando en la dirección indicada en la figura 5-9.

Introduzca el mensaje (un carácter en cada casilla). Si ha indicado que desea configurar más de un mensaje, haga clic en el botón "Mensaje siguiente >" para introducir el texto de los demás mensajes.

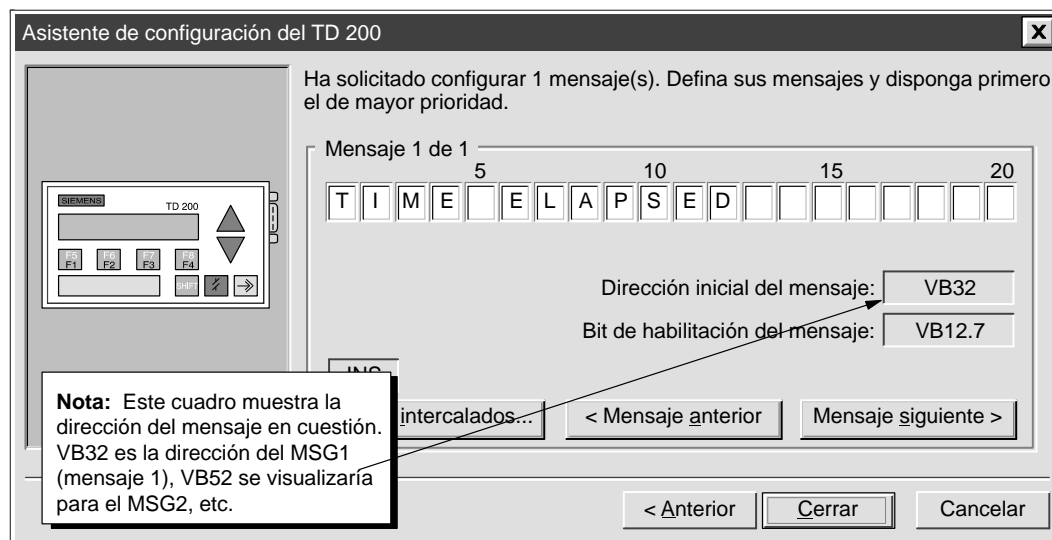


Figura 5-9 Cuadro de diálogo para configurar los mensajes del TD 200

### Intercalar valores de datos en un mensaje de texto

Un valor de datos se puede intercalar en un mensaje para visualizarlo en el TD 200. Por ejemplo, es posible crear un mensaje que visualice un valor de tiempo transcurrido cuando lo lea la CPU. A fin de visualizar un valor de datos es preciso reservar en el mensaje suficiente espacio.

Con objeto de insertar un comodín para una valor de datos variable, sitúe el cursor en el dígito inicial y haga clic en el botón "Datos intercalados..." que aparece en el borde inferior izquierdo del cuadro de diálogo. Aparecerá un cuadro de diálogo donde se debe definir el formato del valor de datos, así como elegir otras opciones, p.ej. si el mensaje se debe confirmar o no, si el valor de datos se puede editar y si se requiere una contraseña para la edición.

### Introducir caracteres internacionales y especiales

Al introducir algunos caracteres internacionales y especiales con el Asistente de configuración del TD 200, es posible que no aparezcan correctamente en el display del TD 200. Si los caracteres no se visualizan correctamente, utilice las combinaciones de números con ALT que figuran en la tabla 5-1 para introducir los caracteres en el Asistente TD 200.

Tabla 5-1 Combinaciones con ALT para introducir caracteres internacionales y especiales

Carácter	Combinación con ALT	Carácter	Combinación con ALT
ü	ALT+0129	ñ	ALT+0164
ä	ALT+0132	Ω	ALT+0234
æ	ALT+0145	Σ	ALT+0228
Æ	ALT+0146	Π	ALT+0227
å	ALT+0134	¥	ALT+0157
ö	ALT+0148	←	ALT+0195 (flecha izquierda ←)
Å	ALT+0143	→	ALT+0180 (flecha derecha →)
°	ALT+0248		ALT+0200 (una barra)
α	ALT+0224		ALT+0201 (dos barras)
β	ALT+0225		ALT+0202 (tres barras)
ε	ALT+0238		ALT+0203 (cuatro barras)
μ	ALT+0230		ALT+0204 (cinco barras)
σ	ALT+0229	↑	ALT+0194 (flecha arriba)
ç	ALT+0155		

### Formatear valores de datos intercalados

La figura 5-10 muestra el cuadro de diálogo donde se definen los parámetros del valor a visualizar. El formato y las opciones que se indiquen se escribirán en una palabra de formato (dos bytes) que precede a cada valor intercalado. Elija el tamaño, el formato de visualización, el número de dígitos a la derecha del punto decimal y las demás opciones para la variable a intercalar.

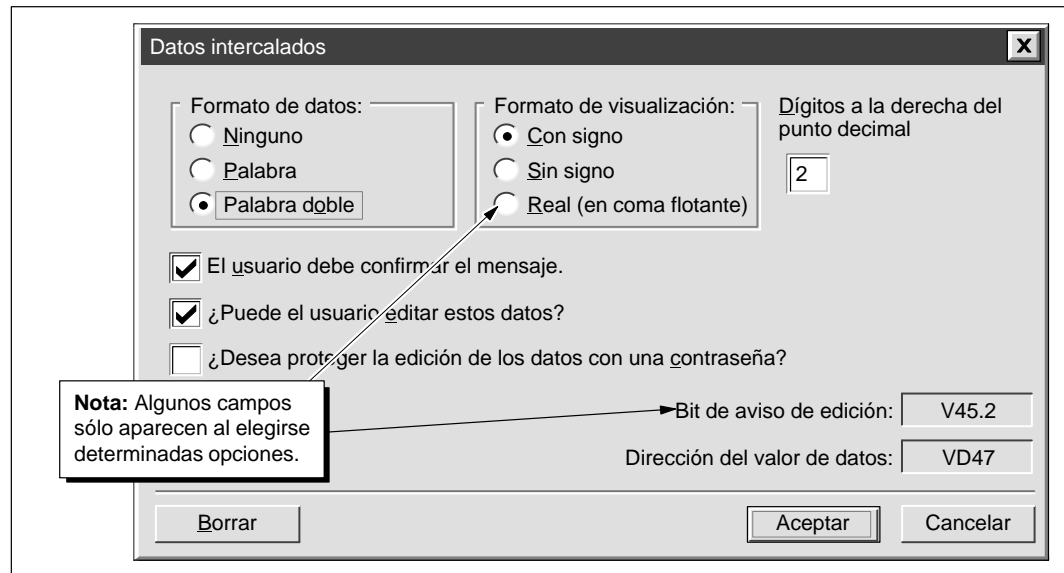


Figura 5-10 Cuadro de diálogo para intercalar datos en mensajes del TD 200

La figura 5-11 muestra el cuadro de diálogo del mensaje tras seleccionarse los parámetros para un valor de datos intercalado. Los espacios grises son comodines para el valor de datos. Si ha indicado que el usuario debe confirmar todos los mensajes, el bit de aviso de confirmación se visualizará en el cuadro de diálogo.

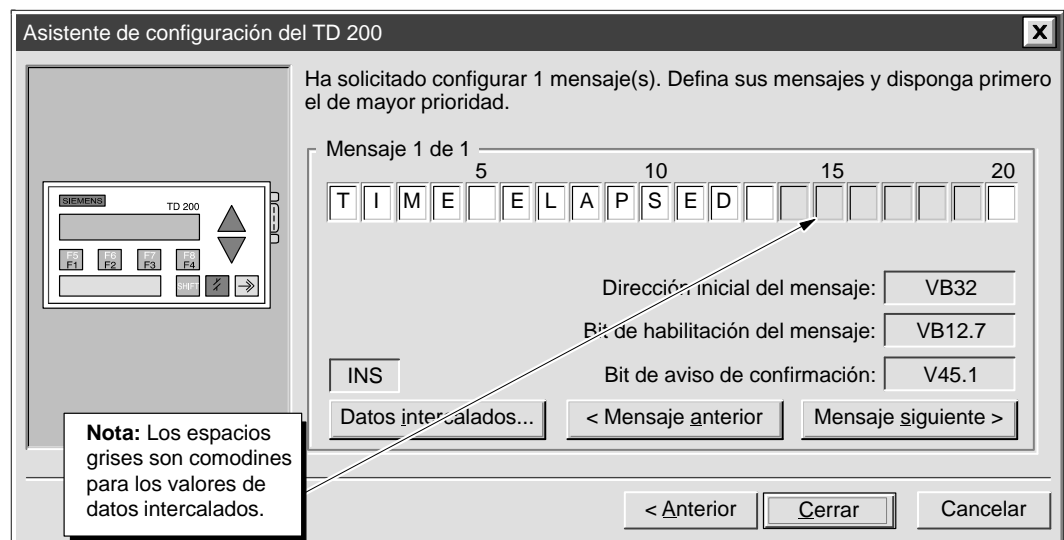


Figura 5-11 Comodines para los valores de datos intercalados en un mensaje del TD 200

### Finalizar la configuración del bloque de parámetros del TD 200

Haga clic en el botón “Mensaje siguiente >” para introducir el texto de los demás mensajes. Una vez introducidos todos los mensajes del TD 200, haga clic en el botón “Cerrar” para guardar el bloque de parámetros y los mensajes configurados en el bloque de datos.

El bloque de parámetros configurado con el Asistente del TD 200 se puede visualizar abriendo el editor de bloques de datos. La figura 5-12 muestra un ejemplo de un bloque de parámetros para un mensaje de 40 caracteres como se visualiza en el editor de bloques de datos.

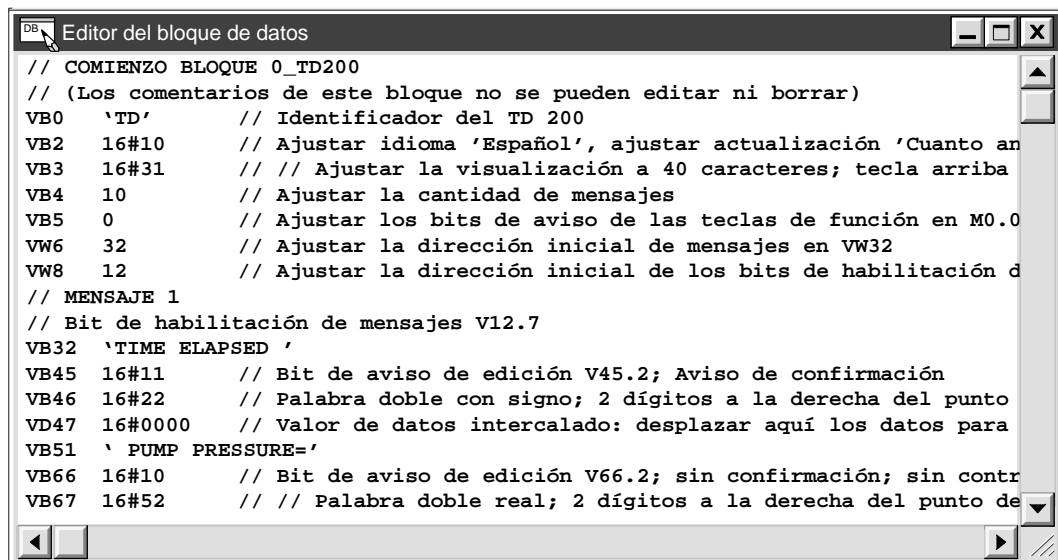


Figura 5-12 Editor de bloques de datos con un ejemplo de un bloque de parámetros del TD 200

## 5.2 Utilizar el Asistente de operaciones S7-200

STEP 7-Micro/WIN incluye un Asistente de operaciones S7-200 que permite configurar las siguientes operaciones complejas de forma rápida y sencilla:

- Configurar operaciones PID.
- Configurar operaciones Leer de la red y Escribir en la red.
- Configurar un algoritmo de muestreo y crear un promedio para filtrar las entradas analógicas.
- Configurar las operaciones de los contadores rápidos.

En el apartado 5.3 se muestra un ejemplo del Asistente para filtrar entradas analógicas.

### Seleccionar el Asistente de operaciones S7-200

Para seleccionar el Asistente de operaciones S7-200:

1. Elija el comando de menú **Herramientas ► Asistente de operaciones...** como muestra la figura 5-13.
2. Haga clic en la operación que desea configurar.
3. Haga clic en el botón "Siguiente >". Si el programa no se ha compilado desde la última vez que se editó, es preciso hacerlo ahora. Puesto que la compilación puede tomar bastante tiempo (si el programa es muy grande), se le pregunta si desea continuar. Aparecerá el mensaje "Compilación necesaria. El programa se debe compilar para poder continuar. ¿Compilar ahora?". En caso afirmativo, haga clic en "Aceptar", o bien en "Cancelar" para salir del Asistente sin compilar el programa.
4. Tras elegir la operación que desea configurar y una vez compilado el programa, aparecerán las pantallas correspondientes.

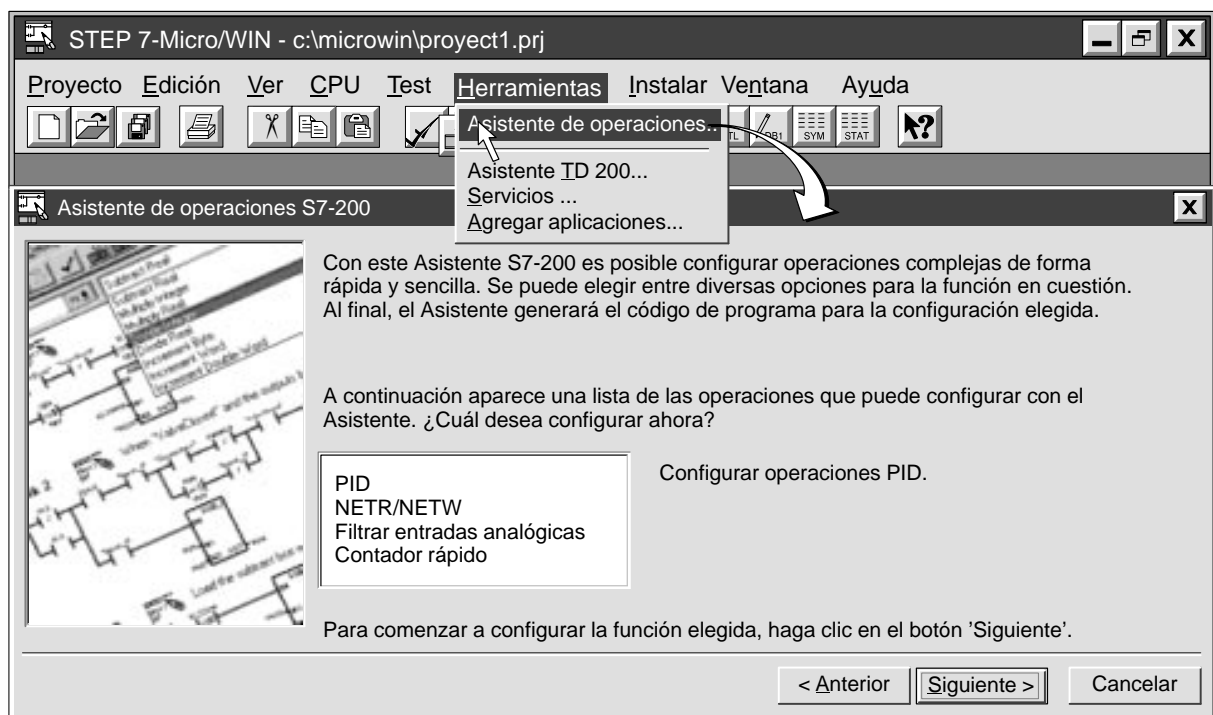


Figura 5-13 Utilizar el Asistente de operaciones S7-200

Tras responder todas las preguntas relativas a la operación en cuestión, aparece la pantalla final del Asistente S7-200 como muestra la figura 5-14. Allí se explica cuáles segmentos del programa se generarán para la configuración elegida. También se ofrece la posibilidad de indicar dónde se debe situar el código dentro del programa principal.

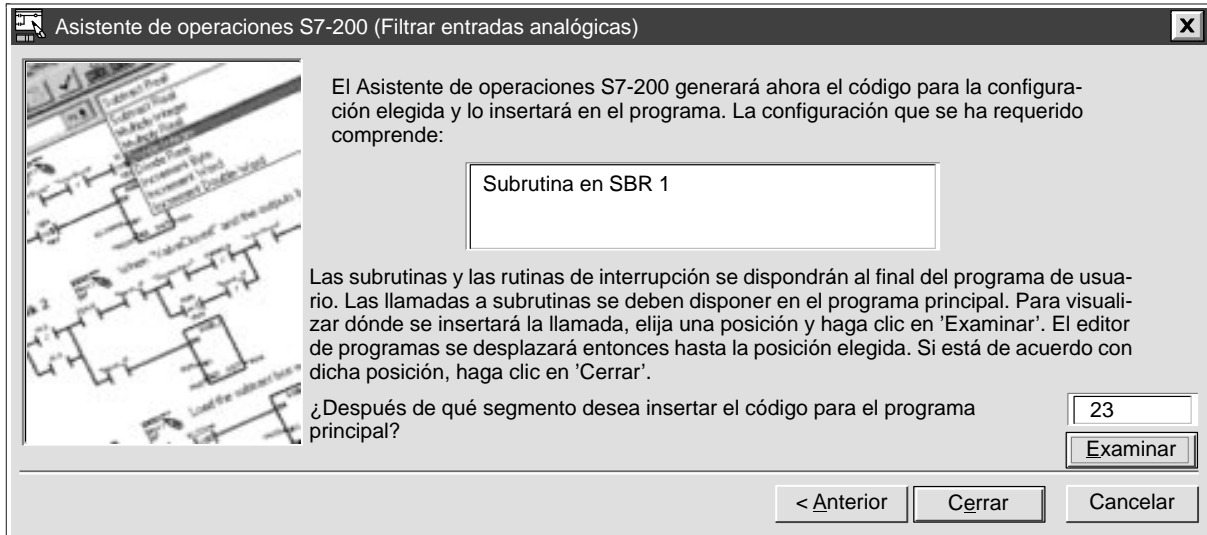


Figura 5-14 Segmentos del programa generados por el Asistente de operaciones S7-200

### 5.3 Utilizar el Asistente para filtrar entradas analógicas

El Asistente para filtrar entradas analógicas se puede utilizar para agregar al programa de usuario una rutina con objeto de crear un promedio. El módulo S7-200 de entradas analógicas es un módulo rápido. Por tanto, 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. Un valor promedio calculado con una cantidad elevada de muestreos puede estabilizar el resultado, ralentizando simultáneamente su respuesta a cambios en la señal de entrada.

#### Filtraje básico

Para poder efectuar un filtraje básico, se deben responder tres preguntas:

1. ¿Qué entrada analógica desea filtrar? (AIW0, AIW2, AIW4,...).
2. ¿En qué dirección se debe escribir el valor filtrado? (VWx, AQWx, ...).
3. ¿En qué dirección desea situar el área de trabajo para los cálculos? El código de filtraje necesita 12 bytes de área en la memoria. (VBx, ..).

#### Opciones adicionales

Para obtener más información acerca de la entrada analógica a filtrar se pueden configurar las siguientes opciones adicionales:

- Cantidad de muestreos
- Comprobación de errores

#### Indicar la entrada y la salida

Como muestra la figura 5-15, indique cuál AIW debe ser la entrada y dónde se debe escribir la salida. Para la salida puede introducir una dirección o un nombre simbólico.

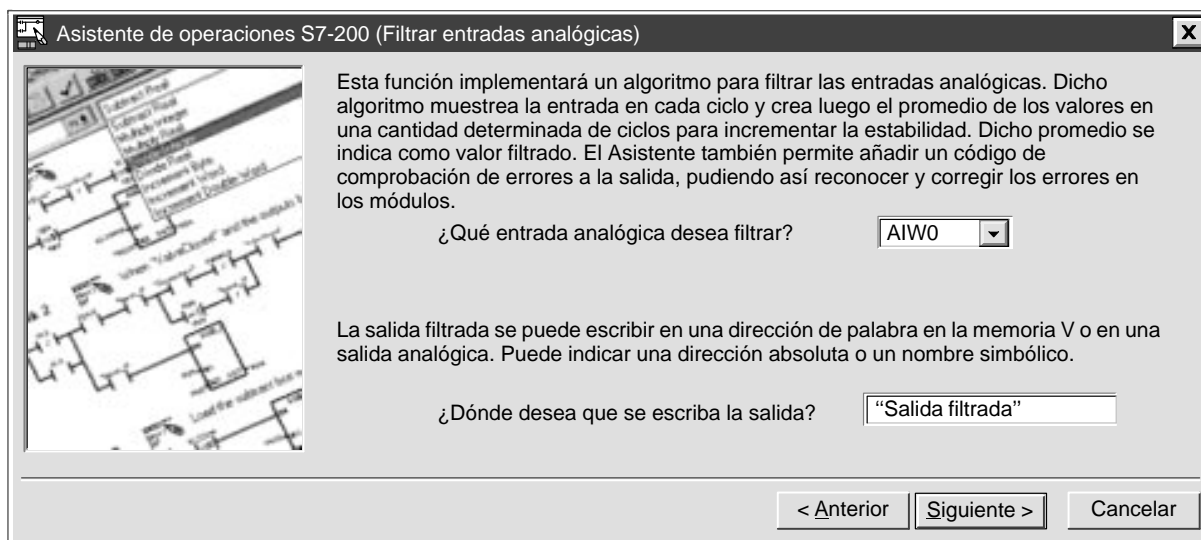


Figura 5-15 Indicar la entrada y la salida en el Asistente para filtrar entradas analógicas

**Elegir la dirección para el área de trabajo de 12 bytes**

Como muestra la figura 5-16, elija dónde debe comenzar el área de trabajo de 12 bytes. También es preciso seleccionar la subrutina a utilizar para generar el código y la cantidad de muestreos.

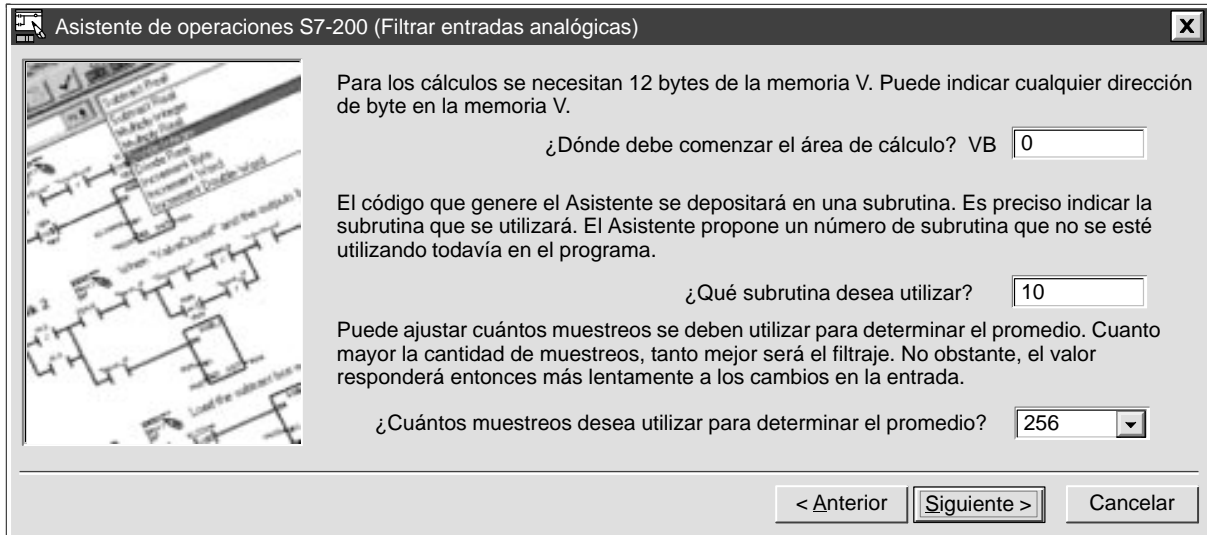


Figura 5-16 Elegir la dirección para el área de trabajo de 12 bytes

## Comprobar errores en los módulos

Puede incluir en la configuración un código de comprobación de errores en los módulos. A tal efecto, debe indicar la posición del módulo analógico utilizado para generar el código que comprobará las direcciones correctas de las marcas especiales (SM). También debe indicar la marca que deberá contener el estado de error del módulo. Si se presenta un error en el módulo, dicha marca se activará. Si desea forzar un valor específico en caso de que se presente un error en el módulo, deberá indicar dicho valor (v. fig. 5-17).

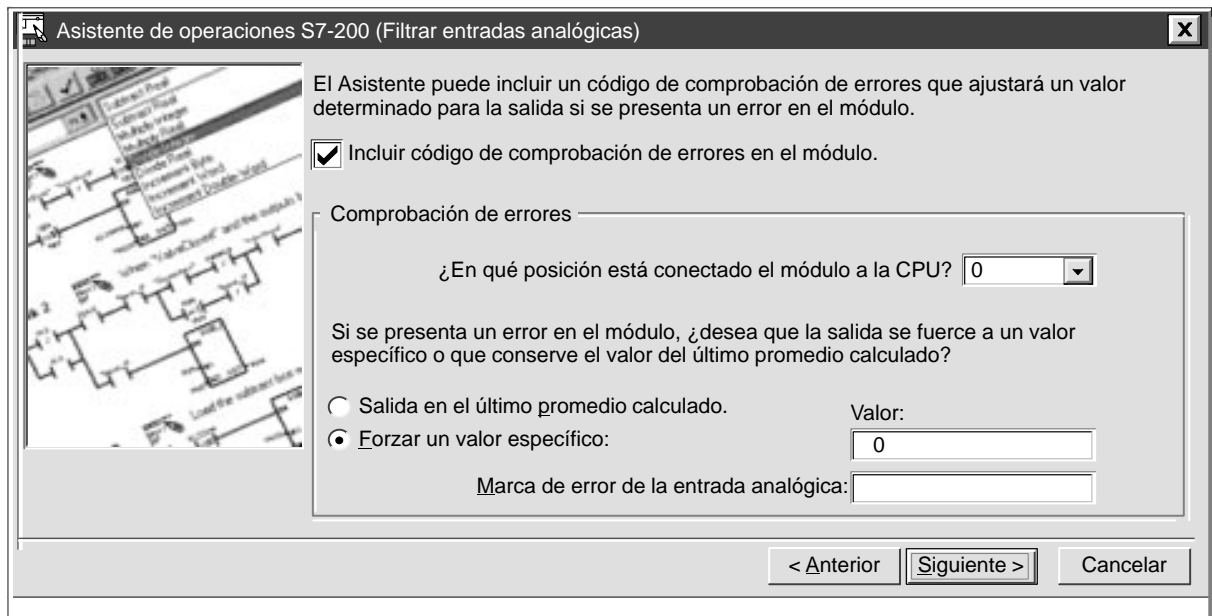


Figura 5-17 Filtrar entradas analógicas: forzar un valor específico si se presenta un error en el módulo

Alternativamente, puede elegir que al presentarse un error en el módulo se adopte en la salida el último promedio calculado (v. fig. 5-18).

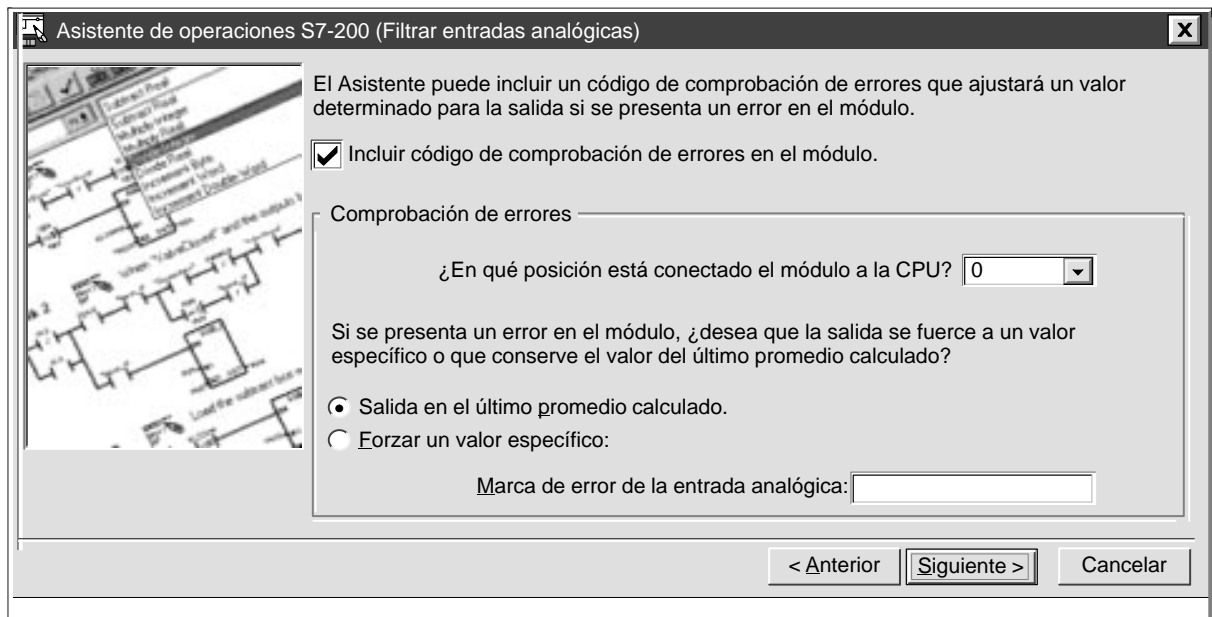


Figura 5-18 Filtrar entradas analógicas: adoptar el último promedio calculado si se presenta un error en el módulo

## 5.4 Utilizar las referencias cruzadas

Utilice la tabla de referencias cruzadas para generar una lista de las direcciones usadas en el programa. Con las referencias cruzadas puede observar las direcciones mientras introduce el programa de usuario. Al desear visualizar las referencias cruzadas, se compila el programa y se genera la tabla en cuestión.

En ella figuran los nombres de los elementos, los números de segmento y las operaciones correspondientes (v. fig. 5-19). Las direcciones indirectas se muestran en la tabla de referencias cruzadas con los símbolos (\*) o (&).

Para generar una tabla de referencias cruzadas:

1. Elija el comando de menú **Ver ► Referencias cruzadas**.
2. El programa se compila y se genera la tabla de referencias cruzadas.
3. La tabla de referencias cruzadas puede permanecer abierta mientras se introduce el programa. Si modifica el programa y hace luego clic en dicha tabla, deberá actualizarla eligiendo la opción "Actualizar" que aparece en el borde superior de la ventana de las referencias cruzadas.
4. Para visualizar un elemento en su programa, haga doble clic en el mismo en la tabla de referencias cruzadas. Ese elemento se destacará entonces en el editor de programas.

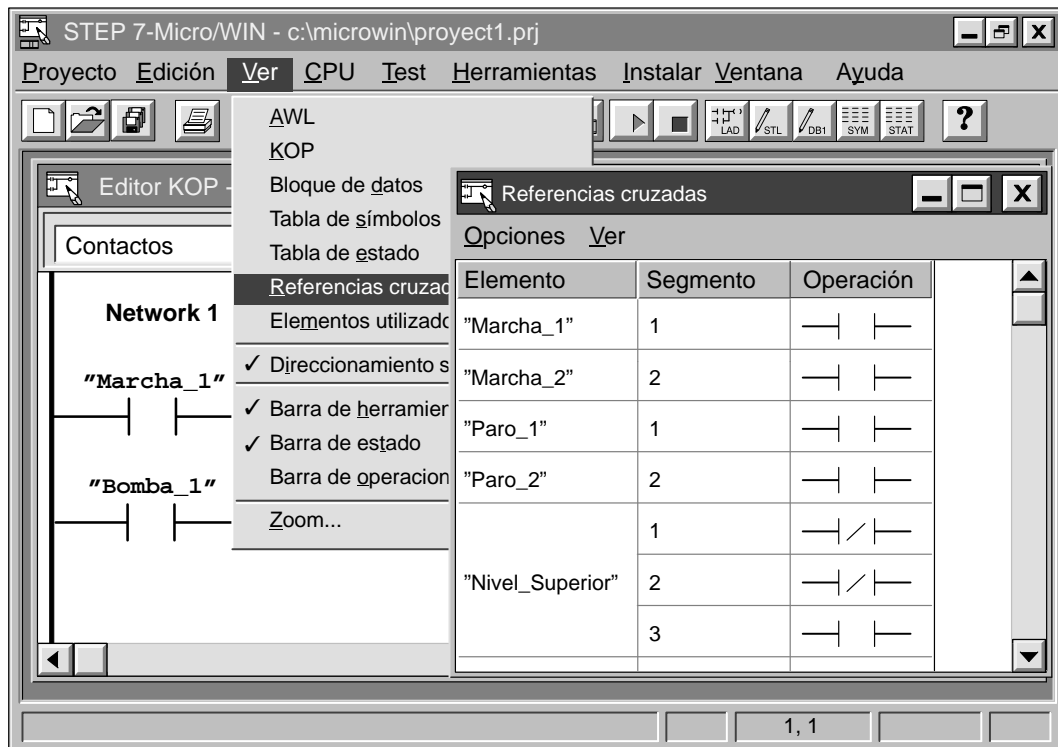


Figura 5-19 Ver la tabla de referencias cruzadas

## 5.5 Usar la lista de elementos utilizados

La lista de elementos utilizados se puede usar para mostrar las direcciones y los márgenes asignados en el programa. En dicha lista, las informaciones figuran de forma más compacta que en la tabla de referencias cruzadas. El margen indicado comienza en la primera dirección utilizada y abarca hasta la última. Las direcciones no utilizadas se muestran como líneas en blanco (v. fig. 5-20).

Hay dos formas de visualizar los elementos utilizados:

- El formato de bit muestra el uso de I, Q, M y S
- El formato de byte muestra el uso de V, AIW, AQW, MB, SMB, T, C y HSC

Consideraciones:

- Al visualizarse en formato de byte, la dirección de una palabra doble se muestra con cuatro letras "D" consecutivas. Si no aparecen cuatro letras "D" consecutivas, es posible que dicha dirección se haya utilizado dos veces o que se haya intentado programarlo así. (Una palabra se muestra con dos letras "W" consecutivas; un byte es una "B" y un bit, una "b").
- Los elementos utilizados que estén marcados con guiones (—) corresponden a referencias que indican un margen resultante de direcciones utilizadas por una operación sin que aparezcan allí explícitamente. Por ejemplo, la operación Leer de la red (NETR) utiliza una tabla de 8 bytes en la memoria V. No obstante, el primer byte es la única referencia explícita.

Para generar una tabla de elementos utilizados, elija el comando de menú **Ver ► Elementos utilizados**. El programa se compila y aparece la tabla de elementos utilizados (v. fig. 5-20). La tabla de elementos utilizados puede permanecer abierta mientras se introduce el programa. Si modifica el programa y hace luego clic en dicha tabla, deberá actualizarla eligiendo la opción "Actualizar" que aparece en el borde superior de la ventana de elementos utilizados.

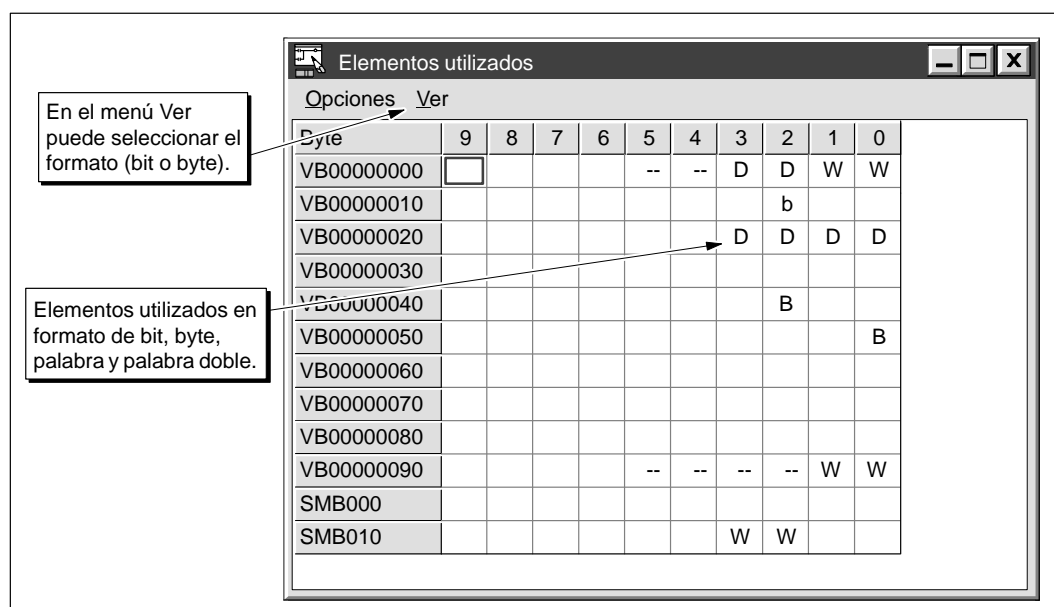


Figura 5-20 Ver la tabla de elementos utilizados

## 5.6 Utilizar la función para buscar y reemplazar

Puede utilizar la función "Buscar" para localizar un parámetro determinado y "Reemplazar" para sustituirlo con otro (v. fig. 5-21).

### Utilizar la función "Buscar" para localizar un parámetro

Para buscar un parámetro determinado:

1. Elija el comando de menú **Edición ► Buscar....** La figura 5-21 muestra el cuadro de diálogo "Buscar".
2. Elija los parámetros que desea buscar.
3. Elija la dirección en la que desea buscar.
4. Pulse el botón "Buscar siguiente" para iniciar la búsqueda.

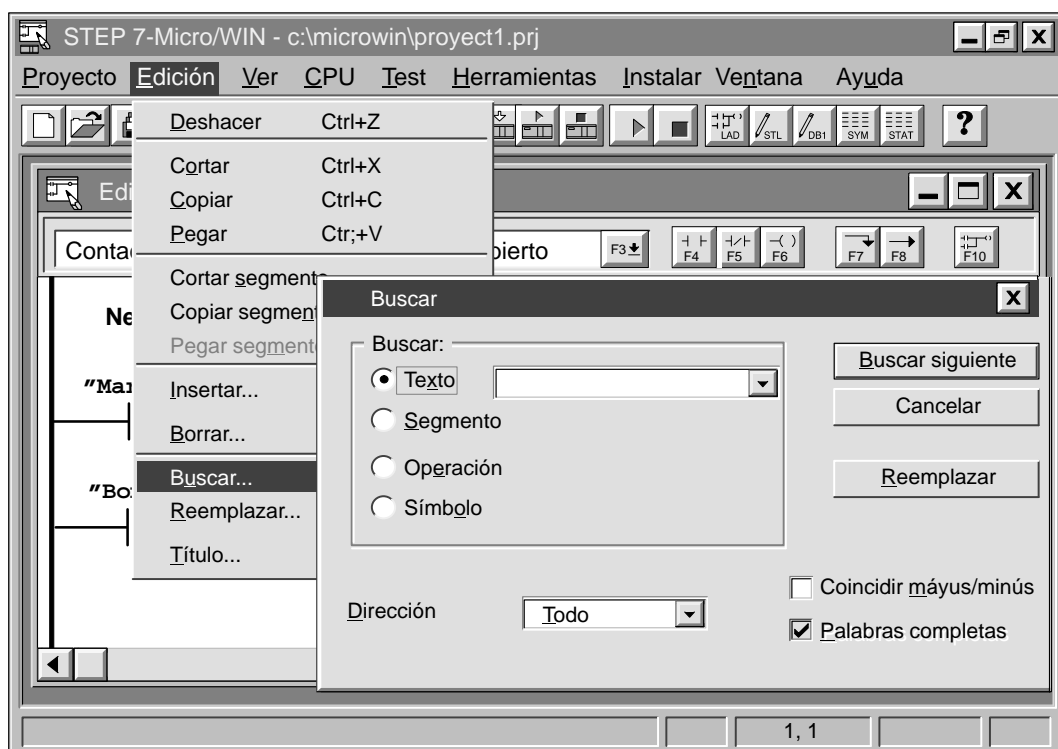


Figura 5-21 Cuadro de diálogo "Buscar"

## Reemplazar un parámetro

Para reemplazar un parámetro determinado:

1. Elija el comando de menú **Edición ► Reemplazar..** La figura 5-22 muestra el cuadro de diálogo "Reemplazar".
2. Defina el parámetro que desea reemplazar.
3. Haga clic en el botón "Reemplazar" para sustituir una ocurrencia. Al pulsar dicho botón, se localiza la primera ocurrencia. Haga nuevamente clic en el botón "Reemplazar" para sustituir dicha ocurrencia y buscar la siguiente.
4. El botón "Reemplazar todo" ignora la selección ("Todo" o "Segmento") indicada y sustituye todas las ocurrencias.

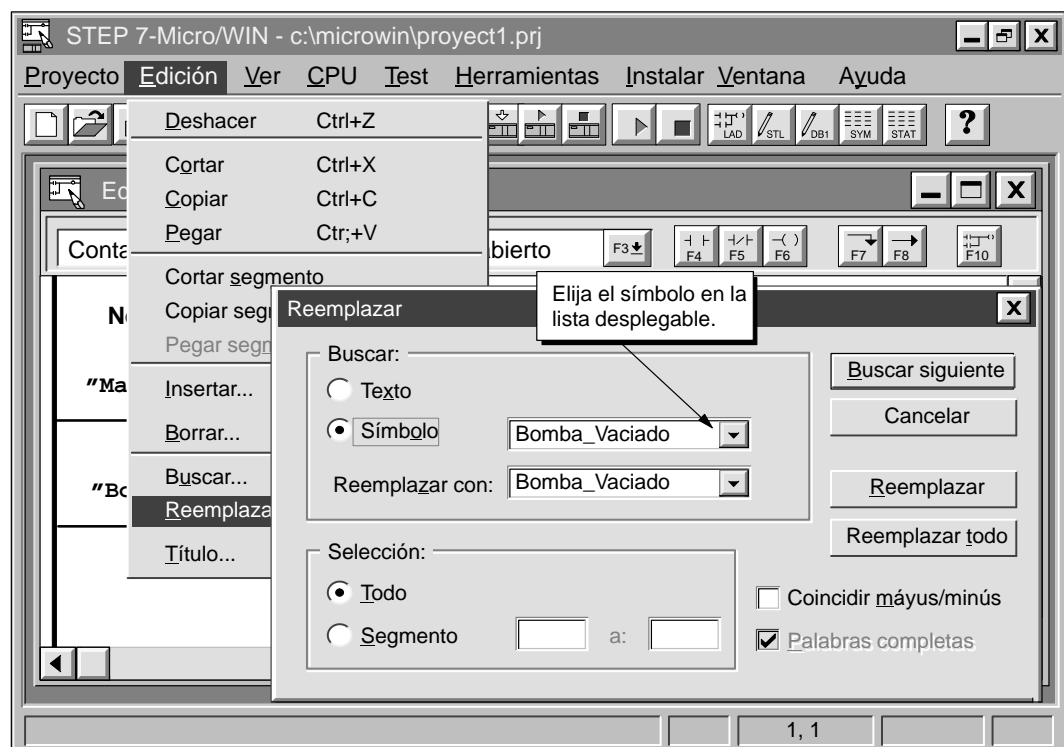


Figura 5-22 Cuadros de diálogo "Reemplazar"

## 5.7 Documentar el programa

El programa KOP se puede documentar utilizando un título para el mismo, así como títulos y comentarios de segmento. El programa AWL se puede documentar con comentarios descriptivos.

### Reglas para documentar programas KOP

El título del programa KOP se utiliza para describir brevemente el proyecto. Para editar el título, elija el comando de menú **Edición ► Título....** Introduzca el título del programa y haga clic en el botón "Aceptar".

El título de un segmento KOP permite resumir la función de éste. Dicho título (que comprende una línea) siempre está visible en la vista KOP. Para editar el título de un segmento, haga doble clic en el campo "Título de segmento" en el programa. Introduzca el resumen en el campo "Título" del editor de títulos de segmento y comentarios KOP. Haga clic en el botón "Aceptar".

Los comentarios de segmento KOP permiten describir más detalladamente la función del segmento en cuestión. Para introducir comentarios de segmento, haga doble clic en el campo "Título de segmento" en el programa. Introduzca sus comentarios en el campo "Comentario" y haga clic en el botón "Aceptar". Los comentarios de segmento no se ven en la pantalla del programa, pero se pueden visualizar haciendo clic en el campo correspondiente al título del segmento.

Para imprimir los comentarios KOP, elija el comando de menú **Proyecto ► Imprimir....** Haga clic en el botón "Preparar página...", seleccione la opción "Imprimir comentarios de segmento" y pulse el botón "Aceptar".

### Reglas para documentar programas AWL

En un programa AWL, cualquier texto de una línea que vaya precedido por dos barras inclinadas (//) se considera un comentario AWL. Los comentarios se pueden utilizar al comienzo del programa para describir la función general del mismo. Dichos comentarios se pueden utilizar en una línea por separado, o bien en la misma línea que la operación con objeto de documentar los detalles del programa (v. fig. 5-23).

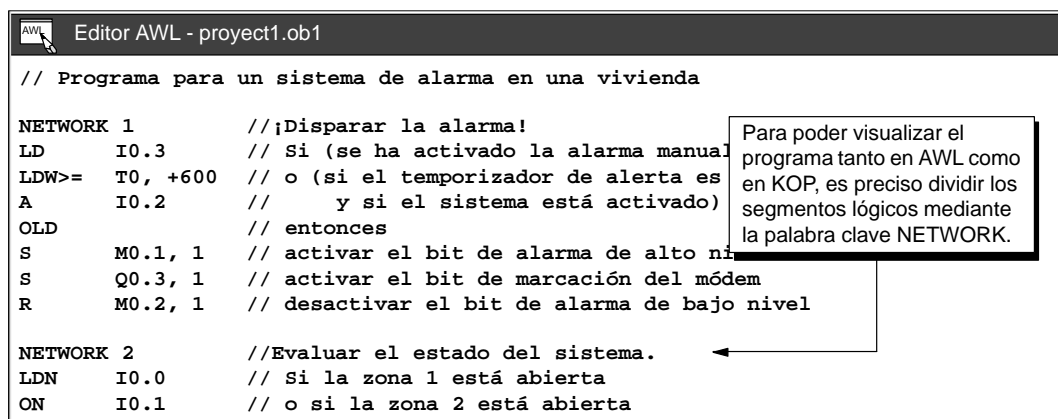


Figura 5-23 Documentar el programa AWL

### Visualizar programas AWL en KOP

Para visualizar un programa AWL en KOP es preciso tener en cuenta las siguientes convenciones al introducir el programa AWL (v. fig. 5-23).

- Los segmentos lógicos del programa AWL se deberán dividir en segmentos independientes, introduciendo a tal efecto la palabra clave NETWORK. Para que el programa se puede visualizar en KOP, entre las palabras claves NETWORK se deberá prever una distancia apropiada. Los números de los segmentos se generan automáticamente al compilar o cargar el programa de usuario.
- Los comentarios AWL que aparezcan antes de la primera palabra clave NETWORK constituirán el título del programa en KOP.
- Los comentarios AWL que aparezcan junto a la palabra clave NETWORK se convertirán en KOP en el título del segmento.
- Los comentarios AWL que aparezcan entre la línea NETWORK y la primera operación de dicho segmento se convertirán en comentarios de segmento en KOP. Ejemplo:

```
NETWORK                      // TÍTULO DEL SEGMENTO
//COMENTARIO DEL SEGMENTO, LÍNEA 1
//COMENTARIO DEL SEGMENTO, LÍNEA 2
LD I0.0
```

## 5.8 Imprimir el programa

Con la función "Imprimir" se puede imprimir el programa completo o partes del mismo.

- Elija el comando de menú **Proyecto ► Imprimir...** para imprimir el programa. Elija lo que desee imprimir y haga clic en el botón "Aceptar" (v. fig. 5-24).
- Pulse el botón "Preparar página" para seleccionar otras opciones adicionales de impresión, a saber: márgenes, direcciones absolutas o nombres simbólicos, comentarios de segmento y encabezados/pies de página.
- Haga clic en el botón "Instalar" para seleccionar la impresora y las opciones relativas al papel.

Para imprimir su programa:

1. Elija el comando de menú **Proyecto ► Imprimir....** Aparecerá el cuadro de diálogo "Imprimir" que muestra la figura 5-24.
2. En el cuadro "Imprimir", elija las opciones deseadas.
3. En el cuadro "Imprimir segmento KOP", elija la selección deseada.
4. Si desea cambiar la instalación de la impresora, puede seleccionar "Preparar página" o "Instalar".
5. Haga clic en el botón "Aceptar".

### Nota

Si desea imprimir la tabla de referencias cruzadas y/o de elementos utilizados, es posible que deba compilar antes el programa. El tiempo necesario para la compilación depende del tamaño del programa.

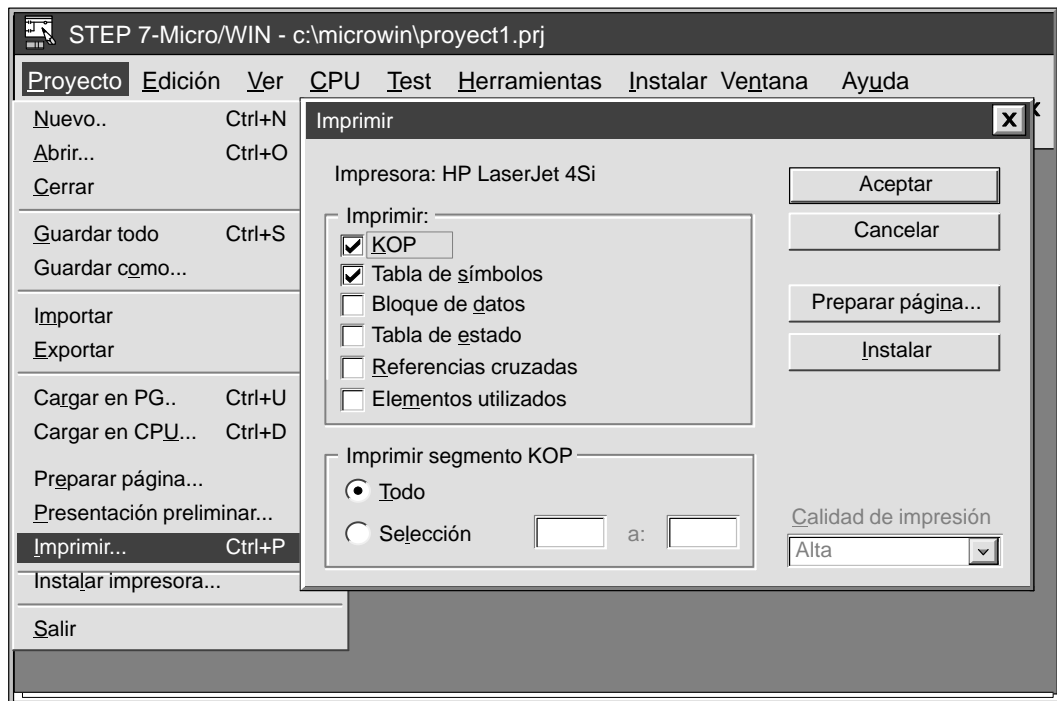


Figura 5-24 Cuadro de diálogo "Imprimir"



# Nociones básicas para programar una CPU S7-200

# 6

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
6.1	Crear una solución de automatización con un Micro-PLC	6-2
6.2	Programas S7-200	6-4
6.3	Lenguajes de programación para las CPUs S7-200	6-5
6.4	Elementos básicos para estructurar un programa	6-8
6.5	El ciclo de la CPU	6-10
6.6	Ajustar el modo de operación de la CPU	6-13
6.7	Determinar una contraseña	6-14
6.8	Comprobar y observar el programa	6-16
6.9	Eliminar errores de las CPUs S7-200	6-19

## 6.1 Crear una solución de automatización con un Micro-PLC

Hay 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 sus propias experiencias. La figura 6-1 muestra los pasos básicos al respecto.

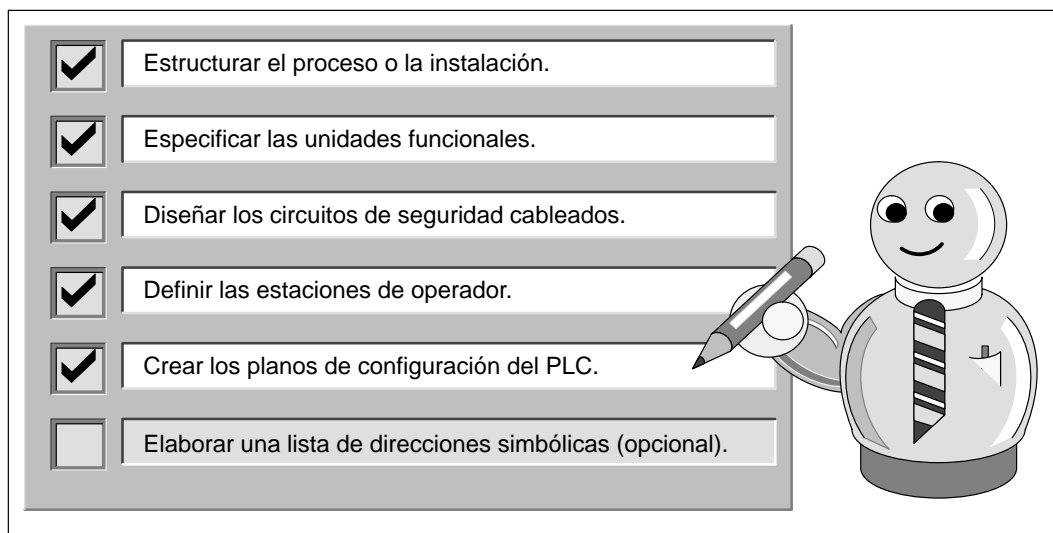


Figura 6-1 Pasos básicos para crear una solución de automatización

### Estructurar el proceso o la instalación

Estructure el proceso o la instalación en secciones independientes entre sí. Dichas secciones determinarán los límites entre los diversos sistemas de automatización e influirán en las descripciones de las áreas de funciones y en la asignación de recursos.

### Especificar las unidades funcionales

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

- Entradas y salidas (E/S)
- Descripción del funcionamiento
- 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 producirse un arranque inesperado o un cambio de funcionamiento de las máquinas que controlan. En tal caso, se pueden causar heridas graves o deteriorar objetos. 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 garanticen el funcionamiento seguro y determine cómo reconocer dichas condiciones, independientemente de la CPU.
- Defina cómo la CPU y los módulos de ampliación deberán influir 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 el 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 del PLC

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.

## 6.2 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 6-2 muestra la conexión de un esquema de circuitos simple con la CPU S7-200. En este ejemplo, el estado del interruptor de la estación de operador para abrir la válvula de vaciado se suma a los estados de otras entradas. Los cálculos de los mismos determinan entonces el estado de la salida para la electroválvula que cierra la válvula de vaciado.

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

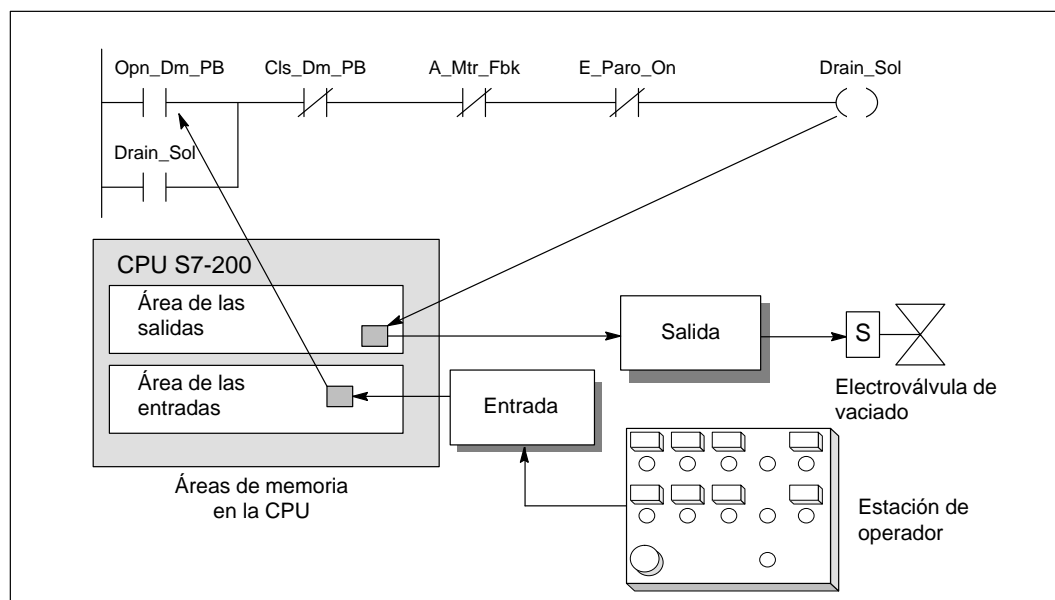


Figura 6-2 Referencias a las entradas y salidas en el programa

### Acceder a los datos en las áreas de memoria

La CPU almacena el estado de las entradas y salidas en determinadas áreas de la memoria. La figura 6-2 muestra el flujo simplificado de la información: entrada → área de memoria → programa → área de memoria → salida. A cada área de la memoria se ha asignado un identificador nemotécnico (p.ej. "I" para las entradas y "Q" para las salidas) que se utiliza para acceder a los datos almacenados allí.

STEP 7-Micro/WIN provee direcciones "absolutas" para todas las áreas de memoria. Para acceder a una dirección determinada es preciso indicar el operando (p.ej. I0.0 que es la primera entrada). STEP 7-Micro/WIN también permite asignar nombres simbólicos a las direcciones absolutas. Una dirección absoluta de un área de memoria incluye no sólo el identificador de área (p.ej. "V"), sino también el tamaño (hasta 4 bytes o 32 bits) de los datos a los que se desea acceder: B (byte), W (palabra = 2 bytes) o D (palabra doble = 4 bytes). La dirección absoluta incluye también un valor numérico: bien sea el número de bytes desde el comienzo del área de memoria (desplazamiento), o bien el número del elemento. (Este valor depende del identificador de área) (v. apt. 7.1).

## 6.3 Lenguajes de programación para las CPUs S7-200

Las CPUs S7-200 (y STEP 7-Micro/WIN) asisten los siguientes lenguajes de programación:

- La lista de instrucciones (AWL) comprende un juego de operaciones nemotécnicas que representan las funciones de la CPU.
- El esquema de contactos (KOP) es un lenguaje de programación gráfico con componentes similares a los elementos de un esquema de circuitos.

STEP 7-Micro/WIN ofrece además dos representaciones nemotécnicas para visualizar las direcciones y las operaciones del programa: internacional y SIMATIC. Tanto la nemotécnica internacional como la de SIMATIC se refieren al mismo juego de operaciones del S7-200. Hay una correspondencia directa entre las dos representaciones, siendo idénticas las funciones de ambas. En el presente manual se utiliza la nemotécnica internacional.

### Elementos básicos de KOP

Al programar con KOP, se crean y se disponen componentes gráficos que conforman un segmento de operaciones lógicas. Como muestra la figura 6-3, se ofrecen los siguientes elementos básicos para crear programas:

- Contactos: un contacto representa un interruptor por el que circula la corriente cuando está cerrado.
- Bobinas: una bobina representa un relé que se excita cuando se le aplica tensión.
- Cuadros: un cuadro representa una función que se ejecuta cuando la corriente circula por él.
- Segmentos: cada uno de estos elementos constituye un circuito completo. La corriente circula desde la barra de alimentación izquierda pasando por los contactos cerrados para excitar las bobinas o cuadros.

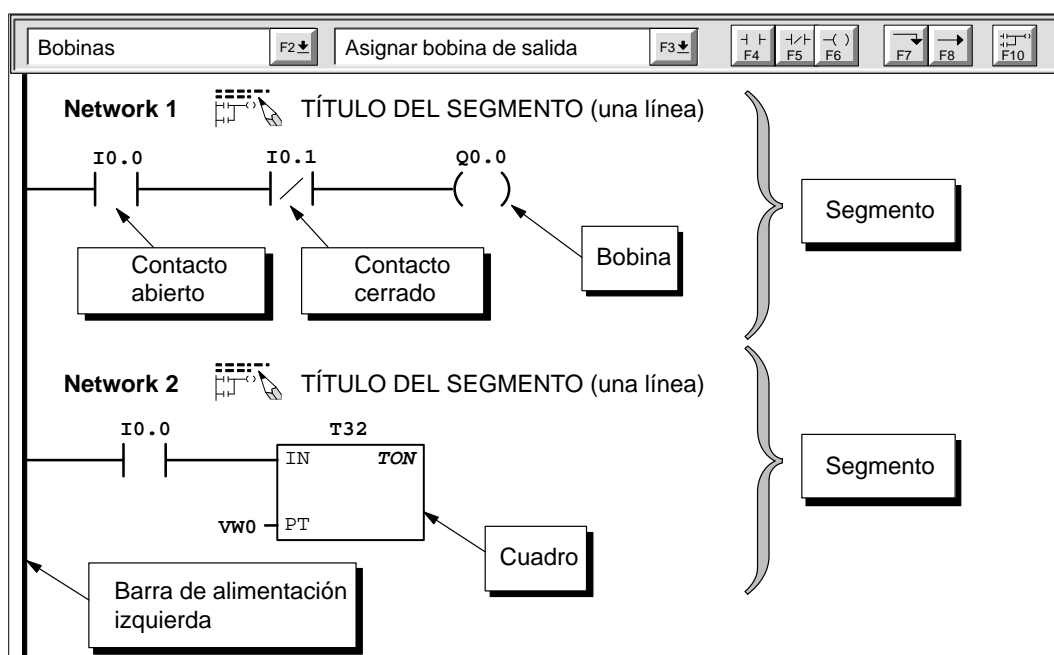


Figura 6-3 Elementos básicos de KOP

## Operaciones de AWL

La lista de instrucciones (AWL) es un lenguaje de programación en el que cada línea del programa contiene una operación que utiliza una abreviatura nemotécnica para representar una función de la CPU. Las operaciones se combinan en un programa, creando así la lógica de control de la aplicación.

La figura 6-4 muestra los elementos básicos de un programa AWL.

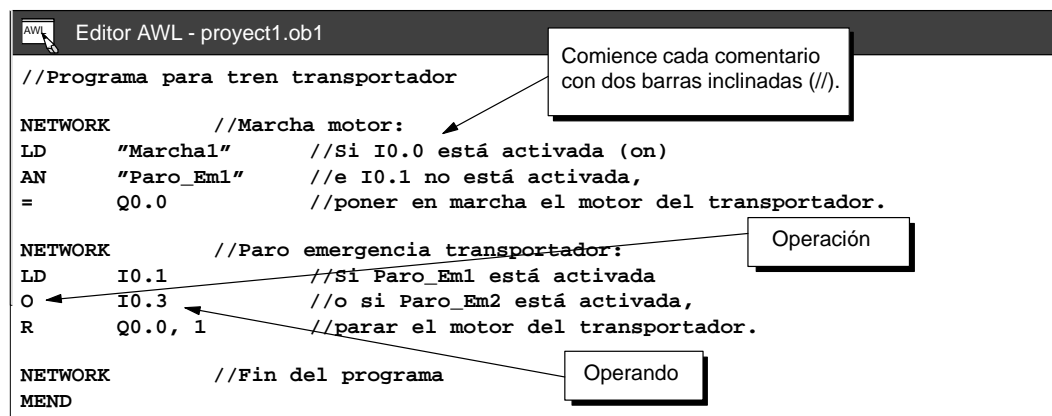


Figura 6-4 Ventana del editor AWL con un programa de ejemplo

Las operaciones AWL utilizan una pila lógica en la CPU para resolver la lógica. Como muestra la figura 6-5, dicha pila tiene nueve bits de profundidad y uno de ancho. La mayoría de las operaciones AWL utilizan el primer bit, o bien el primero y el segundo bit de la pila. Combinando los primeros dos bits de la pila, se pierde el valor del nivel superior y en su lugar se coloca el valor del nivel siguiente, es decir que a la pila se le resta un bit.

En tanto que la mayoría de las operaciones AWL leen sólo los valores de la pila lógica, muchas otras modifican también los valores ahí almacenados. La figura 6-5 muestra ejemplos de cómo tres operaciones utilizan la pila lógica.

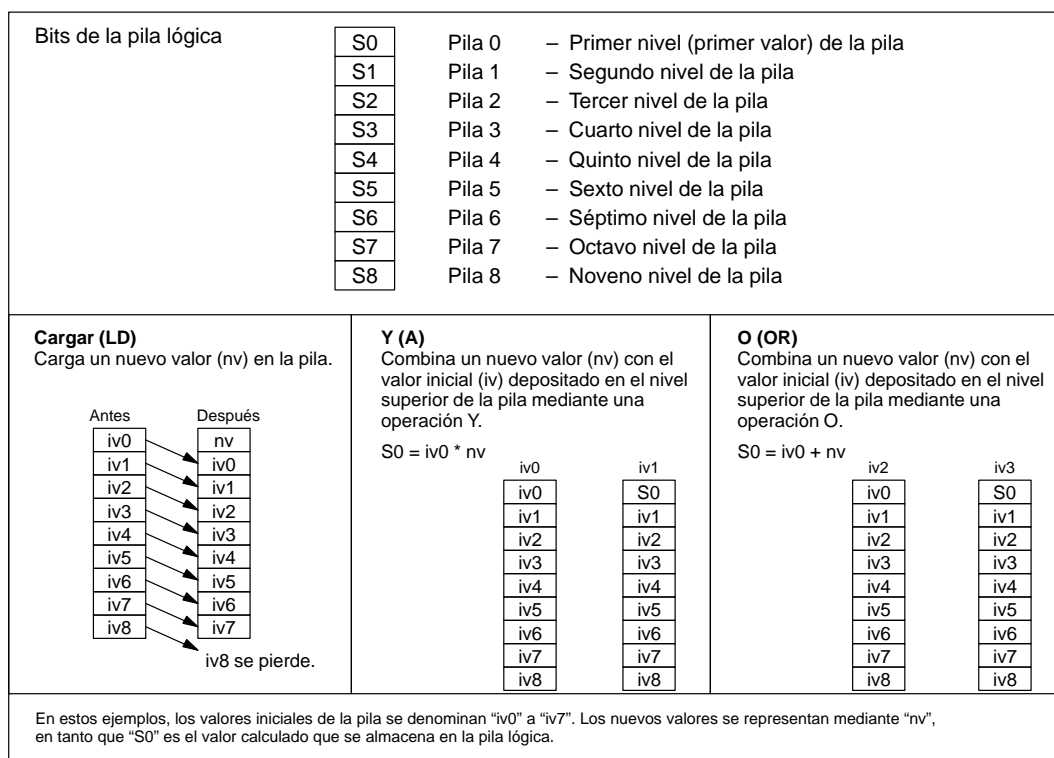


Figura 6-5 Pila lógica de la CPU S7-200

## 6.4 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 y se carga en la CPU. Desde el programa principal se pueden llamar 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). Como muestra la figura 6-6, un programa S7-200 se estructura mediante 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. Para terminar el programa principal, utilice en KOP una bobina absoluta Finalizar programa principal, o en AWL una operación Finalizar programa principal (MEND). Véase (1) en la figura 6-6.
- **Subrutinas:** Estos elementos opcionales del programa se ejecutan sólo cuando se llaman desde el programa principal. Se deben añadir siempre al final del programa principal (detrás de la bobina absoluta Finalizar programa principal en KOP o detrás de la operación MEND en AWL). Utilice siempre una operación Retorno absoluto (RET) para terminar cada subrutina. Véase (2) en la figura 6-6.
- **Rutinas de interrupción:** Estos elementos opcionales del programa se ejecutan cada vez que se presente el correspondiente evento de interrupción. Se deben añadir siempre al final del programa principal (detrás de la bobina absoluta Finalizar programa principal en KOP o detrás de la operación MEND en AWL). Utilice siempre una operación Retorno absoluto desde rutina de interrupción (RETI) para terminar cada rutina de interrupción. Véase (3) en la figura 6-6.

Las subrutinas y las rutinas de interrupción se deben añadir detrás de la bobina absoluta Finalizar programa principal en KOP o detrás de la operación MEND en AWL. No hay reglas adicionales en lo relativo a su disposición en el programa de usuario. Las subrutinas y las rutinas de interrupción se pueden mezclar a voluntad después del programa principal. No obstante, para que la estructura del programa sea fácil de leer y comprender, es recomendable agrupar al final del programa principal primero todas las subrutinas y, después, todas las rutinas de interrupción.

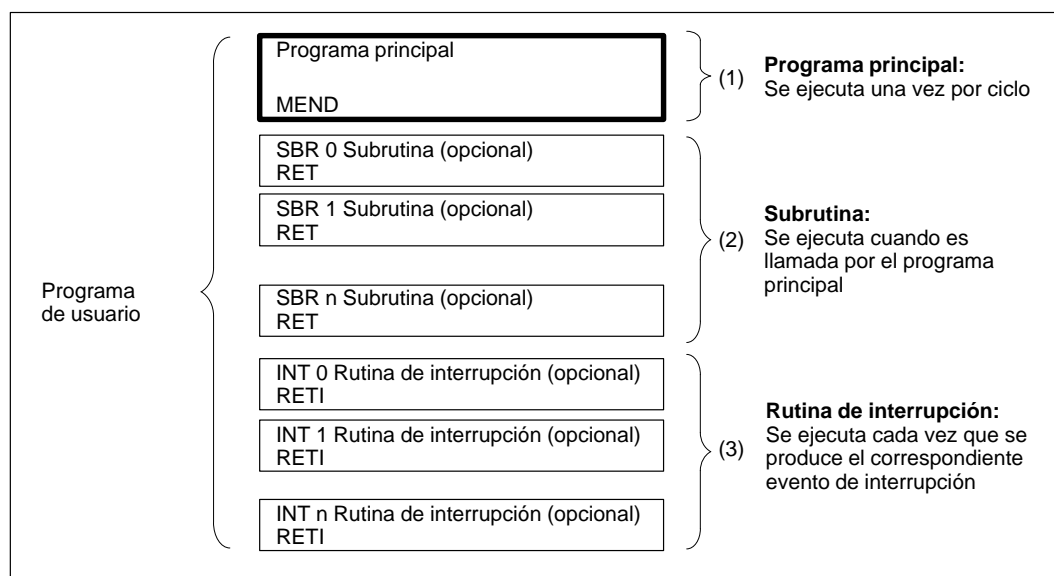


Figura 6-6 Estructura del programa de una CPU S7-200

### Programa de ejemplo con subrutinas y rutinas de interrupción

El programa de ejemplo representado en la figura 6-7 muestra una interrupción temporizada que se puede utilizar p.ej. para leer el valor de una entrada analógica. En este ejemplo, el intervalo de muestreo de la entrada analógica es de 100 ms.



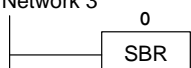
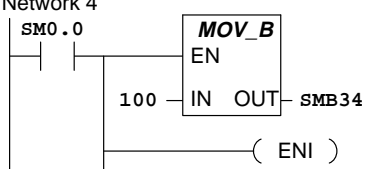
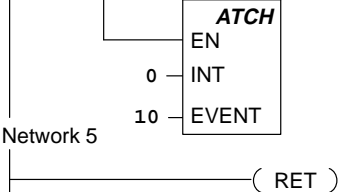
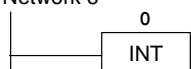
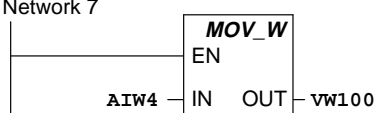
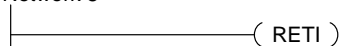
KOP	AWL
<b>Programa principal</b>	
<p>Network 1</p>  <p>Network 2</p> 	<p>Network 1</p> <pre>LD    SM0.1      //Si se activa la                 //marca del primer ciclo CALL  0          //Llamar subrutina 0.</pre> <p>Network 2</p> <pre>MEND</pre>
<b>Subrutinas</b>	
<p>Network 3</p>  <p>Network 4</p>  <p>Network 5</p> 	<p>Network 3</p> <pre>SBR  0          //Comenzar subrutina 0</pre> <p>Network 4</p> <pre>LD    SM0.0      //Marca de funcionamiento                 //continuo, MOVB  100, SMB34  //Ajustar a 100 ms el intervalo                 //de la int. temporiz. 0. ENI                    //Habilitar todos los eventos ATCH  0, 10       //Asociar int. temp. 0 a                 //rutina de interrup. 0.</pre> <p>Network 5</p> <pre>RET            //Fin subrutina.</pre>
<b>Rutinas de interrupción</b>	
<p>Network 6</p>  <p>Network 7</p>  <p>Network 8</p> 	<p>Network 6</p> <pre>INT  0          //Comenzar rutina de int. 0.</pre> <p>Network 7</p> <pre>MOVW AIW4,VW100 //Mostrar entrada analógica 4</pre> <p>Network 8</p> <pre>RETI           //Finalizar rutina                 de interrup.</pre>

Figura 6-7 Programa de ejemplo con una subrutina y una rutina de interrupción

## 6.5 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 se muestra en la figura 6-8, la CPU ejecuta la mayoría de las tareas siguientes (o todas ellas):

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

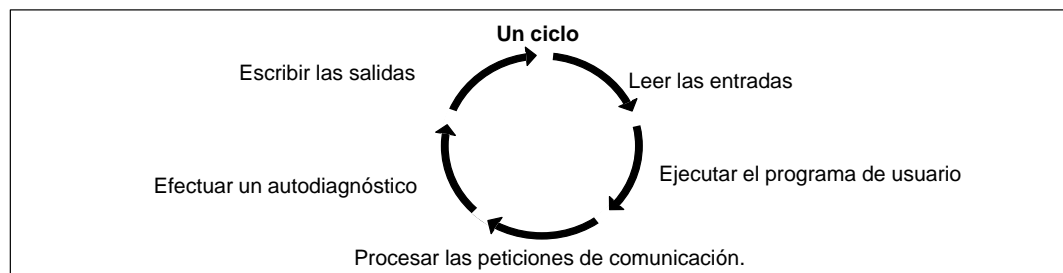


Figura 6-8 Ciclo de la CPU S7-200

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 proporcionan 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 asiste 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 automáticamente las entradas analógicas como parte del ciclo y no ofrece una imagen del proceso para las mismas. A las entradas analógicas se debe acceder directamente desde el programa de usuario.

### **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. 6.4). 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 interface 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 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. No obstante, los bits no utilizados de la imagen del proceso de las salidas se pueden usar como marcas internas adicionales.

La CPU no actualiza automáticamente las salidas analógicas como parte del ciclo y no ofrece una imagen del proceso para las mismas. A las salidas analógicas se debe acceder directamente desde el programa de usuario.

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. 8.3). 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.

Otra ventaja es que las imágenes del proceso son lo suficientemente grandes para poder procesar el número máximo de entradas y salidas. Puesto que un sistema real comprende tanto entradas como salidas, en la imagen del proceso existe siempre un número de direcciones que no se utilizan. Estas direcciones libres pueden utilizarse como marcas internas adicionales (v. apt. 8.1).

### **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.

## 6.6 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. Cuando está en modo RUN, no es posible cargar programas ni configurar la CPU.

El diodo luminoso (LED) en la parte frontal de la CPU indica el modo de operación actual. Para poder cargar un programa en la memoria de la CPU es preciso cambiar a modo STOP.

### 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 a 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 (terminal), no cambiará el modo de operación de la CPU. No obstante, será posible cambiarlo utilizando el software de programación (STEP 7-Micro/WIN).

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

Como muestra la figura 6-9, el modo de operación de la CPU se puede cambiar también con STEP 7-Micro/WIN. Para que ello sea posible, el selector de la CPU deberá estar en posición TERM o RUN.

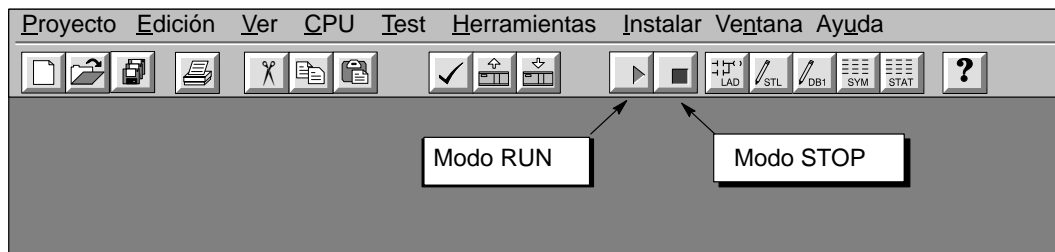


Figura 6-9 Utilizar STEP 7-Micro/WIN para cambiar el modo de operación de la CPU

### 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 10.

## 6.7 Determinar 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 determinar la contraseña.

### Restringir el acceso a la CPU

Como muestra la tabla 6-1, 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 las CPUs 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.

Tabla 6-1 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			
Leer y escribir el reloj de tiempo real			
Leer los datos forzados en la CPU			Restringido
Cargar en la PG el programa de usuario, los datos y la configuración			
Cargar en la CPU			
Borrar el programa de usuario, los datos y la configuración <sup>1</sup>			
Forzar datos o ejecutar uno/varios ciclo(s)			
Copiar en el cartucho de memoria			

<sup>1</sup> La contraseña "clearplc" puede imponerse a la protección contra el borrado.

### Configurar la contraseña para la CPU

STEP 7-Micro/WIN permite determinar una contraseña para acceder a las funciones de la CPU. Elija el comando de menú **CPU ► Configurar** y seleccione la ficha "Contraseña" (v. fig. 6-10). Indique el nivel de protección deseado. Introduzca y verifique luego la contraseña.

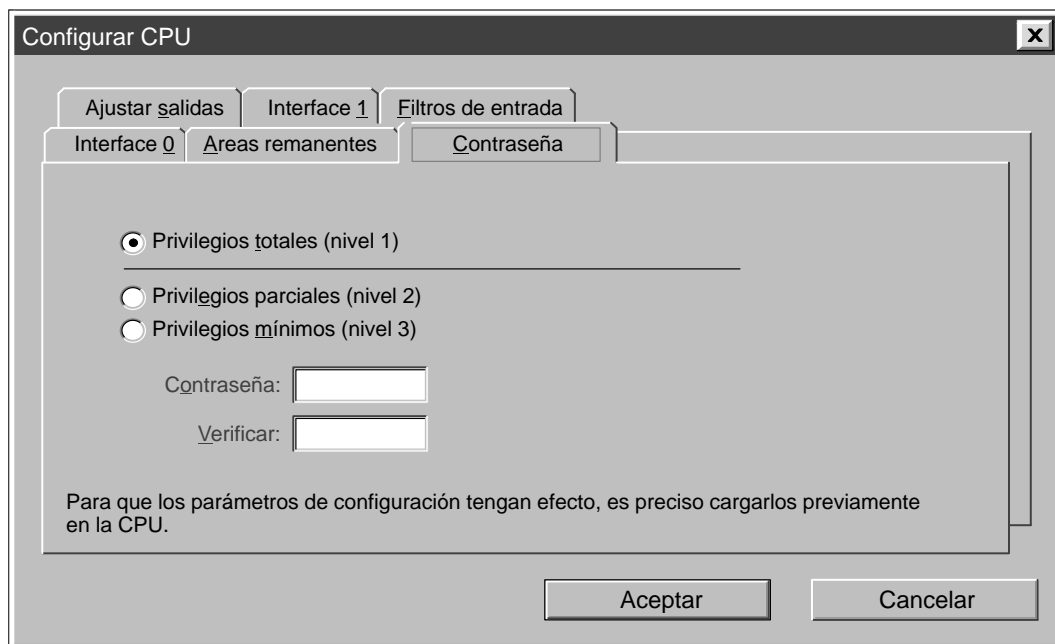


Figura 6-10 Configurar una contraseña para la CPU

**Remedio si se olvida la contraseña**

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 predeterminados, con excepción de la dirección de estación y del reloj de tiempo real.

Si desea borrar el programa de la CPU, elija el comando de menú **CPU ► Borrar...** para visualizar el cuadro de diálogo "Borrar CPU". Elija la opción "Todo" y confirme su acción haciendo clic en el botón "Aceptar". Entonces se visualizará un cuadro de diálogo donde deberá introducir la contraseña "clearplc" que permitirá 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 (las salidas analógicas 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 conectados, 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.

## 6.8 Comprobar y observar el programa

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

### Ejecutar uno 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). Al seleccionar el número de ciclos que la CPU debe ejecutar, se puede observar el programa a medida que van cambiando las variables del proceso. Para indicar el número de ciclos a ejecutar, elija el comando de menú **Test ► Ejecutar ciclos**. La figura 6-11 muestra el cuadro de diálogo para introducir el número de ciclos a ejecutar por la CPU.

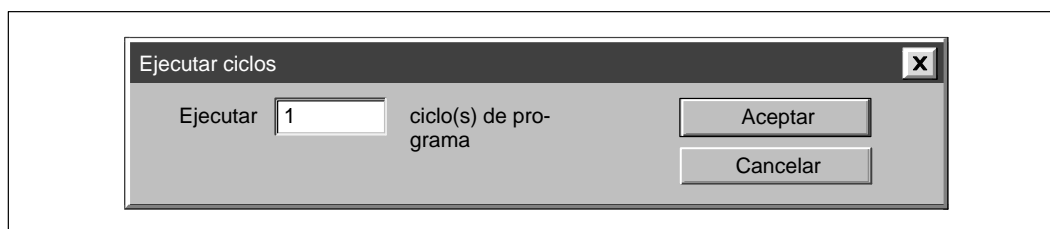


Figura 6-11 Ejecutar el programa un número determinado de ciclos

### Utilizar una tabla de estado para observar y modificar el programa

Como muestra la figura 6-12, la tabla de estado se puede utilizar para leer, escribir, forzar y observar las variables mientras se ejecuta el programa. Para obtener más información acerca de cómo crear dicha tabla, consulte el apartado 3.8.

Tabla de estado				
Dirección	Formato	Valor actual	Valor nuevo	
"Marcha_1"	Binario	2#0		
"Marcha_2"	Binario	2#0	1	
"Paro_1"	Binario	2#0		
"Paro_2"	Binario	2#0		
"Nivel_Superior"	Binario	2#0		
"Nivel_Inferior"	Binario	2#0		
"Desactivar"	Binario	2#0		
"Bomba_1"	Binario	2#0		
"Bomba_2"	Binario	2#0		
"Motor_Mezclador"	Binario	2#0		
"Válvula_Vapor"	Binario	2#0		
"Válvula_Vaciado"	Binario	2#0		
"Bomba_Vaciado"	Binario	2#0		
"Niv_Sup_Alcanz"	Binario	2#0		
"Temporiz_Mezcla"	Con signo	+0		
"Contador_Ciclos"	Con signo	+0		

Figura 6-12 Observar y modificar variables con una tabla de estado

### Visualizar el estado del programa en KOP

Como muestra la figura 6-13, el editor de programas de STEP 7-Micro/WIN permite observar el estado del programa online. (El programa se debe visualizar en KOP). Así es posible observar el estado de las operaciones del programa a medida que se ejecutan en la CPU.

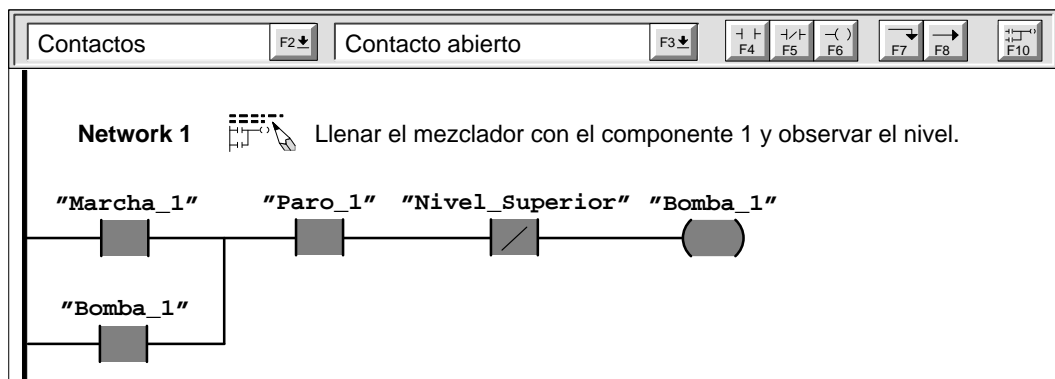


Figura 6-13 Visualizar el estado de un programa en KOP

### Utilizar la tabla de estado para forzar valores determinados

La CPU S7-200 permite forzar algunas o todas las entradas y salidas (I y Q), así como las variables para que adopten determinados valores. 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 sólo en formato de palabras y siempre en bytes pares (p.ej. AIW6 o AIW14). 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 6-14 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, se impone a una salida que se haya configurado para que adopte un valor determinado cuando la CPU cambie a STOP. En este último caso, la salida conservará el valor forzado y no el valor configurado.

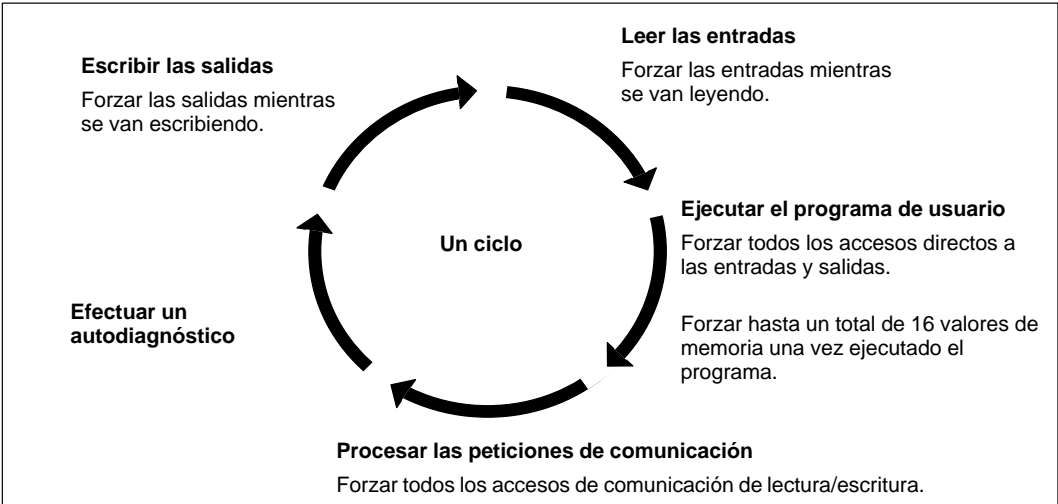


Figura 6-14 Ciclo de la CPU S7-200

La figura 6-15 muestra un ejemplo de la tabla de estado. Para obtener más información al respecto, consulte el apartado 3.8.

La imagen muestra una ventana de software titulada 'Tabla de estado'. En la parte superior hay una barra de herramientas con iconos para guardar, imprimir, borrar, bloquear, desbloquear, mover y zoom. La tabla principal contiene la siguiente información:

Dirección	Formato	Valor actual	Valor nuevo
"Marcha_1"	Binario	2#0	
"Marcha_2"	Binario	2#0	1
"Paro_1"	Binario	2#0	
"Paro_2"	Binario	2#0	
"Nivel_Superior"	Binario	2#0	
"Nivel_Inferior"	Binario	2#0	
"Desactivar"	Binario	2#0	
"Bomba_1"	Binario	2#0	
"Bomba_2"	Binario	2#0	
"Motor_Mezclador"	Binario	2#0	
"Válvula_Vapor"	Binario	2#0	
"Válvula_Vaciado"	Binario	2#0	
"Bomba_Vaciado"	Binario	2#0	
"Niv_Sup_Alcanz"	Binario	2#0	
"Temporiz_Mezcla"	Con signo	+0	
"Contador_Ciclos"	Con signo	+0	

Figura 6-15 Forzar variables mediante la tabla de estado

## 6.9 Eliminar errores de las CPUs S7-200

Las CPUs S7-200 clasifican los errores en errores fatales y no fatales. STEP 7-Micro/WIN permite visualizar los códigos generados por los errores. La figura 6-16 muestra un cuadro de diálogo donde se visualizan el código y la descripción del error. El Anexo C incluye una lista completa de los códigos de error.

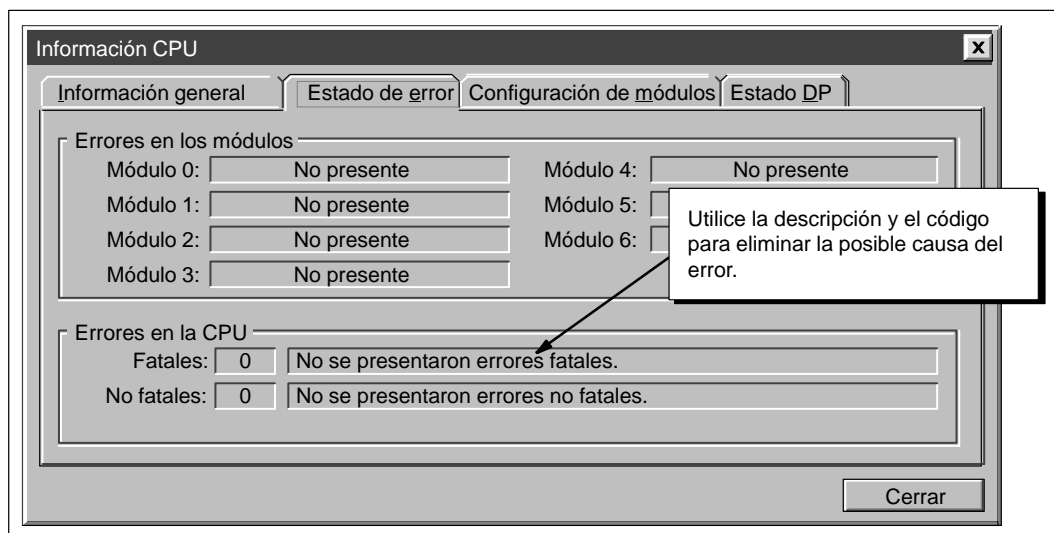


Figura 6-16 Cuadro de diálogo "Información CPU": ficha "Estado de error"

### 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. A tal efecto se puede desconectar y volver a conectar la alimentación de la CPU, o bien cambiar la posición del selector de modos de operación de RUN o TERM a STOP. 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 un defecto del hardware, por lo que será necesario reparar la CPU. Estas condiciones no se pueden remediar modificando el programa o efectuando un borrado total 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 6-16, STEP 7-Micro/WIN 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. Si desea obtener más información acerca de las marcas especiales utilizadas para indicar los errores no fatales durante el tiempo de ejecución, consulte el Anexo D.  
Durante el arranque, la CPU 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 usuario (o el programa) pueden crear condiciones de error mientras se ejecuta el programa. Por ejemplo, un puntero de direccionamiento indirecto que era válido cuando se compiló el programa se puede haber modificado durante la ejecución del programa, señalando entonces una dirección fuera de área. Esto es considerado un error de programación durante el tiempo de ejecución. Para determinar el tipo de error que se ha presentado, utilice el cuadro de diálogo que muestra la figura 6-16.

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 6-17 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.

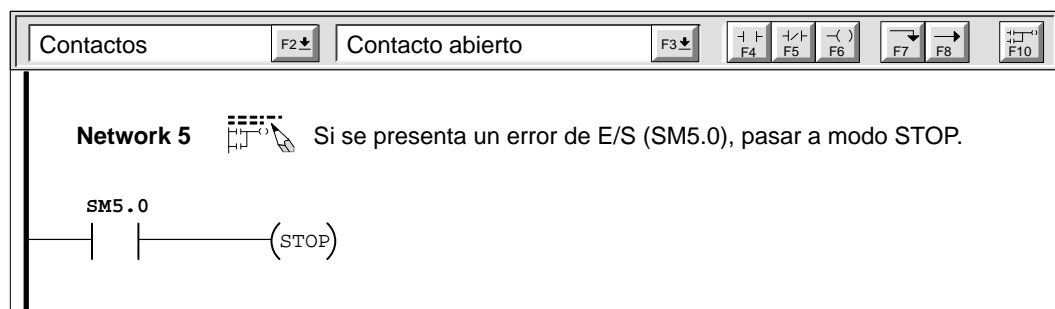


Figura 6-17 Detectar errores no fatales mediante el programa de usuario

# Memoria de la CPU: tipos de datos y direccionamiento

# 7

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
7.1	Direccionamiento directo de las áreas de memoria de la CPU	7-2
7.2	Direccionamiento indirecto de las áreas de memoria de la CPU	7-9
7.3	Respaldo de datos en la CPU S7-200	7-11
7.4	Utilizar el programa para guardar datos en la memoria no volátil	7-16
7.5	Guardar el programa en el cartucho de memoria	7-17

## 7.1 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 7-1 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).

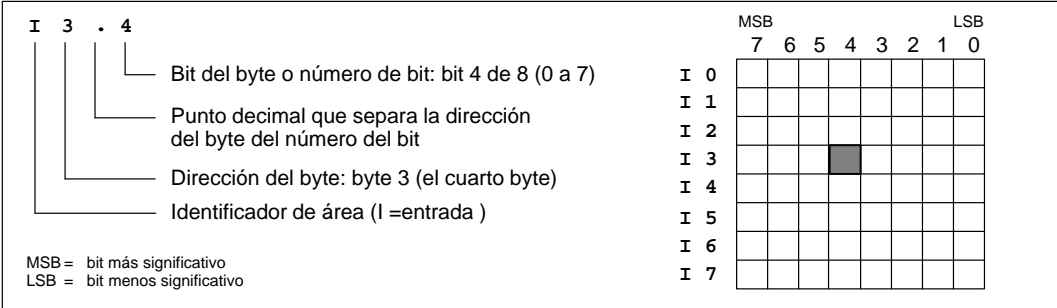


Figura 7-1 Acceder a un bit de datos en la memoria de la CPU (direccionamiento "byte.bit")

Utilizando la dirección del byte se puede acceder a los datos de diversas áreas de memoria de la CPU (V, I, Q, M y SM) en formato de byte, palabra o palabra doble. 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 KEIN MERKER. 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.

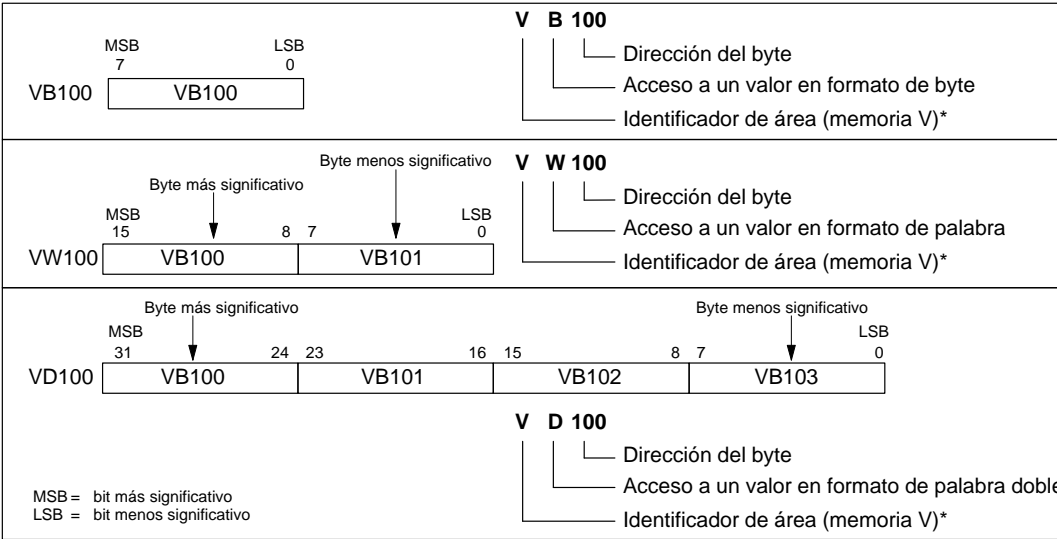


Figura 7-2 Acceso a la misma dirección en formato de byte, palabra y palabra doble

## Representación numérica

La tabla 7-1 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, cuyo formato se describe en la norma ANSI/IEEE 754-1985. A los valores de números reales se accede en formato de palabra doble.

Tabla 7-1 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 6.5, 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)

Las marcas internas (área de marcas M) se pueden utilizar como relés de control para almacenar el estado intermedio de una operación u otras informaciones 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. Dichos relés 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:

- Un bit que se activa sólo en el primer ciclo.
- Bits que se activan y se desactivan en determinados intervalos.
- Bits que muestran el estado de operaciones matemáticas y de otras operaciones.

Para obtener más información acerca de las marcas especiales, consulte el Anexo D. 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]M0.1*  
 Byte, palabra, palabra doble *SM[tamaño][direcc. del byte inicial] SMB86*

### 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 (se pone a 1) cuando el valor actual del temporizador es mayor o igual al valor predeterminado. (Éste último se introduce como parte de la operación).

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 7-3, la operación Contacto 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 del juego de operaciones del S7-200, consulte el capítulo 10.

Formato: *T[número del temporizador] T24*

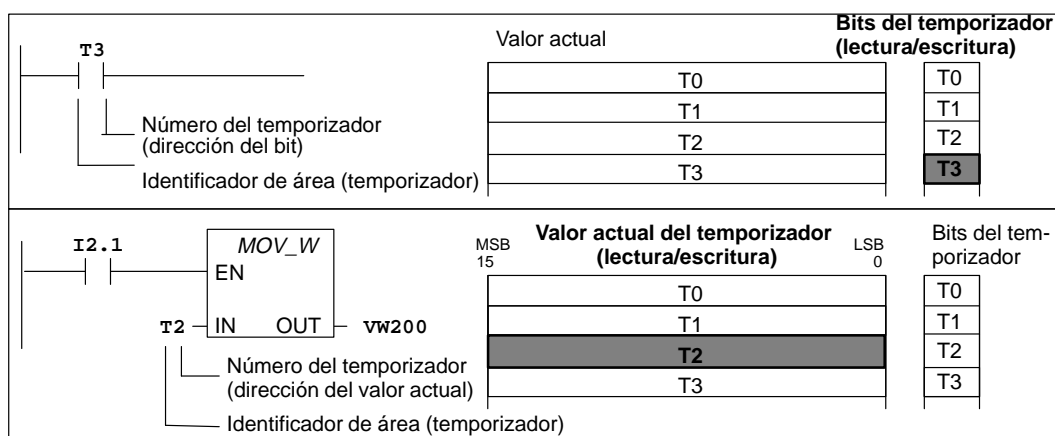


Figura 7-3 Acceder a los datos del temporizador

### 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 dos tipos de contadores: uno que cuenta sólo adelante 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 (se pone a 1) cuando el valor actual del contador es mayor o igual al valor predeterminado. (Éste último se introduce como parte de la operación).

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 7-4, la operación Contacto 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 del juego de operaciones del S7-200, consulte el capítulo 10.

Formato: **C[número del contador]** **C20**

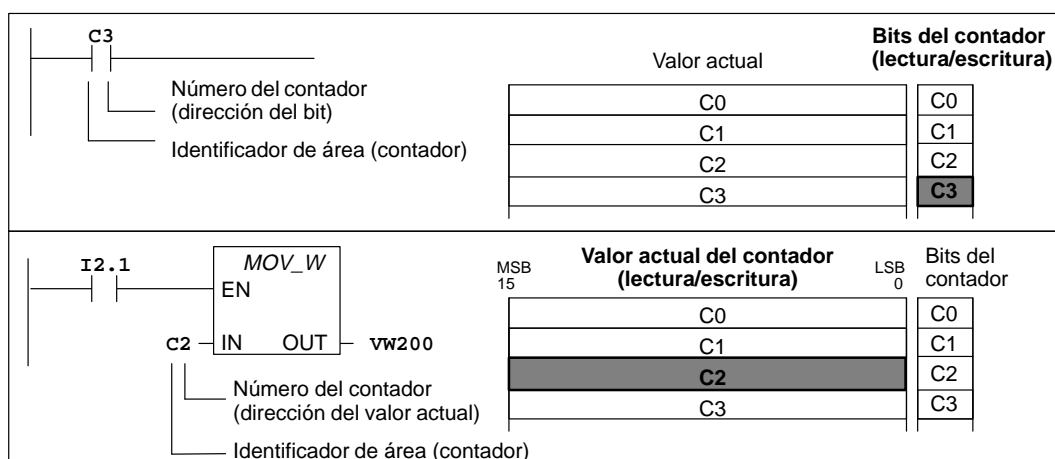


Figura 7-4 Acceder a los datos del contador

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 7-5. Las entradas analógicas son valores de sólo lectura.

Formato: AIW[dirección del byte inicial]      AIW4

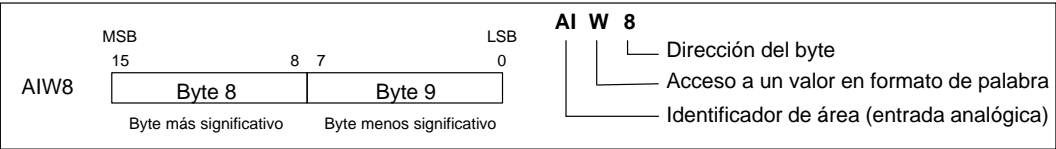


Figura 7-5      Acceso a una entrada analógica

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. corriente o voltaje), 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 7-6. El programa no puede leer los valores de las salidas analógicas.

Formato: AQW[dirección del byte inicial]      AQW4

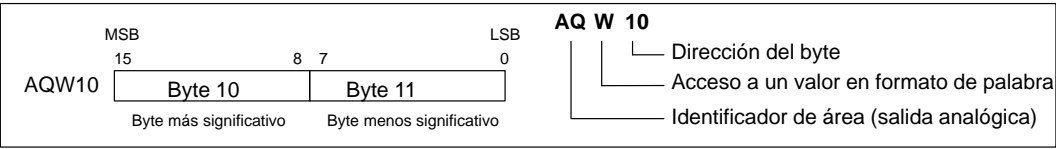


Figura 7-6      Acceso a una salida analógica

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 7-7, 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 10.14 para obtener más información acerca de cómo utilizar los acumuladores con rutinas de interrupción.

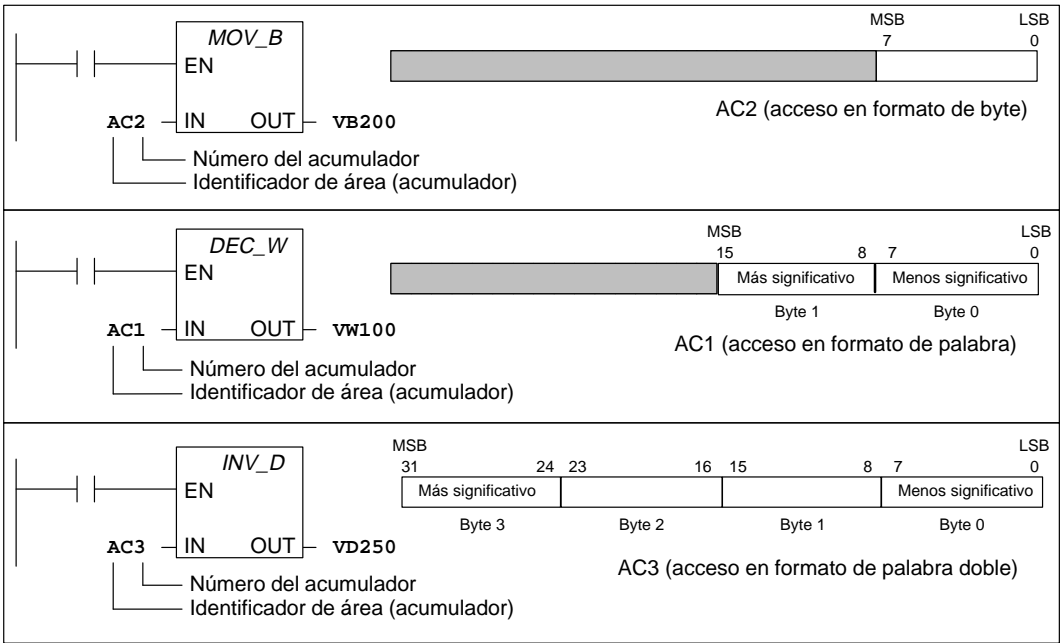


Figura 7-7 Acceso a los acumuladores

**Direccionamiento de los contadores rápidos (HC)**

Los contadores rápidos son elementos que cuentan los eventos más deprisa de lo que puede explorarlos 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 7-8.

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

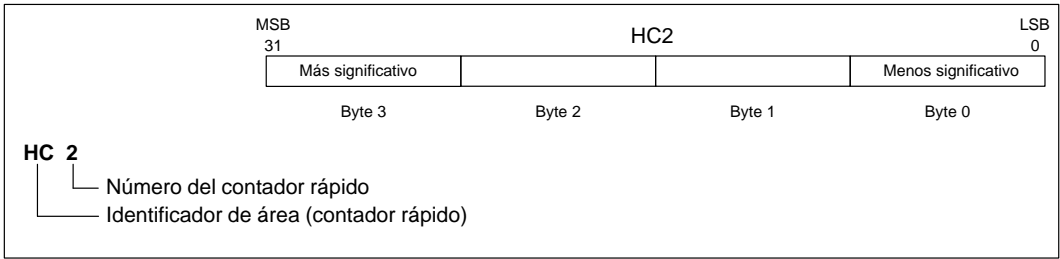


Figura 7-8 Acceso a los valores actuales del contador rápido

### 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 o ASCII.

Formato decimal:        **[valor decimal]**

Formato hexadecimal: **16#[valor hexadecimal]**

Formato ASCII:        **'[texto ASCII]'**

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 y ASCII:

- Constante decimal:        **20047**
- Constante hexadecimal: **16#4E4F**
- Constante ASCII:        **'El texto aparece entre comillas sencillas.'**

## 7.2 Direccionamiento indirecto 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 o 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:**

<b>MOVD</b>	<b>&amp;VB100, VD204</b>
<b>MOVD</b>	<b>&amp;MB4, AC2</b>
<b>MOVD</b>	<b>&amp;C4, VD6</b>

### Nota

Si desea acceder indirectamente a un valor de palabra o de palabra doble en las áreas de memoria I, Q, V, M o S, debe indicar la dirección del byte inicial del valor como operando de entrada de la operación MOVD utilizada para crear el puntero. Por ejemplo, VB100 es la dirección del byte inicial de VW100, en tanto que MB4 es la dirección del byte inicial de MD4. Si al valor de palabra o de palabra doble se le ha asignado un nombre simbólico, éste último no se puede usar en la operación MOVD utilizada para crear el puntero, puesto que la dirección del byte inicial del valor se debe indicar en el operando de entrada de la operación. En estos casos es preciso asignarle un nombre simbólico diferente a la dirección del byte inicial de la palabra o de la palabra doble utilizada para crear el puntero.

Ejemplo: `''Velocidad_Bomba''` es el nombre simbólico de VW100  
`''Velocidad_Bomba_IB''` es el nombre simbólico de VB100  
 (que es el byte inicial del valor de palabra almacenado en VW100)

<b>MOVD &amp;"Velocidad_Bomba", AC1</b>	no válido (&VW100 no está permitido)
<b>MOVD &amp;"Velocidad_Bomba_IB", AC1</b>	correcto (&VB100 está permitido)

### 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 7-9, \*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.

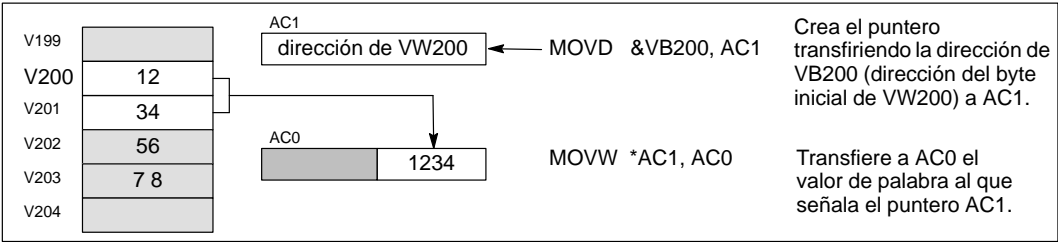


Figura 7-9 Utilizar un puntero para el direccionamiento indirecto

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 una 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 7-10 muestra un ejemplo de cómo crear un puntero de direccionamiento indirecto y de cómo acceder indirectamente a los datos e incrementar el puntero.

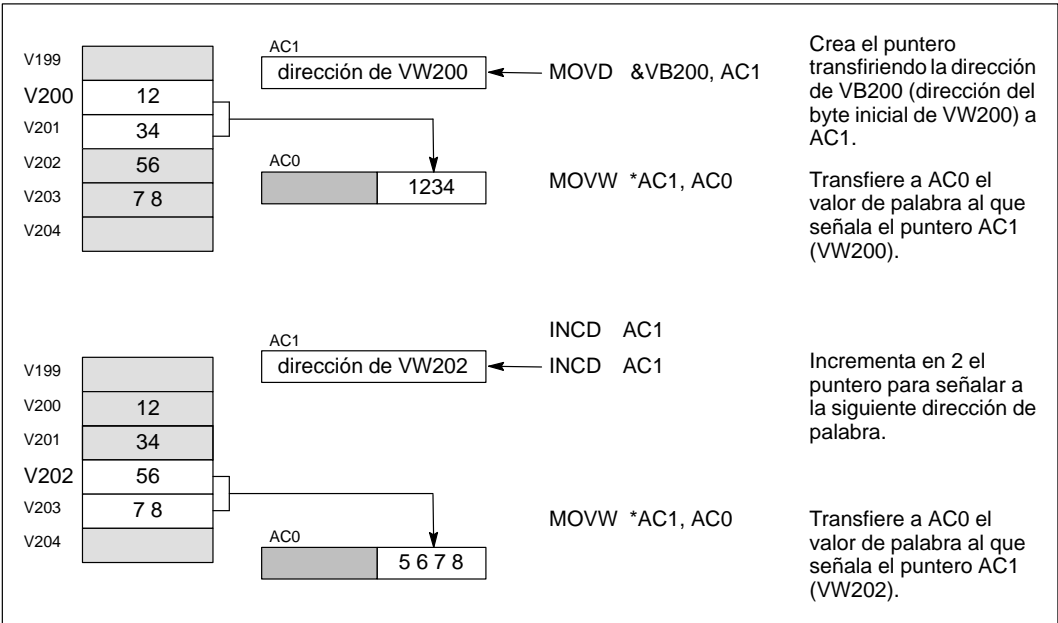


Figura 7-10 Modificar un puntero al acceder a un valor de palabra

### 7.3 Respaldo de 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:

- La CPU dispone de una EEPROM no volátil para almacenar todo el programa, así como algunas áreas de datos y la configuración de la CPU (v. fig. 7-11).
- 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.
- Algunas CPUs asisten 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 bajo diversas circunstancias.

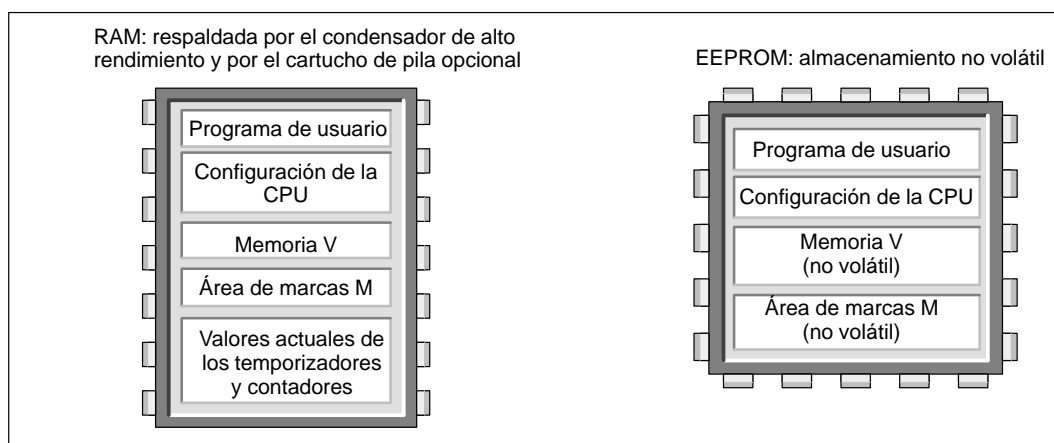


Figura 7-11 Áreas de memoria de la CPU S7-200

#### Cargar programas en la CPU y en la PG

El programa comprende tres elementos: el programa de usuario, el bloque de datos (opcional) y la configuración de la CPU (opcional). Como muestra la figura 7-12, cargando el programa 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.

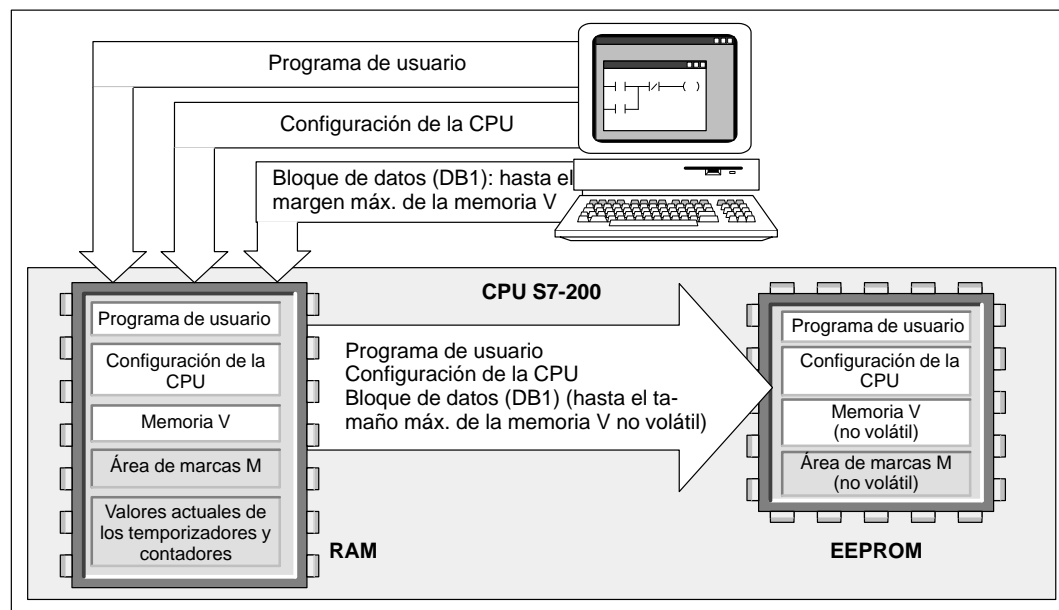


Figura 7-12 Cargar los elementos del programa en la CPU

Como muestra la figura 7-13, cuando un programa se carga de la CPU en el PC (o la PG), el programa de usuario y la configuración de la CPU se cargan de la RAM en el PC (o la PG). Cuando se carga el bloque de datos en el PC (o la PG), el área no volátil del bloque de datos (almacenada en la EEPROM) se fusiona con el resto del bloque de datos (si existe) que esté almacenado en la RAM. El bloque de datos completo se transfiere entonces al PC (ó a la PG). El tamaño de la memoria V no volátil depende de la CPU (v. apt. 10.1).

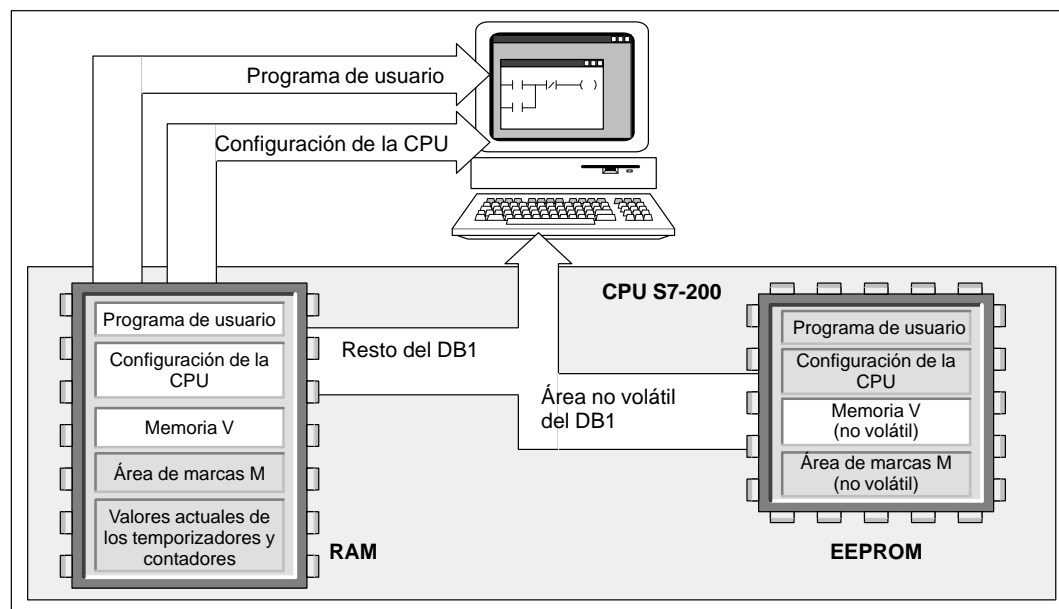


Figura 7-13 Cargar los elementos del programa en el PC (o la PG)

### 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 no volátil cuando se produzca un corte de alimentación de la CPU. Como muestra la figura 7-14, la CPU transfiere dichas áreas remanentes del área de marcas a la EEPROM.

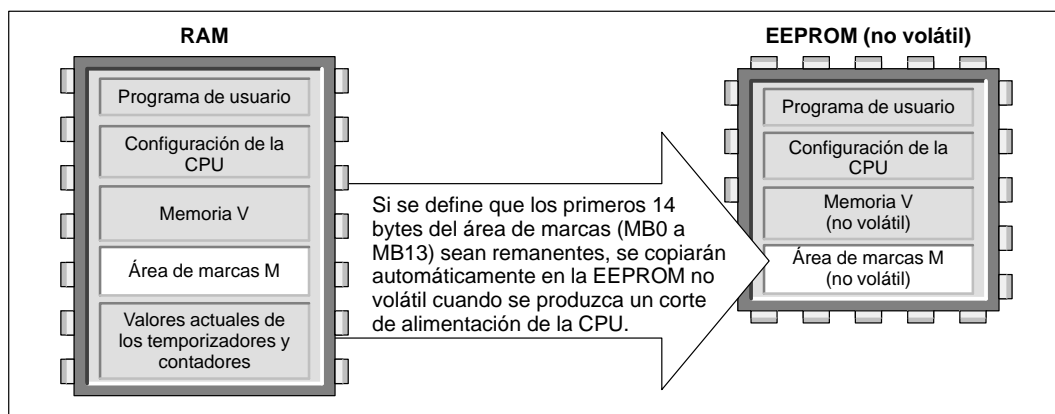


Figura 7-14 Almacenar partes del área de marcas (M) en la EEPROM en caso de un corte de alimentación

### Respalidar 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. 7-15).

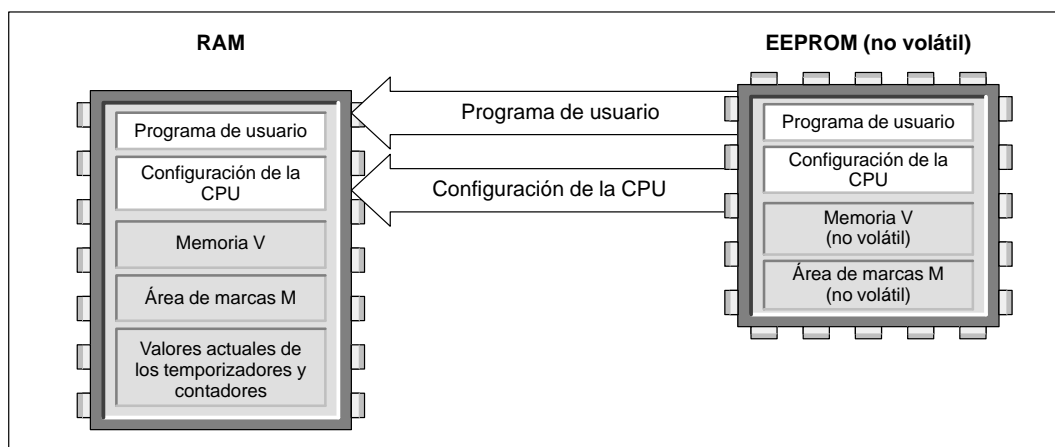


Figura 7-15 Restablecer el programa de usuario y la configuración de la CPU al conectar la alimentación

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 7-16, 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.

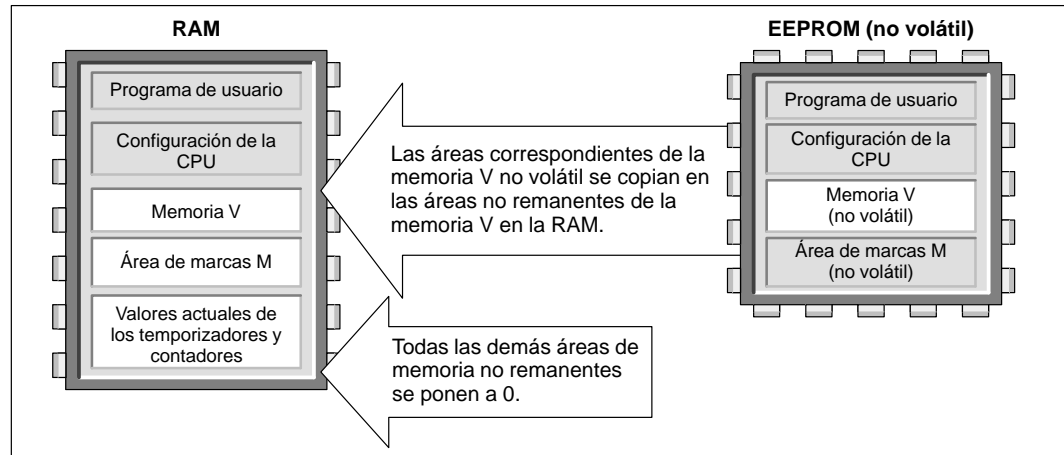


Figura 7-16 Restablecer los datos del programa al conectarse la alimentación (habiéndose respaldado los datos en la RAM)

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 7-17, los datos almacenados en la EEPROM no volátil se copian entonces en la memoria RAM.

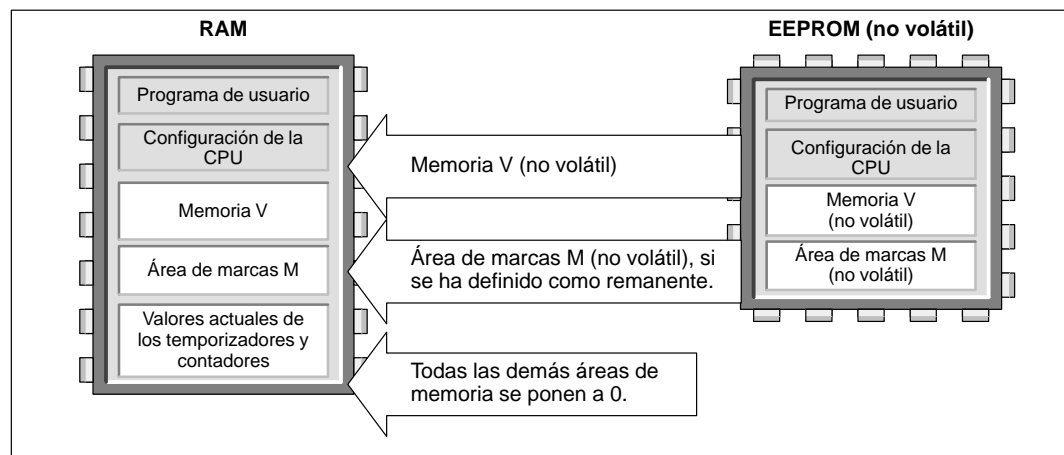


Figura 7-17 Restablecer los datos del programa al conectarse la alimentación (sin haberse respaldado los datos en la RAM)

### Definir las áreas remanentes

Como muestra la figura 7-18, el usuario puede definir hasta seis áreas remanentes para elegir las áreas de memoria que se deberán respaldar cuando se interrumpa la alimentación. Se puede 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).

#### 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 de la memoria, elija el comando de menú **CPU ► Configurar** y haga clic en la ficha "Áreas remanentes". La figura 7-18 muestra el cuadro de diálogo donde se definen dichas áreas. Para visualizar las áreas remanentes predeterminadas de su CPU, haga clic en el botón **Estándar**.

	Área de datos	Offset	Número de elementos	
Área 0:	<input type="text"/>	<input type="text"/>	<input type="text"/>	Borrar
Área 1:	<input type="text"/>	<input type="text"/>	<input type="text"/>	Borrar
Área 2:	<input type="text"/>	<input type="text"/>	<input type="text"/>	Borrar
Área 3:	<input type="text"/>	<input type="text"/>	<input type="text"/>	Borrar
Área 4:	<input type="text"/>	<input type="text"/>	<input type="text"/>	Borrar
Área 5:	<input type="text"/>	<input type="text"/>	<input type="text"/>	Borrar

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

Estándar

Aceptar Cancelar

Figura 7-18 Configurar las áreas remanentes de la memoria de la CPU

## 7.4 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.

La operación de guardar en EEPROM prolonga el tiempo de ciclo en unos 15 ms a 20 ms. Si en esa 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 7-19 muestra el formato de SMB31 y SMW32. Si desea programar la CPU para guardar o escribir un valor determinado en la memoria V:

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. 7-19).
3. Active la marca SM31.7.

Al final de cada ciclo, la CPU comprueba 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.

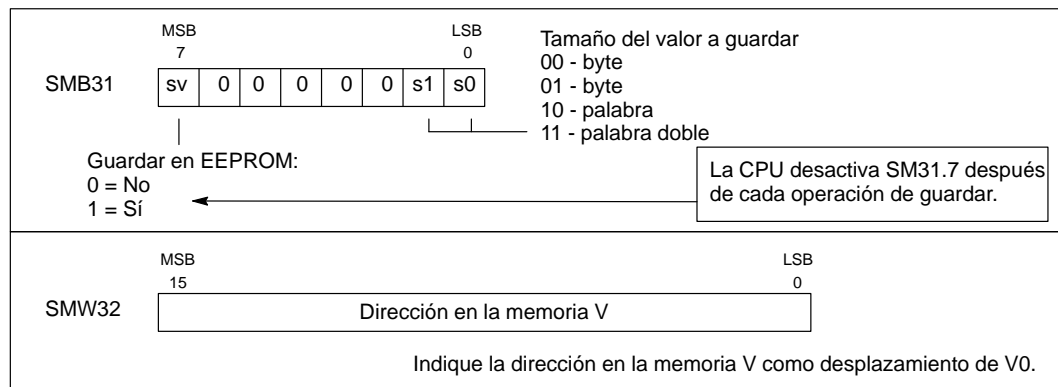


Figura 7-19 Formato de SMB31 y SMW32

### 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 muy frecuentemente.

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.

## 7.5 Guardar el programa en un cartucho de memoria

Algunas CPUs asientan un cartucho de memoria opcional que permite guardar el programa en una EEPROM portátil. Dicho cartucho se puede utilizar como un disquete. 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

Para obtener más información sobre el cartucho de memoria adecuado para su CPU, 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 y si está insertado dicho cartucho.



#### Cuidado

Las descargas electrostáticas 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 instalar o extraer estando conectada la alimentación de la CPU. Para instalarlo, retire la cinta protectora del receptáculo e inserte allí el cartucho. El receptáculo está ubicado debajo de la tapa de acceso de la CPU. (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. Cargue el programa en la CPU si no lo ha hecho todavía.
2. Elija el comando de menú **CPU ► Cartucho de memoria** para copiar el programa en el cartucho de memoria. La figura 7-20 muestra los componentes de la memoria de la CPU que se almacenan en dicho cartucho.
3. Extraiga el cartucho de memoria del receptáculo (opcional).

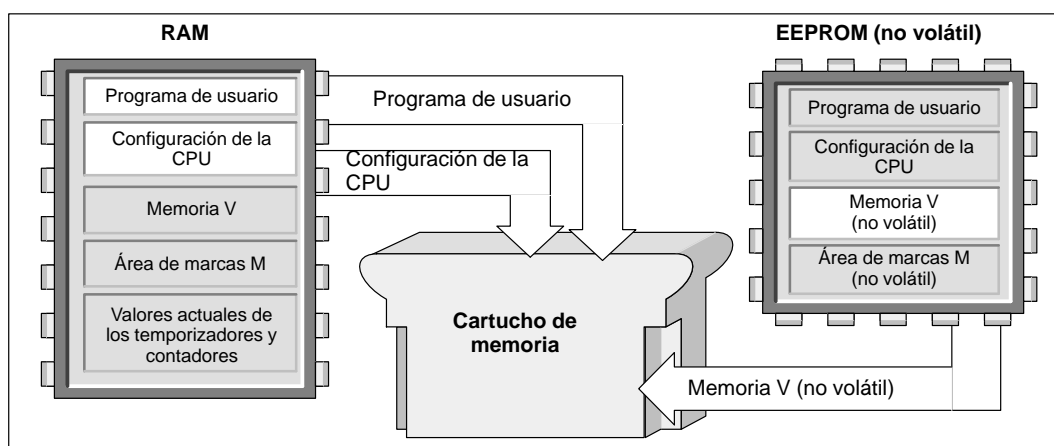


Figura 7-20 Copiar la memoria de la CPU en el cartucho de memoria

### 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 7-21, 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 (hasta el tamaño máximo del área no volátil de la misma) 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 producirá un error. Retire el cartucho de memoria y arranque la CPU de nuevo. Entonces será posible insertar y programar el cartucho.

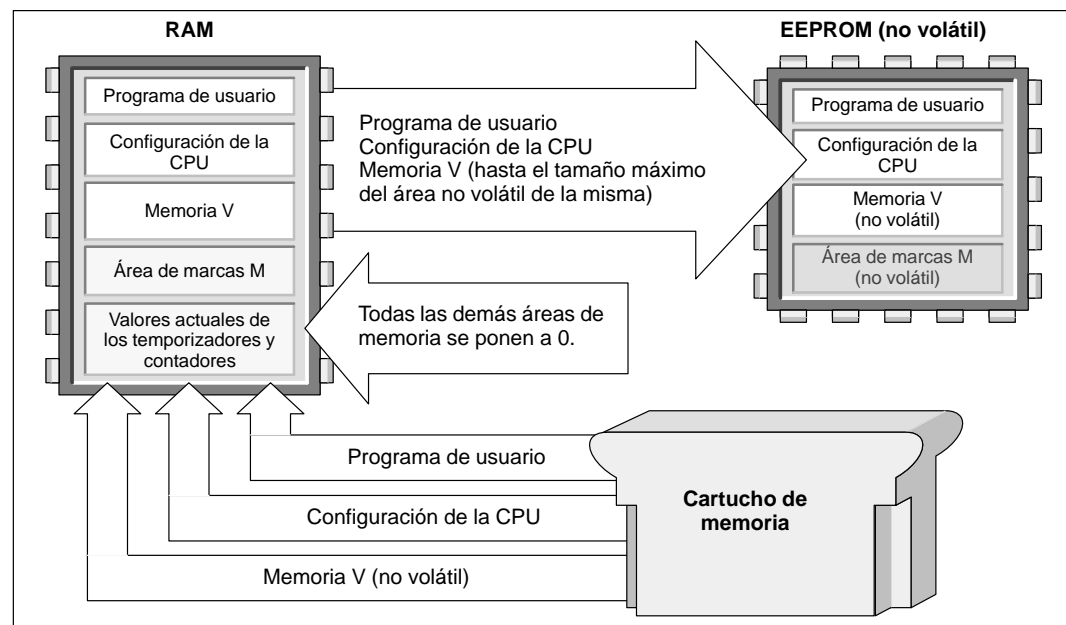


Figura 7-21 Restablecer la memoria durante el arranque (estando insertado un cartucho de memoria)

## 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
8.1	Entradas y salidas integradas y adicionales	8-2
8.2	Utilizar filtros de entrada para suprimir interferencias	8-5
8.3	Configurar los estados de señal para las salidas	8-6
8.4	Entradas y salidas rápidas	8-7
8.5	Potenciómetros analógicos	8-8

## 8.1 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 S7-200 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 de la unidad central (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 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 las direcciones de un módulo de entradas y viceversa. Igualmente, los módulos analógicos no afectan el 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. Por lo que respecta a los módulos de salidas, los bits no utilizados en los bytes reservados pueden servir de marcas internas (M). 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, por lo que no pueden servir de marcas internas.

Los módulos de ampliación analógicos se prevén 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. Puesto que para las E/S analógicas no se prevé imagen del proceso, no hay ninguna manera de aprovechar estas E/S no utilizadas. Todos los accesos de E/S analógicas se establecen en el mismo instante de ejecutarse la operación.

### Ejemplos de E/S integradas y adicionales

Las figuras 8-1, 8-2 y 8-3 muestran ejemplos de cómo las diferentes configuraciones del hardware afectan la numeración de las entradas y salidas. Observe que algunas configuraciones contienen intervalos de direcciones que el programa no puede utilizar, en tanto que otras direcciones de E/S se pueden emplear de igual manera que las marcas internas (M).

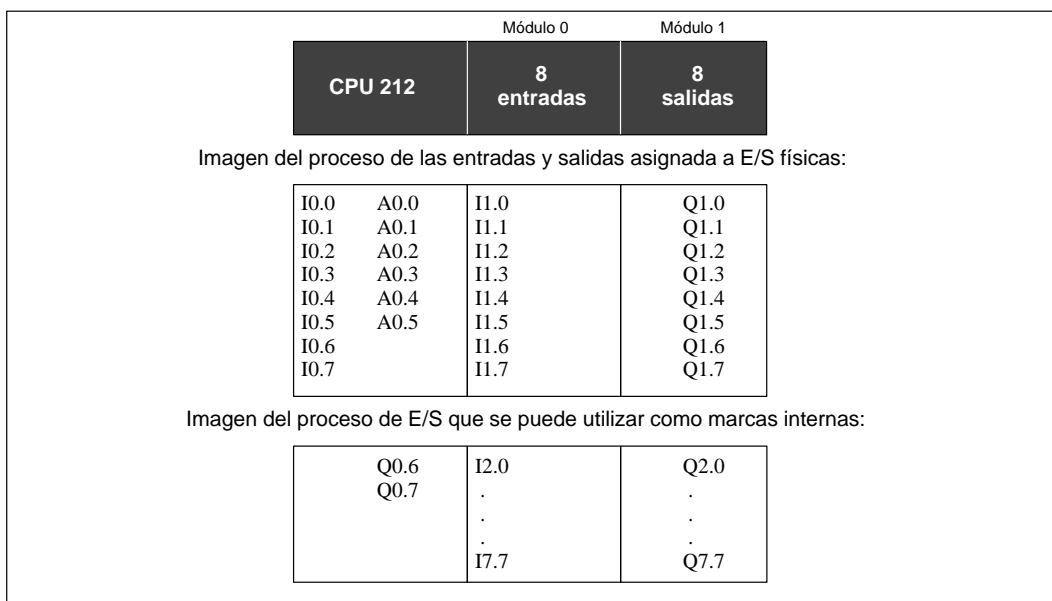


Figura 8-1 Ejemplos de numeración de E/S para una CPU 212

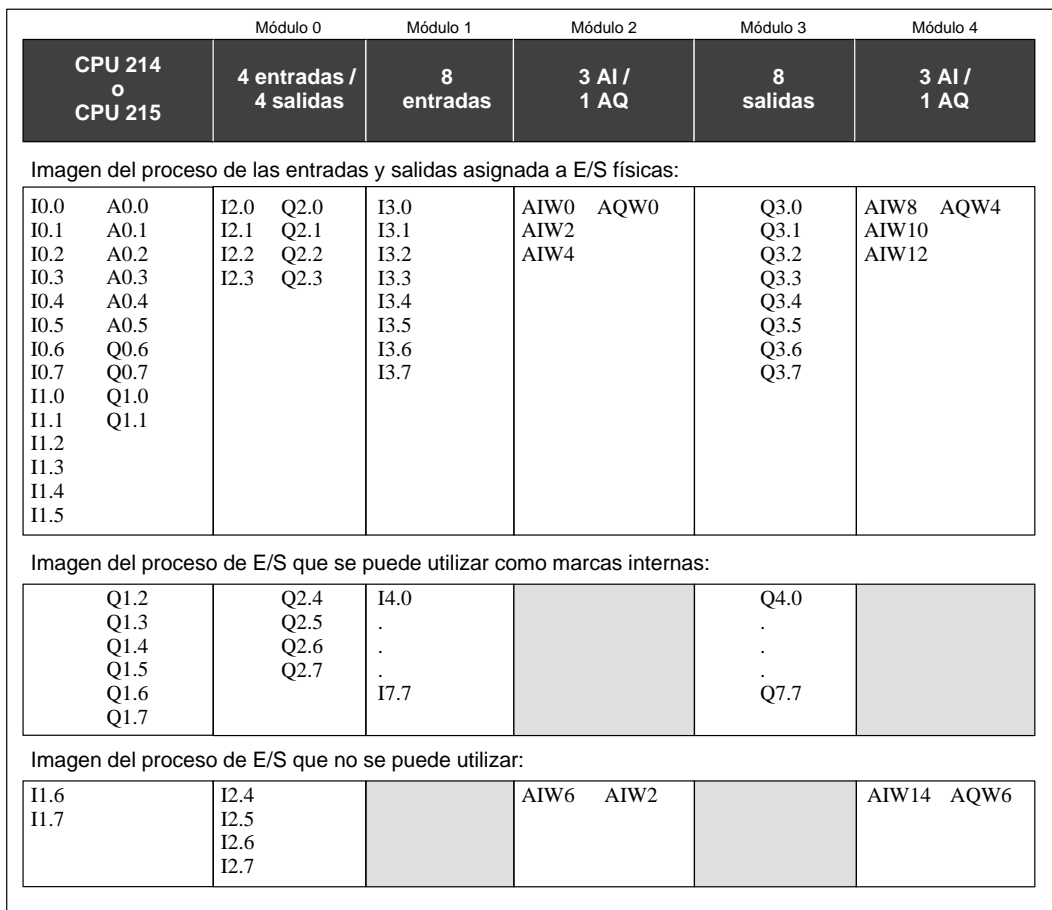


Figura 8-2 Ejemplo de numeración de E/S para una CPU 214 ó 215

Módulo 0		Módulo 1		Módulo 2	
CPU 216		8 entradas / 8 salidas	16 entradas / 16 salidas	16 entradas / 16 salidas	
Imagen del proceso de las entradas y salidas asignada a E/S físicas:					
I0.0	A0.0	I3.0	Q2.0	I4.0	Q3.0
I0.1	A0.1	I3.1	Q2.1	I4.1	Q3.1
I0.2	A0.2	I3.2	Q2.2	I4.2	Q3.2
I0.3	A0.3	I3.3	Q2.3	I4.3	Q3.3
I0.4	A0.4	I3.4	Q2.4	I4.4	Q3.4
I0.5	A0.5	I3.5	Q2.5	I4.5	Q3.5
I0.6	Q0.6	I3.6	Q2.6	I4.6	Q3.6
I0.7	Q0.7	I3.7	Q2.7	I4.7	Q3.7
I1.0	Q1.0			I5.0	Q4.0
I1.1	Q1.1			I5.1	Q4.1
I1.2	Q1.2			I5.2	Q4.2
I1.3	Q1.3			I5.3	Q4.3
I1.4	Q1.4			I5.4	Q4.4
I1.5	Q1.5			I5.5	Q4.5
I1.6	Q1.6			I5.6	Q4.6
I1.7	Q1.7			I5.7	Q4.7
I2.0					
I2.1					
I2.2					
I2.3					
I2.4					
I2.5					
I2.6					
I2.7					

Figura 8-3 Ejemplos de numeración para una CPU 216

## 8.2 Utilizar filtros de entrada para suprimir interferencias

Algunas CPUs S7-200 permiten seleccionar un filtro de entrada que define un tiempo de retardo (comprendido entre 0,2 ms y 8,7 ms) para algunas o todas las entradas digitales integradas. (Para obtener más información acerca de su CPU, consulte el Anexo A). Como muestra la figura 8-4, el tiempo de retardo se añade al tiempo de respuesta estándar para 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.

Elija el comando de menú **CPU ► Configurar...** y haga clic en la ficha "Filtros de entrada" para configurar los tiempos de retardo.

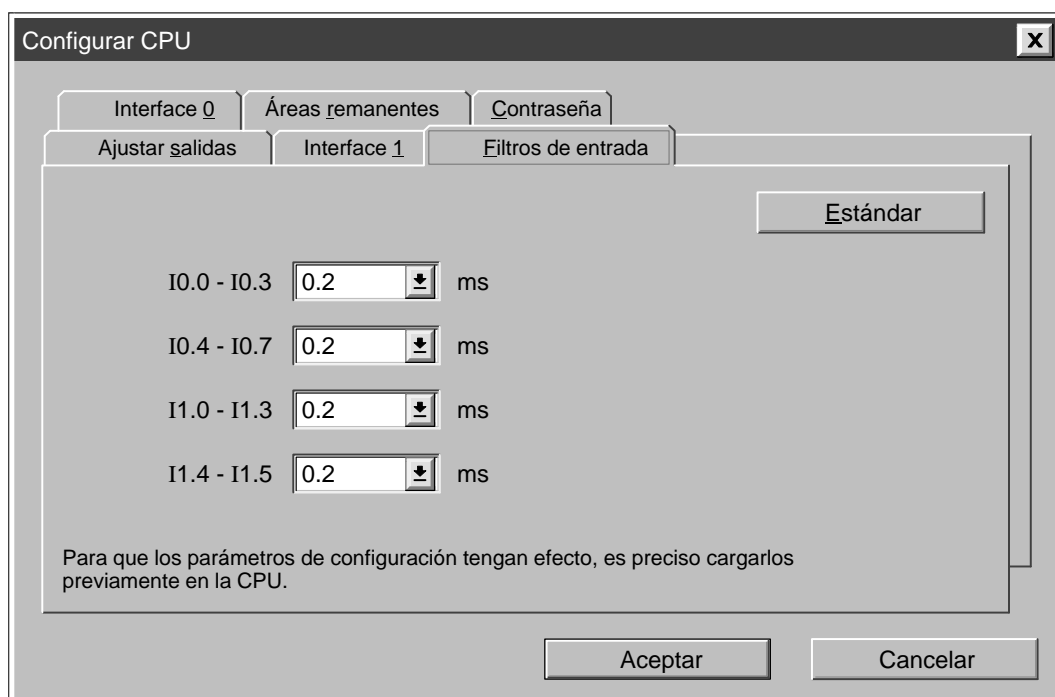


Figura 8-4 Configurar el filtro de entrada para suprimir interferencias

### 8.3 Configurar los estados de señal para 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. Ello se debe a que la actualización de los valores de las salidas analógicas corre a cargo del programa. 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.

Elija el comando de menú **CPU ► Configurar...** y haga clic en la ficha "Ajustar salidas" para acceder al correspondiente cuadro de diálogo (v. fig. 8-5). 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.

El ajuste predeterminado de la CPU es copiar los valores de la tabla en las salidas. En dicha tabla, todas las salidas están desactivadas (puestas a 0) por defecto.

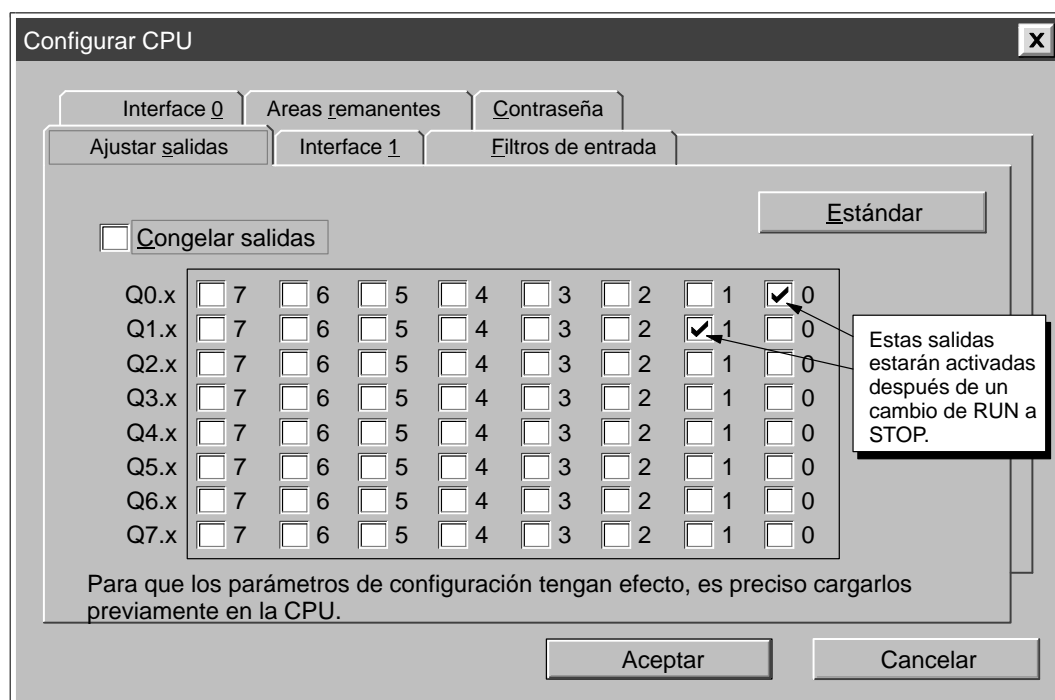


Figura 8-5 Configurar el estado de las salidas

## 8.4 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

Los contadores rápidos cuentan eventos que se ejecutan más deprisa de lo que pueden controlarlos las CPUs S7-200. Según el modelo de CPU, se asiste un contador rápido en el software y hasta dos contadores rápidos en el hardware:

- HSC0 es un contador (software) que permite contar adelante y atrás, asistiendo una sola entrada de reloj. El programa controla el sentido de conteo (adelante o atrás) mediante el bit del control del sentido. La frecuencia máxima de este contador es de 2 kHz.
- HSC1 y HSC2 son contadores universales (hardware) que se pueden configurar según uno de doce modos de operación distintos. La tabla 10-5 muestra los modos de los contadores. La frecuencia máxima de conteo de HSC1 y HSC2 depende de la CPU. Consulte el Anexo A.

Cada contador dispone de entradas que asisten 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 del uso de los contadores rápidos, consulte el apartado 10.5.

### Salidas de impulsos rápidos

Las CPUs S7-200 asisten salidas de impulsos rápidos. En dichas CPUs, 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 250  $\mu$ s a 65.535  $\mu$ 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 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 250  $\mu$ s y 65.535  $\mu$ s, o bien entre 2 ms y 65.535 ms. El tiempo del ancho de impulsos puede estar comprendido entre 0  $\mu$ s y 65.535  $\mu$ 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 las salidas rápidas, consulte el apartado 10.5.

8.5 Potenciómetros analógicos

Las CPUs S7-200 incorporan uno o dos potenciómetros analógicos (dependiendo del tipo de CPU) ubicados debajo de la tapa de protección del módulo. 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 predeterminados, 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. Los potenciómetros analógicos tienen un margen nominal comprendido entre 0 y 255, y un margen garantizado de 10 a 200.

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 8-6 muestra un programa de ejemplo donde se utiliza el potenciómetro analógico.

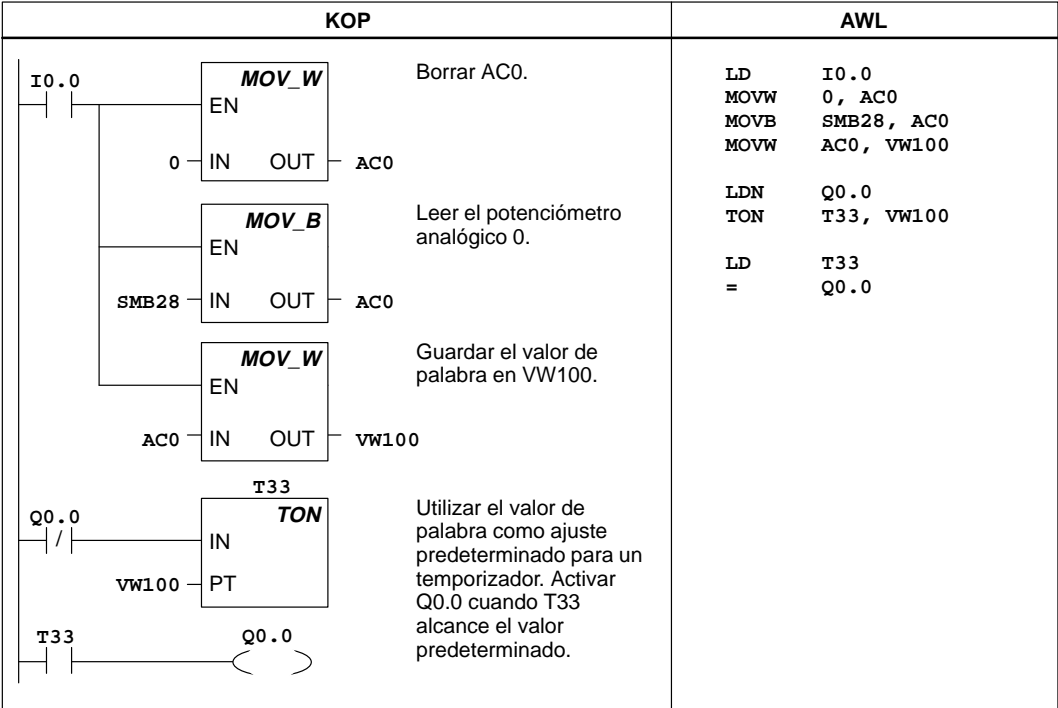


Figura 8-6 Ejemplo de un potenciómetro analógico

# Comunicación en redes con CPUs S7-200

## 9

Las CPUs S7-200 asisten diversos métodos de comunicación, incluyendo los siguientes:

- Comunicación punto a punto (PPI)
- Comunicación en una red multimaestro
- Comunicación en redes de periferia descentralizada (DP)

### Índice del capítulo

Apartado	Descripción	Página
9.1	Protocolos de comunicación de las CPUs S7-200	9-2
9.2	Componentes para la comunicación en redes	9-6
9.3	Utilizar un cable PC/PPI para la comunicación	9-9
9.4	Utilizar una tarjeta MPI o CP para la comunicación	9-13
9.5	Comunicación en redes de periferia descentralizada (DP)	9-15
9.6	Rendimiento de la red	9-28

## 9.1 Protocolos de comunicación de las CPUs S7-200

### Protocolos para la comunicación en redes

Las CPUs S7-200 asisten 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-DP

En la tabla 9-1 se indican más detalles al respecto.

Tabla 9-1 Posibilidades de comunicación de las CPUs S7-200

CPU	Inter- face	Esclavo PPI	Maestro PPI	Esclavo PROFIBUS-DP	Esclavo MPI	Freeport	Velocidad de transferencia
212	0	Sí	No	No	No	Sí	9,6 kbit/s, 19,2 kbit/s
214	0	Sí	Sí	No	No	Sí	9,6 kbit/s, 19,2 kbit/s
215	0	Sí	Sí	No	Sí	Sí	9,6 kbit/s, 19,2 kbit/s
	DP, DPV2	No	No	Sí	Sí	No	9,6 kbits/s, 19,2 kbits/s, 93,75 kbits/s, 187,5 kbits/s, 500 kbits/s, 1 Mbit/s, 1,5 Mbit/s, 3 Mbit/s, 6 Mbit/s, 12 Mbit/s
216	0	Sí	Sí	No	Sí	Sí	9,6 kbit/s, 19,2 kbit/s
	1	Sí	Sí	No	Sí	Sí	9,6 kbit/s, 19,2 kbit/s

Estos protocolos se basan en la intercomunicación de sistemas abiertos (OSI) de la arquitectura de siete capas. Los protocolos PPI, MPI y PROFIBUS-DP 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 dichos 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. Si se usan repetidores, las redes pueden tener una longitud de hasta 9.600 m, dependiendo de la velocidad de transferencia (v. apt. 9.2).

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 asisten 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 para los PCs con STEP 7-Micro/WIN es la dirección "0". El visualizador 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". La dirección predeterminada del interface DP de la CPU 215 es "126".

### 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 D). 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 capítulo 10 se describen 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 dicho 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 comunicados entre sí. 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 en la CPU, cada CPU puede asistir sólo una cantidad limitada de enlaces. La tabla 9-2 muestra la cantidad y el tipo de enlaces MPI asistidos por las diversas CPUs S7-200. Cada CPU reserva algunos de sus enlaces para las unidades de programación SIMATIC y para los paneles de operador. El enlace reservado para una unidad de programación (PG) SIMATIC o para un PC con STEP 7-Micro/WIN garantiza que el usuario pueda conectar siempre por lo menos una PG SIMATIC a la CPU para poder acceder a ésta última. Algunas CPUs reservan también un enlace para un panel de operador. Dichos enlaces reservados no pueden ser utilizados por otros maestros (p.ej. CPUs).

Tabla 9-2 Cantidad y tipo de enlaces lógicos asistidos por las CPUs S7-200

CPU	Inter- face	Cantidad total de enlaces	Cantidad y tipo de enlaces lógicos reservados
215	0	Cuatro	Dos: Uno para la unidad de programación Uno para el panel de operador
	DP, DPV2	Seis	Dos: Uno para la unidad de programación Uno para el panel de operador
216	0	Cuatro	Dos: Uno para la unidad de programación Uno para el panel de operador
	1	Cuatro	Dos: Uno para la unidad de programación Uno para el panel de operador

Las CPUs S7-300 y S7-400 se pueden comunicar con las CPUs S7-200 estableciendo una conexión por los enlaces no reservados de éstas últimas. 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).

**Nota**

El protocolo MPI no se puede utilizar para la comunicación con CPUs S7-200 en las que se haya habilitado el modo maestro PPI. El protocolo MPI clasifica dichas CPUs como estaciones maestras e intenta comunicarse con ellas vía un protocolo maestro/maestro no asistido por ellas.

**Protocolo PROFIBUS-DP**

El protocolo PROFIBUS-DP se ha diseñado para la comunicación rápida con unidades periféricas descentralizadas. Hay numerosos dispositivos PROFIBUS ofrecidos por diversos fabricantes. Dichos dispositivos abarcan desde módulos sencillos de entradas o de salidas hasta controladores de motores y sistemas de automatización.

Por lo general, las redes PROFIBUS-DP incorporan un maestro y varios esclavos. La configuración del maestro le permite reconocer cuáles tipos de esclavos 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.

La CPU 215 dispone de un interface que actúa de puerto PROFIBUS-DP (v. fig. 9-1). Para obtener informaciones más detalladas acerca de la función DP de la CPU 215, consulte el apartado 9.5.

**Protocolos definidos por el usuario (Freeport)**

La comunicación Freeport es un modo de operación con el que el programa de usuario puede controlar el interface 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 interface 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 las marcas SMB30 (interface 0) y SMB130 (interface 1), estando activo únicamente cuando la CPU se encuentre en modo RUN. Cuando la CPU retorna a modo STOP, la comunicación Freeport se detiene y el interface de comunicación vuelve a utilizar el protocolo PPI normal. Para obtener informaciones más detalladas acerca de la utilización del modo Freeport, consulte el apartado 10.14.

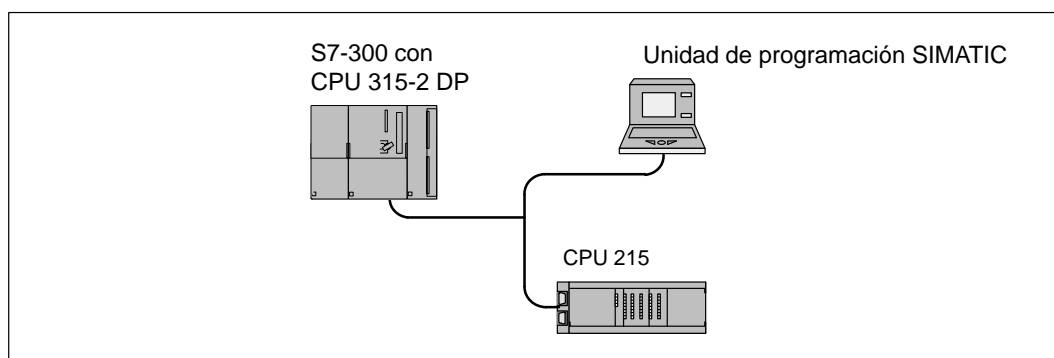


Figura 9-1 CPU 215 conectada a una CPU S7-300 y a una unidad de programación vía interface DP

## 9.2 Componentes para la comunicación en redes

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

### Interface de comunicación

Los interfaces 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 9-2 muestra el conector que ofrece el enlace físico para el interface de comunicación y la tabla 9-3 describe las señales.

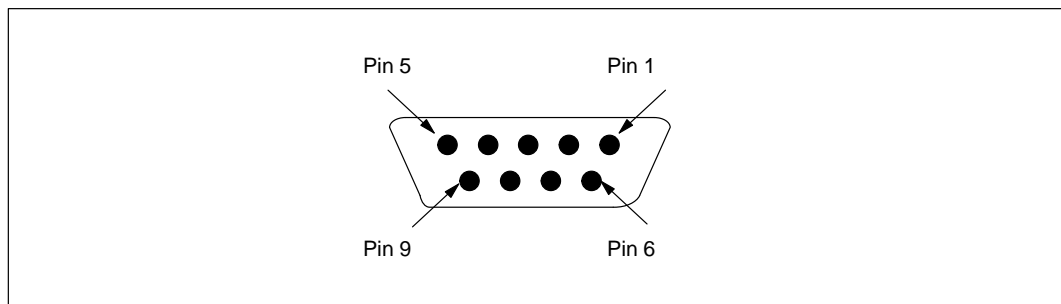


Figura 9-2 Pines del interface de comunicación de la CPU S7-200

Tabla 9-3 Asignación de pines del interface de comunicación de la CPU S7-200

Pin	Denominación PROFIBUS	Interfaces 0 y 1	Interface DP
1	Blindaje	Hilo lógico	Hilo lógico
2	Hilo de retorno 24 V	Hilo lógico	Hilo lógico
3	Señal B RS-485	Señal B RS-485	Señal B RS-485
4	Request-to-Send	Sin conexión	Request-to-send <sup>1</sup>
5	Hilo de retorno 5 V	Hilo lógico	Isolated +5 V Return <sup>2</sup>
6	+5 V	+5 V, 100 $\Omega$ series limit	+5 V, con separación galvánica, 90 mA
7	+24 V	+24 V	+24 V
8	Señal A RS-485	Señal A RS-485	Señal A RS-485
9	No aplicable	Sin conexión	Sin conexión
Carcasa del enchufe	Blindaje	Hilo lógico (CPU 212/214) Tierra (CPU 215/216)	Tierra

<sup>1</sup>  $V_{OH}=3,5$  V, 1,6 mA,  $V_{OL}=0,6$  V, 1,6 mA, señal =  $V_{OH}$  cuando la CPU esté enviando.

<sup>2</sup> Las señales A, B y la petición de enviar (request-to-send) en el interface DP están separadas galvánicamente de la lógica de la CPU, teniendo como tensión de referencia el hilo de retorno de 5 V con separación galvánica.

## 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 interface de programación (v. fig. 9-3). En el Anexo G se indican las correspondientes referencias.

El conector que provee un interface de programación permite añadir a la red una unidad de programación SIMATIC o un panel de operador, sin perturbar ningún enlace existente. Dicho conector transmite todas las señales de la CPU a través del interface 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 interface de programación.



### Cuidado

Al interconectar equipos con potenciales de referencia diferentes, pueden circular corrientes indeseadas por el cable de enlace.

Dichas 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 común 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.

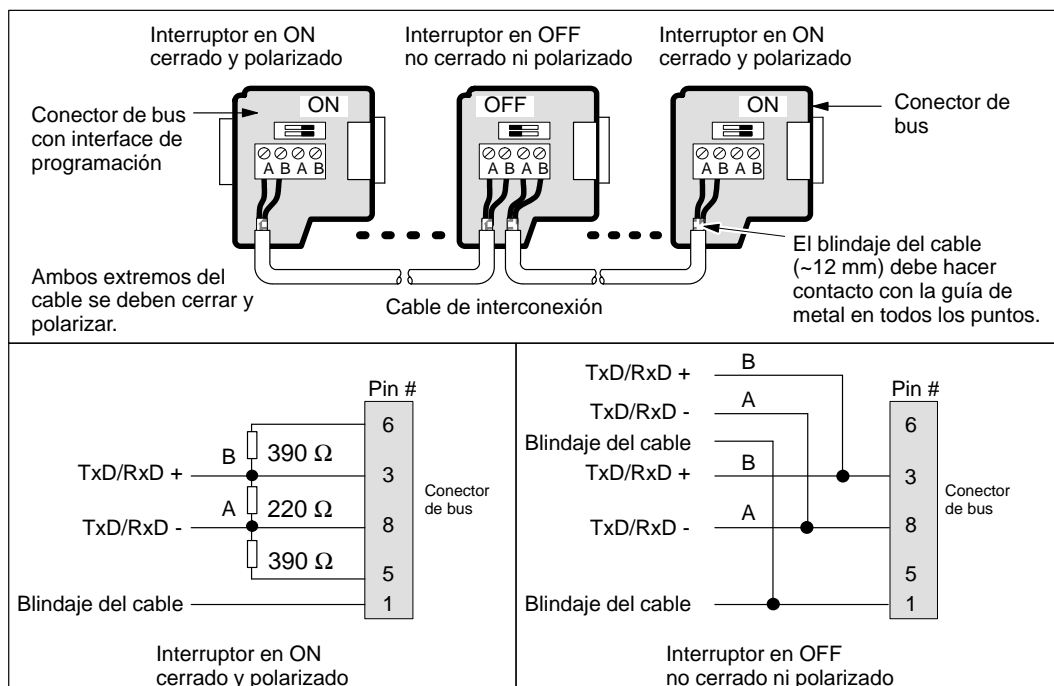


Figura 9-3 Polarizar y cerrar el cable de interconexión

Cable para una red PROFIBUS

La tabla 9-4 muestra los datos técnicos generales de un cable para una red PROFIBUS. En el Anexo G se indica la referencia de Siemens de cables PROFIBUS con las propiedades indicadas.

Tabla 9-4 Datos técnicos generales de un cable para una red PROFIBUS

Propiedades generales	Datos técnicos
Tipo	Apantallado, con par trenzado
Sección transversal del cable	24 AWG (0,22 mm <sup>2</sup> ) 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 9-5 figuran las longitudes máximas de los segmentos para el cable indicado en la tabla 9-4.

Tabla 9-5 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
1,5 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. 9-4). Utilizando repetidores es posible ampliar la longitud total de la red y/o agregar dispositivos a la misma. El protocolo PROFIBUS admite 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 agregar 32 dispositivos adicionales a la red y ampliar ésta última en 1.200 m a una velocidad de transferencia de 9.600 bit/s. En una red se pueden utilizar 9 repetidores como máximo. Cada repetidor permite polarizar y cerrar el segmento de red en cuestión. En el Anexo G se indican las correspondientes referencias.

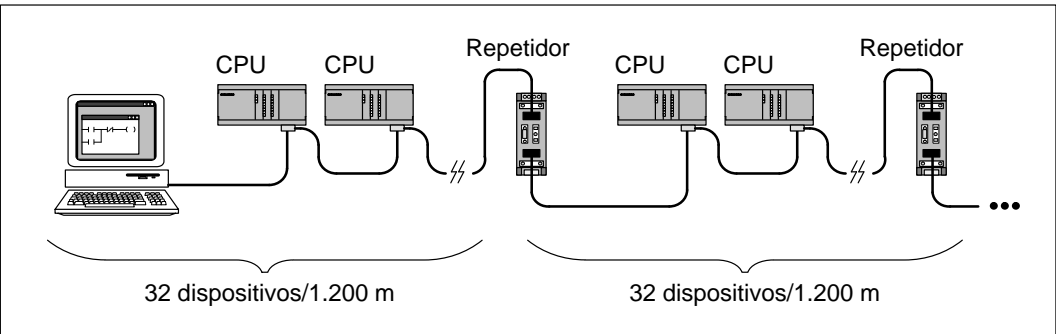


Figura 9-4 Red con repetidores

### 9.3 Utilizar un cable PC/PPI para la comunicación

#### Cable PC/PPI

Por lo general, los puertos de comunicación de un PC son compatibles con el estándar RS-232. Los interfaces de comunicación de la CPU S7-200 utilizan el estándar RS-485 para poder agregar varios dispositivos a una misma red. El cable PC/PPI permite conectar el puerto RS-232 de un PC al interface RS-485 de una CPU S7-200 (v. fig. 9-5). Dicho cable se puede utilizar también para conectar el interface de comunicación de una CPU S7-200 a otros dispositivos compatibles con el estándar RS-232.

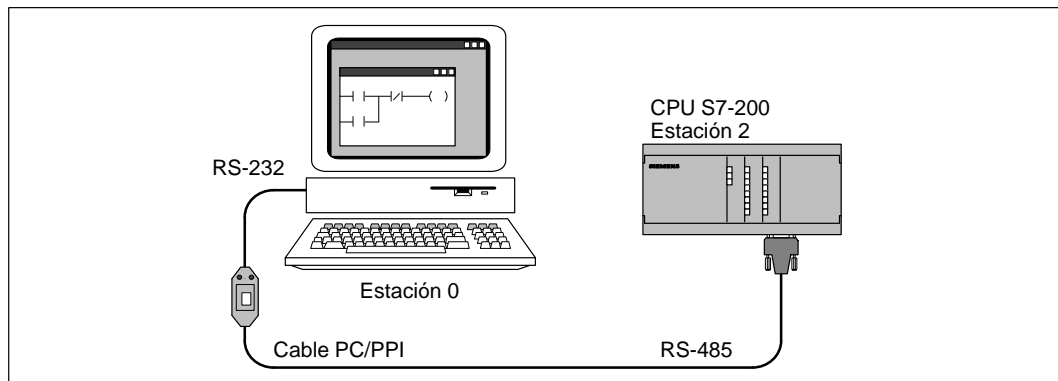


Figura 9-5 Utilizar un cable PC/PPI para la comunicación con una CPU S7-200

#### Utilizar STEP 7-Micro/WIN con un cable PC/PPI

STEP 7-Micro/WIN puede utilizar el cable PC/PPI para comunicarse con una o varias CPUs S7-200 (v. fig. 9-6). Al utilizar STEP 7-Micro/WIN, verifique que la velocidad de transferencia del cable PC/PPI se ajuste a la exigida en la red. STEP 7-Micro/WIN asiste sólo velocidades de transferencia de 9.600 bit/s y 19.200 bit/s.

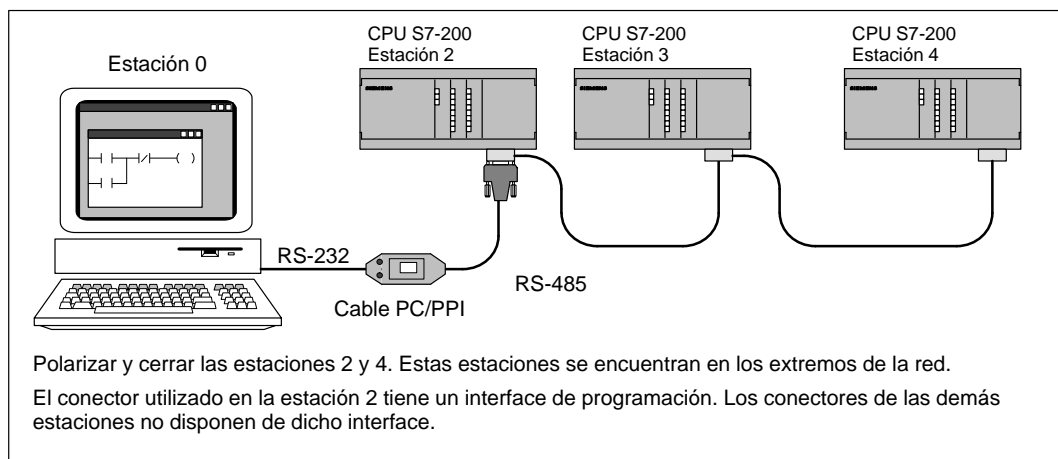


Figura 9-6 Utilizar un cable PC/PPI para la comunicación con una CPU a la vez

Al comunicarse con CPUs S7-200, el ajuste estándar de STEP 7-Micro/WIN es el protocolo PPI multimaestro. Dicho protocolo permite que STEP 7-Micro/WIN coexista con otros maestros (visualizadores de texto TD 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 al que se accede desde el cuadro de diálogo "Ajustar interface PG/PC" (v. apt. 3.3).

STEP 7-Micro/WIN asiste también el protocolo PPI con un maestro único. Al utilizar éste último, STEP 7-Micro/WIN asume 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 vía módems o en redes con muchas interferencias. El protocolo con un maestro único se selecciona borrando la marca de verificación en la casilla "Red multimaestro" en el cuadro de propiedades del cable PC/PPI al que se accede desde el cuadro de diálogo "Ajustar interface PG/PC" (v. apt. 3.3).

Para obtener información acerca de los datos técnicos del cable PC/PPI, consulte el apartado A.40 (en el Anexo A). Su referencia se indica en el Anexo G.

### 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.

El cable PC/PPI asiste 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 9-6 muestra las velocidades de transferencia y las correspondientes posiciones de los interruptores.

Tabla 9-6 Seleccionar la velocidad de transferencia del cable PC/PPI

Velocidad de transferencia	Interruptor DIP (1 = arriba)
38400	0000
19200	0010
9600	0100
4800	0110
2400	1000
1200	1010
600	1100

El interface RS-232 del cable PC/PPI se considera un equipo de comunicación de datos (DCE o Data Communications Equipment). Las únicas señales presentes en dicho interface son: transmitir datos, recibir datos y tierra. La tabla 9-7 muestra los números de los pines y las funciones del interface RS-232 del cable PC/PPI. El cable PC/PPI no utiliza ni envía ninguna de las señales de control del RS-232, tales como Request to Send (RTS) y Clear to Send (CTS).

Tabla 9-7 Cable PC/PPI: definición de pines del interface RS-232

Nº de pin	Función
2	Recibir datos (de DCE)
3	Transmitir datos (de DTE a DCE)
5	Tierra

El cable PC/PPI se encuentra en el modo de transmisión cuando los datos se envían del interface RS-232 al RS-485. En cambio, se encuentra en modo de recepción al estar en vacío, o bien cuando los datos se transmiten del interface 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 en vacío durante el tiempo de inversión del cable. Dicho tiempo depende de la velocidad de transferencia seleccionada con los interruptores DIP del cable (v. tabla 9-8).

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 situaciones siguientes:

- 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 que 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 que 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 interface RS-485 al RS-232.

Tabla 9-8 Tiempo de inversión del cable PC/PPI (cambio de transmisión a recepción)

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

### Utilizar un módem con un cable PC/PPI

El cable PC/PPI se puede utilizar para conectar el interface 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 usa ninguna de dichas señales. Por tanto, al utilizar un módem con un cable PC/PPI, el módem se deberá configurar para que funcione sin dichas señales. Como mínimo, se deberán ignorar las señales RTS y DTR. Consulte el manual del módem para determinar los comandos necesarios para configurarlo.

Al conectar un módem a un cable PC/PPI, se deberá enchufar un adaptador de módem nulo entre el módem y el interface RS-232 del cable PC/PPI. Los módems se consideran equipos de comunicación de datos (DCE o Data Communications Equipment). El interface RS-232 del cable PC/PPI 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. El adaptador de módem nulo invierte los canales de transmisión y recepción. La figura 9-7 muestra una instalación típica y la asignación de pines de un adaptador de módem nulo.

STEP 7-Micro/WIN sólo se puede utilizar con un módem dúplex que asista cadenas de 11 caracteres. Consulte el apartado 3.3 para obtener informaciones más detalladas acerca de la utilización de STEP 7-Micro/WIN con un módem. Si se utiliza un protocolo Freeport definido por el usuario, se puede emplear cualquier módem que asista el tamaño de la cadena de caracteres del protocolo.

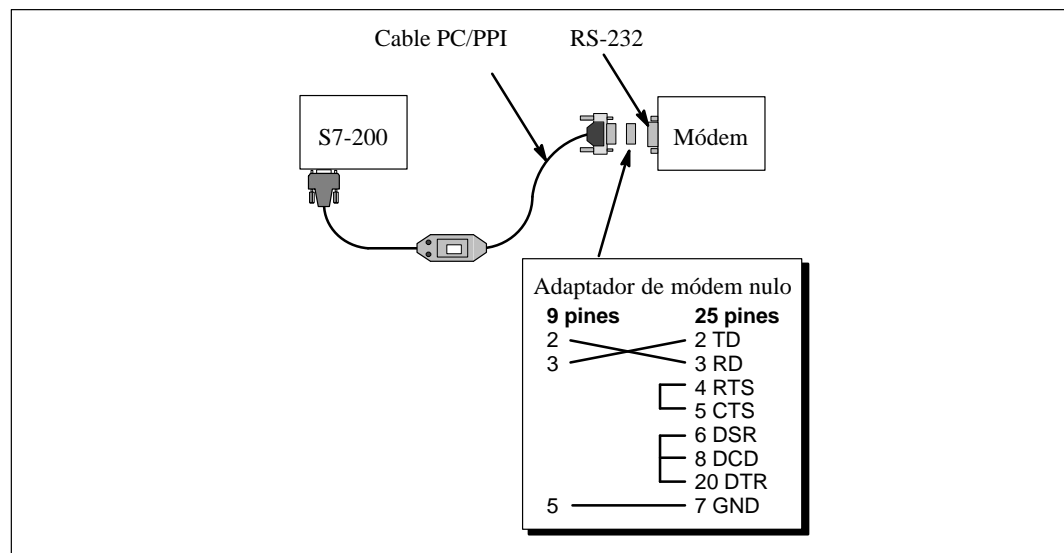


Figura 9-7 Módem con adaptador de módem nulo

## 9.4 Utilizar una tarjeta MPI o CP para la comunicación

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. Dichas 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 varias velocidades de transferencia (v. tabla 9-9).

Tabla 9-9 Tarjetas de conexión a una red multimaestro

Nombre	Tipo	Sistemas operativos asistidos	Comentarios
MPI	Short AT ISA o integrado en PG	MS-DOS Windows 3.1x	Asiste el protocolo PPI, 9.600 bit/s y 19.200 bit/s
		Windows 95 Windows NT	Asiste los protocolos PPI, <sup>1</sup> MPI y PROFIBUS-DP, 9.600 bit/s a 1,5 Mbit/s para PCs y PGs
CP 5411	Short AT ISA	Windows 95 Windows NT	Asiste los protocolos PPI, <sup>1</sup> MPI y PROFIBUS-DP, 9.600 bit/s a 12 Mbit/s para PCs y PGs
CP 5511	PCMCIA, tipo II Hardware "plug & play"	Windows 95 Windows NT	Asiste los protocolos PPI, <sup>1</sup> MPI y PROFIBUS-DP, 9.600 bit/s a 12 Mbit/s para PCs portátiles
CP 5611	Short PCI Hardware "plug & play"	Windows 95 Windows NT	Asiste los protocolos PPI, <sup>1</sup> MPI y PROFIBUS-DP, 9.600 bit/s a 12 Mbit/s para PCs

<sup>1</sup> sólo a 9.600 bit/s ó 19.200 bit/s

La tarjeta y el protocolo en cuestión se ajustan en el cuadro de diálogo "Ajustar interface PG/PC" accesible desde STEP 7-Micro/WIN, o bien en el Panel de control de Windows (v. apt. 3.3).

En Windows 95 ó Windows NT se puede seleccionar cualquier protocolo (PPI, MPI o PROFIBUS) a utilizar con cualquiera de las tarjetas disponibles. Como regla general, se deberá seleccionar el protocolo PPI a 9.600 bit/s ó a 19.200 bit/s para la comunicación con las CPUs S7-200. La única excepción la constituye la CPU 215. Al comunicarse con dicha CPU a través del interface DP, es preciso seleccionar el protocolo MPI. El interface DP de la CPU 215 asiste velocidades de transferencia comprendidas entre 9.600 bit/s y 12 Mbit/s. Dicho interface determina automáticamente la velocidad de transferencia del maestro (CP o tarjeta MPI) y se autosincroniza para utilizar la misma velocidad.

Cada tarjeta incorpora un interface 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 interface RS-485 de la tarjeta y el otro, al conector del interface de programación en la red (v. fig. 9-8). 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*.

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

Una tarjeta de interface multipunto (tarjeta MPI) o un procesador de comunicaciones (CP) permite crear numerosas configuraciones. Una estación donde se ejecute el software de programación STEP 7-Micro/WIN (PC o PG con tarjeta MPI o CP) se puede conectar a una red que incorpore 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 (TD 200). La figura 9-8 muestra una configuración con dos visualizadores de textos TD 200 que se han incorporado a la red.

Esta configuración ofrece las siguientes posibilidades de comunicación:

- STEP 7-Micro/WIN (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 214 (estaciones 3 y 4, respectivamente).
- Ambas CPUs 214 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 212) y 4 (CPU 214).
- La estación 4 puede leer datos de y escribir datos en las estaciones 2 (CPU 212) y 3 (CPU 214).

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.

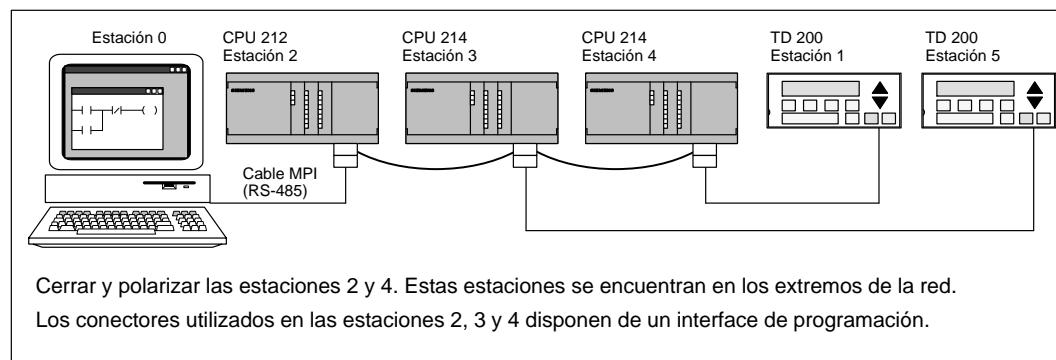


Figura 9-8 Utilizar una tarjeta MPI o un CP para la comunicación con CPUs S7-200

## 9.5 Comunicación en redes de periferia descentralizada (DP)

### Estándar PROFIBUS-DP

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 la CPU 215 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 la CPU 215 en calidad de esclavo DP

La CPU 215 se puede conectar a una red PROFIBUS-DP para actuar allí de esclavo DP. El interface 1 de la CPU 215 (que lleva el letrero DP) es el puerto DP. Éste último puede funcionar a una velocidad de transferencia cualquiera comprendida entre 9.600 bit/s y 12 Mbit/s. En calidad de esclavo DP, la CPU 215 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, la CPU 215 no se limita a transferir datos de E/S. Antes bien, utiliza un bloque de la memoria de variables para transferir los datos al y del maestro. 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 215. De igual manera, los datos recibidos del maestro se almacenan en la memoria de variables de la CPU 215, pudiéndose transferir de allí a otras áreas de datos.

El interface DP de la CPU 215 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 9-9 muestra una red PROFIBUS con una CPU 215. 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 215 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 215 mediante las operaciones XGET contenidas en el programa de usuario de la CPU S7-400.

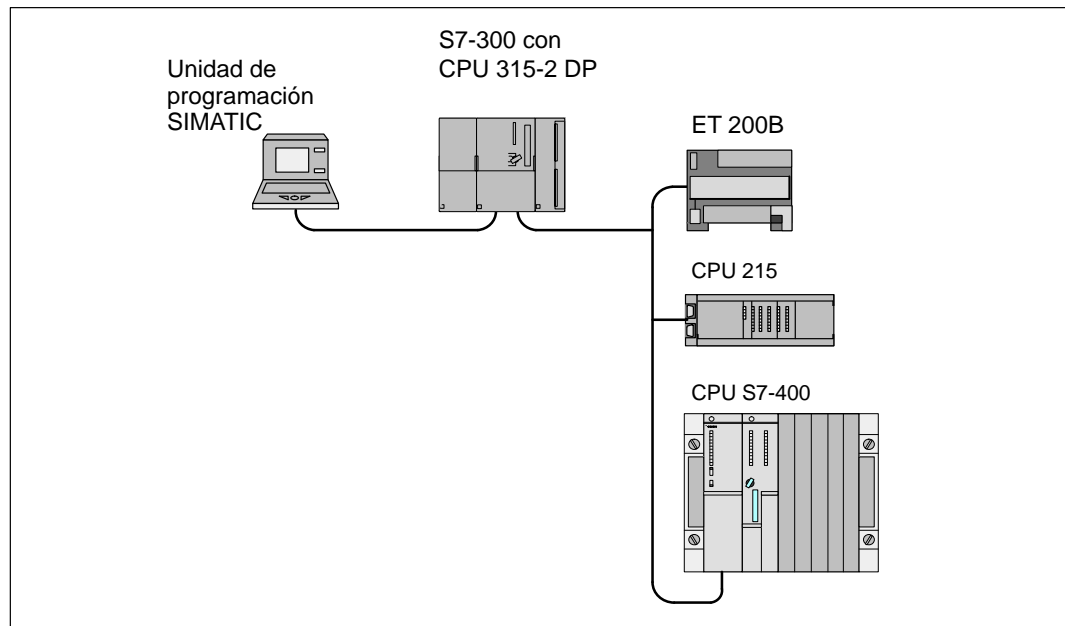


Figura 9-9 CPU 215 en una red PROFIBUS

## Configuración

La dirección de estación del interface DP es el único parámetro que se debe ajustar en la CPU 215 para que ésta actúe de esclavo PROFIBUS. Dicha dirección debe coincidir con la indicada en la configuración del maestro. STEP 7-Micro/WIN se puede utilizar para modificar la dirección del interface DP en la configuración de la CPU, cargándose luego la nueva configuración en la CPU 215.

La dirección del interface DP de la CPU 215 también se puede ajustar mediante un dispositivo de configuración DP conectado al interface DP. La dirección de éste último sólo se podrá parametrizar con uno de los dispositivos DP si en la configuración de la CPU indicada en STEP 7-Micro/WIN se ha ajustado la dirección estándar 126 para el interface DP. La dirección del interface DP configurada en STEP 7-Micro/WIN tiene prioridad sobre una dirección ajustada mediante un dispositivo de configuración DP.

---

### Nota

Para restablecer la dirección estándar del interface DP tras haberla modificado con un dispositivo de configuración DP:

1. Utilizando STEP 7-Micro/WIN, cambie la dirección del interface DP en la configuración de la CPU, ajustando un valor no utilizado hasta ahora (que no sea 126).
  2. Cargue dicha configuración en la CPU 215.
  3. Utilizando STEP 7-Micro/WIN nuevamente, ajuste la dirección estándar (126) del interface DP en la configuración de la CPU.
  4. Cargue dicha configuración en la CPU 215.
- 

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. 9-10).

El maestro DP puede configurar la CPU 215 para que ésta reciba datos de salida del maestro y retorne datos de entrada al mismo. Los búfer de salida y de entrada se almacenan en la memoria de variables (memoria V) de la CPU 215. 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 la CPU 215. Asimismo, se define la configuración de E/S como la cantidad de datos de salida a escribir en la CPU 215 y de datos de entrada a leer de la misma. La CPU 215 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 la CPU 215.

La figura 9-10 muestra un ejemplo de la memoria V en una CPU 215, 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 215, 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*.

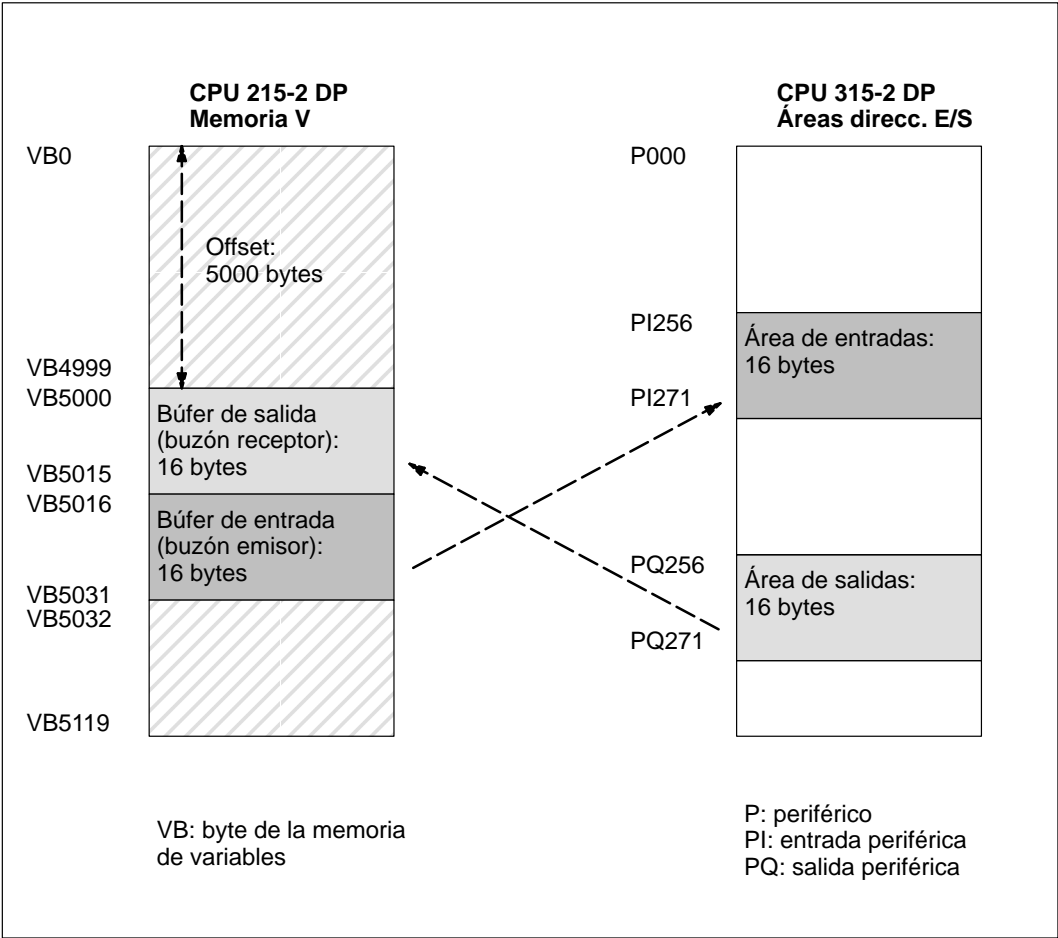


Figura 9-10 Ejemplo: Memoria V de la CPU 215 y área de direcciones de E/S de un maestro PROFIBUS-DP

En la tabla 9-10 figuran las configuraciones asistidas por la CPU 215.

Tabla 9-10 Configuraciones de E/S asistidas por la CPU 215

Configuración	Tamaño del búfer de entrada (datos para el maestro)	Tamaño del búfer de salida (datos del maestro)	Coherencia de datos
1	1 palabra	1 palabra	Coherencia de palabras
2 (estándar)	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 215, 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 215. 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 ET 200 (COM PROFIBUS).
- Para los maestros SIMATIC S7, utilice el software de programación STEP 7.
- Para los maestros SIMATIC 505, utilice COM ET 200 (COM PROFIBUS) y TISOFT2.

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*. (En el Anexo G se indica el número de referencia de dicho manual).

### 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 9-11 muestra los diversos tipos de coherencias.

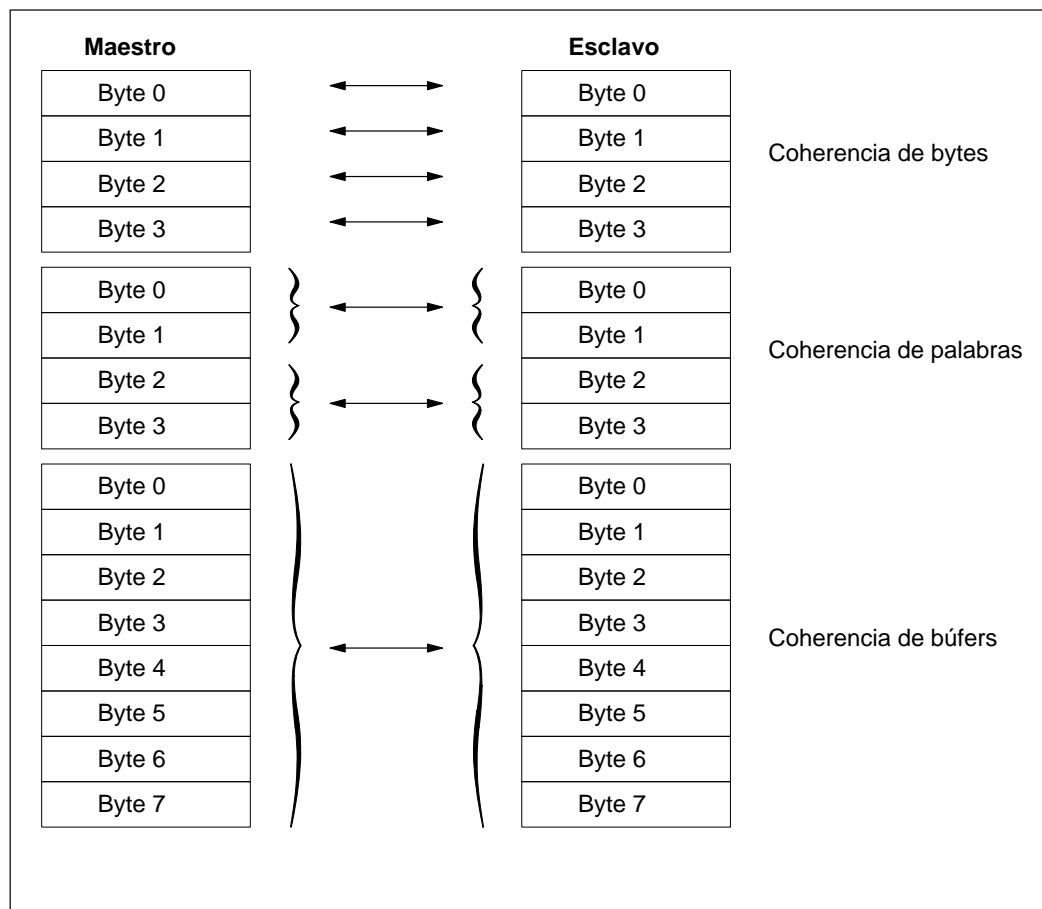


Figura 9-11 Coherencia de bytes, palabras y búfers

### Consideraciones relativas al programa de usuario

Una vez que un maestro DP haya configurado correctamente la CPU 215, ambos equipos pasan al modo de intercambio de datos. En dicho modo, el maestro escribe datos de salida en la CPU 215 y ésta responde con datos de entrada. 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 215. El programa de usuario contenido en dicha CPU debe transferir los datos de salida que envíe 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 envíe 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 un área interna 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 SMB110 y SMB115 proporcionan informaciones de estado sobre la CPU 215 que actúa de esclavo DP. Dichas marcas tienen valores estándar al no haberse establecido la comunicación DP con un maestro. Una vez que un maestro haya escrito los parámetros y la configuración de E/S en la CPU 215, dichas marcas adoptan la configuración ajustada por el maestro DP. Es preciso comprobar SMB110 para verificar que la CPU 215 se encuentre actualmente en modo de intercambio de datos con el maestro antes de utilizar las informaciones contenidas en los bytes de marcas especiales SMB111 a SMB115 (v. tabla 9-11).

#### Nota

Los tamaños o las direcciones de los búfers de E/S de la CPU 215 no se pueden configurar escribiendo en las marcas especiales SMB112 a SMB115. El maestro DP es el único que puede configurar la CPU 215 para la comunicación DP.

Tabla 9-11 Información sobre el estado DP

Byte de marcas	Descripción									
SMB110	<div><div><div>MSB</div><div>LSB</div></div><div><div>7</div><div>0</div></div><table><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>s</td><td>s</td></tr></table></div> <div>Interface 1: Byte de estado del protocolo estándar DP</div> <div>ss: Byte de estado del protocolo estándar DP 00 = Comunicación DP no iniciada desde el arranque 01 = Error de configuración o de parametrización 10 = Intercambio de datos activado 11 = Intercambio de datos desactivado</div> <div>SM111 a SM115 se actualizan cada vez que la CPU acepta datos de configuración y/o de parametrización. Dichas direcciones se actualizan aun si se detecta un error de configuración y/o de parametrización. Las direcciones se borran en cada arranque de la CPU.</div>	0	0	0	0	0	0	s	s	
0	0	0	0	0	0	s	s			
SMB111	Este byte indica la dirección del maestro del esclavo (0 a 126).									
SMB112	Estos bytes indican la dirección en la memoria V del búfer de salida (desplazamiento de VB0).									
SMB113	SMB112 es el byte más significativo (MSB) y SMB113 es el byte menos significativo (LSB).									
SMB114	Este byte indica el número de bytes de los datos de salida.									
SMB115	Este byte indica el número de bytes de los datos de entrada.									

**Indicadores de estado (LEDs) para la comunicación DP**

La CPU 215 dispone de un LED en el panel frontal que indica el estado del interface DP:

- Después del arranque de la CPU, el LED DP permanecerá apagado mientras que no se intente establecer la comunicación DP.
- Una vez iniciada correctamente la comunicación DP (la CPU 215 pasa entonces a modo de intercambio de datos con el maestro), el LED DP se encenderá en verde y permanecerá encendido hasta que se desactive el modo de intercambio de datos.
- Si se interrumpe la comunicación, lo que obliga a la CPU 215 a salir del modo de intercambio de datos, el LED DP se encenderá en rojo. Dicho estado persistirá hasta que la CPU 215 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 la CPU 215, el LED DP parpadea en rojo.

La tabla 9-12 resume los diversos estados del LED DP.

Tabla 9-12 Estados del LED DP y su significado

Estado del LED	Descripción del estado
OFF	Desde el último arranque no se ha intentado establecer la comunicación DP
Parpadeante (rojo)	Error de parametrización o configuración, la CPU no está en modo de intercambio de datos
Verde	Modo de intercambio de datos activado
Rojo	Modo de intercambio de datos desactivado

**Archivo de datos maestros de los dispositivos (archivo GSD)**

Los diversos dispositivos PROFIBUS tienen diferentes propiedades de rendimiento. Dichas 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. Dichos archivos GSD 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 ET 200 (llamado ahora COM PROFIBUS) o del software STEP 7 incluyen archivos de configuración para la CPU 215. Si su versión del software no incluye un archivo de configuración para la CPU 215, puede utilizar un módem para acceder al PROFIBUS Bulletin Board Service (BBS) y copiar de allí el archivo GSD para la CPU 215. En el BBS, introduzca en los indicadores los datos que se le soliciten para acceder a la base de datos de la CPU 215 y copie el archivo. Dicho archivo se descomprime automáticamente, conteniendo los archivos necesarios para las redes PROFIBUS. Para acceder al BBS, llame a los siguientes números de teléfono:

- En Norteamérica y Suramérica: (423) 461-2751  
Nombre del archivo a copiar: S7215.EXE
- En Europa: (49) (911) 73 79 72  
Nombre del archivo a copiar: W32150AX.200

Alternativamente, puede obtener el archivo GSD más reciente a través de Internet. La dirección es: [www.profibus.com](http://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 la CPU 215**

La tabla 9-13 contiene un listado del archivo GSD actual (archivo de datos maestros de los dispositivos) para la CPU 215.

Tabla 9-13 Archivo GSD para maestros no SIMATIC

```

=====
; GSD-Data for the S7-215 DP slave with SPC3
; MLFB : 6ES7 215-2.D00-0XB0
; Date : 05-Oct-1996/release 14-March-97/09/29/97 (45,45)
; Version: 1.2 GSD
; Model-Name, Freeze_Mode_supp, Sync_mode_supp, 45,45k
; File : SIE_2150
=====
#Profibus_DP
; Unit-Definition-List:
GSD_Revision=1
Vendor_Name="Siemens"
Model_Name="CPU 215-2 DP"
Revision="REV 1.00"
Ident_Number=0x2150
Protocol_Ident=0
Station_Type=0
Hardware_Release="A1.0"
Software_Release="Z1.0"
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
Implementation_Type="SPC3"
Bitmap_Device="S7_2150"
;
; Slave-Specification:
OrderNumber="6ES7 215-2.D00-0XB0"
Periphery="SIMATIC S5"
;
Freeze_Mode_supp=1
Sync_Mode_supp=1
Set_Slave_Add_supp=1
Min_Slave_Intervall=1

```

Tabla 9-13 Archivo GSD para maestros no SIMATIC, continuación

```

Max_Diag_Data_Len=6
Slave_Family=3@TdF@SIMATIC
;
; UserPrmData-Definition
ExtUserPrmData=1 "I/O Offset in the V-memory"
Unsigned16 0 0-5119
EndExtUserPrmData
; UserPrmData: Length and Preset:
User_Prms_Data_Len=3
User_Prms_Data= 0,0,0
Ext_User_Prms_Data_Ref(1)=1
;
Modular_Station=1
Max_Module=1
Max_Input_Len=64
Max_Output_Len=64
Max_Data_Len=128
;
; Module-Definitions:
;
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 215 esclava**

La tabla 9-14 contiene un listado de un programa de ejemplo AWL para una CPU 215 que utiliza la información del interface DP en el área de marcas especiales. La figura 9-12 muestra ese mismo programa en KOP. El programa averigua la dirección de los búfers DP mediante SMB112 y lee los tamaños de los mismos de SMB114 y SMB115. Dichas 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 215. De forma similar, los datos contenidos en la imagen del proceso de las entradas de la CPU 215 se copian en el búfer de entrada DP.

Tabla 9-14 Programa de ejemplo AWL para la comunicación DP con una CPU 215 esclava

```

Listado del programa

//Los datos de configuración DP en el área de marcas especiales indican cómo el
//maestro ha configurado el esclavo DP. El programa utiliza los datos siguientes:
//  SMB110      Estado DP
//  SMB111      Dirección del maestro
//  SMB112      Offset en la memoria de variables para los datos de salida
//  SMB114      Número de bytes de salida
//  SMB115      Número de bytes de entrada
//  VD1000      Puntero de datos de salida
//  VD1004      Puntero de datos de entrada

NETWORK
LD      SM0.0                      //En cada ciclo:
MOVD    &VB0, VD1000              //Crear un puntero a los datos de salida,
MOVW    SMW112, VW1002            //añadir al offset de los datos de salida,
MOVD    &VB0, VD1004              //crear un puntero a los datos de entrada,
MOVW    SMW112, VW1006            //añadir al offset de los datos de salida,
MOVW    +0, AC0                   //borrar el acumulador,
MOVB    SMB114, AC0               //cargar el número de bytes de salida.
+I      AC0, VW1006               //Puntero de desplazamiento

NETWORK
LDB>=   SMB114, 9                  //Si el número de bytes de salida > 8,
MOVB    8, VB1008                 //contaje de salida = 8
NOT      //En caso contrario
MOVB    SMB114, VB1008            //contaje de salida = número de bytes de salida.

NETWORK
LDB>=   SMB115, 9                  //Si el número de bytes de entrada > 8,
MOVB    8, VB1009                 //contaje de entrada = 8
NOT      //En caso contrario
MOVB    SMB115, VB1009           //contaje de entrada = número de bytes de entrada.

NETWORK
LD      SM0.0                      //En cada ciclo:
BMB     *VD1000, QB0, VB1008       //Copiar las salidas DP en las salidas,
BMB     IB0, *VD1004, VB1009       //copiar las entradas en las entradas DP.

NETWORK
MEND

```

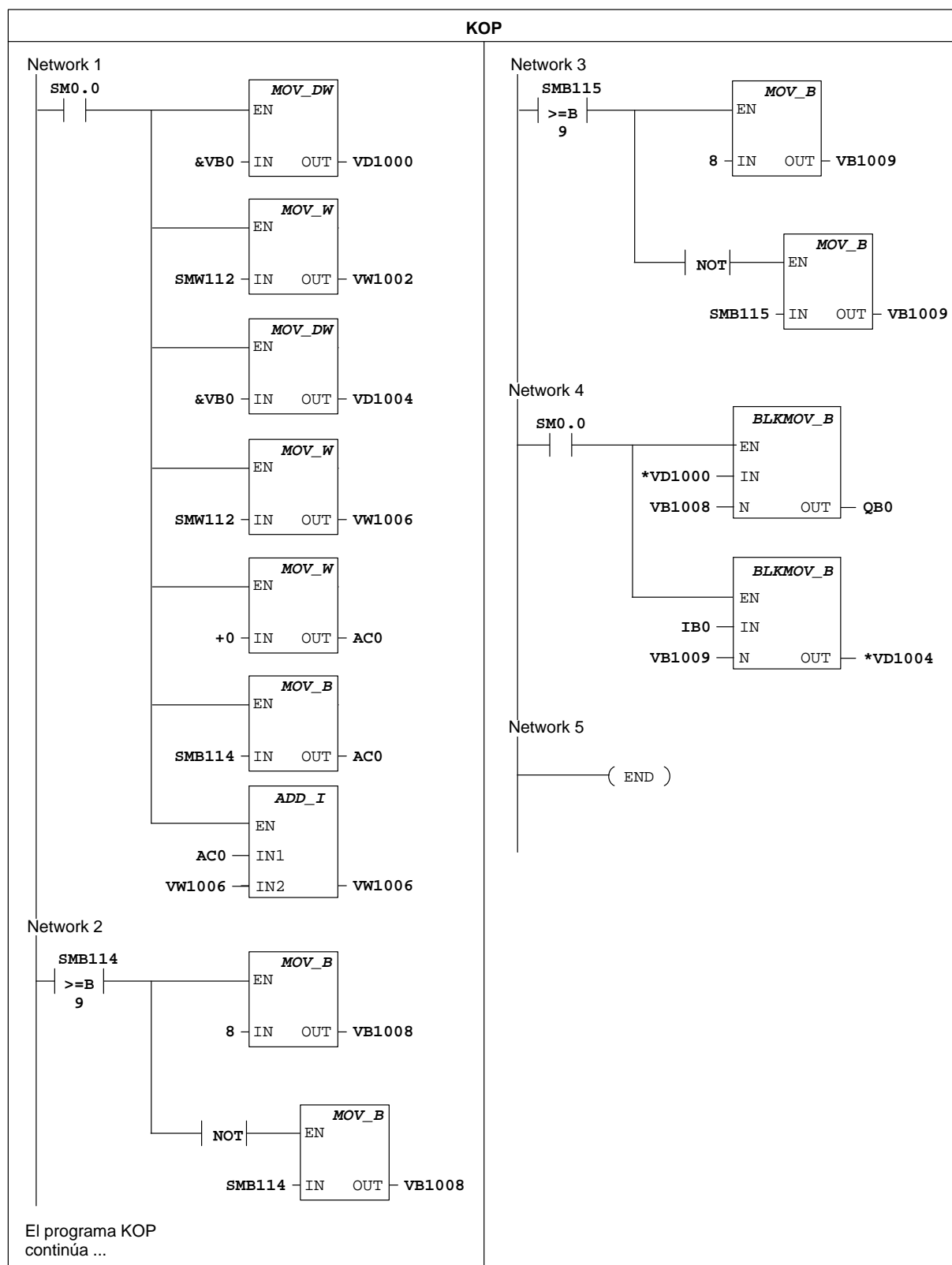


Figura 9-12 Programa de ejemplo KOP para la comunicación DP con una CPU 215 esclava

## 9.6 Rendimiento de la red

### Limitaciones

El rendimiento de la red depende de numerosas variables complejas. No obstante, dos factores básicos lo determinan: la velocidad de transferencia y el número de estaciones conectadas a la red.

### Ejemplo de una red con token passing

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 conforman. El ejemplo de la figura 9-13 muestra el funcionamiento de una red multimaestro.

La red de la figura 9-13 comprende cuatro CPUs S7-200, teniendo cada una de ellas su propio TD 200. Dos CPUs 214 recopilan datos de las demás CPUs.

#### Nota

El ejemplo indicado se basa en la configuración que muestra la figura 9-13. Dicha configuración incluye visualizadores de textos TD 200. Las CPUs 214 utilizan operaciones NETR y NETW. Las fórmulas para calcular el tiempo de posesión y de rotación del testigo que muestra la figura 9-14 se basan también en dicha configuración.

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

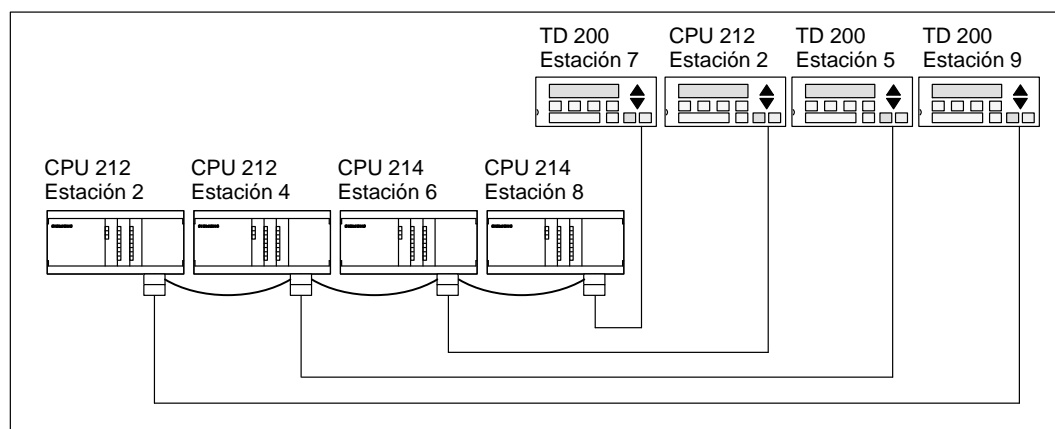


Figura 9-13 Ejemplo de una red con token passing

En esta configuración, un TD 200 (estación 3) se comunica con una CPU 212 (estación 2), otro TD 200 (estación 5) se comunica con la otra CPU 212 (estación 4), etc. Además, una CPU 214 (estación 6) envía mensajes a las estaciones 2, 4 y 8, y la otra CPU 214 (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 214), así como dos estaciones esclavas (las dos CPUs 212).

## 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 bits/s. Si aumenta el número de bytes de datos a los que se debe acceder por mensaje o si se incorporan más estaciones, se incrementará el tiempo de rotación del testigo.

## Tiempo de rotación del testigo (token)

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 las CPUs 214, 215 ó 216. (Consulte la descripción de dichas operaciones en el capítulo 10). Si envía mensajes utilizando las operaciones NETR y NETW, puede utilizar la fórmula que muestra la figura 9-14 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 +  $n$  datos) \* 11 bits/carácter \* 1/velocidad de transferencia

Tiempo de rotación del testigo ( $T_{rot}$ ) =  $T_{pos}$  del maestro 1 +  $T_{pos}$  del maestro 2 + ... +  $T_{pos}$  del maestro  $m$

donde  $n$  es el número de caracteres de datos (bytes)  
y  $m$  es 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 9-14 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 9-15 y 9-16 muestran el tiempo de rotación del testigo en función del número de estaciones y del volumen de datos a transferir a una velocidad de 19,2 kbit/s y 9,6 kbit/s, respectivamente. Dichos tiempos son válidos al utilizarse las operaciones Leer de la red (NETR) y Escribir en la red (NETW) con una CPU 214, 215 ó 216.

Tabla 9-15 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 9-16 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

### Optimizar el rendimiento de redes

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 asistida 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 incrementa el tiempo de procesamiento en la red. Por tanto, dicho tiempo se acortará cuanto menor sea el número de maestros.

Los siguientes factores afectan también 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 intervalos entre las mismas. Al haber un intervalo (GAP) entre las direcciones de los maestros, éstos comprueban continuamente las direcciones del GAP para averiguar si hay otro maestro que desee conectarse online. Dicha comprobación aumenta el tiempo de procesamiento de la red. Si no existe un intervalo 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 se afecte 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 intervalos entre las direcciones de los maestros.

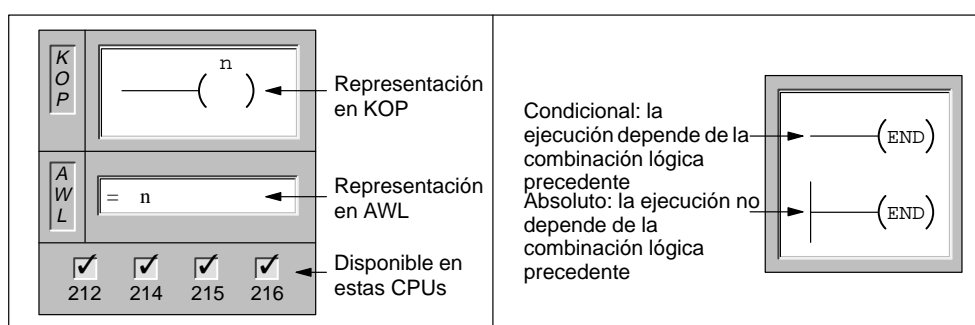
Las CPUs S7-200 se pueden configurar para que comprueben sólo periódicamente si hay intervalos entre las direcciones. A tal efecto, en STEP 7-Micro/WIN se ajusta el factor de actualización GAP cuando se configure el correspondiente interface de la CPU. El factor de actualización GAP le indica a la CPU la frecuencia con la que debe comprobar el intervalo de direcciones para determinar si hay otros maestros. Si se elige "1" como factor de actualización GAP, la CPU comprobará el intervalo de direcciones cada vez que tenga el testigo en su poder. Si se elige "2", la CPU comprobará el intervalo 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 intervalos entre las direcciones de los maestros. Si no existen intervalos, 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 menos 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 dicho valor se limita el intervalo de direcciones que el último maestro (la dirección más alta) debe comprobar en la red. Al limitarse el tamaño del intervalo 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 en 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 al configurar el interface de la CPU en cuestión.

Como regla general, se deberá ajustar en todos los maestros un mismo valor para la dirección de estación más alta. Dicha dirección debería ser mayor que 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 "126".

## Juego de operaciones

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



### Índice del capítulo

Apartado	Descripción	Página
10.1	Márgenes válidos para las CPUs S7-200	10-2
10.2	Operaciones con contactos	10-4
10.3	Operaciones de comparación	10-7
10.4	Operaciones con salidas	10-10
10.5	Operaciones con temporizadores, contadores, contadores rápidos, reloj de tiempo real y salida de impulsos	10-13
10.6	Operaciones aritméticas y de regulación PID	10-50
10.7	Operaciones para incrementar y decrementar	10-66
10.8	Operaciones de transferencia, inicializar memoria y tabla	10-68
10.9	Operaciones de desplazamiento y rotación	10-78
10.10	Operaciones de control del programa	10-84
10.11	Operaciones lógicas de pilas	10-99
10.12	Operaciones lógicas	10-102
10.13	Operaciones de conversión	10-108
10.14	Operaciones de interrupción y comunicación	10-114

## 10.1 Márgenes válidos para las CPUs S7-200

Tabla 10-1 Resumen de las áreas de memoria y funciones de las CPUs S7-200

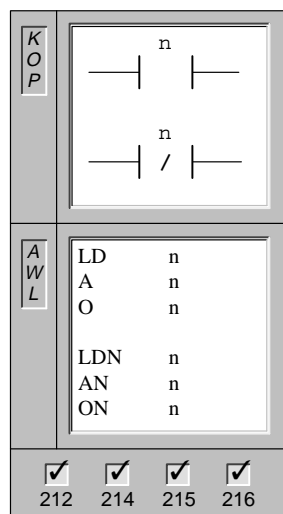
Descripción	CPU 212	CPU 214	CPU 215	CPU 216
Tamaño del programa de usuario	512 palabras	2K palabras	4K palabras	4K palabras
Tamaño de los datos de usuario	512 palabras	2K palabras	2.5K palabras	2.5K palabras
Imagen del proceso de las entradas	I0.0 a I7.7	I0.0 a I7.7	I0.0 a I7.7	I0.0 a I7.7
Imagen del proceso de las salidas	Q0.0 a Q7.7	Q0.0 a Q7.7	Q0.0 a Q7.7	Q0.0 a Q7.7
Entradas analógicas (sólo lectura)	AIW0 a AIW30	AIW0 a AIW30	AIW0 a AIW30	AIW0 a AIW30
Salidas analógicas (sólo escritura)	AQW0 a AQW30	AQW0 a AQW30	AQW0 a AQW30	AQW0 a AQW30
Memoria de variables (V) Área no volátil (máx).	V0.0 a V1023.7 V0.0 a V199.7	V0.0 a V4095.7 V0.0 a V1023.7	V0.0 a V5119.7 V0.0 a V5119.7	V0.0 a V5119.7 V0.0 a V5119.7
Área de marcas (M) Área no volátil (máx).	M0.0 a M15.7 MB0 a MB13	M0.0 a M31.7 MB0 a MB13	M0.0 a M31.7 MB0 a MB13	M0.0 a M31.7 MB0 a MB13
Marcas especiales (SM) Sólo lectura	SM0.0 a SM45.7 SM0.0 a SM29.7	SM0.0 a SM85.7 SM0.0 a SM29.7	SM0.0 a SM194.7 SM0.0 a SM29.7	SM0.0 a SM194.7 SM0.0 a SM29.7
Temporizadores	64 (T0 a T63)	128 (T0 a T127)	256 (T0 a T255)	256 (T0 a T255)
Retardo a la conexión memorizado 1 ms	T0	T0, T64	T0, T64	T0, T64
Retardo a la conexión memorizado 10 ms	T1 a T4	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	T5 a T31, T69 a T95	T5 a T31, T69 a T95	T5 a T31, T69 a T95
Retardo a la conexión 1 ms	T32	T32, T96	T32, T96	T32, T96
Retardo a la conexión 10 ms	T33 a T36	T33 a T36, T97 a T100	T33 a T36, T97 a T100	T33 a T36, T97 a T100
Retardo a la conexión 100 ms	T37 a T63	T37 a T63, T101 a T127	T37 a T63, T101 a T255	T37 a T63, T101 a T255
Contadores	C0 a C63	C0 a C127	C0 a C255	C0 a C255
Contadores rápidos	HC0	HC0 a HC2	HC0 a HC2	HC0 a HC2
Relés de control secuencial	S0.0 a S7.7	S0.0 a S15.7	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 63	0 a 255	0 a 255	0 a 255
Llamadas a subrutinas	0 a 15	0 a 63	0 a 63	0 a 63
Rutinas de interrupción	0 a 31	0 a 127	0 a 127	0 a 127
Eventos de interrupción	0, 1, 8 a 10, 12	0 a 20	0 a 23	0 a 26
Lazos PID	No asistidos	No asistidos	0 a 7	0 a 7
Interfaces	0	0	0	0 y 1

Tabla 10-2 Áreas de operandos de las CPUs S7-200

Tipo de acceso	CPU 212		CPU 214		CPU 215		CPU 216	
Bit (byte.bit)	V	0.0 a 1023.7	V	0.0 a 4095.7	V	0.0 a 5119.7	V	0.0 a 5119.7
	I	0.0 a 7.7	I	0.0 a 7.7	I	0.0 a 7.7	I	0.0 a 7.7
	Q	0.0 a 7.7	Q	0.0 a 7.7	Q	0.0 a 7.7	Q	0.0 a 7.7
	M	0.0 a 15.7	M	0.0 a 31.7	M	0.0 a 31.7	M	0.0 a 31.7
	SM	0.0 a 45.7	SM	0.0 a 85.7	SM	0.0 a 194.7	SM	0.0 a 194.7
	T	0 a 63	T	0 a 127	T	0 a 255	T	0 a 255
	C	0 a 63	C	0 a 127	C	0 a 255	C	0 a 255
	S	0.0 a 7.7	S	0.0 a 15.7	S	0.0 a 31.7	S	0.0 a 31.7
Byte	VB	0 a 1023	VB	0 a 4095	VB	0 a 5119	VB	0 a 5119
	IB	0 a 7	IB	0 a 7	IB	0 a 7	IB	0 a 7
	QB	0 a 7	QB	0 a 7	QB	0 a 7	QB	0 a 7
	MB	0 a 15	MB	0 a 31	MB	0 a 31	MB	0 a 31
	SMB	0 a 45	SMB	0 a 85	SMB	0 a 194	SMB	0 a 194
	AC	0 a 3	AC	0 a 3	AC	0 a 3	AC	0 a 3
	SB	0 a 7	SB	0 a 15	SB	0 a 31	SB	0 a 31
	constante		constante		constante		constante	
Palabra	VW	0 a 1022	VW	0 a 4094	VW	0 a 5118	VW	0 a 5118
	T	0 a 63	T	0 a 127	T	0 a 255	T	0 a 255
	C	0 a 63	C	0 a 127	C	0 a 255	C	0 a 255
	IW	0 a 6	IW	0 a 6	IW	0 a 6	IW	0 a 6
	QW	0 a 6	QW	0 a 6	QW	0 a 6	QW	0 a 6
	MW	0 a 14	MW	0 a 30	MW	0 a 30	MW	0 a 30
	SMW	0 a 44	SMW	0 a 84	SMW	0 a 193	SMW	0 a 193
	AC	0 a 3	AC	0 a 3	AC	0 a 3	AC	0 a 3
	AIW	0 a 30	AIW	0 a 30	AIW	0 a 30	AIW	0 a 30
	AQW	0 a 30	AQW	0 a 30	AQW	0 a 30	AQW	0 a 30
	SW	0 a 6	SW	0 a 14	SW	0 a 30	SW	0 a 30
	constante		constante		constante		constante	
Palabra doble	VD	0 a 1020	VD	0 a 4092	VD	0 a 5116	VD	0 a 5116
	ID	0 a 4	ID	0 a 4	ID	0 a 4	ID	0 a 4
	QD	0 a 4	QD	0 a 4	QD	0 a 4	QD	0 a 4
	MD	0 a 12	MD	0 a 28	MD	0 a 28	MD	0 a 28
	SMD	0 a 42	SMD	0 a 82	SMD	0 a 191	SMD	0 a 191
	AC	0 a 3	AC	0 a 3	AC	0 a 3	AC	0 a 3
	HC	0	HC	0 a 2	HC	0 a 2	HC	0 a 2
	SD	0 a 4	SD	0 a 12	SD	0 a 28	SD	0 a 28
	constante		constante		constante		constante	

## 10.2 Operaciones con contactos

### Contactos estándar



El **contacto abierto** se cierra (se activa) si el valor binario de la dirección  $n = 1$ .

En AWL, el contacto abierto se representa con las operaciones **Cargar** (LD), **Y** (AND) y **O** (OR). Dichas operaciones cargan el valor binario de la dirección  $n$  en el nivel superior de la pila.

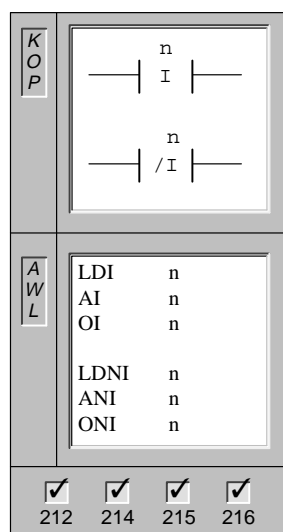
El **contacto cerrado** se cierra (se activa) si el valor binario de la dirección  $n = 0$ .

En AWL, el contacto cerrado se representa con las operaciones **Cargar valor negado** (LDN), **Y-NO** (AN) y **O-NO** (ON). Dichas operaciones cargan el valor binario invertido de la dirección  $n$  en el nivel superior de la pila.

Operandos:  $n$ : I, Q, M, SM, T, C, V, S

Ambas operaciones leen el valor direccionado de la imagen del proceso cuando ésta se actualiza al comienzo de cada ciclo.

### Contactos directos



El **contacto abierto directo** se cierra (se activa) si el valor binario de la entrada física direccionada  $n = 1$ .

En AWL, el contacto cerrado directo se representa con las operaciones **Cargar directamente** (LDI), **Y directa** (AI) y **O directa** (OI). Estas operaciones cargan directamente el valor binario de la entrada física direccionada  $n$  en el nivel superior de la pila y lo combinan mediante Y u O.

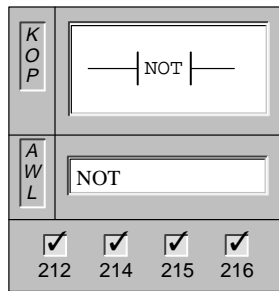
El **contacto cerrado directo** se cierra (se activa) si el valor binario de la entrada física direccionada  $n = 0$ .

En AWL, el contacto abierto directo se representa con las operaciones **Cargar valor negado directamente** (LDNI), **Y-NO directa** (ANI) y **O-NO directa** (ONI). Estas operaciones cargan directamente el valor binario negado de la entrada física direccionada  $n$  en el nivel superior de la pila y lo combinan mediante Y u O.

Operandos:  $n$ : I

Ambas operaciones leen el valor direccionado de la entrada física al ejecutarse la operación, pero la imagen del proceso no se actualiza.

## NOT

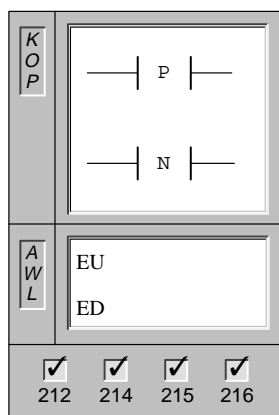


El contacto **NOT** invierte el sentido de circulación de la corriente. La corriente se detiene al alcanzar el contacto NOT. Si no logra alcanzar el contacto, entonces hace circular la corriente.

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

## Detectar flanco positivo y negativo



El contacto **Detectar flanco positivo** permite que fluya la corriente durante un ciclo cada vez que se produce un cambio de 0 a 1 (de "off" a "on").

En AWL, dicho contacto se representa con la operación **Detectar flanco positivo** (EU). 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.

El contacto **Detectar flanco negativo** permite que fluya la corriente durante un ciclo cada vez que se produce un cambio de 1 a 0 (de "on" a "off").

En AWL, dicho contacto se representa con la operación **Detectar flanco negativo** (ED). 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.

Operandos: ninguno

Ejemplos de operaciones con contactos

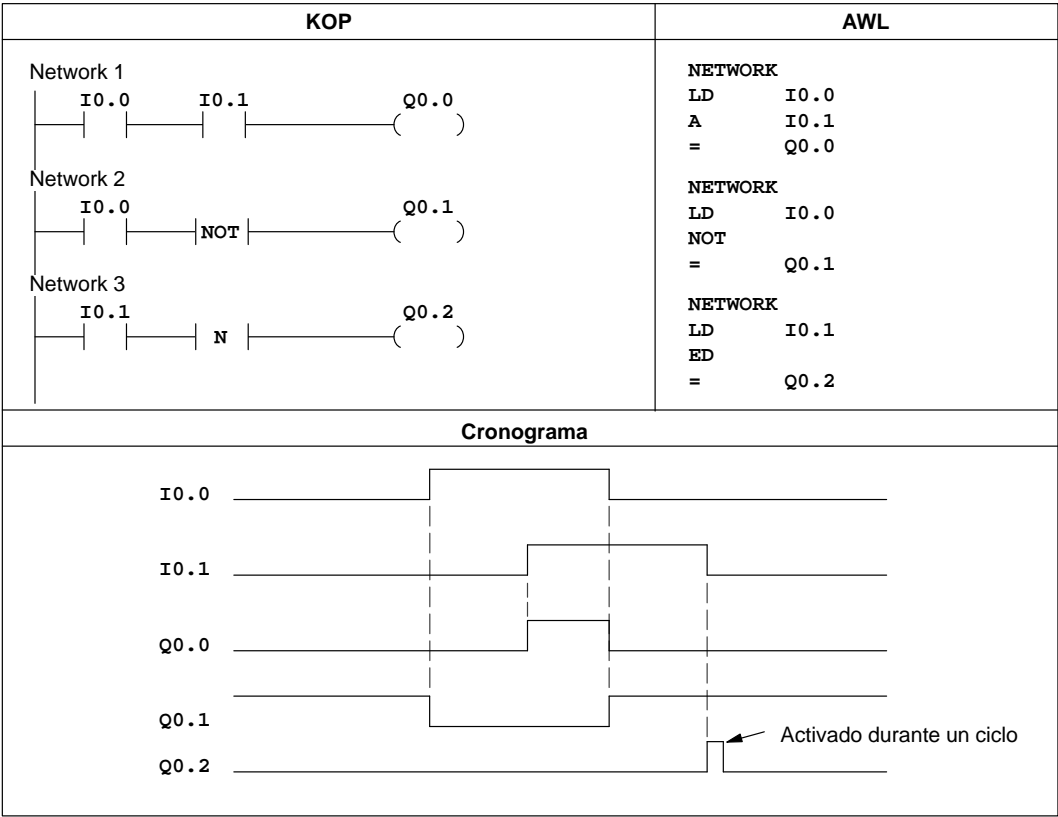
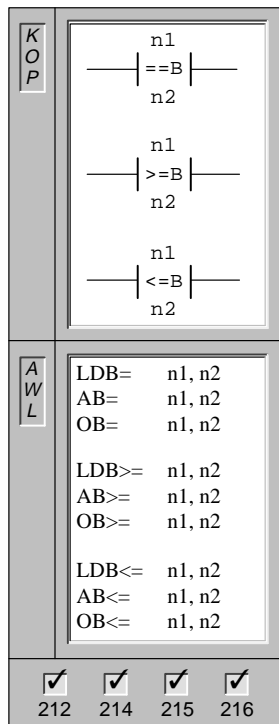


Figura 10-1 Ejemplos de operaciones lógicas con contactos en KOP y AWL

## 10.3 Operaciones de comparación

### Comparar byte



La operación **Comparar byte** se utiliza para comparar dos valores: n1 y n2. Se puede comparar  $n1 = n2$ ,  $n1 \geq n2$  y  $n1 \leq n2$ .

Operandos: n1, n2: VB, IB, QB, MB, SMB, AC, constante, \*VD, \*AC, SB

En KOP, el contacto se activa si la comparación es verdadera.

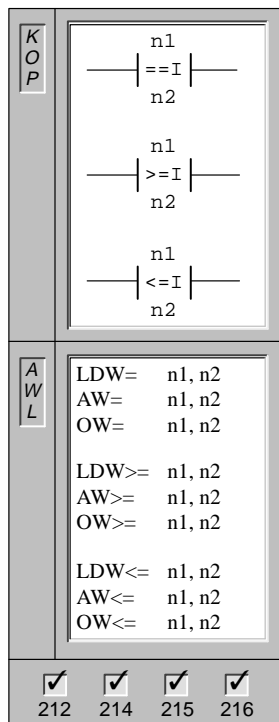
En AWL, las operaciones cargan un "1" en el nivel superior de la pila y combinan el valor "1" con el primer valor de la pila mediante Y u O cuando la comparación es verdadera.

Las comparaciones de bytes no llevan signo.

Nota: Se pueden crear comparaciones  $<>$ ,  $<$  ó  $>$  utilizando la operación NOT con una operación  $=$ ,  $\geq$  ó  $\leq$ . La siguiente secuencia de operaciones corresponde a una comparación  $<>$  de VB100 con 50:

```
LDB= VB100, 50
NOT
```

### Comparar entero palabra



La operación **Comparar entero palabra** se utiliza para comparar dos valores: n1 y n2. Se puede comparar  $n1 = n2$ ,  $n1 \geq n2$  ó  $n1 \leq n2$ .

Operandos: n1, n2: VW, T, C, IW, QW, MW, SMW, AC, AIW, constante, \*VD, \*AC, SW

En KOP, el contacto se activa si la comparación es verdadera.

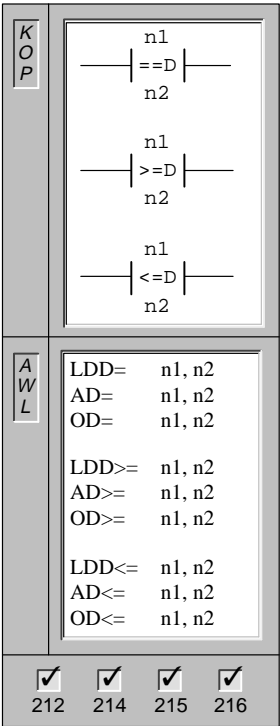
En AWL, las operaciones cargan un "1" en el nivel superior de la pila y combinan el valor "1" con el primer valor de la pila mediante Y u O cuando la comparación es verdadera.

Las comparaciones de palabras llevan signo ( $16\#7FFF > 16\#8000$ ).

Nota: Se pueden crear comparaciones  $<>$ ,  $<$  ó  $>$  utilizando la operación NOT con una operación  $=$ ,  $\geq$  ó  $\leq$ . La siguiente secuencia de operaciones equivale a una comparación  $<>$  de VW100 con 50:

```
LDW= VW100, 50
NOT
```

Comparar entero palabra doble



La operación **Comparar entero palabra doble** se utiliza para comparar dos valores: n1 y n2. Se puede comparar  $n1 = n2$ ,  $n1 \geq n2$  ó  $n1 \leq n2$ .

Operandos: n1, n2: VD, ID, QD, MD, SMD, AC, HC, constante, \*VD, \*AC, SD

En KOP, el contacto se activa si la comparación es verdadera.

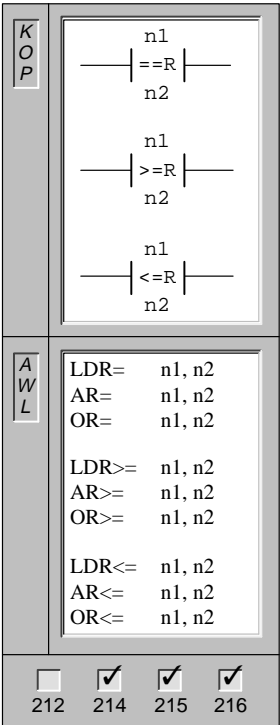
En AWL, las operaciones cargan un "1" en el nivel superior de la pila y combinan el valor "1" con el primer valor de la pila mediante Y u O cuando la comparación es verdadera.

Las comparaciones de palabras dobles llevan signo (16#7FFFFFFF > 16#80000000).

Nota: Se pueden crear comparaciones <>, < ó > utilizando la operación NOT con una operación =, >= ó <=. La siguiente secuencia de operaciones equivale a una comparación <> de VD100 con 50:

LDD= VD100, 50  
NOT

Comparar real



La operación **Comparar real** se utiliza para comparar dos valores: n1 y n2. Se puede comparar  $n1 = n2$ ,  $n1 \geq n2$  y  $n1 \leq n2$ .

Operandos: n1, n2: VD, ID, QD, MD, SMD, AC, constante, \*VD, \*AC, SD

En KOP, el contacto 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.

Las comparaciones de números reales llevan signo.

Nota: Se pueden crear comparaciones <>, < ó > utilizando la operación NOT con una operación =, >= ó <=. La siguiente secuencia de operaciones equivale a una comparación <> de VD100 con 50:

LDR= VD100, 50.0  
NOT

### Ejemplos de operaciones de comparación

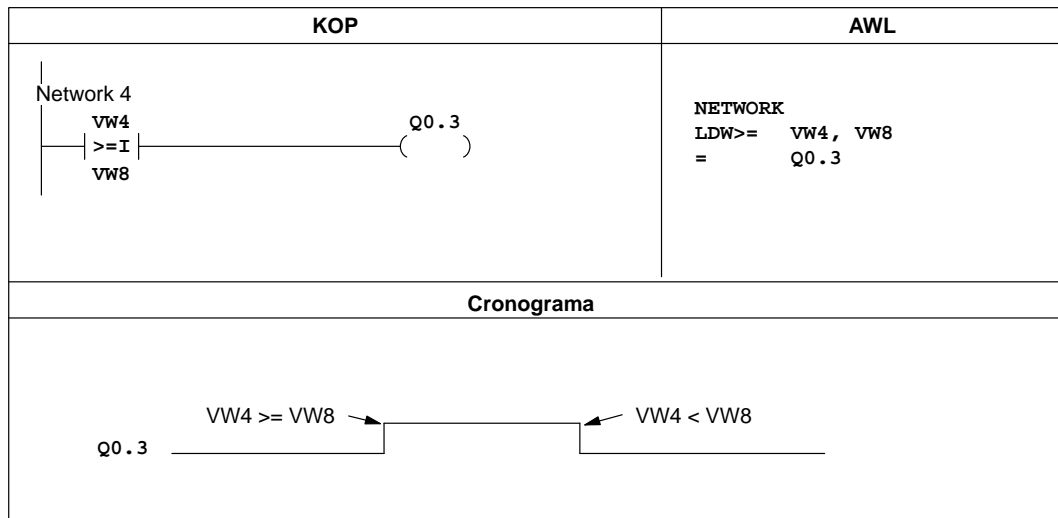
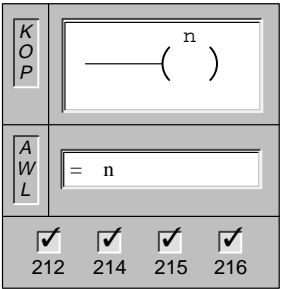


Figura 10-2 Ejemplos de operaciones de comparación en KOP y AWL

### 10.4 Operaciones con salidas

#### Asignar

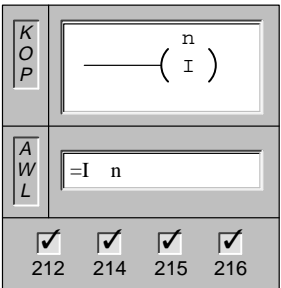


Al ejecutar la operación *Asignar* se activa el parámetro indicado (n).

En AWL, la operación Asignar (=) copia el primer valor de la pila en el parámetro indicado (n).

Operandos: n: I, Q, M, SM, T, C, V, S

#### Asignar directamente



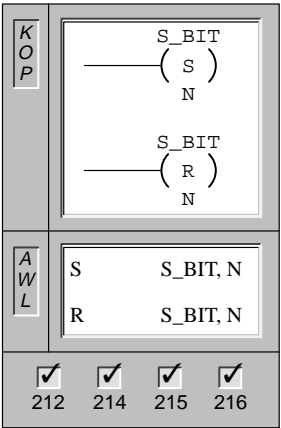
Al ejecutar la operación **Asignar directamente** se activa directamente la salida física indicada (n).

En AWL, la operación Asignar directamente (=I) copia el primer valor de la pila directamente en la salida física indicada (n).

Operandos: n: Q

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.

#### Poner a 1, Poner a 0

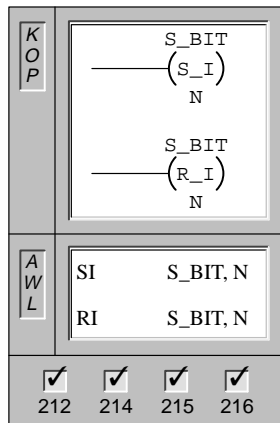


Al ejecutar las operaciones **Poner a 1** y **Poner a 0**, se activa (se pone a 1) o se desactiva (se pone a 0) el número indicado de entradas y/o salidas (N) a partir de S\_BIT, respectivamente.

Operandos: S\_BIT: I, Q, M, SM, T, C, V, S  
N: IB, QB, MB, SMB, VB, AC, constante, \*VD, \*AC, SB

El margen de entradas y/o salidas que se pueden activar o desactivar está comprendido entre 1 y 255. Al utilizarse la operación Poner a 0, si S\_BIT es un bit T (bit de temporización) o un bit C (bit de conteo), se desactivará dicho bit y se borrará el valor del temporizador o contador, respectivamente.

### Poner a 1 directamente, Poner a 0 directamente



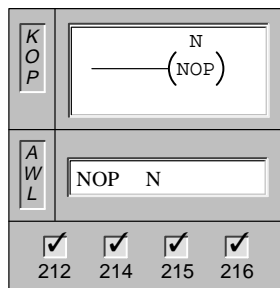
Cuando se ejecutan las operaciones **Poner a 1 directamente** y **Poner a 0 directamente**, se activa (se pone a 1) o se desactiva (se pone a 0) directamente el número indicado de salidas físicas (N) a partir de S\_BIT, respectivamente.

Operandos: S\_BIT: Q  
N: IB, QB, MB, SMB, VB, AC, constante, \*VD, \*AC, SB

Se pueden activar o desactivar un margen comprendido entre 1 y 64 salidas.

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.

### Operación nula



La **Operación nula** (NOP) no tiene efecto alguno en la ejecución del programa. El operando N es un número comprendido entre 0 y 255.

Operandos: N: 0 a 255

La operación NOP se debe hallar dentro del programa principal o en una subrutina o bien, en una rutina de interrupción.

Ejemplos de operaciones con salidas

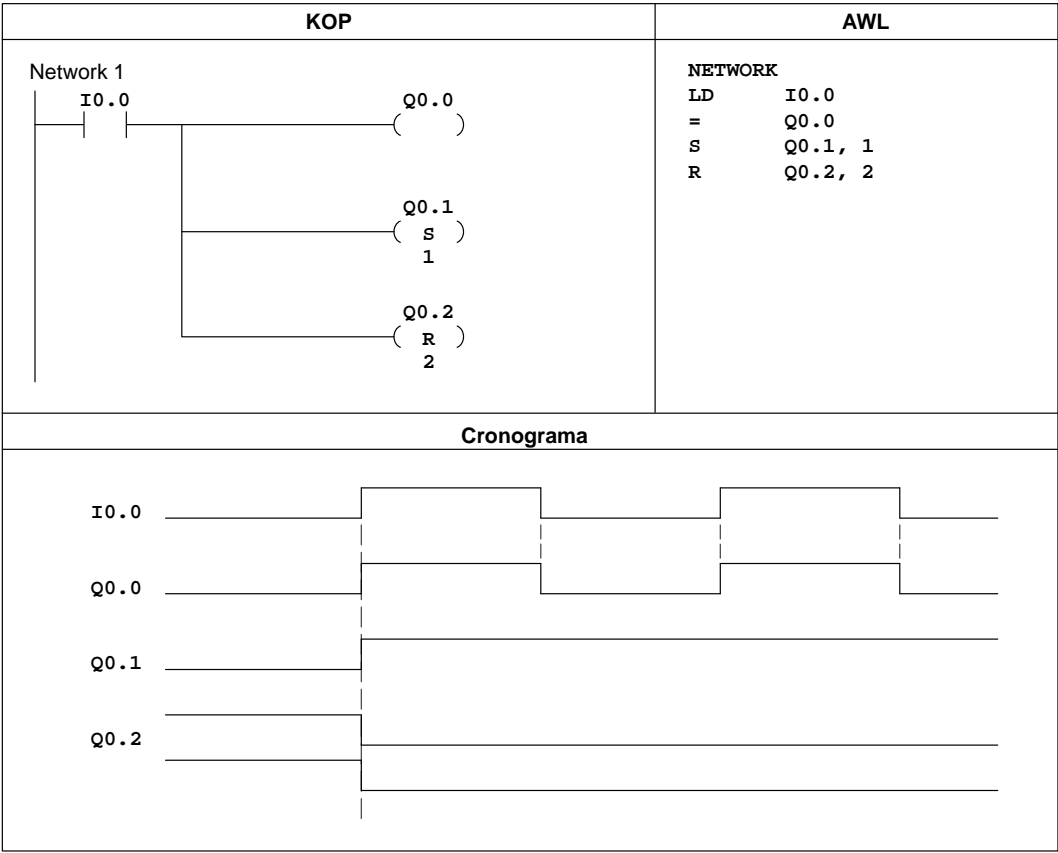
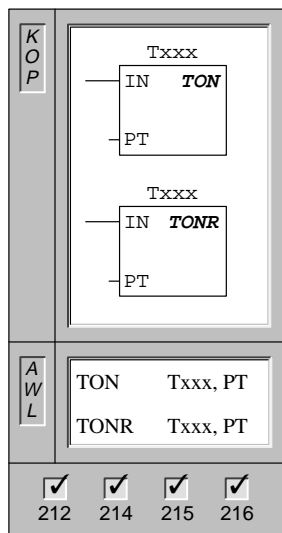


Figura 10-3 Ejemplos de operaciones con salidas en KOP y AWL

## 10.5 Operaciones con temporizadores, contadores, contadores rápidos, reloj de tiempo real y salida de impulsos

### Temporizador de retardo a la conexión, Temporizador de retardo a la conexión memorizado



Las operaciones **Temporizador de retardo a la conexión** y **Temporizador de retardo a la conexión memorizado** empiezan a contar hasta el valor máximo al ser habilitadas. Si el valor actual (Txxx) es mayor o igual al valor de preselección (PT), se activa el bit de temporización.

Cuando se inhibe la operación, el temporizador de retardo a la conexión se pone a 0, en tanto que el temporizador de retardo a la conexión memorizado se detiene. Ambos temporizadores se detienen al alcanzar el valor máximo.

Operandos: Txxx: TON TONR  
1 ms T32, T96  
T0, T64

10 ms T33 a T36 T1 a T4  
T97 a T100 T65 a T68

100 ms T37 a T63 T5 a T31  
T101 a T255 T69 a T95

PT: VW, T, C, IW, QW, MW, SMW, AC,  
AIW, constante, \*VD, \*AC, SW

Hay disponibles temporizadores TON y TONR con tres resoluciones distintas. 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 milisegundos (ms) equivale a 500 ms.

Tabla 10-3 Temporizadores y sus resoluciones

Temporizador	Resolución	Valor máximo	CPU 212	CPU 214	CPU 215/216
TON	1 ms	32,767 segundos (s)	T32	T32, T96	T32, T96
	10 ms	327,67 s	T33 a T36	T33 a T36, T97 a T100	T33 a T36, T97 a T100
	100 ms	3276,7 s	T37 a T63	T37 a T63, T101 a T127	T37 a T63, T101 a T255
TONR	1 ms	32,767 s	T0	T0, T64	T0, T64
	10 ms	327,67 s	T1 a T4	T1 a T4, T65 a T68	T1 a T4, T65 a T68
	100 ms	3276,7 s	T5 a T31	T5 a T31, T69 a T95	T5 a T31, T69 a T95

### Operaciones de temporización del S7-200

Es posible utilizar temporizadores para implementar funciones controladas por tiempo. En el S7-200 se ofrecen dos operaciones de temporización diferentes: el Temporizador de retardo a la conexión (TON) y el Temporizador de retardo a la conexión memorizado (TONR). Estos dos tipos de temporizadores (TON y TONR) difieren en cuanto a su reacción al estado de la entrada de habilitación. Ambos temporizadores cuentan adelante mientras esté activada dicha entrada, pero no actúan al estar desactivada la misma. Al desactivarse la entrada de habilitación, el temporizador TON se pone a 0 automáticamente, pero no el temporizador TONR, conservando éste su último valor. Por consiguiente, el temporizador TON resulta especialmente apropiado para temporizar intervalos individuales. En cambio, el temporizador TONR se adecúa para acumular varios intervalos temporizados.

Los temporizadores del S7-200 presentan las siguientes características:

- Los temporizadores se controlan mediante una sola entrada de habilitación y disponen de un valor actual que almacena el tiempo transcurrido desde que fueron habilitados. Los temporizadores disponen también de un valor de preselección (PT) que se compara con el valor actual cada vez que se actualiza éste y tras ejecutarse la operación de temporización.
- Conforme al resultado de la comparación entre el valor de preselección y el valor actual, se activa o desactiva un bit de temporización (bit T).
- Si el valor actual es mayor que o igual al valor de preselección, se activa el bit T.

---

#### Nota

Se pueden definir como remanentes algunos valores actuales de un temporizador. Los bits de temporización no son remanentes, activándose únicamente como resultado de la comparación entre el valor actual y el de preselección.

---

Al inicializar un temporizador, se pone a 0 su valor actual y se desactiva su bit T. Es posible borrar cualquier temporizador mediante la operación Poner a 0, pero un temporizador TONR sólo puede inicializarse a través de esta operación. Si se escribe un cero en el valor actual de un temporizador, su bit de temporización no se desactivará. Igualmente, si se escribe un cero en el bit T de un temporizador, su valor actual no se desactivará.

Diversos temporizadores de 1 ms también se pueden utilizar para generar un evento de interrupción. Para obtener más información sobre las interrupciones temporizadas, consulte el apartado 10.14.

### Actualizar los temporizadores con una resolución de 1 ms

La CPU S7-200 dispone de temporizadores que son actualizados cada milisegundo (temporizadores de 1 ms) por la rutina de interrupción del sistema que mantiene la base de tiempo del mismo. Estos temporizadores permiten controlar las operaciones de forma precisa.

El valor actual de un temporizador de 1 ms activo se actualiza automáticamente en una rutina del sistema. Una vez habilitado un temporizador de 1 ms, la ejecución de la operación TON/TONR que lo controla se necesita sólo para supervisar el estado "on" u "off" (habilitado o inhibido) del mismo.

Puesto que los temporizadores de 1 ms se actualizan mediante una rutina de interrupción, el valor actual y el bit T de estos temporizadores se pueden actualizar en cualquier instante del ciclo, así como varias veces dentro de un ciclo, si éste dura más de 1 milisegundo. Por tal causa, es posible que esos valores no permanezcan constantes al ejecutarse el programa principal de usuario.

Al inicializar un temporizador de 1 ms habilitado, se desactiva el mismo, se pone a 0 su valor actual y se borra su bit T.

---

**Nota**

La rutina que mantiene la base de tiempo de 1 ms del sistema no depende de si los temporizadores se habilitan o no. Un temporizador con una resolución de 1 ms se puede habilitar en cualquier momento del intervalo actual de 1 ms. Por lo tanto, el intervalo para un temporizador con una resolución de 1 ms puede tener una duración máxima de 1 ms. Es necesario programar el valor de preselección a un valor que supere en 1 al 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.

---

### Actualizar los temporizadores con una resolución de 10 ms

La CPU S7-200 dispone de temporizadores que cuentan la cantidad de intervalos de 10 ms transcurridos tras haberse habilitado el temporizador de 10 ms activo. Estos temporizadores se actualizan al comienzo de cada ciclo, añadiendo la cantidad de intervalos de 10 ms transcurridos (desde la última actualización) al valor actual del temporizador.

El valor actual de un temporizador de 10 ms activo se actualiza automáticamente al comenzar el ciclo. Una vez habilitado un temporizador de 10 ms, la ejecución de la operación TON/TONR que lo controla se necesita sólo para supervisar el estado "on" u "off" (habilitado o inhibido) del mismo. A diferencia de los temporizadores de 1 ms, el valor actual del temporizador de 10 ms sólo se actualiza una vez por ciclo y no se modifica mientras se ejecuta el programa principal de usuario.

Al inicializar un temporizador de 10 ms habilitado, se desactiva el mismo, se pone a 0 su valor actual y se borra su bit T.

---

**Nota**

Como la acumulación de los intervalos de 10 ms no depende de si los temporizadores se habilitan o no, los temporizadores con una resolución de 10 ms se habilitan dentro de un determinado intervalo de 10 ms. Por lo tanto, el intervalo para un temporizador con una resolución de 10 ms puede tener una duración máxima de 10 ms. Es necesario programar el valor de preselección a un valor que supere en 1 al 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.

---

### Actualizar los temporizadores con una resolución de 100 ms

La mayoría de los temporizadores disponibles para la CPU S7-200 utilizan una resolución de 100 ms. Estos temporizadores cuentan el número de intervalos de 100 ms que han transcurrido desde que se actualizó el temporizador de 100 ms. Estos temporizadores se actualizan añadiendo el valor acumulativo de 100 ms al valor actual del temporizador cuando se ejecuta la operación del mismo.

Los temporizadores de 100 ms no se actualizan automáticamente, ya que el valor actual de un temporizador con una resolución de 100 ms sólo se actualiza al ejecutar la operación de temporización. 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 añadirá también varias veces al valor actual del temporizador, con lo cual aumenta 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. Cuando se inicializa un temporizador de 100 ms, se pone a 0 su valor actual y se borra su bit T.

---

#### Nota

Como la acumulación de los intervalos de 100 ms no depende de si los temporizadores se habilitan o no, un determinado temporizador con una resolución de 100 ms se habilita en un punto cualquiera dentro del actual intervalo de 100 ms. Por lo tanto, el intervalo para un determinado temporizador con una resolución de 100 ms puede tener una duración máxima de 100 ms. Es necesario programar el valor de preselección a un valor que supere en 1 al 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 10-4.

- Si se utiliza un temporizador con una resolución de 1 ms, Q0.0 se activará durante un ciclo, siempre que el valor actual del temporizador se actualice tras ejecutarse el contacto cerrado T32 y antes de ejecutarse el contacto abierto T32.
- Si se utiliza un temporizador con una resolución de 10 ms, 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 abierto T33, se desactivarán T33 y Q0.0.
- Si se utiliza un temporizador con una resolución de 100 ms, Q0.0 se activará 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 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 (v. fig.10-4). Las figuras 10-5 y 10-6 muestran ejemplos de las operaciones de temporización en KOP y AWL.

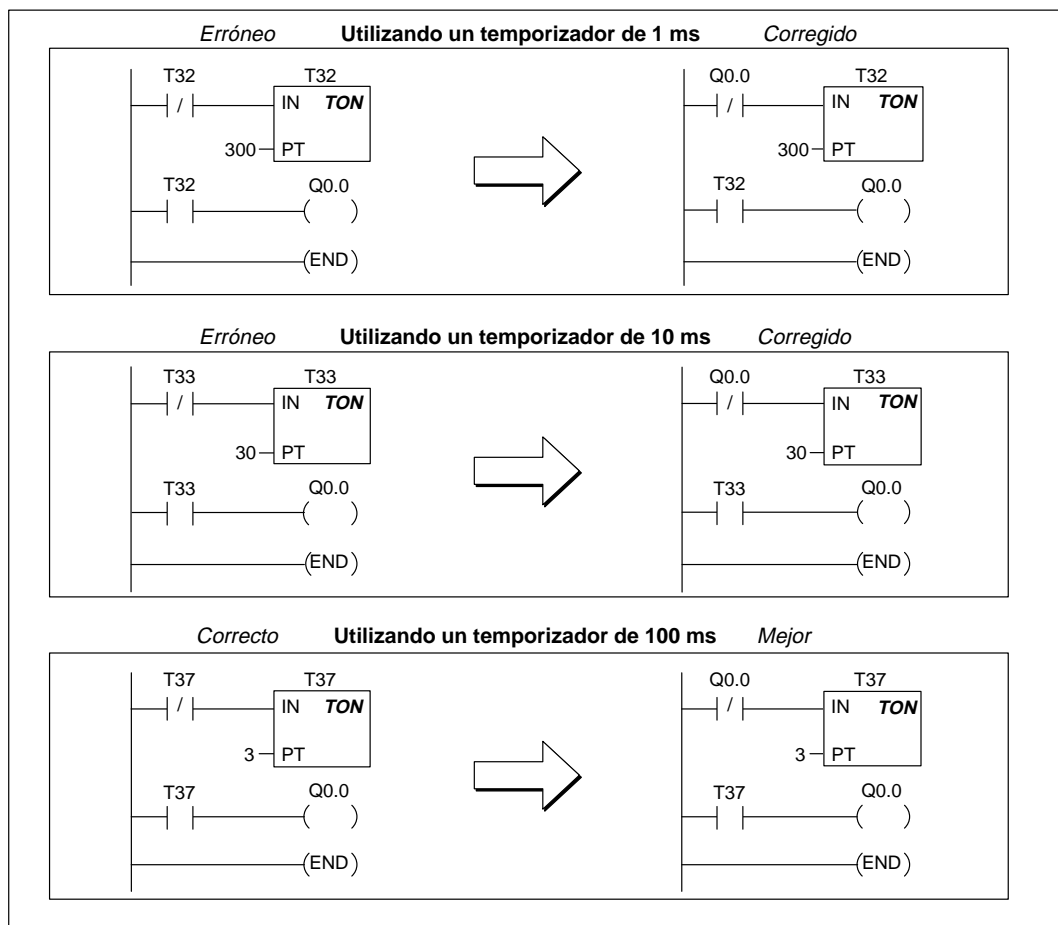


Figura 10-4 Ejemplo del redisparo automático de un temporizador

### Ejemplo de un temporizador de retardo a la conexión

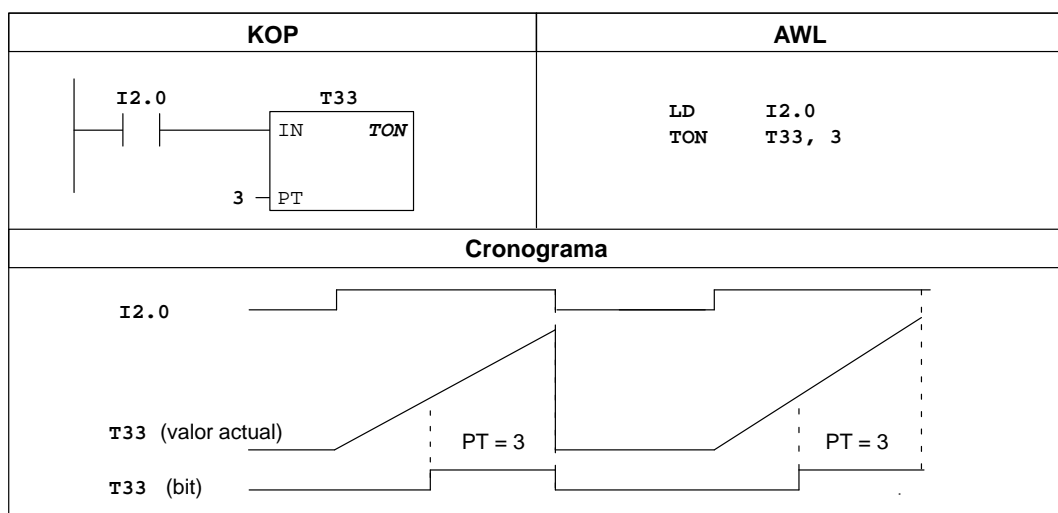


Figura 10-5 Ejemplo de un temporizador de retardo a la conexión en KOP y AWL

Ejemplo de un temporizador de retardo a la conexión memorizado

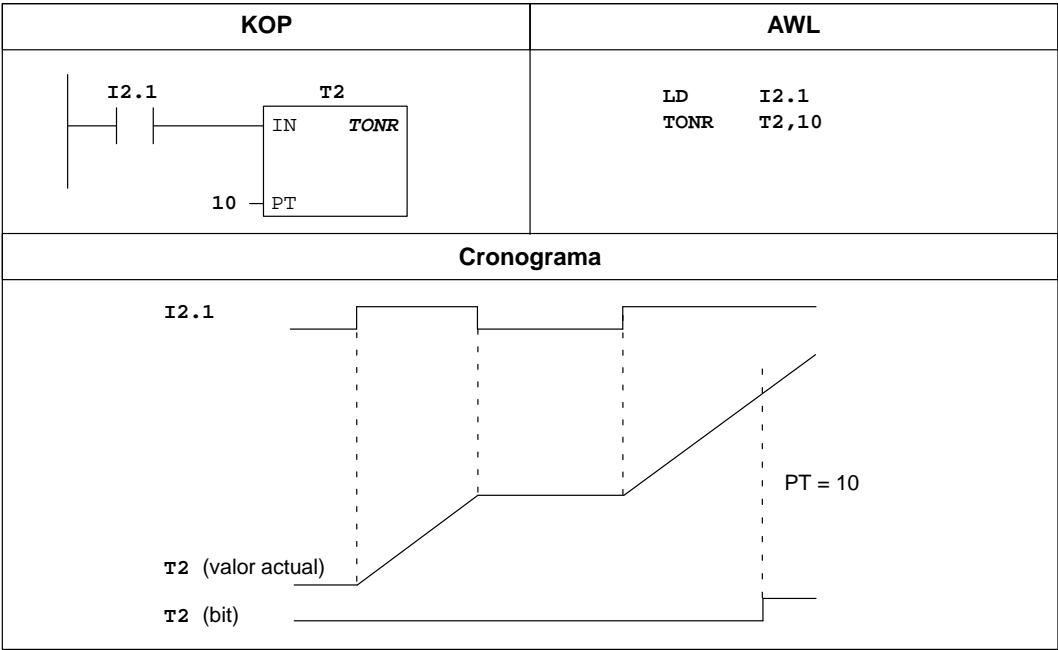
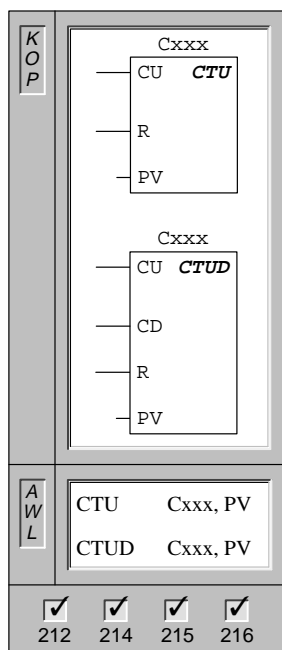


Figura 10-6    Ejemplo de un temporizador de retardo a la conexión memorizado en KOP y AWL

## Contar adelante, Contar adelante/atrás



La operación **Contar adelante** empieza a contar hasta el valor máximo cuando se produce un flanco positivo en la entrada de conteo adelante (CU). Si el valor actual (Cxxx) es mayor o igual al valor de preselección (PV), se activa el bit de conteo (Cxxx). El contador se inicializa al activarse la entrada de desactivación (R).

En AWL, la entrada de desactivación es el primer valor de la pila, en tanto que la entrada de conteo adelante se carga en el segundo nivel de la pila.

La operación **Contar adelante/atrás** empieza a contar adelante cuando se produce un flanco positivo en la entrada de conteo adelante (CU). Por el contrario, empieza a contar atrás cuando se produce un flanco positivo en la entrada de conteo atrás (CD). Si el valor actual (Cxxx) es mayor o igual al valor de preselección (PV), se activa el bit de conteo (Cxxx). El contador se inicializa al activarse la entrada de desactivación (R).

En AWL, la entrada de desactivación es el primer valor de la pila, la entrada de conteo atrás se carga en el segundo nivel de la pila y la entrada de conteo adelante, en el tercero.

Operandos: Cxxx: 0 a 255  
PV: VW, T, C, IW, QW, MW, SMW, AC, AIW, constante, \*VD, \*AC, SW

## Operaciones de conteo 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 conteo 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 conteo adelante, y empieza a contar atrás cuando se produce un flanco positivo en la entrada de conteo 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 conteo adelante invertirá el conteo hasta alcanzar el valor mínimo (–32.768). Igualmente, cuando se alcanza el valor mínimo (–32.768), el siguiente flanco positivo en la entrada de conteo atrás invertirá el conteo hasta alcanzar el valor máximo (32.767).

Cuando se inicializa un contador con la operación Poner a 0, se desactivan tanto el bit de conteo como el valor actual del contador.

Los contadores Contar adelante y Contar adelante/atrás tienen un valor actual que almacena el valor de conteo actual. También disponen de un valor de preselección (PV) que se compara con el valor actual cuando se ejecuta la operación de conteo. Si el valor actual es mayor o igual al valor de preselección, se activa el bit de conteo (bit C). En caso contrario, dicho bit se desactiva.

El número del contador se debe utilizar para direccionar tanto el valor actual como el bit C de dicho 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 y Contar adelante/atrás acceden a un mismo valor actual).

Ejemplo de una operación de conteaje

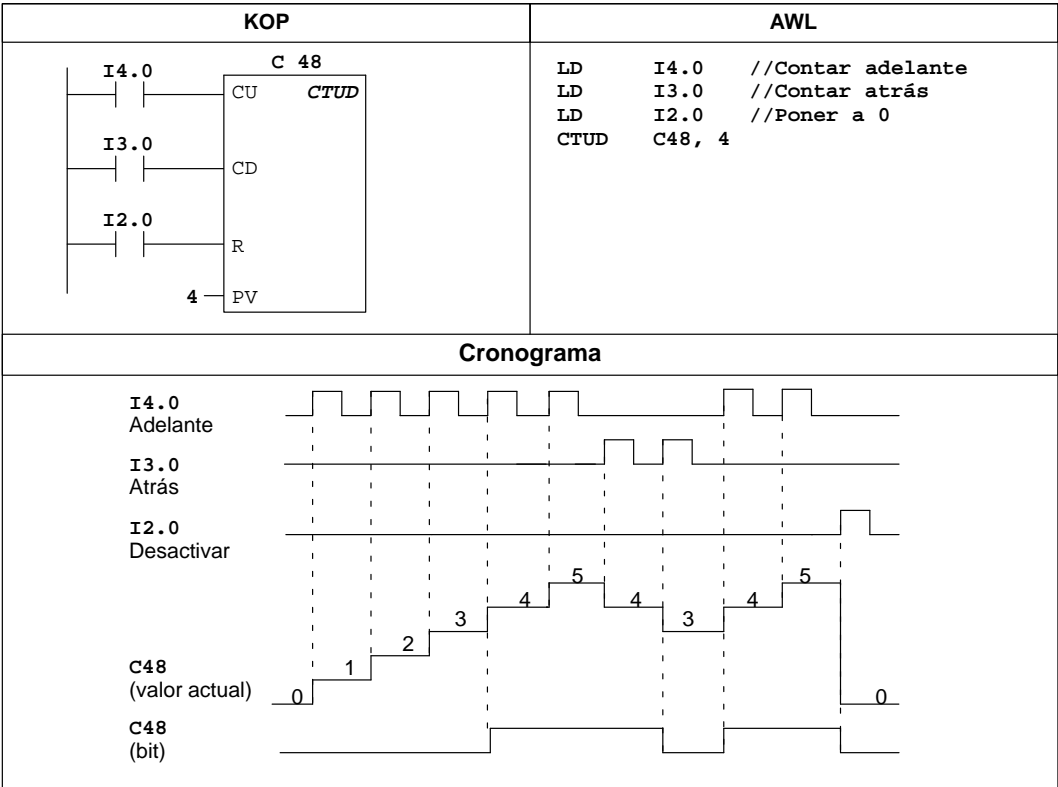
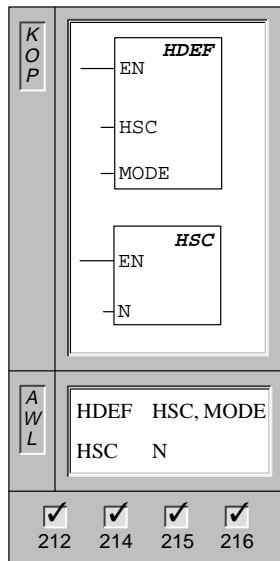


Figura 10-7 Ejemplo de una operación de conteaje en KOP y AWL

### Definir modo para contador rápido, Activar contador rápido



La operación **Definir modo para contador rápido** (HDEF) asigna un modo (MODE) al contador direccionado (HSC) (v. tabla 10-5).

La operación **Activar contador rápido** (HSC) configura y controla el funcionamiento del contador direccionado, basándose en el estado de las marcas especiales del mismo. El parámetro N indica el número del contador rápido.

Por cada contador rápido sólo puede utilizarse un cuadro HDEF.

Operandos:

HSC:	0 a 2
MODE:	0 (HSC0) 0 a 11 (HSC1 ó 2)
N:	0 a 2

### Descripción de las operaciones con contadores rápidos

Los contadores rápidos cuentan eventos que se ejecutan más deprisa de lo que puede controlarlos el ciclo normal de la CPU.

- HSC0 es un contador (software) que permite contar adelante y atrás, asistiendo una sola entrada de reloj. El programa controla el sentido de conteo (adelante o atrás) mediante el bit del control del sentido. La frecuencia máxima de este contador es de 2 kHz.
- HSC1 y HSC2 son contadores universales (hardware) que se pueden configurar según uno de doce modos de operación distintos. La tabla 10-5 muestra los modos de los contadores. La frecuencia máxima de conteo de HSC1 y HSC2 depende de la CPU (consulte el Anexo A).

Cada contador dispone de entradas que asisten 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. Los contadores A/B permiten elegir una velocidad simple (1x) o cuádruple (4x) 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.

La figura 10-16 muestra un ejemplo de la inicialización de HSC1.

### 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 codificador o captador 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 captador 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 "Operaciones de interrupción".

### Diagramas de impulsos para los contadores rápidos

Los siguientes diagramas de impulsos (figuras 10-8, 10-9, 10-10 y 10-11) 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 y es aplicable a todos los contadores 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.

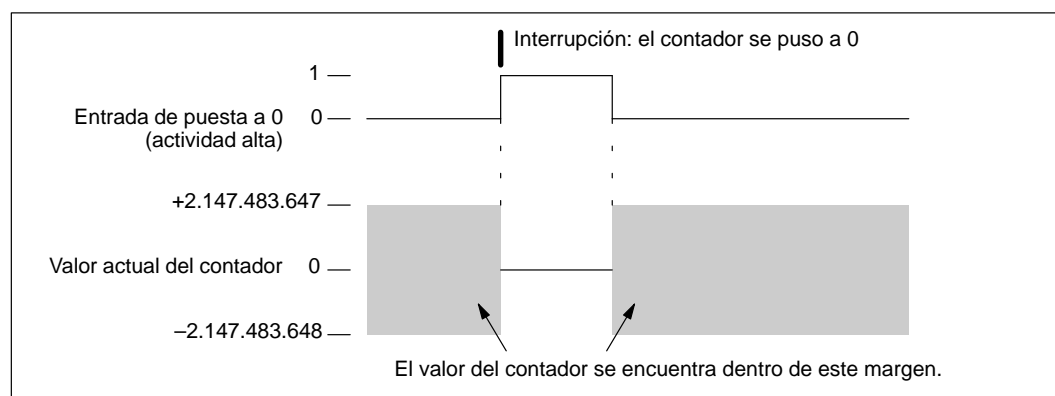


Figura 10-8 Ejemplo del funcionamiento con puesta a 0 y sin arranque

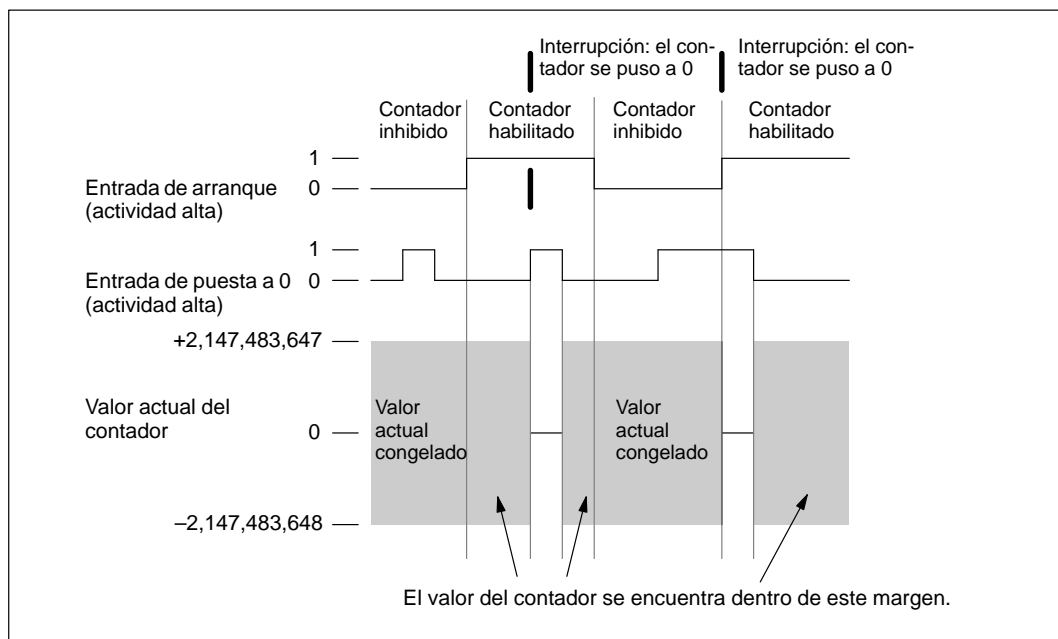


Figura 10-9 Ejemplo del funcionamiento con puesta a 0 y arranque

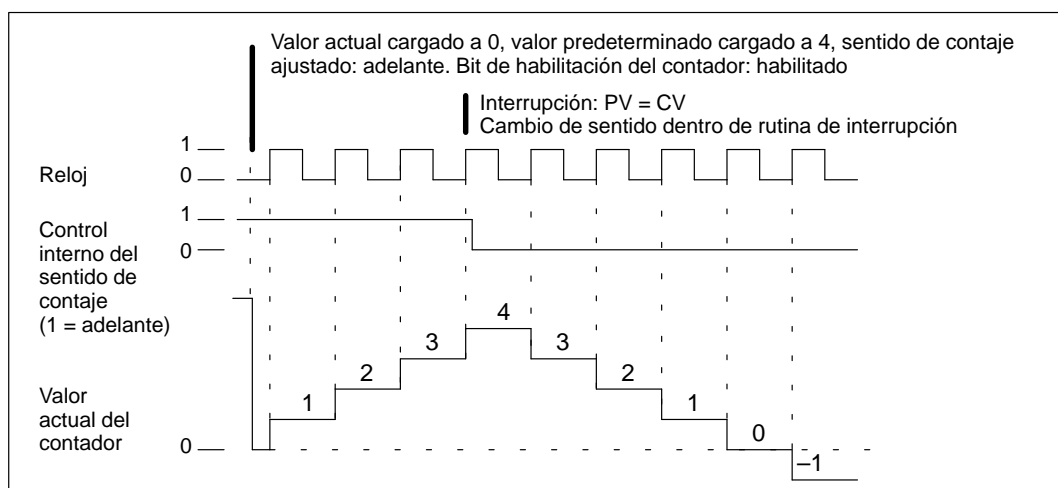


Figura 10-10 Ejemplo del funcionamiento de HSC0 en modo 0 y HSC1 o HSC2 en modo 0, 1, ó 2

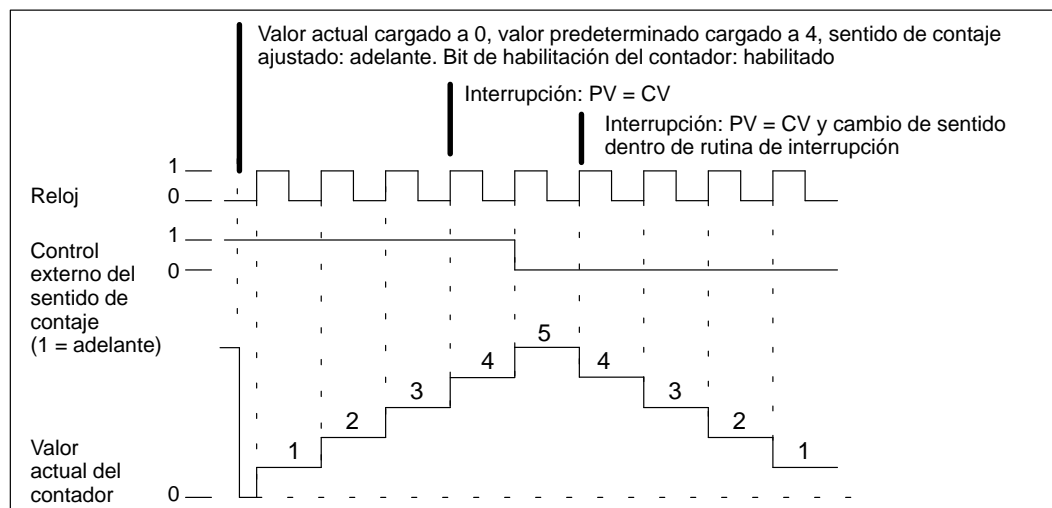


Figura 10-11 Ejemplo del funcionamiento de HSC1 o HSC2 en modo 3, 4 ó 5

Si para HSC1 o HSC2 se utilizan los modos de conteo 6, 7 u 8 y se produce un flanco positivo tanto en la entrada de conteo adelante como en la entrada de conteo atrás en menos de 0,3 microsegundos de diferencia, puede ocurrir que el contador rápido considere ambos eventos simultáneos. 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 los dos casos se produce un error (v. figuras 10-12, 10-13 y 10-14).

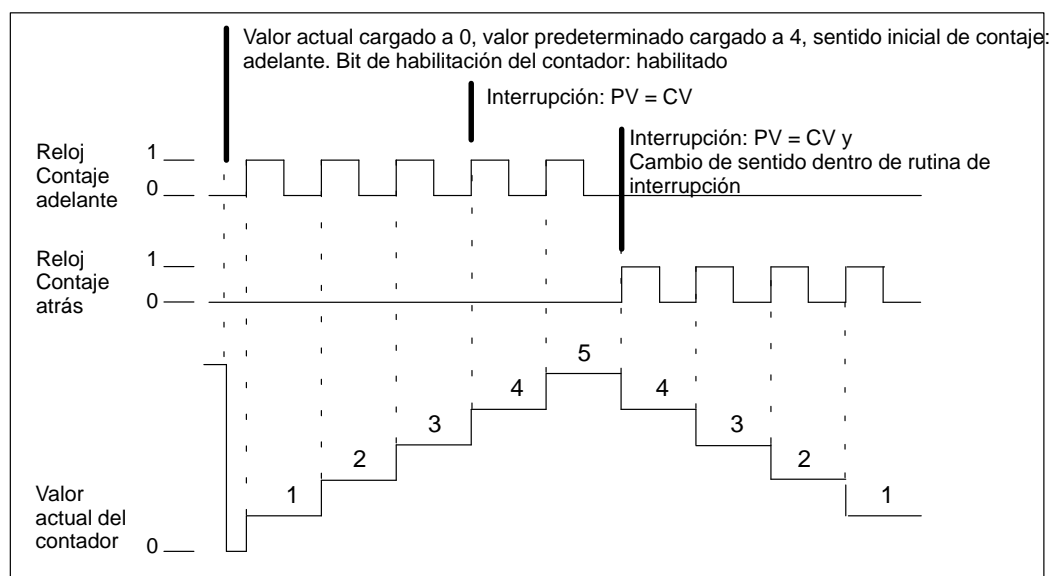


Figura 10-12 Ejemplo del funcionamiento de HSC1 o HSC2 en modo 6, 7 u 8

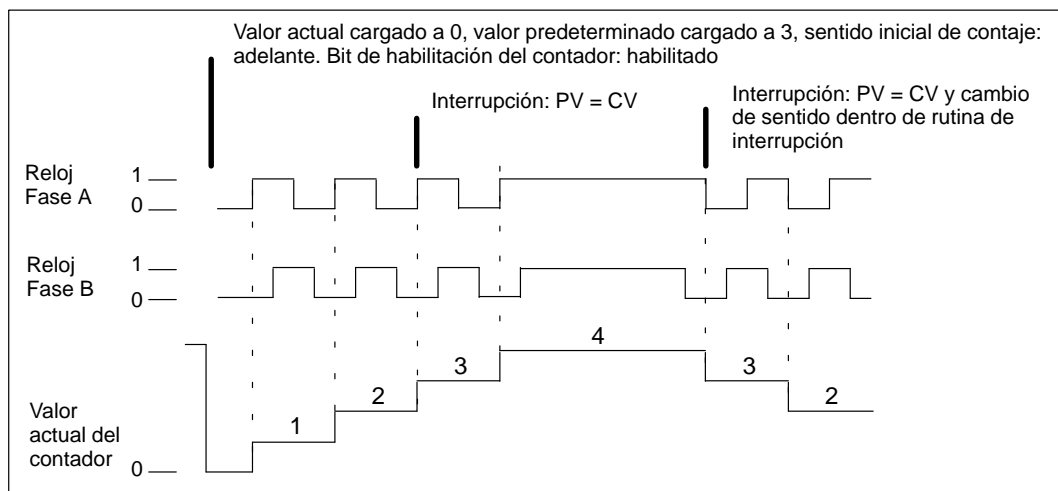


Figura 10-13 Ejemplo del funcionamiento de HSC1 o HSC2 en modo 9, 10 u 11 (contadores A/B, frecuencia simple)

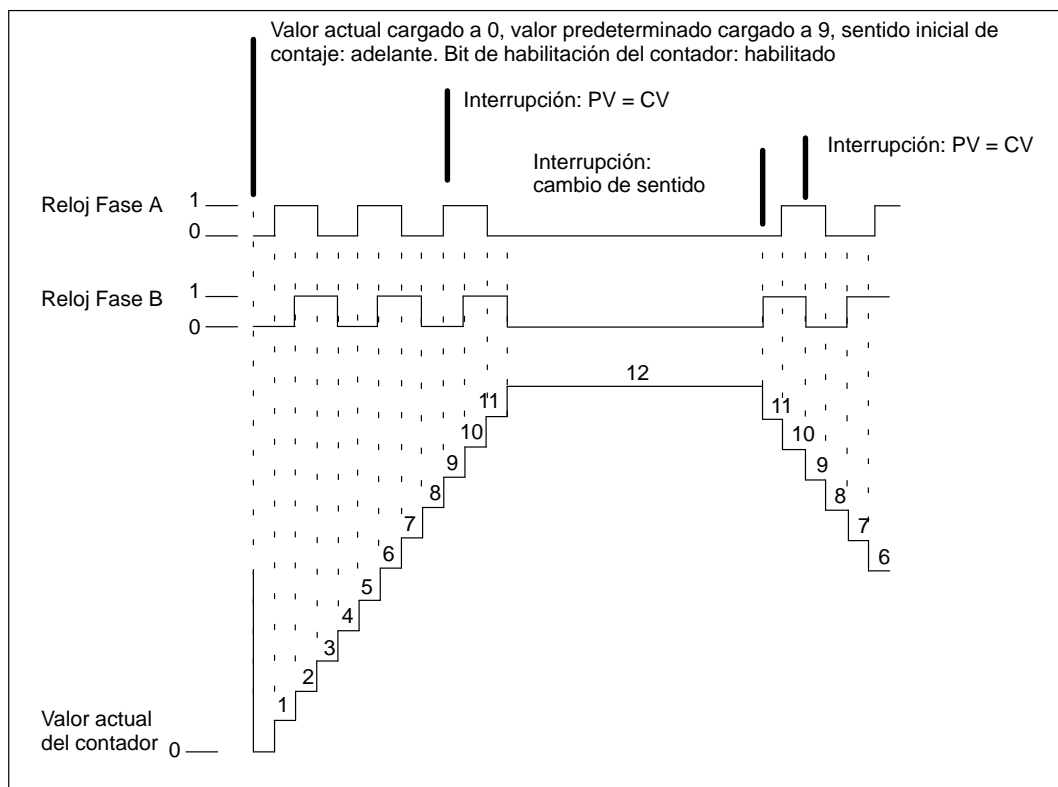


Figura 10-14 Ejemplo del funcionamiento de HSC1 o HSC2 en modo 9, 10 u 11 (contadores A/B, frecuencia cuádruple)

Conectar el cableado de las entradas de los contadores rápidos

La tabla 10-4 muestra las entradas correspondientes al reloj, el control del sentido, la puesta a 0 y las funciones de arranque de los contadores rápidos. En la tabla 10-5 se describen dichas funciones.

Tabla 10-4 Entradas para los contadores rápidos

Contador rápido	Entradas utilizadas
HSC0	I0.0
HSC1	I0.6, I0.7, I1.0, I1.1
HSC2	I1.2, I1.3, I1.4, I1.5

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 10-15.

Formato: HC[número del contador rápido] HC1

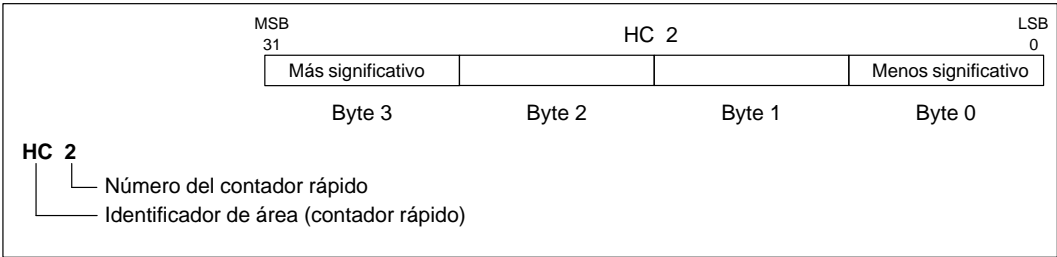


Figura 10-15 Acceso a los valores actuales del contador rápido

Tabla 10-5 Modos de operación de los contadores rápidos

HSC0					
Modo	Descripción	I0.0			
0	Contador adelante/atrás de fase simple con control interno del sentido de contaje. SM37.3 = 0, contaje atrás SM37.3 = 1, contaje adelante	Reloj			
HSC1					
Modo	Descripción	I0.6	I0.7	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	
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	

### Descripción de los diferentes contadores rápidos (HSC0, HSC1, HSC2)

Todos los contadores (HSC0, HSC1 y HSC2) funcionan de la misma manera en el mismo modo de operación. Como muestra la tabla 10-5, hay cuatro tipos básicos de contadores HSC1 y HSC2. Cada contador se puede 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 dicha 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 esté 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. A tal efecto se utiliza la operación HDEF (Definir modo para contador rápido). HDEF establece el enlace entre un contador rápido (HSC0, HSC1 o HSC2) 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

HSC1 y HSC2 disponen de tres bits de control para configurar el nivel de actividad de las entradas de puesta a 0 y de arranque, así como para elegir los modos de conteo (simple o cuádruple, sólo en el caso de los contadores A/B). Dichos bits se encuentran en el byte de control del respectivo contador y se emplean solamente cuando se ejecuta la operación HDEF. La tabla 10-6 muestra los bits.

Antes de poder ejecutar la operación HDEF es preciso ajustar los bits de control de HSC1 y HSC2 al estado deseado. De lo contrario, el contador adoptará la configuración predeterminada del modo de conteo elegido. En el caso de los contadores HSC1 y HSC2, los ajustes predeterminados de las entradas de puesta a 0 y de arranque es de actividad alta. En el caso de los contadores A/B, la velocidad de conteo está ajustada a modo cuádruple (cuatro veces la velocidad de entrada). 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 10-6 Nivel de actividad de las entradas de puesta a 0 y de arranque; bits para elegir la velocidad simple o cuádruple de los contadores HSC1 y HSC2

HSC1	HSC2	Descripción (sólo cuando se ejecuta HDEF)
SM47.0	SM57.0	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
SM47.2	SM57.2	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. Cada contador rápido dispone de un byte que lo 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 10-7 describe cada uno de los bits del byte de control.

Tabla 10-7 Bits de control de HSC0, HSC1 y HSC2

HSC0	HSC1	HSC2	Descripción
SM37.0	SM47.0	SM57.0	No se utiliza tras ejecutar HDEF (nunca utilizado por HSC0).
SM37.1	SM47.1	SM57.1	No se utiliza tras ejecutar HDEF (nunca utilizado por HSC0).
SM37.2	SM47.2	SM57.2	No se utiliza tras ejecutar HDEF (nunca utilizado por HSC0).
SM37.3	SM47.3	SM57.3	Bit de control para el sentido de conteo: 0 = conteo atrás; 1 = conteo adelante
SM37.4	SM47.4	SM57.4	Escribir el sentido de conteo en el contador rápido: 0 = no actualizar; 1 = actualizar el sentido de conteo
SM37.5	SM47.5	SM57.5	Escribir el nuevo valor predeterminado en el contador rápido: 0 = no actualizar; 1 = actualizar el valor predeterminado
SM37.6	SM47.6	SM57.6	Escribir el nuevo valor actual en el contador rápido: 0 = no actualizar; 1 = actualizar el valor actual
SM37.7	SM47.7	SM57.7	Habilitar el contador rápido: 0 = inhibir el contador rápido; 1 = habilitar el contador rápido

### Ajustar los valores actuales y predeterminados

Cada contador rápido dispone de un valor actual y un valor predeterminado de 32 bits cada uno. 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 10-8 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 y actuales, también es posible leer el valor actual de cada contador rápido, utilizando el tipo de datos HC (valor actual del contador rápido) seguido del número de contador (0, 1 ó 2). Ello permite acceder directamente al valor actual para operaciones de lectura. Por el contrario, dicho valor sólo se puede escribir utilizando la operación HSC que se describe más arriba.

Tabla 10-8 Valores actuales y predeterminados de los contadores HSC0, HSC1 y HSC2

Valor actual de HSC0, HSC1 y HSC2			
HSC0	HSC1	HSC2	Descripción
SM38	SM48	SM58	Byte más significativo del nuevo valor actual de 32 bits
SM39	SM49	SM59	Segundo byte más significativo del nuevo valor actual de 32 bits
SM40	SM50	SM60	Segundo byte menos significativo del nuevo valor actual de 32 bits
SM41	SM51	SM61	Byte menos significativo del nuevo valor actual de 32 bits
Valor predeterminado de HSC0, HSC1 y HSC2			
HSC0	HSC1	HSC2	Descripción
SM42	SM52	SM62	Byte más significativo del nuevo valor predeterminado de 32 bits
SM43	SM53	SM63	Segundo byte más significativo del nuevo valor predeterminado de 32 bits
SM44	SM54	SM64	Segundo byte menos significativo del nuevo valor predeterminado de 32 bits
SM45	SM55	SM65	Byte menos significativo del nuevo valor predeterminado de 32 bits

## Byte de estado

Cada contador rápido dispone 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 10-9 muestra los bits de estado de los contadores rápidos.

Tabla 10-9 Bits de estado de los contadores HSC0, HSC1 y HSC2

HSC0	HSC1	HSC2	Descripción
SM36.0	SM46.0	SM56.0	No utilizado
SM36.1	SM46.1	SM56.1	No utilizado
SM36.2	SM46.2	SM56.2	No utilizado
SM36.3	SM46.3	SM56.3	No utilizado
SM36.4	SM46.4	SM56.4	No utilizado
SM36.5	SM46.5	SM56.5	Bit de estado para sentido de conteo actual: 0 = conteo atrás; 1 = conteo adelante
SM36.6	SM46.6	SM56.6	Bit de estado para valor actual igual a valor predeterminado: 0 = diferente, 1 = igual
SM36.7	SM46.7	SM56.7	Bit de estado para valor actual mayor que valor predeterminado: 0 = menor o igual, 1 = mayor que

### Nota

Los bits de estado para HSC0, HSC1 y HSC2 son válidos únicamente mientras se está procesando 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

HSC0 asiste una condición de interrupción (si el valor actual es igual al valor predeterminado). HSC1 y HSC2 asisten tres condiciones de interrupción: si el valor actual es igual al valor predeterminado, si se activa una entrada de puesta a 0 externa y si cambia el 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 "Operaciones de interrupción".

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 SM47 en la subrutina de inicialización conforme a la operación de control deseada. Ejemplo:  
 SM47 = 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 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 SM48 (valor de palabra doble) (cargue 0 para borrar la marca).
5. Cargue el valor predeterminado deseado en SM52 (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 "Operaciones de interrupción".
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 de HSC1.
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 SM47 en la subrutina de inicialización conforme a la operación de control deseada. Ejemplo:  

**SM47 = 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 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 SM48 (valor de palabra doble) (cargue 0 para borrar la marca).
5. Cargue el valor predeterminado deseado en SM52 (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 "Operaciones de interrupción".
7. Para poder detectar un cambio del sentido de conteo, programe una interrupción asociando el evento de interrupción 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 de HSC1.
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 SM47 en la subrutina de inicialización conforme a la operación de control deseada. Ejemplo:  
 SM47 = 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 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 SM48 (valor de palabra doble) (cargue 0 para borrar la marca).
5. Cargue el valor predeterminado deseado en SM52 (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 "Operaciones de interrupción".
7. Para poder detectar un cambio del sentido de conteo, programe una interrupción asociando el evento de interrupción 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 de HSC1.
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. Llamando a una 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 SM47 en la subrutina de inicialización conforme a la operación de control deseada.

Ejemplo (frecuencia simple):

SM47 = 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 contaje adelante.  
Se ajusta la actividad alta de las entradas de arranque y de puesta a 0.

Ejemplo (frecuencia cuádruple):

SM47 = 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 contaje 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 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 SM48 (valor de palabra doble) (cargue 0 para borrar la marca).
5. Cargue el valor predeterminado deseado en SM52 (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 "Operaciones de interrupción".
7. Para poder detectar un cambio del sentido de contaje, programe una interrupción asociando el evento de interrupción 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 de HSC1.
10. Ejecute la operación HSC para que el sistema de automatización S7-200 programe el contador HSC1.
11. Finalice la subrutina.

### Cambio de sentido (modos 0, 1 ó 2)

Para configurar el cambio de sentido HSC1 como contador de fase simple con control interno del sentido de conteo (modos 0,1 ó 2):

1. Cargue SM47 para ajustar el sentido de conteo deseado:
  - SM47 = 16#90    Habilita el contador.  
Ajusta el sentido de conteo atrás.
  - SM47 = 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)

Para modificar el valor actual del contador HSC1 (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.

1. Cargue SM47 para escribir el valor actual deseado.
  - SM47 = 16#C0    Habilita el contador.  
Escribe el nuevo valor actual.
2. Cargue el valor actual deseado en SM48 (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 SM47 para escribir el valor predeterminado deseado.
  - SM47 = 16#A0    Habilita el contador.  
Escribe el nuevo valor predeterminado.
2. Cargue el valor predeterminado deseado en SM52 (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 SM47 para inhibir el contador.
  - SM47 = 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 conteo, el valor actual o el valor predeterminado. No obstante, también es posible modificar todos o sólo algunos de dichos ajustes en ese mismo orden, definiendo el valor de SM47 conforme a ello y ejecutando luego la operación HSC.

Ejemplo contador rápido



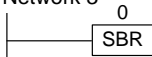
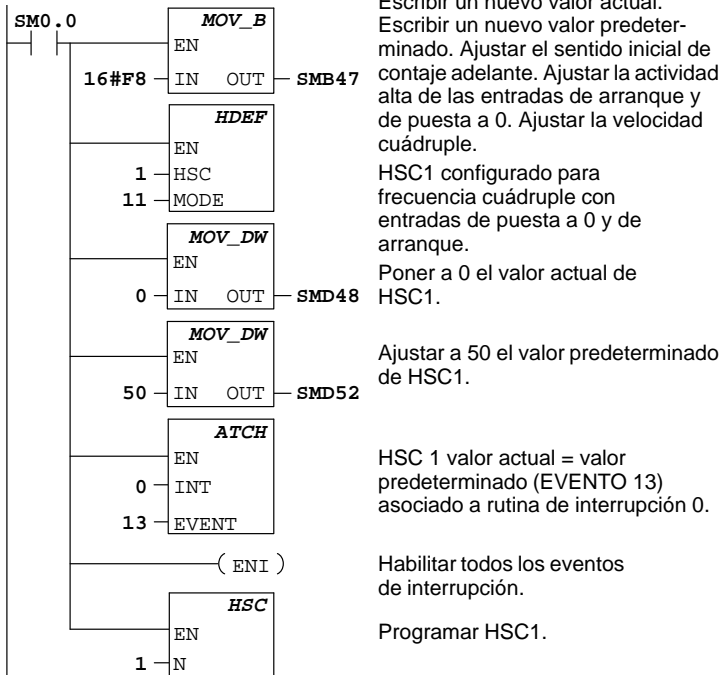


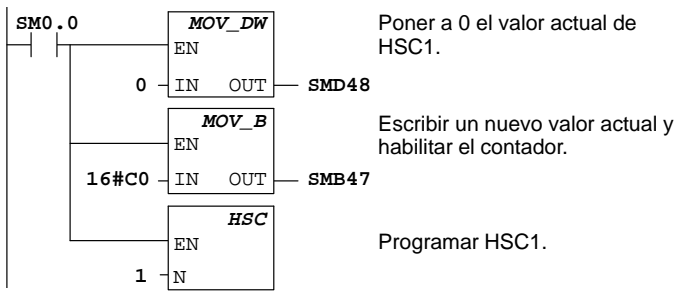

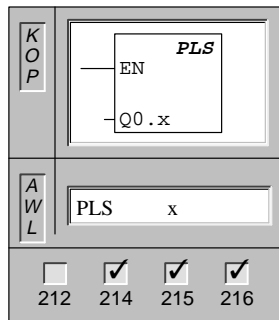
KOP			AWL
Network 1		Llamar subrutina 0 en el primer ciclo.	Network 1 LD SM0.1 CALL 0
Network 2		Fin programa principal.	Network 2 MEND
Network 3		Comienzo subrutina 0.	Network 3 SBR 0
Network 4		Habilitar el contador. Escribir un nuevo valor actual. Escribir un nuevo valor predeterminado. Ajustar el sentido inicial de conteo adelante. Ajustar la actividad alta de las entradas de arranque y de puesta a 0. Ajustar la velocidad cuádruple. HSC1 configurado para frecuencia cuádruple con entradas de puesta a 0 y de arranque. Poner a 0 el valor actual de HSC1. Ajustar a 50 el valor predeterminado de HSC1. HSC 1 valor actual = valor predeterminado (EVENTO 13) asociado a rutina de interrupción 0. Habilitar todos los eventos de interrupción. Programar HSC1.	Network 4 LD SM0.0 MOVB 16#F8, SMB47 HDEF 1, 11 MOVD 0, SMD48 MOVD 50, SMD52 ATCH 0, 13 ENI HSC 1
Network 5		Fin subrutina.	Network 5 RET
Network 6		Comienzo interrupción 0.	Network 6 INT 0
Network 7		Poner a 0 el valor actual de HSC1. Escribir un nuevo valor actual y habilitar el contador. Programar HSC1.	Network 7 LD SM 0.0 MOVD 0, SMD48 MOVB 16#C0, SMB47 HSC 1
Network 8		Finalizar rutina de interrupción.	Network 8 RETI

Figura 10-16 Ejemplo de inicialización de HSC1

## Salida de impulsos



La operación **Salida de impulsos** comprueba las marcas especiales de esta salida de impulsos (0.x). A continuación se llama la operación de impulsos definida por las marcas especiales.

Operandos: x: 0 a 1

## Operaciones rápidas de salida del S7-200

Algunas CPUs permiten utilizar Q0.0 y Q0.1 para generar trenes de impulsos rápidos (PTO) o para 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 (de 250 a 65.535 microsegundos) o en milisegundos (de 2 a 65.535 milisegundos). Si se indica un número impar en microsegundos o milisegundos se distorsiona el factor de trabajo relativo (relación impulso-pausa).

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 se puede indicar en microsegundos (de 250 a 65.535 microsegundos) o en milisegundos (de 2 a 65.535 milisegundos). El tiempo del ancho de impulsos puede estar comprendido entre 0 y 65.535 microsegundos o bien, entre 0 y 65.535 milisegundos. 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.

Si se indica un tiempo de ciclo inferior a dos unidades de tiempo, el tiempo de ciclo se predeterminará en dos unidades de tiempo.

### Nota

En las funciones PTO y PWM, los tiempos de cambio de las salidas de "off" (desactivada) a "on" (activada) y viceversa no son idénticos. Esta diferencia se manifiesta como distorsión del factor de trabajo relativo (relación impulso-pausa). En el Anexo A se indican los tiempos de cambio. 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.

### **Cambiar el ancho de impulsos**

PWM es una función continua. Si se cambia el ancho de impulsos, se deberá inhibir la función PWM provisionalmente durante su actualización. Ello se efectúa de forma asíncrona al ciclo PWM y podría producir vibraciones no deseadas en la unidad que se está controlando. Si se requieren actualizaciones sincrónicas del ancho de impulsos, la salida de impulsos se reconduce a una de las entradas de interrupción (I0.0 a I0.4). El ciclo PWM se sincroniza habilitando la interrupción Flanco positivo en la entrada de la misma, cuando sea necesario cambiar el ancho de impulsos (v. fig. 10-19).

El ancho de impulsos se cambia durante la rutina de interrupción. Ello impide que se produzcan interrupciones excepto cuando se desee cambiar el ancho de impulsos.

### **Llamar a la operación PTO/PWM**

Cada generador PTP/PWM dispone de un byte de control (8 bits), de un valor de tiempo de ciclo y de un valor de ancho de impulsos (ambos son valores de 16 bits sin signo), así como de un valor de conteo de impulsos que es un valor de 32 bits sin signo. Estos valores están almacenados en determinadas áreas de la memoria de marcas especiales. Una vez que se disponga de las direcciones de marcas especiales para la función deseada, ésta se llama ejecutando la operación Salida de impulsos (PLS). Con esta operación, la CPU S7-200 lee las direcciones de las marcas especiales y programa el generador PTO/PWM.

### **Pipeline PTO**

Además de las informaciones de control, existen dos bits de estado que se utilizan con la función PTO. Estos bits indican si se ha generado el número de impulsos indicado o si se ha producido un desbordamiento de la pipeline.

La función PTO permite combinar las definiciones de dos salidas de impulsos, o bien encadenarlas conforme al sistema pipeline. Así es posible asistir la continuidad de trenes de impulsos de salida. La pipeline se carga configurando en primer lugar la primera definición PTO y ejecutando luego la operación PLS. Inmediatamente después de ejecutar la operación PLS se puede configurar la segunda definición y ejecutar otra operación PLS.

En caso de configurar una tercera definición antes de finalizar la primera operación PTO (es decir, antes de generarse todos los impulsos de salida de la primera definición), se activa (se pone a 1) la marca Desbordamiento pipeline PTO (SM66.6 o SM76.6). Al pasar a RUN, se vuelve a poner a 0. Si se detecta un desbordamiento, el programa debe poner el bit a 0 para que se puedan detectar otros desbordamientos posteriores.

La tabla 10-10 muestra las direcciones de las marcas especiales (SM) para las salidas de impulsos 0 y 1.

---

#### **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.

---

Tabla 10-10 Direcciones PTO/PWM para el encadenamiento pipeline de dos salidas de impulsos

Q0.0	Q0.1	Bits de estado para las salidas de impulsos
SM66.6	SM76.6	Desbordamiento pipeline PTO 0 = sin desbordamiento; 1 = desbordamiento
SM66.7	SM76.7	PTO en vacío 0 = ejecución; 1 = PTO en vacío
Q0.0	Q0.1	Bits de control para salidas PTO/PWM
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 conteaje de impulsos PTO 0 = no actualizar; 1 = actualizar valor de conteaje de impulsos
SM67.3	SM77.3	Elegir base de tiempo PTO/PWM 0 = 1 $\mu$ s/ciclo; 1 = 1ms/ciclo
SM67.4	SM77.4	No utilizado
SM67.5	SM77.5	No utilizado
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	Valores de tiempo de ciclo para salidas PTO/PWM (margen: 2 a 65.535)
SM68	SM78	Byte más significativo del valor de tiempo de ciclo PTO/PWM
SM69	SM79	Byte menos significativo del valor de tiempo de ciclo PTO/PWM
Q0.0	Q0.1	Valores de ancho de impulsos para salidas PWM (margen: 0 a 65.535)
SM70	SM80	Byte más significativo del valor de ancho de impulsos PWM
SM71	SM81	Byte menos significativo del valor de ancho de impulsos PWM
Q0.0	Q0.1	Valores de conteaje para salidas de impulsos (margen: 1 a 4.294.967.295)
SM72	SM82	Byte más significativo del valor de conteaje de impulsos PTO
SM73	SM83	Segundo byte más significativo del valor de conteaje de impulsos PTO
SM74	SM84	Segundo byte menos significativo del valor de conteaje de impulsos PTO
SM75	SM85	Byte menos significativo del valor de conteaje de impulsos PTO

Con ayuda de la tabla 10-11 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.

Tabla 10-11 Tabla de referencia PTO/PWM para valores hexadecimales

Registro de control (valor hexadecimal)	Resultado de la operación PLS					
	Habilitar	Modo	Base de tiempo	Valor de conteo	Ancho de impulso	Tiempo de ciclo
16#81	Sí	PTO	1 $\mu$ s/ciclo			Cargar
16#84	Sí	PTO	1 $\mu$ s/ciclo	Cargar		
16#85	Sí	PTO	1 $\mu$ s/ciclo	Cargar		Cargar
16#89	Sí	PTO	1 ms/ciclo			Cargar
16#8C	Sí	PTO	1 ms/ciclo	Cargar		
16#8D	Sí	PTO	1 ms/ciclo	Cargar		Cargar
16#C1	Sí	PWM	1 $\mu$ s/ciclo			Cargar
16#C2	Sí	PWM	1 $\mu$ s/ciclo		Cargar	
16#C3	Sí	PWM	1 $\mu$ s/ciclo		Cargar	Cargar
16#C9	Sí	PWM	1 ms/ciclo			Cargar
16#CA	Sí	PWM	1 ms/ciclo		Cargar	
16#CB	Sí	PWM	1 ms/ciclo		Cargar	Cargar

### Inicialización y operaciones de las funciones PTO/PWM

Para comprender mejor el funcionamiento de las funciones PTO y PWM, se describe seguidamente su inicialización paso a paso, así como las correspondientes operaciones. En las descripciones se utiliza la salida 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 con otra condición y no con la marca del primer ciclo.

## Inicializar la función PWM

Para inicializar la función PWM para Q0.0:

1. Con la marca del primer ciclo, ponga la salida a 1 y llame a la subrutina requerida 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#C3 para PWM en la marca SM67 para incrementar en microsegundos (o cargue 16#CB 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 SM68 (valor de palabra).
4. Cargue el ancho de impulsos deseado en SM70 (valor de palabra).
5. Ejecute la operación PLS para que el S7-200 programe el generador PTO/PWM.
6. Cargue el valor 16#C2 en SM67 para incrementar en microsegundos (o 16#CA si desea incrementar en milisegundos). Así se borra el tiempo de ciclo actualizado en el byte de control, pudiéndose modificar el ancho de impulsos. Se carga un nuevo ancho de impulsos y luego se ejecuta la operación PLS. El byte de control permanece inalterado.
7. Finalice la subrutina.

Pasos opcionales para actualizaciones síncronas. A continuación se explican los pasos opcionales para las actualizaciones síncronas:

1. Ejecute la operación Habilitar todos los eventos de interrupción (ENI).
2. Asocie una rutina de interrupción (ATCH) al evento Flanco positivo mediante la condición que se va a utilizar para actualizar el ancho de impulsos. La condición sólo deberá estar activa durante un ciclo.
3. Agregue una rutina de interrupción que actualice el ancho de impulso y que inhiba luego la interrupción de flanco.

---

### Nota

Los pasos opcionales para actualizaciones síncronas exigen que la salida PWM se vuelva a conducir a una de las entradas de interrupción.

---

## Cambiar el ancho de impulsos para las salidas PWM

Para cambiar el ancho de impulsos para salidas PWM en una subrutina:

1. Llame a una subrutina para cargar el ancho de impulsos deseado en SM70 (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**

Para inicializar la función PTO:

1. Con la marca del primer ciclo, ponga la salida a 0 y llame a la subrutina requerida para ejecutar la inicialización. Si se llama a una 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#85 para PTO en la marca SM67 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 SM68 (valor de palabra).
4. Cargue el ancho de impulsos deseado en SM72 (valor de palabra doble).
5. El siguiente 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 y ejecutando la operación Habilitar todos los eventos de interrupción. Para obtener más información acerca de la utilización de interrupciones, consulte el apartado 10.14 ("Operaciones de interrupción").
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**

Para cambiar el tiempo de ciclo PTO en una subrutina o en una rutina de interrupción:

1. Cargue 16#81 para PTO en SM67 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 SM68 (valor de palabra).
3. Ejecute la operación PLS para que el S7-200 programe el generador PTO/PWM.
4. Finalice la subrutina o la rutina de interrupción. (Las subrutinas no se pueden llamar desde rutinas de interrupción).

### **Cambiar el valor de conteo PTO**

Para cambiar el valor de conteo PTO en una subrutina o en una rutina de interrupción:

1. Cargue 16#84 para PTO en SM67 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 ancho de impulsos deseado en SM72 (valor de palabra doble).
3. Ejecute la operación PLS para que el S7-200 programe el generador PTO/PWM.
4. Finalice la subrutina o la rutina de interrupción. (Las subrutinas no se pueden llamar desde rutinas de interrupción).

**Cambiar el tiempo de ciclo PTO y el valor de conteo de impulsos**

Para cambiar el tiempo de ciclo PTO y el valor de conteo de impulsos en una subrutina o en una rutina de interrupción:

1. Cargue 16#85 para PTO en SM67 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 valor de conteo de impulsos.
2. Cargue el tiempo de ciclo deseado en SM68 (valor de palabra).
3. Cargue el ancho de impulsos deseado en SM72 (valor de palabra doble).
4. Ejecute la operación PLS para que el S7-200 programe el generador PTO/PWM.
5. Finalice la subrutina o la rutina de interrupción. (Las subrutinas no se pueden llamar desde rutinas de interrupción).

**Funciones PTO/PWM activas**

Si una de las operaciones PTO o PWM está activada en Q0.0 o en Q0.1, entonces se inhibe el uso normal de ambas salidas. Durante ese tiempo, ni los valores almacenados en la imagen del proceso de las salidas ni los valores que se hayan forzado para las mismas, se transfieren a la salida correspondiente. La operación PTO está activa cuando se ha habilitado sin haberse terminado aún su ejecución. Mediante operaciones directas de salida que escriben valores en dichas salidas mientras está activada la función PTO/PWM se impide que se distorsione la forma de onda de PTO o PWM.

---

**Nota**

Si una función PTO se inhibe antes de finalizar, termina el tren de impulsos actual y la entrada Q0.0 ó Q0.1 retorna a la ejecución normal de la imagen del proceso. Al habilitarse nuevamente la función PTO, el tren de impulsos reanuda desde el comienzo, utilizando el último valor de salida de impulsos que se haya cargado.

---

## Efecto en las salidas

La operación PTO/PWM y la imagen del proceso utilizan conjuntamente las salidas Q0.0 y Q0.1. Los estados inicial y final de la forma de onda de las operaciones PTO y PWM pueden quedar afectados por el valor de la correspondiente imagen del proceso. Si se emite un tren de impulsos por Q0.0 o Q0.1, la imagen del proceso determina los estados inicial y final de la salida y hace que la salida de impulsos se efectúe desde un nivel superior o inferior.

Debido a que ambas operaciones (PTO y PWM) se bloquean temporalmente cuando se cambia de la pipeline PTO al ancho de impulsos, en dichos puntos de transición se puede presentar una pequeña discontinuidad de la forma de onda de las salidas. Para minimizar los efectos negativos de dicha discontinuidad, el bit de la imagen del proceso siempre deberá estar puesto a 0 al utilizarse la operación PTO, o bien puesto a 1 en el caso de la operación PWM. La figura 10-17 muestra la forma de onda resultante de ambas operaciones. En el caso de la operación PTO también se debe tener en cuenta que en el punto de transición se reduce el último medio ciclo a un ancho de impulsos de aproximadamente 120  $\mu$ s. En el caso de la operación PWM con actualización síncrona opcional, el primer impulso alto de tiempo se prolonga tras el punto de transición aprox. 120  $\mu$ s.

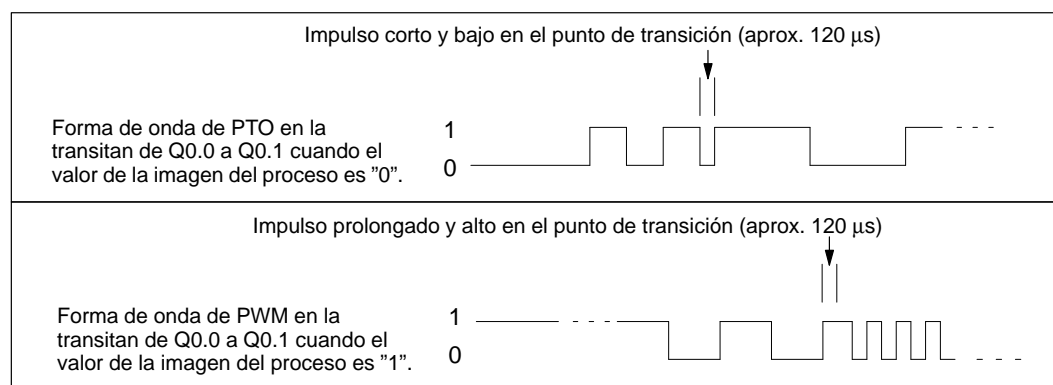


Figura 10-17 Ejemplo de la forma de onda de un tren de impulsos en Q0.0 y Q0.1

### Ejemplo de la función Tren de impulsos (PTO)

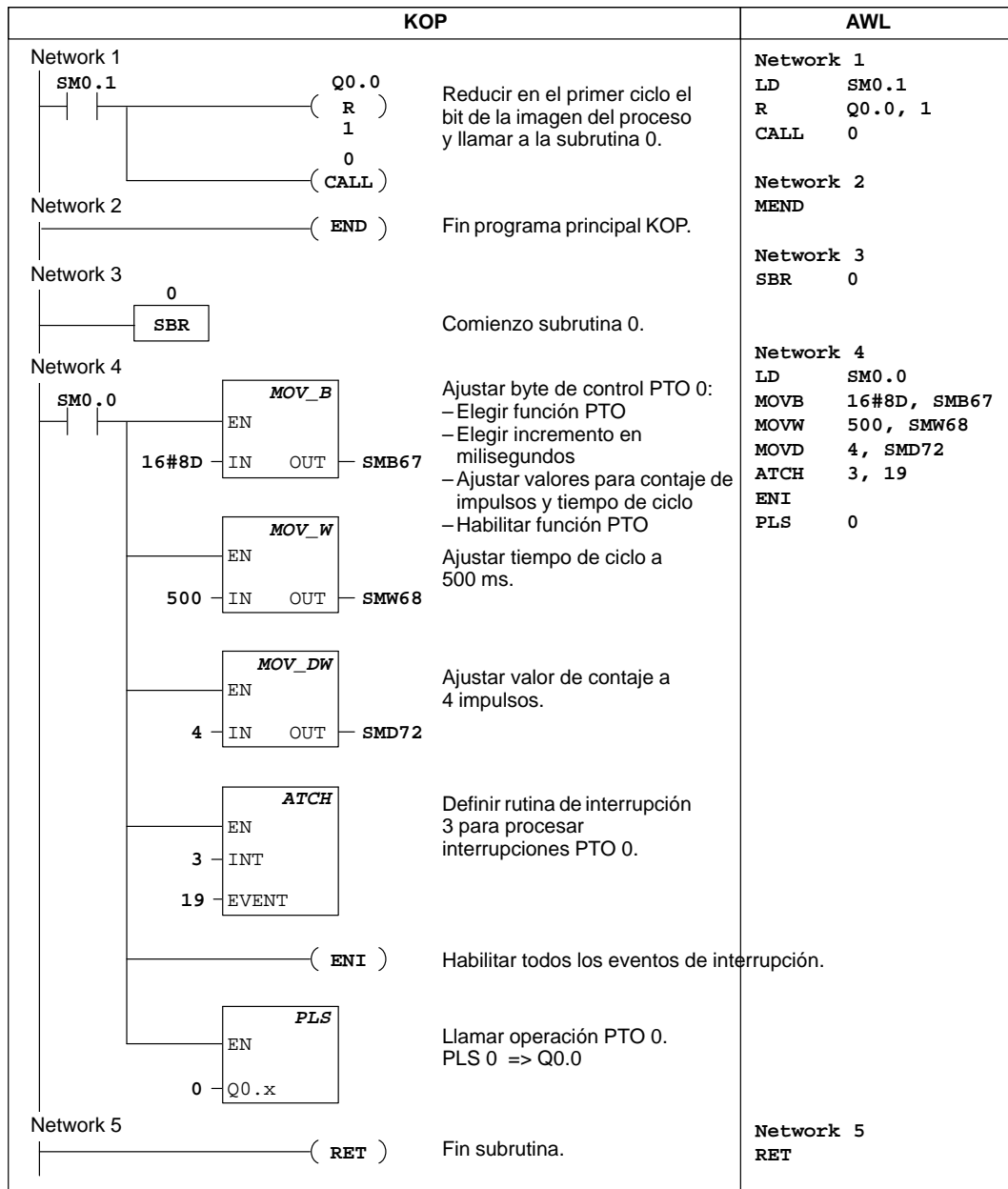


Figura 10-18 Ejemplo de la función Tren de impulsos (PTO)

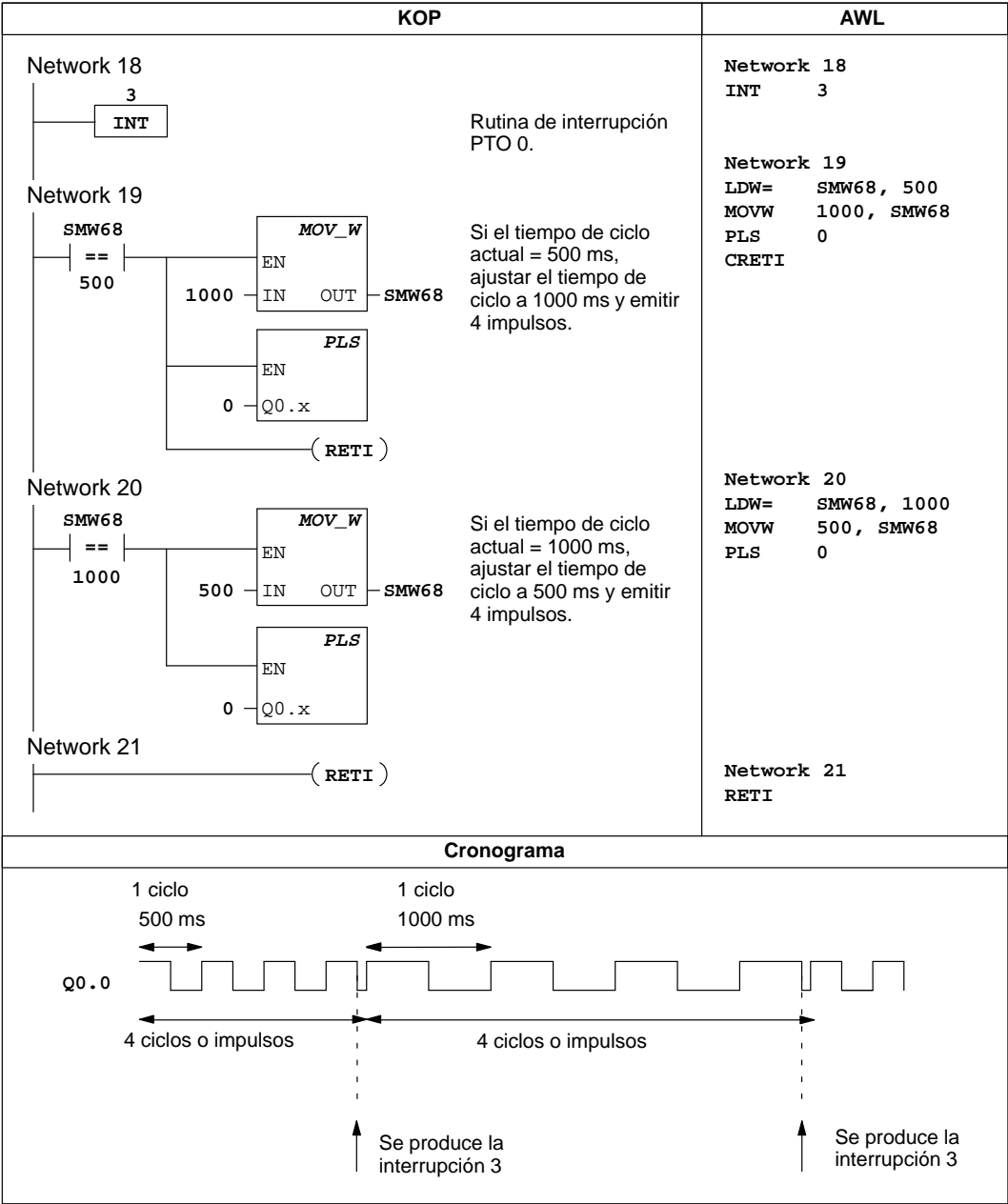


Figura 10-18Ejemplo de la función Tren de impulsos (continuación)

### Ejemplo de la función Modulación del ancho de impulsos (PWM)

La figura 10-19 muestra un ejemplo de la operación Modulación del ancho de impulsos (PWM). Si se cambia el ancho de impulsos, se deberá inhibir la función PWM provisionalmente durante su actualización. Ello se efectúa de forma asíncrona al ciclo PWM y podría producir vibraciones no deseadas en la unidad que se está controlando. Si se requieren actualizaciones síncronas del ancho de impulsos, la salida de impulsos se deberá reconducir a la entrada de interrupción (I0.0). Cuando sea necesario modificar el ancho de impulsos, se habilita la entrada de interrupción. El ancho de impulsos cambiará entonces de forma síncrona con el ciclo PWM cuando se produzca el siguiente flanco positivo en I0.0.

El ancho de impulsos se cambia durante la rutina de interrupción. Ello impide que se produzcan interrupciones excepto cuando se desee cambiar el ancho de impulsos.

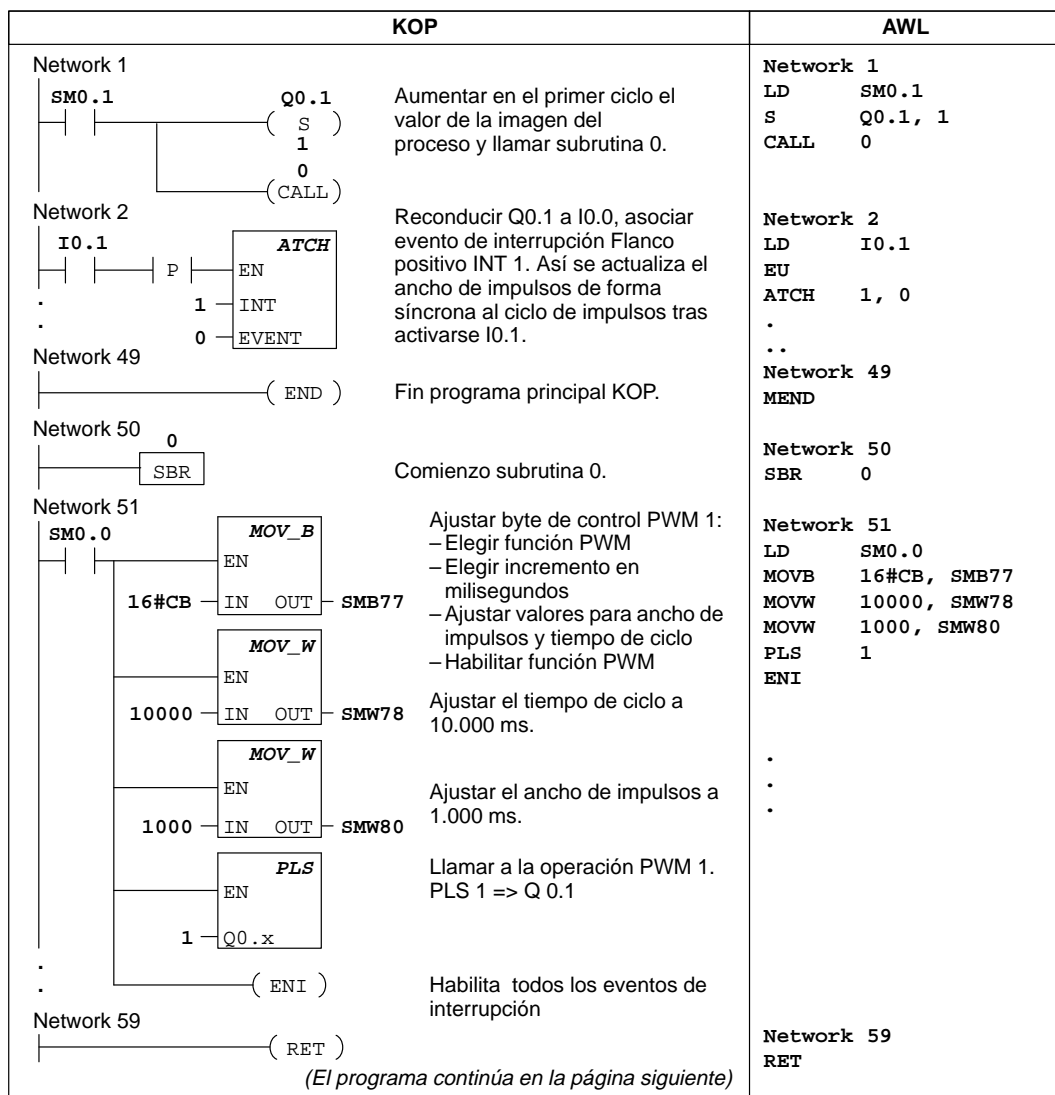


Figura 10-19 Ejemplo de operaciones rápidas de salida con modulación del ancho de impulsos

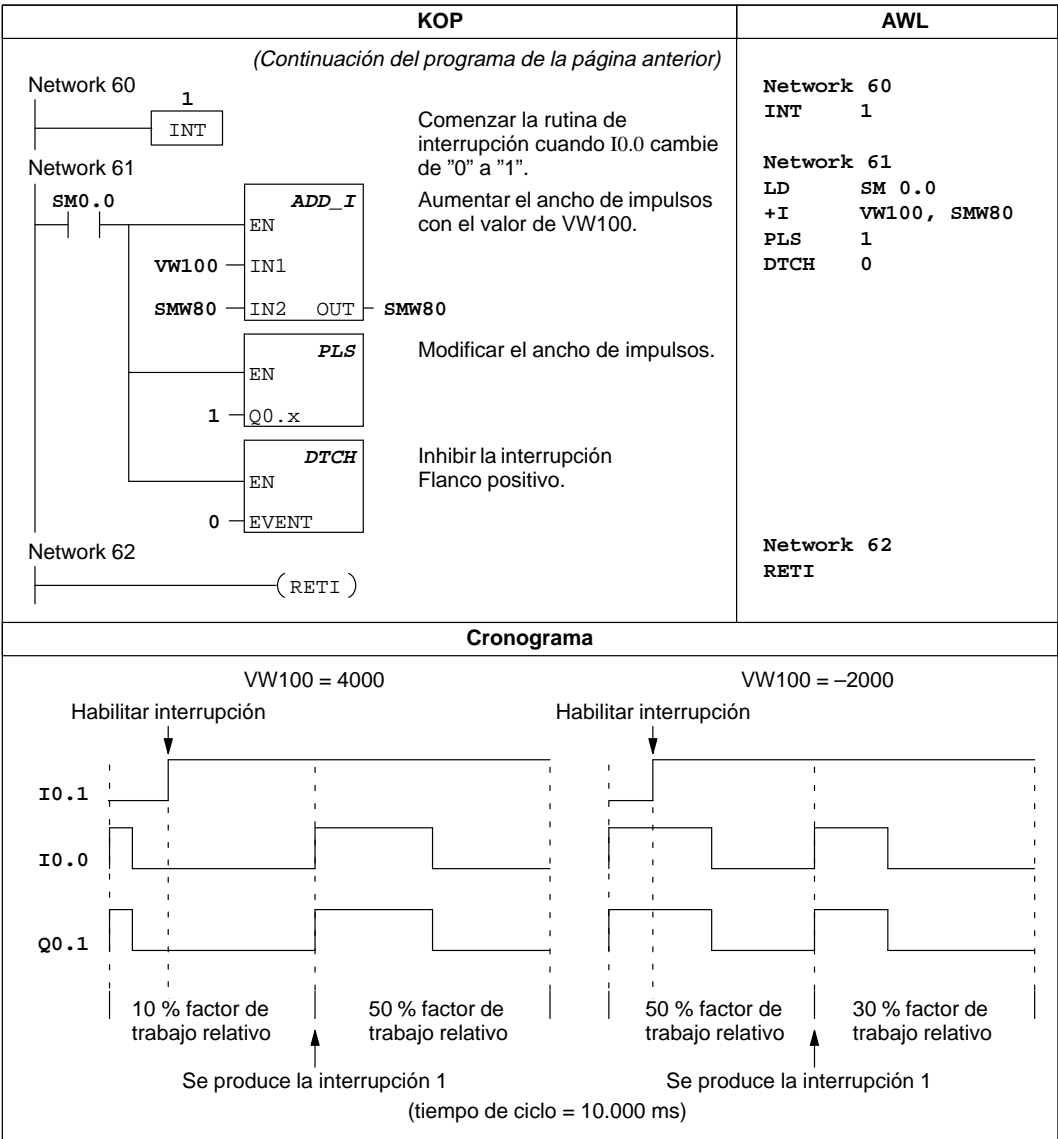
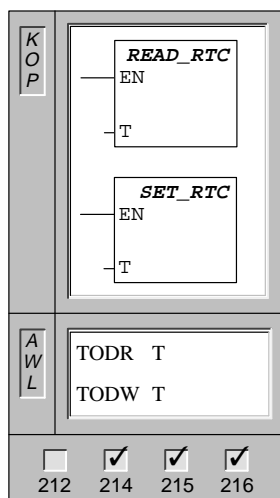


Figura 10-19 Ejemplo de operaciones rápidas de salida con modulación del ancho de impulsos (continuación)

## 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 READ\_RTC y SET\_RTC se representan mediante las operaciones TODR (Leer reloj de tiempo real) y TODW (Escribir reloj de tiempo real).

Operandos: T: VB, IB, QB, MB, SMB, \*VD, \*AC, SB

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 representará como "00" (el reloj pasará de 99 a 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	000d	d – 0 a 7	1 = Domingo
			0 = desactiva el día de la semana (permanece 0)

### Nota

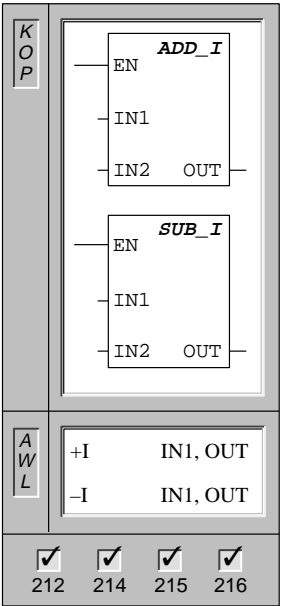
La CPU S7-200 no comprueba si el día de la semana coincide con la fecha. Así puede ocurrir que se acepten fechas no válidas, p.ej. el 30 de febrero. Asegúrese de que los datos introducidos sean correctos.

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 ejecutará. Entonces se activará SM4.3, indicando que dos operaciones intentaron acceder simultáneamente al reloj.

El sistema de automatización S7-200 no utiliza la información relativa al año de ninguna forma y no se verá 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.

## 10.6 Operaciones aritméticas y de regulación PID

### 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, dando como resultado 16 bits (OUT).

Operandos: IN1, IN2: VW, T, C, IW, QW, MW, SMW, AC, AIW, constante, \*VD, \*AC, SW

OUT: VW, T, C, IW, QW, MW, SMW, AC, \*VD, \*AC, SW

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

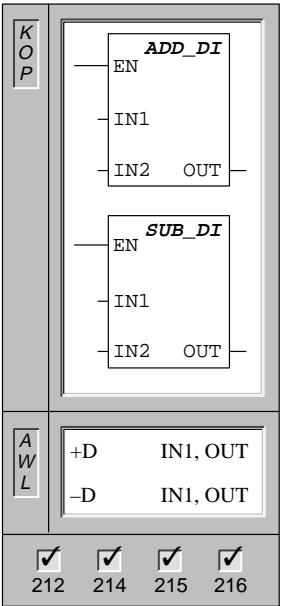
En AWL:  $IN1 + OUT = OUT$   
 $OUT - IN1 = OUT$

Nota: Al programar en KOP se puede reducir el espacio de memoria necesario indicando que IN1 sea igual a OUT.

Estas operaciones afectan a las siguientes marcas especiales:

SM1.0 (cero); SM1.1 (desbordamiento); SM1.2 (negativo)

### 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, dando como resultado 32 bits (OUT).

Operandos: IN1, IN2: VD, ID, QD, MD, SMD, AC, HC, constante, \*VD, \*AC, SD

OUT: VD, ID, QD, MD, SMD, AC, \*VD, \*AC, SD

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

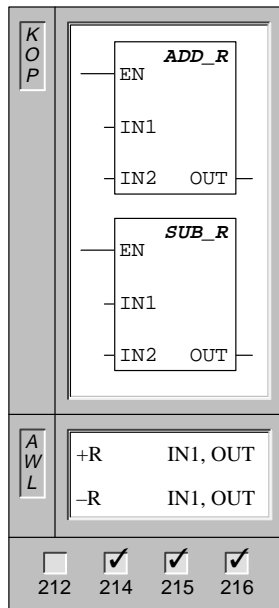
En AWL:  $IN1 + OUT = OUT$   
 $OUT - IN1 = OUT$

Nota: Al programar en KOP se puede reducir el espacio de memoria necesario indicando que IN1 sea igual a OUT.

Estas operaciones afectan a las siguientes marcas especiales:

SM1.0 (cero); SM1.1 (desbordamiento); SM1.2 (negativo)

## Sumar y restar reales



Las operaciones **Sumar reales** / **Restar reales** suman / restan dos números reales de 32 bits, dando como resultado un número real de 32 bits (OUT).

Operandos: IN1, IN2: VD, ID, QD, MD, SMD, AC, constante  
\*VD, \*AC, SD

OUT: VD, ID, QD, MD, SMD, AC, \*VD, \*AC, SD

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

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

Nota: Al programar en KOP se puede reducir el espacio de memoria necesario indicando que IN1 sea igual a OUT.

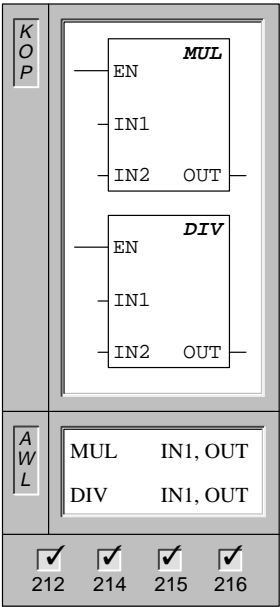
Estas operaciones afectan a las siguientes marcas especiales:

SM1.0 (cero); SM1.1 (desbordamiento); SM1.2 (negativo)

### 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 enteros de 16 bits



La operación **Multiplicar enteros de 16 bits** multiplica dos números enteros de 16 bits, dando un resultado 32 bits (OUT).

En AWL, la palabra menos significativa (16 bits) de OUT se utiliza como uno de los factores.

La operación **Dividir enteros de 16 bits** divide dos números enteros de 16 bits, dando un resultado de 32 bits (OUT) compuesto de un cociente de 16 bits (los menos significativos) y un resto de 16 bits (los más significativos).

En AWL, la palabra menos significativa (16 bits) de OUT se utiliza como dividendendo.

Operandos: IN1, IN2: VW, T, C, IW, QW, MW, SMW, AC, AIW, constante, \*VD, \*AC, SW

OUT: VD, ID, QD, MD, SMD, AC, \*VD, \*AC, SD

En KOP:  $IN1 * IN2 = OUT$   
 $IN1 / IN2 = OUT$

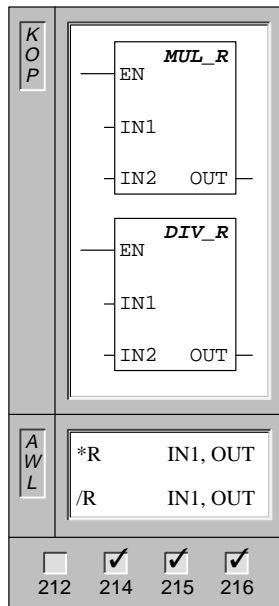
En AWL:  $IN1 * OUT = OUT$   
 $OUT / IN1 = OUT$

Nota: Al programar en KOP se puede reducir el espacio de memoria necesario indicando que IN1 sea igual a OUT.

Estas operaciones afectan a las siguientes marcas especiales:

SM1.0 (cero); SM1.1 (desbordamiento); SM1.2 (negativo); SM1.3 (división por cero)

## Multiplicar y dividir reales



La operación **Multiplicar reales** multiplica dos números reales de 32 bits, dando como resultado un número real de 32 bit (OUT).

La operación **Dividir reales** divide entre sí dos números reales de 32 bits, dando como resultado un cociente de número real de 32 bits.

Operandos: IN1, IN2: VD, ID, QD, MD, SMD, AC, constante, \*VD, \*AC, SD

OUT: VD, ID, QD, MD, SMD, AC, \*VD, \*AC, SD

En KOP:  $IN1 * IN2 = OUT$   
 $IN1 / IN2 = OUT$

En AWL:  $IN1 * OUT = OUT$   
 $OUT / IN1 = OUT$

Nota: Al programar en KOP se puede reducir el espacio de memoria necesario indicando que IN1 sea igual a OUT.

Estas operaciones afectan a las siguientes marcas especiales:

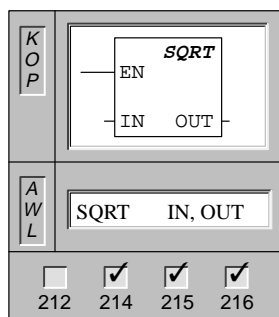
SM1.0 (cero); SM1.1 (desbordamiento); SM1.2 (negativo); SM1.3 (división por cero)

Si se activan SM1.1 o SM1.3, no se modifican los demás bits de estado para operaciones aritméticas ni los operandos originales de las entradas.

### 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.

## Raíz cuadrada



La operación **Raíz cuadrada de números reales** 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:

$$\sqrt{IN} = OUT$$

Operandos: IN: VD, ID, QD, MD, SMD, AC, constante, \*VD, \*AC, SD

OUT: VD, ID, QD, MD, SMD AC, \*VD, \*AC, SD

Estas operaciones afectan a las siguientes marcas especiales:

SM1.0 (cero); SM1.1 (desbordamiento); SM1.2 (negativo)

Ejemplos de operaciones aritméticas

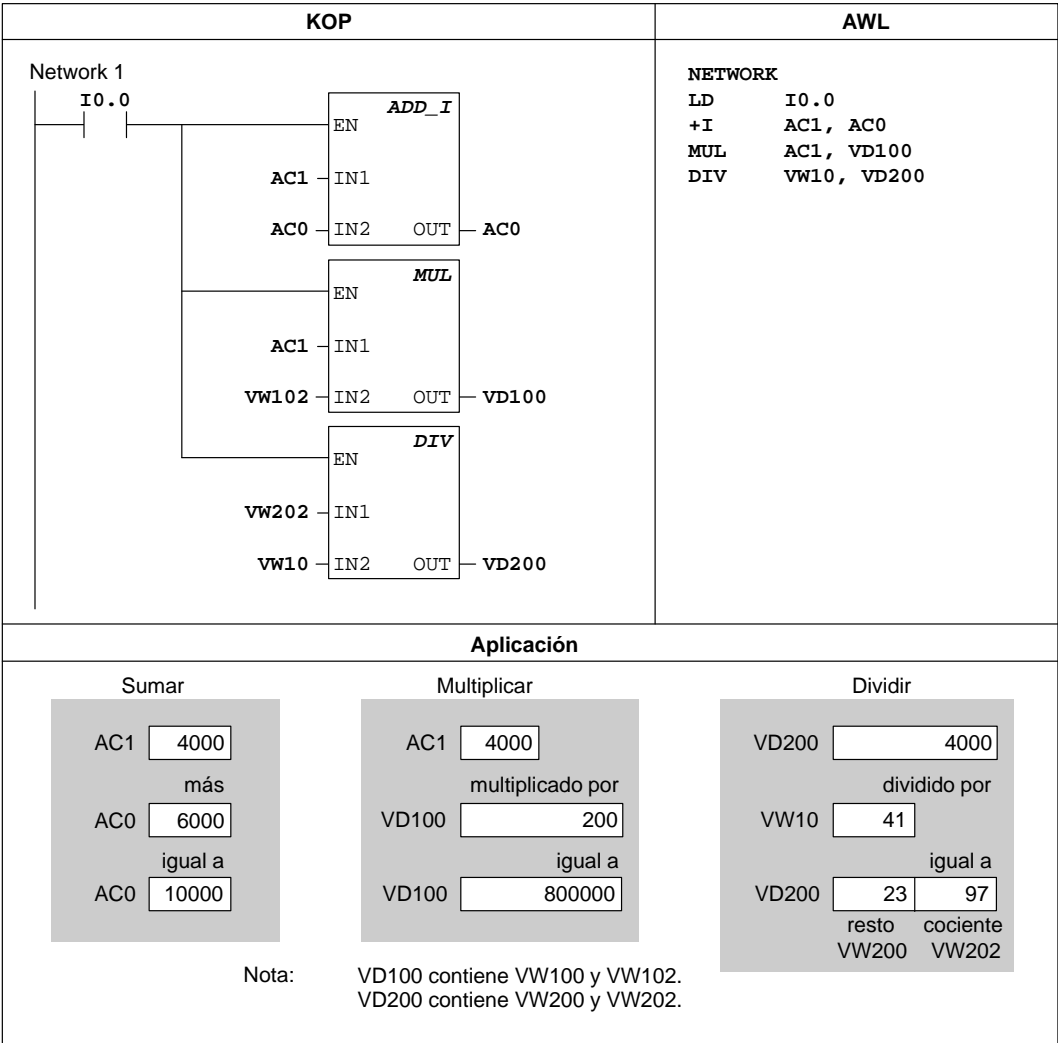
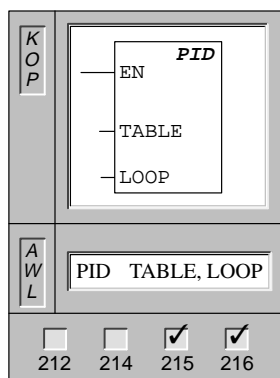


Figura 10-20 Ejemplos de operaciones aritméticas en KOP y AWL

## 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 la TABLE.

Operandos:   Tabla:    VB  
                   Loop:    0 a 7

Estas operaciones afectan a la siguiente marca especial:  
 SM1.1 (desbordamiento)

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 en 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.

## 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<sub>C</sub>     es la ganancia del lazo  
 e        es el error de regulación (diferencia entre consigna y variable de proceso)  
 M<sub>inicial</sub> 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:

$M_n$	=	$K_C * e_n$	+	$K_I * \sum_1^n$	+	$M_{inicial}$	+	$K_D * (e_n - e_{n-1})$
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, sólo es necesario 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 es como la siguiente:

$M_n$	=	$MP_n$	+	$MI_n$	+	$MD_n$
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"). 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 como 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 sólo es necesario 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:

$R_{\text{Norm}}$	es la representación como número real normalizado del valor físico
$R_{\text{No norm}}$	es la representación como valor 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 puede 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:

$R_{\text{Scal}}$	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:

TRUNC	AC0, AC0	// Convertir el número real en un entero de 32 bits.
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 a las fórmulas siguientes:

$$MX = 1.0 - (MP_n + MD_n) \quad \text{si la salida calculada, } M_n > 1.0$$

o

$$MX = - (MP_n + MD_n) \quad \text{si la salida calculada, } M_n < 0.0$$

donde:

MX	es el valor de la suma integral ajustada
MP <sub>n</sub>	es el valor del término proporcional de salida del lazo en el muestreo n-ésimo
MD <sub>n</sub>	es el valor del término diferencial de la salida del lazo en el muestreo n-ésimo
M <sub>n</sub>	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 cuando se finaliza cada 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 dicho 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 dicho bit de historial para detectar una transición de circulación de corriente de 0 a 1. Cuando se detecta dicha transición, la operación ejecuta una serie de acciones destinadas a lograr un cambio sin choques de modo manual a 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<sub>n</sub> 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 circulación de corriente de 0 a 1:

- Ajusta consigna (SP<sub>n</sub>) = variable de proceso (PV<sub>n</sub>)
- Ajusta variable del proceso antigua (PV<sub>n-1</sub>) = variable del proceso (PV<sub>n</sub>)
- Ajusta suma integral (MX) = valor de salida (M<sub>n</sub>)

El estado por defecto de los bits de historial PID es "activado"; dicho 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 transición de 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, éste 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 verifica si ciertos valores de entrada en la tabla del lazo son conformes con 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 dichos valores y corregir el valor de entrada que ha causado el error matemático antes de efectuar la siguiente ejecución de la operación de regulación PID).

### Tabla del lazo

La tabla de lazo tiene 36 bytes de longitud y el formato que muestra la tabla 10-12:

Tabla 10-12 Formato de la tabla del lazo

Offset	Campo	Formato	Tipo de entrada	Descripción
0	Variable del proceso ( $PV_n$ )	Palabra doble – real	entrada	Contiene la variable del proceso que debe estar escalada entre 0.0 y 1.0.
4	Consigna ( $SP_n$ )	Palabra doble – real	entrada	Contiene la consigna que debe estar escalada entre 0.0 y 1.0.
8	Salida ( $M_n$ )	Palabra doble – real	entrada /salida	Contiene la salida calculada, escalada entre 0.0 y 1.0.
12	Ganancia ( $K_C$ )	Palabra doble – real	entrada	Contiene la ganancia, que es una constante proporcional. Puede ser un número positivo o negativo.
16	Tiempo de muestreo ( $T_S$ )	Palabra doble – real	entrada	Contiene, en segundos, el tiempo de muestreo. Tiene que ser un número positivo.
20	Tiempo de acción integral ( $T_I$ )	Palabra doble – real	entrada	Contiene, en minutos, el tiempo de acción integral. Tiene que ser un número positivo.
24	Tiempo de acción derivada ( $T_D$ )	Palabra doble – real	entrada	Contiene, en minutos, el tiempo de acción derivada. Tiene que ser un número positivo.
28	Suma integral (MX)	Palabra doble – real	entrada /salida	Contiene el valor de la suma integral entre 0.0 y 1.0.
32	Variable del proceso previa ( $PV_{n-1}$ )	Palabra doble – real	entrada /salida	Contiene el valor previo de la variable del proceso almacenada desde la última ejecución de la operación PID.

### Ejemplo de programa PID

En este ejemplo se utiliza un tanque para mantener una presión de agua constante. Para ello se toma continuamente agua del tanque en una cantidad variable. Una bomba de velocidad variable se utiliza para añadir agua al tanque con un caudal apto para mantener una presión adecuada del agua y evitar así el vaciado del tanque.

La consigna de este sistema es el nivel de agua; en este caso, un valor equivalente al 75% de llenado del tanque. La variable del proceso la entrega un sensor de flotador que señala el nivel de llenado del tanque; equivale a 0 % cuando está vacío y 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 de 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 vale 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 valen:

$K_C$  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 tanque esté lleno en un 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 10-21 muestra el programa de control (regulación) para esta aplicación.

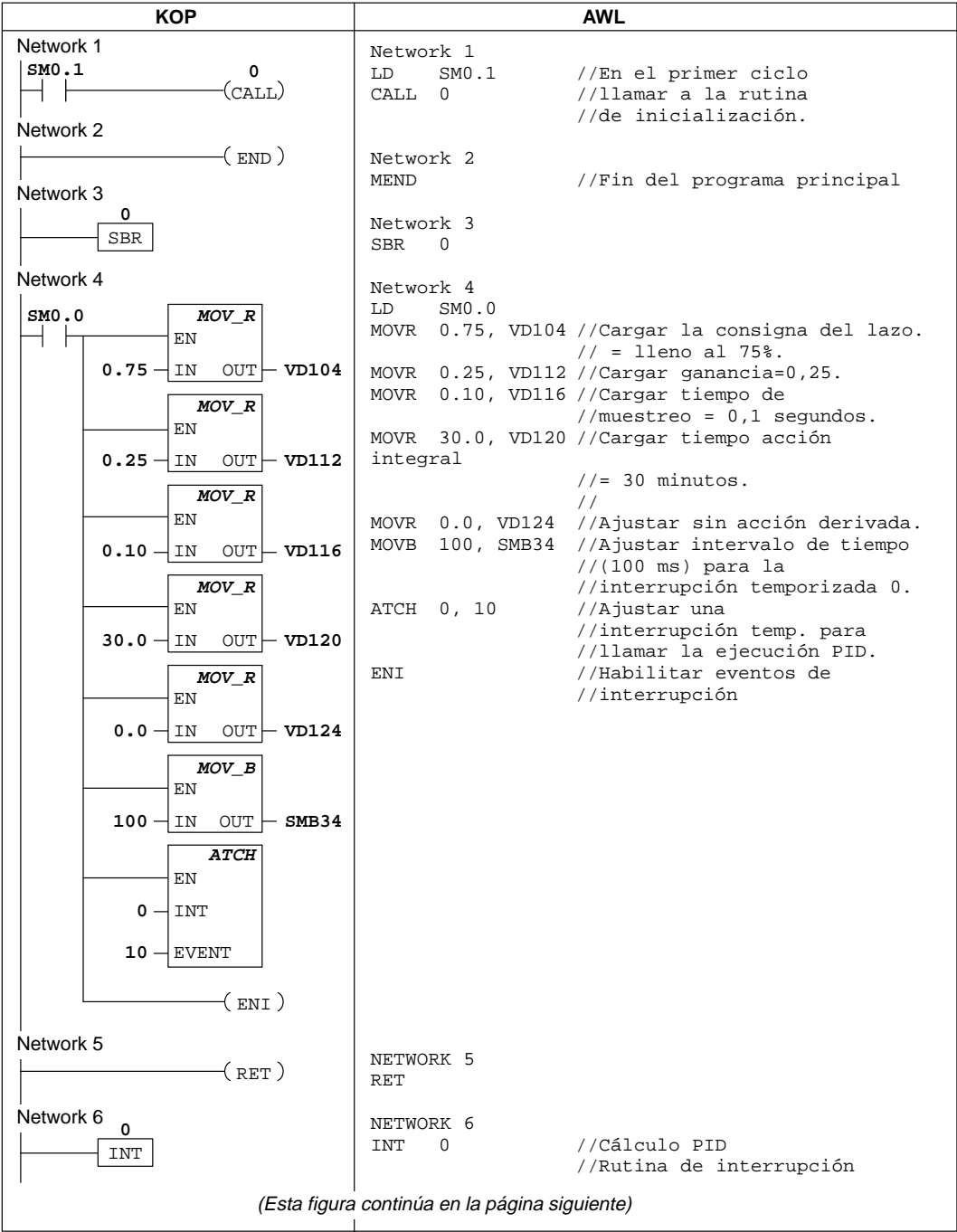


Figura 10-21 Ejemplo de regulación PID

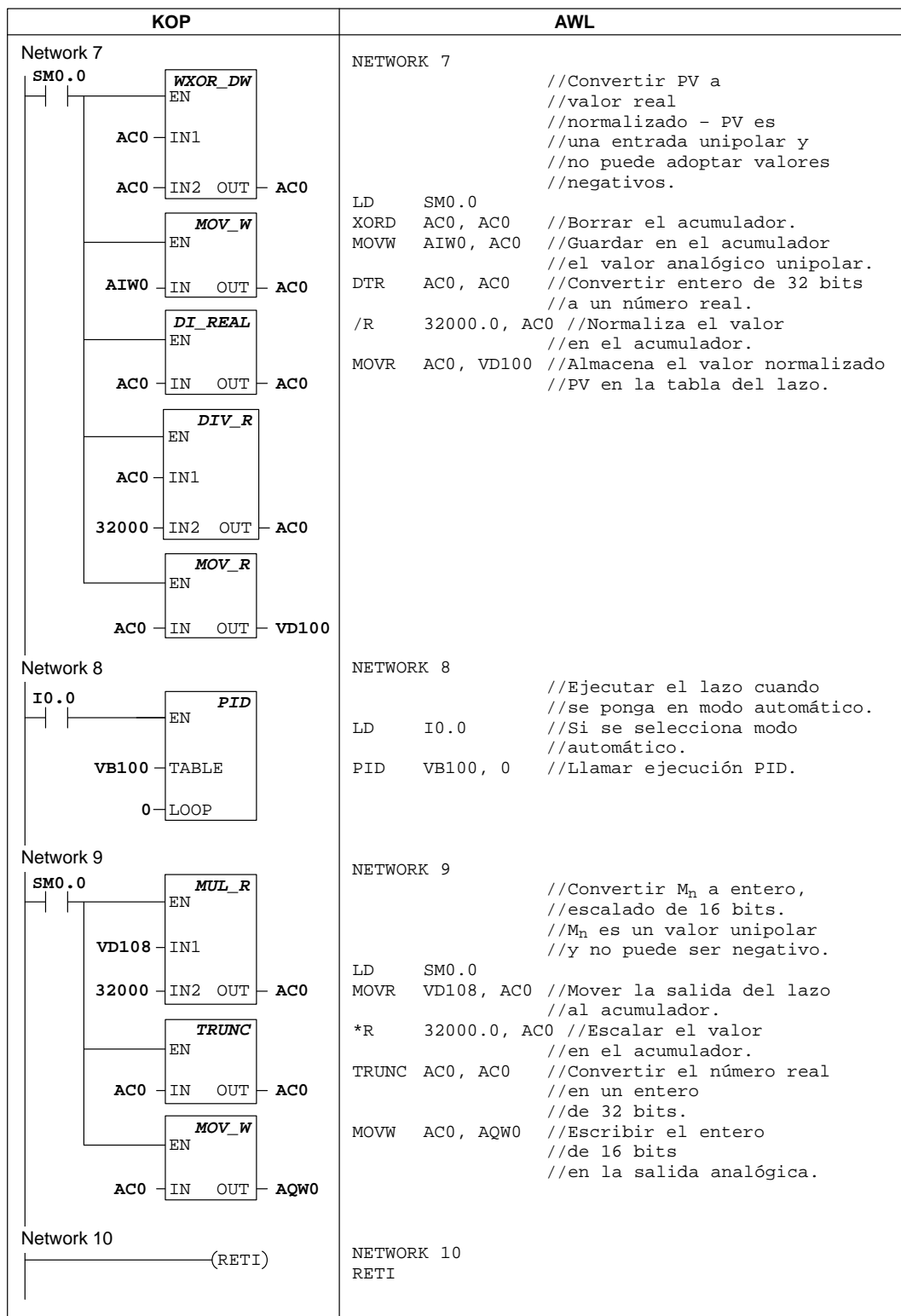
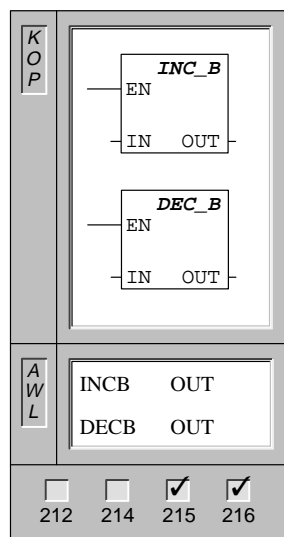


Figura 10-21 Ejemplo de regulación PID (continuación)

## 10.7 Operaciones para incrementar y decrementar

### Incrementar byte, Decrementar byte



Las operaciones **Incrementar byte** / **Decrementar byte** suman / restan 1 al valor del byte de entrada.

Operandos: IN: VB, IB, QB, MB, SMB, SB, AC, constante, \*VD, \*AC, SB

OUT: VB, IB, QB, MB, SMB, SB, AC, \*VD, \*AC, SB

En KOP:  $IN + 1 = OUT$   
 $IN - 1 = OUT$

En AWL:  $OUT + 1 = OUT$   
 $OUT - 1 = OUT$

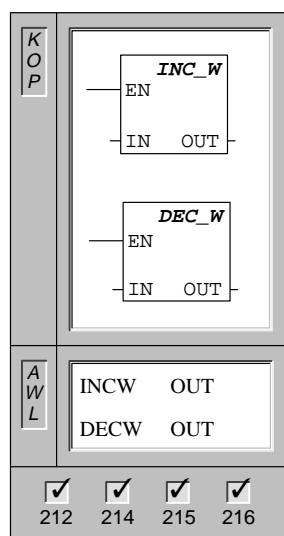
Las operaciones Incrementar y Decrementar byte no llevan signo.

Nota: Al programar en KOP se puede reducir el espacio de memoria necesario indicando que IN sea igual a OUT.

Estas operaciones afectan a las siguientes marcas especiales:

SM1.0 (cero); SM1.1 (desbordamiento)

### Incrementar palabra, Decrementar palabra



Las operaciones **Incrementar palabra** / **Decrementar palabra** suman / restan 1 al valor de la palabra de entrada.

Operandos: IN: VW, T, C, IW, QW, MW, SMW, AC, AIW, constante, \*VD, \*AC, SW

OUT: VW, T, C, IW, QW, MW, SMW, AC, \*VD, \*AC, SW

En KOP:  $IN + 1 = OUT$   
 $IN - 1 = OUT$

En AWL:  $OUT + 1 = OUT$   
 $OUT - 1 = OUT$

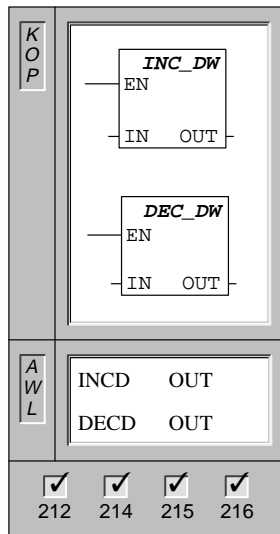
Las operaciones Incrementar y Decrementar palabra llevan signo (16#7FFF > 16#8000).

Nota: Al programar en KOP se puede reducir el espacio de memoria necesario indicando que IN sea igual a OUT.

Estas operaciones afectan a las siguientes marcas especiales:

SM1.0 (cero); SM1.1 (desbordamiento); SM1.2 (negativo)

### Incrementar palabra doble, Decrementar palabra doble



Las operaciones **Incrementar palabra doble** y **Decrementar palabra doble** suman / restan 1 al valor de la palabra doble de entrada.

Operandos: IN: VD, ID, QD, MD, SMD, AC, HC, constante, \*VD, \*AC, SD  
OUT: VD, ID, QD, MD, SMD, AC, \*VD, \*AC, SD

En KOP:  $IN + 1 = OUT$   
 $IN - 1 = OUT$

En AWL:  $OUT + 1 = OUT$   
 $OUT - 1 = OUT$

Las operaciones Incrementar y Decrementar palabra doble llevan signo (16#7FFFFFFF > 16#80000000).

Nota: Al programar en KOP se puede reducir el espacio de memoria necesario indicando que IN sea igual a OUT.

Estas operaciones afectan a las siguientes marcas especiales:

SM1.0 (cero); SM1.1 (desbordamiento); SM1.2 (negativo)

### Ejemplos de las operaciones Incrementar y Decrementar

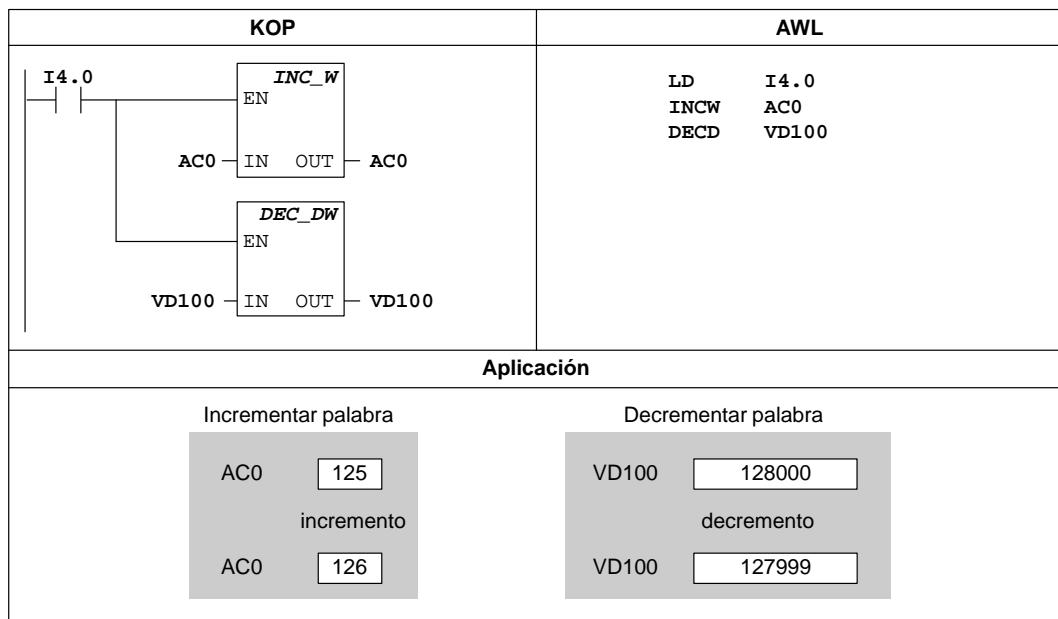
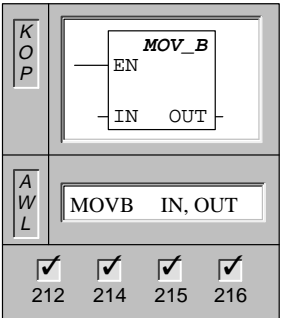


Figura 10-22 Ejemplo de las operaciones Incrementar y Decrementar en KOP y AWL

## 10.8 Operaciones de transferencia, inicializar memoria y tabla

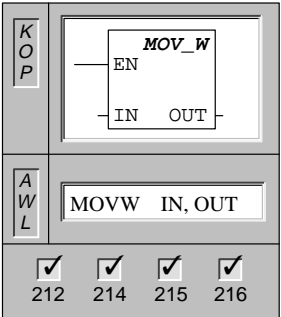
### Transferir byte



La operación **Transferir byte** transfiere el byte de entrada (IN) al byte de salida (OUT). El byte de entrada permanece inalterado.

Operandos: IN: VB, IB, QB, MB, SMB, AC, constante, \*VD, \*AC, SB  
OUT: VB, IB, QB, MB, SMB, AC, \*VD, \*AC, SB

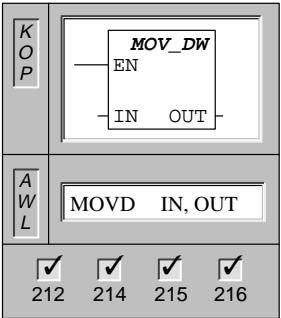
### Transferir palabra



La operación **Transferir palabra** transfiere la palabra de entrada (IN) a la palabra de salida (OUT). La palabra de entrada permanece inalterada.

Operandos: IN: VW, T, C, IW, QW, MW, SMW, AC, AIW, constante, \*VD, \*AC, SW  
OUT: VW, T, C, IW, QW, MW, SMW, AC, AQW, \*VD, \*AC, SW

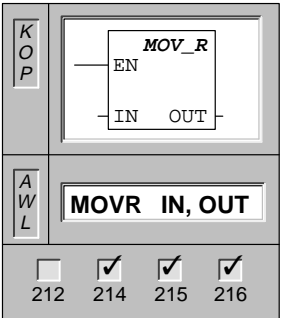
### Transferir palabra doble



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.

Operandos: IN: VD, ID, QD, MD, SMD, AC, HC, constante, \*VD, \*AC, &VB, &IB, &QB, &MB, &T, &C, &SB, SD  
OUT: VD, ID, QD, MD, SMD, AC, \*VD, \*AC, SD

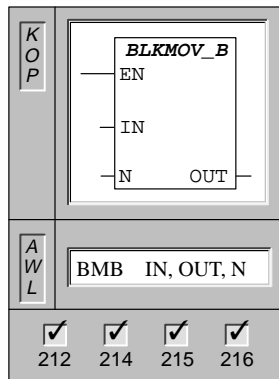
### Transferir real



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.

Operandos: IN: VD, ID, QD, MD, SMD, AC, constante, \*VD, \*AC, SD  
OUT: VD, ID, QD, MD, SMD, AC, \*VD, \*AC, SD

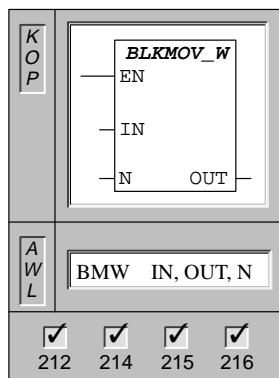
### Transferir bytes en bloque



La operación **Transferir bytes en bloque** transfiere un número determinado de bytes (N) del campo de entrada que comienza en IN al campo de salida que comienza en OUT. N puede estar comprendido entre 1 y 255.

Operandos: IN, OUT: VB, IB, QB, MB, SMB, \*VD, \*AC, SB  
N: VB, IB, QB, MB, SMB, AC, constante, \*VD, \*AC, SB

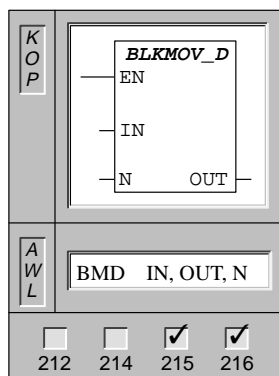
### Transferir palabras en bloque



La operación **Transferir palabras en bloque** transfiere un número determinado de palabras (N) del campo de entrada que empieza en IN al campo de salida que comienza en OUT. N puede estar comprendido entre 1 y 255.

Operandos: IN: VW, T, C, IW, QW, MW, SMW, AIW, \*VD, \*AC, SW  
OUT: VW, T, C, IW, QW, MW, SMW, AQW, \*VD, \*AC, SW  
N: VB, IB, QB, MB, SMB, AC, constante, \*VD, \*AC, SB

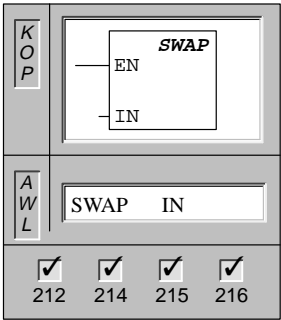
### Transferir palabras dobles en bloque



La operación **Transferir palabras dobles en bloque** transfiere un número determinado de palabras dobles (N) del campo de entrada que empieza en IN al campo de salida que comienza en OUT. N puede estar comprendido entre 1 y 255.

Operandos: IN, OUT: VD, ID, QD, MD, SMD, \*VD, \*AC, SD  
N: VB, IB, QB, MB, SMB, AC, constante, \*VD, \*AC, SB

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).

Operandos: IN: VW, T, C, IW, QW, MW, SMW, AC, \*VD, \*AC, SW

Ejemplos de operaciones de transferir e invertir

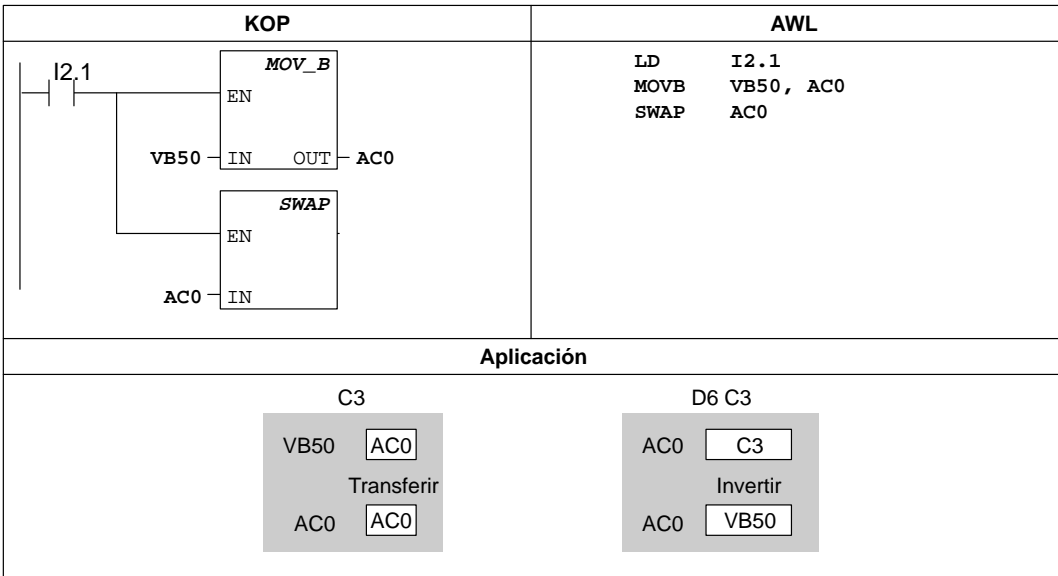


Figura 10-23 Ejemplo de operaciones de transferir e invertir en KOP y AWL

### Ejemplo de una operación de transferencia de bloques

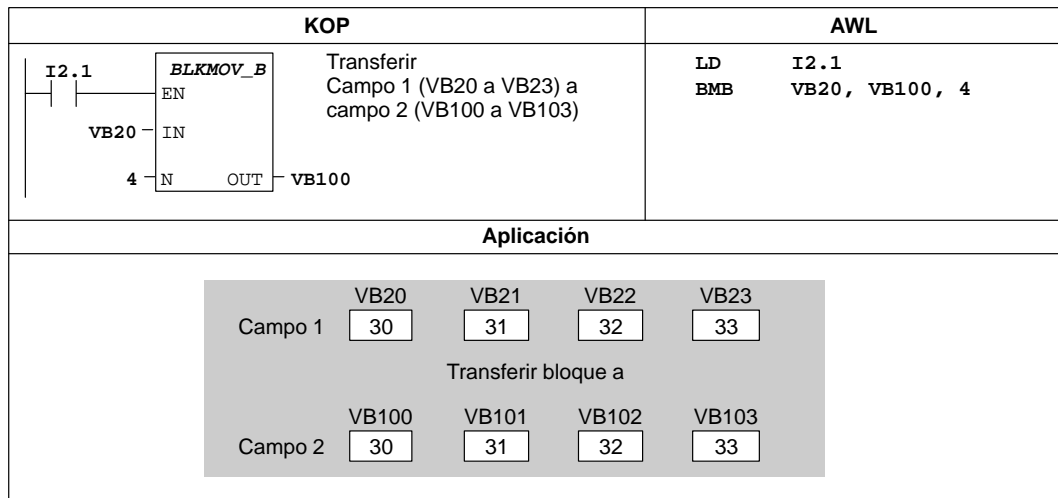
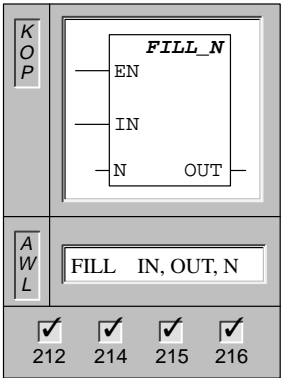


Figura 10-24 Ejemplo de una operación de transferencia en bloque en KOP y AWL

Inicializar memoria



La operación **Inicializar memoria** rellena la memoria que comienza en la palabra de salida (OUT) con la configuración de la palabra de entrada (IN) para el área indicada por el número de palabras N. N puede estar comprendida entre 1 y 255.

- Operandos:
- IN: VW, T, C, IW, QW, MW, SMW, AIW, constante, \*VD, \*AC, SW
  - OUT: VW, T, C, IW, QW, MW, SMW, AQW, \*VD, \*AC, SW
  - N: VB, IB, QB, MB, SMB, AC, constante, \*VD, \*AC, SB

Ejemplo de la operación Inicializar memoria

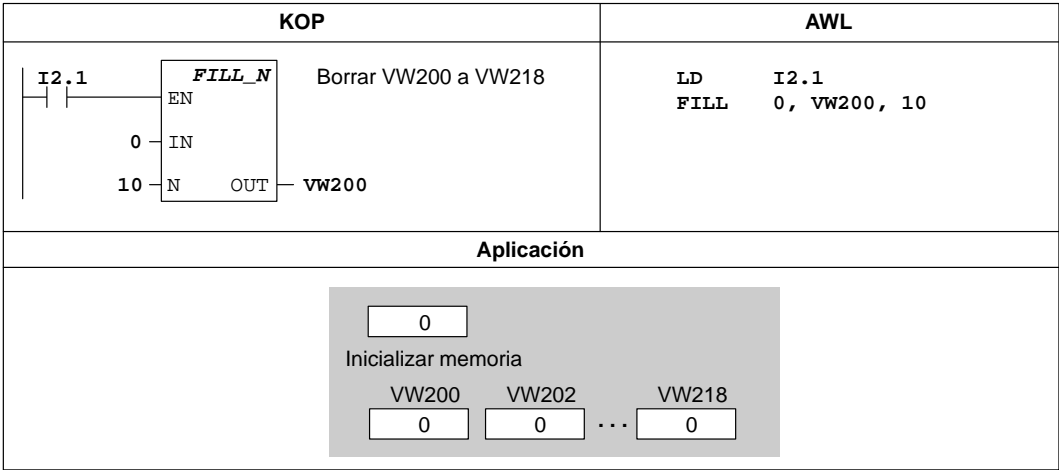
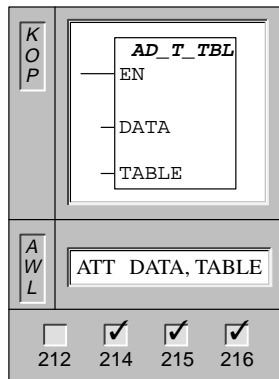


Figura 10-25 Ejemplo de una operación Inicializar memoria en KOP y AWL

## Registrar valor en tabla



La operación **Registrar valor en tabla** (ATT) registra valores de palabra (DATA) en la tabla (TABLE).

Operandos: DATA: VW, T, C, IW, QW, MW, SMW, AC, AIW, constante, \*VD, \*AC, SW  
TABLE: VW, T, C, IW, QW, MW, SMW, \*VD, \*AC, SW

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. fig. 10-26). Los nuevos datos se agregan al final de la tabla, debajo del último registro. Cada vez que se agrega un registro a la tabla, se incrementa el número efectivo de registros. Una tabla puede tener 100 registros como máximo, exceptuando los parámetros que especifican la longitud máxima de la tabla y el número actual de registros.

Esta operación afecta a la siguiente marca especial:

SM1.4 se activa si se intenta introducir demasiados registros en la tabla.

## Ejemplo de la operación Registrar valor en tabla

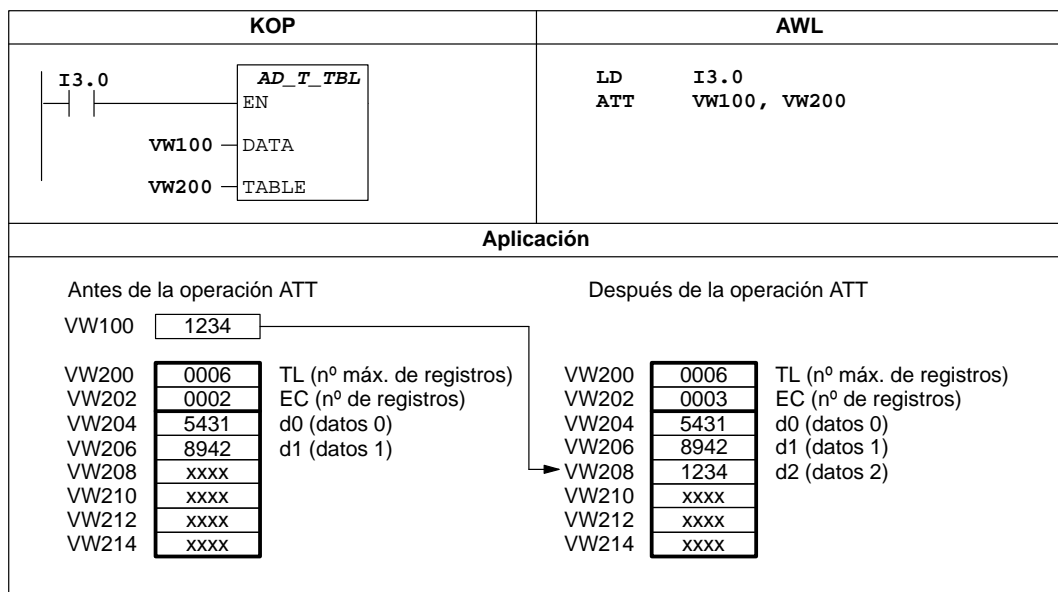
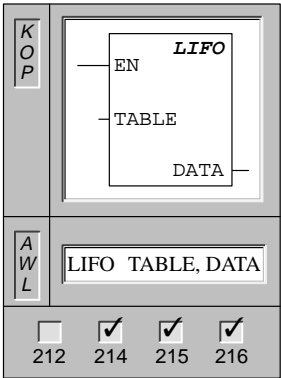


Figura 10-26 Ejemplo de la operación Registrar valor en tabla (ATT)

Borrar último registro de la tabla



La operación **Borrar último registro de la tabla** borra el último registro de la tabla (TABLE) y transfiere el valor a la dirección (DATA). El número de registros (EC) de la tabla se decrementa cada vez que se ejecuta esta operación.

Operandos: TABLE: VW, T, C, IW, QW, MW, SMW, \*VD, \*AC, SW  
DATA: VW, T, C, IW, QW, MW, SMW, AC, AQW, \*VD, \*AC, SW

Esta operación afecta a la siguiente marca especial:  
SM1.5 se activa si se intenta borrar un registro de una tabla vacía.

Ejemplo de la operación Borrar último registro de la tabla

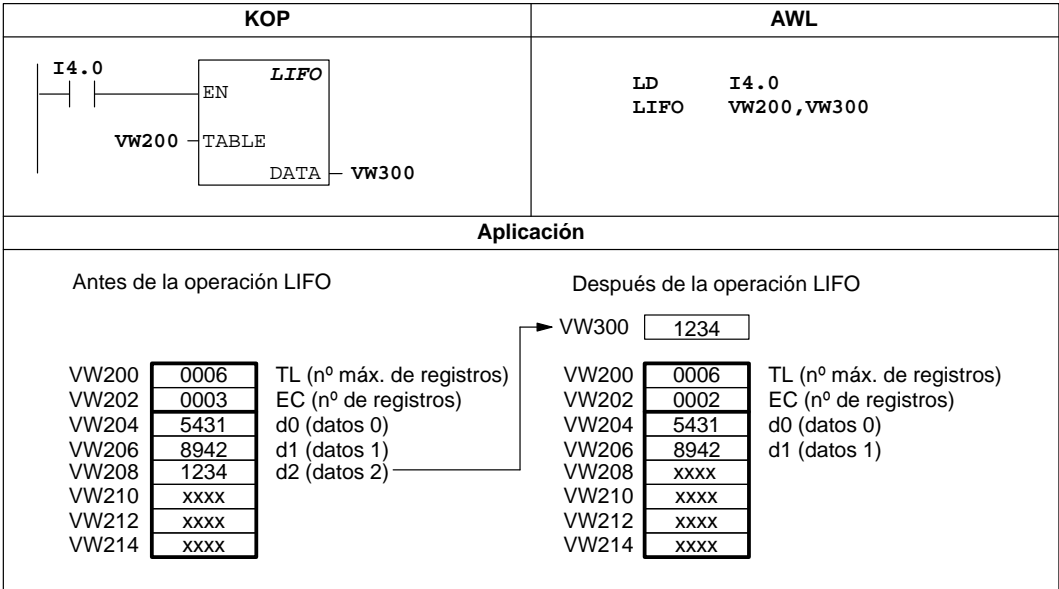
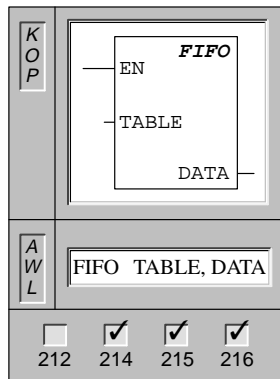


Figura 10-27 Ejemplo de la operación Borrar último registro de la tabla (LIFO)

### Borrar primer registro de la tabla



La operación **Borrar primer registro de la tabla** borra el primer registro de la tabla (TABLE) 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 se decrementa cada vez que se ejecuta esta operación.

Operandos: TABLE: VW, T, C, IW, QW, MW, SMW, \*VD, \*AC, SW  
DATA: VW, T, C, IW, QW, MW, SMW, AC, AQW, \*VD, \*AC, SW

Esta operación afecta a la siguiente marca especial:

SM1.5 se activa si se intenta borrar un registro de una tabla vacía.

### Ejemplo de la operación Borrar primer registro de la tabla

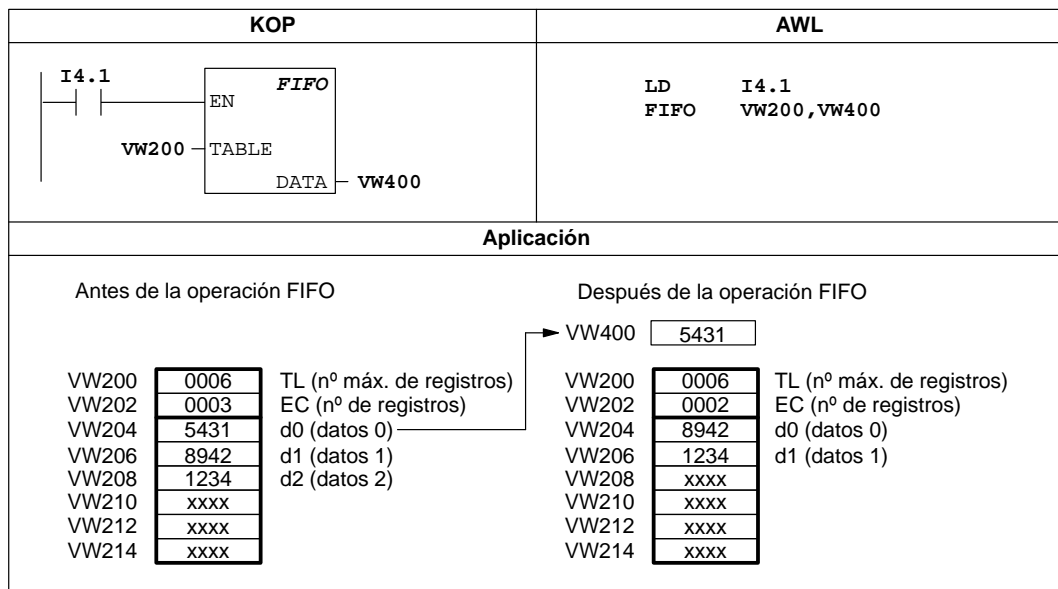


Figura 10-28 Ejemplo de la operación Borrar primer registro de la tabla (FIFO)

Buscar valor en tabla

K  
O  
P

TBL\_FIND

— EN

— SRC

— PATRN

— INDX

— CMD

A  
W  
L

FND= SRC, PATRN, INDX

FND<> SRC, PATRN, INDX

FND< SRC, PATRN, INDX

FND> SRC, PATRN, INDX

☐ 212

☒ 214

☒ 215

☒ 216

La operación **Buscar valor en tabla** rastrea la tabla (SRC) comenzando con el registro indicado por INDX y busca el valor (PATRN) que corresponda a los criterios de búsqueda (=, <>, <, y >).

En KOP, el parámetro de comando (CMD) indica un valor numérico comprendido entre 1 y 4 que corresponde a la relación =, <>, <, y >, respectivamente.

Operandos: SRC: VW, T, C, IW, QW, MW, SMW, \*VD, \*AC, SW

PATRN: VW, T, C, IW, QW, MW, SMW, AC, AIW, constante, \*VD, \*AC, SW

INDX: VW, T, C, IW, QW, MW, SMW, AC, \*VD, \*AC, SW

CMD: 1 (=) 2 (<>) 3 (<) 4 (>)

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.

Los registros de la tabla (el área donde se desea buscar) están numerados de 0 hasta el valor máximo (99). Una tabla puede tener 100 registros como máximo, exceptuando los parámetros que especifican la longitud permitida de la tabla y el número actual de registros.

**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 dicha palabra. Por consiguiente, la dirección del operando SRC de una operación de búsqueda supera en una palabra (dos bytes) al operando TABLE correspondiente a una operación ATT, LIFO y FIFO, como muestra la figura 10-29.

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 10-29 Diferencia de los formatos de tabla entre las operaciones de búsqueda y las operaciones ATT, LIFO y FIFO

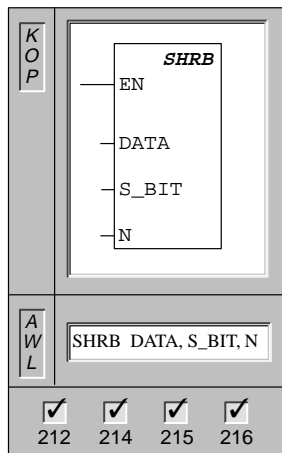
### Ejemplo de la operación Buscar valor en tabla

KOP		AWL															
<div><div><div>I2.1</div><div></div><div></div><div>VW202</div><div>16#3130</div><div>AC1</div><div>1</div></div><div><div>TBL_FIND</div><div>EN</div><div>SRC</div><div>PATRN</div><div>INDX</div><div>CMD</div></div></div>	Si I2.1 está activada, buscar un valor en la tabla que sea igual a 3130 HEX.	LD	I2.1														
		FND=	VW202, 16#3130, AC1														
Aplicación																	
Esta es la tabla que se va a rastrear. Si la tabla se creó utilizando las operaciones ATT, LIFO y FIFO, VW200 contendrá el número máximo de registros posibles y no será requerido por las operaciones de búsqueda.																	
	<table><tr><td>VW202</td><td>0006</td></tr><tr><td>VW204</td><td>3133</td></tr><tr><td>VW206</td><td>4142</td></tr><tr><td>VW208</td><td>3130</td></tr><tr><td>VW210</td><td>3030</td></tr><tr><td>VW212</td><td>3130</td></tr><tr><td>VW214</td><td>4541</td></tr></table>	VW202	0006	VW204	3133	VW206	4142	VW208	3130	VW210	3030	VW212	3130	VW214	4541	EC (nº de registros)	
VW202	0006																
VW204	3133																
VW206	4142																
VW208	3130																
VW210	3030																
VW212	3130																
VW214	4541																
		d0 (datos 0)															
		d1 (datos 1)															
		d2 (datos 2)															
		d3 (datos 3)															
		d4 (datos 4)															
		d5 (datos 5)															
AC1	<input type="text" value="0"/>	AC1 se debe poner a 0 para poder iniciar la búsqueda desde el primer registro de la tabla.															
Buscar																	
AC1	<input type="text" value="2"/>	AC1 contiene el número del primer registro que corresponde al criterio de búsqueda (d2).															
AC1	<input type="text" value="3"/>	Incrementar INDX en pasos de 1 antes de buscar los demás registros de la tabla.															
Buscar																	
AC1	<input type="text" value="4"/>	AC1 contiene el número del segundo registro que corresponde al criterio de búsqueda (d4).															
AC1	<input type="text" value="5"/>	Incrementar INDX en pasos de 1 antes de buscar los demás registros de la tabla.															
Buscar																	
AC1	<input type="text" value="6"/>	AC1 contiene un valor igual al número de registros. Se ha rastreado toda la tabla sin encontrar otro registro que corresponda al criterio de búsqueda.															
AC1	<input type="text" value="0"/>	Antes de que la tabla se pueda rastrear de nuevo, es preciso poner a 0 el valor de INDX.															

Figura 10-30 Ejemplo de la operación Buscar valor en tabla en KOP y AWL

## 10.9 Operaciones de desplazamiento y rotación

### Registro de desplazamiento



La operación **Registro de desplazamiento** desplaza el valor de DATA al registro de desplazamiento. S\_BIT indica el bit menos significativo de dicho registro. N indica la longitud del registro y el sentido de desplazamiento (valor positivo = N, valor negativo = -N).

Operandos: DATA, S\_BIT: I, Q, M, SM, T, C, V, S  
N: VB, IB, QB, MB, SMB, SB, AC, Constante, \*VD, \*AC, SB

### 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 10-32 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 10-31 muestra el desplazamiento de bits de un valor N positivo y de un valor N negativo.

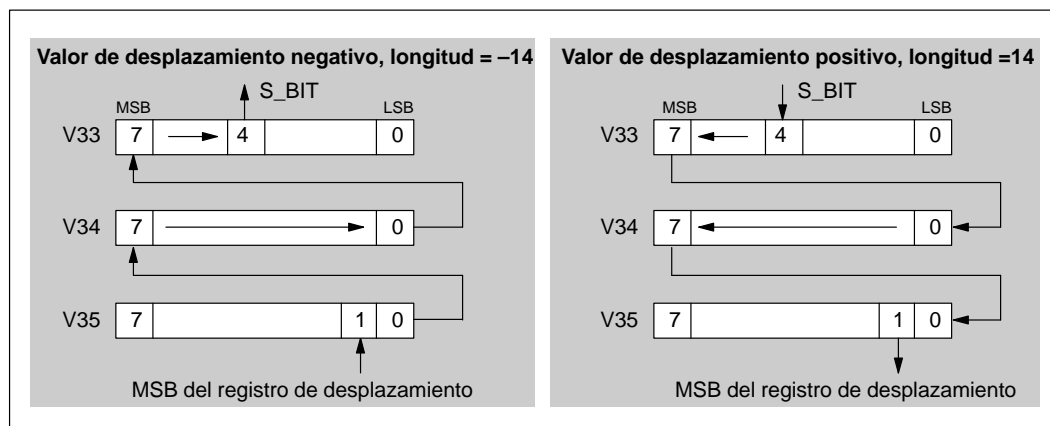


Figura 10-31 Entrada y salida de valores positivos y negativos en el registro de desplazamiento

### Ejemplo de la operación Registro de desplazamiento

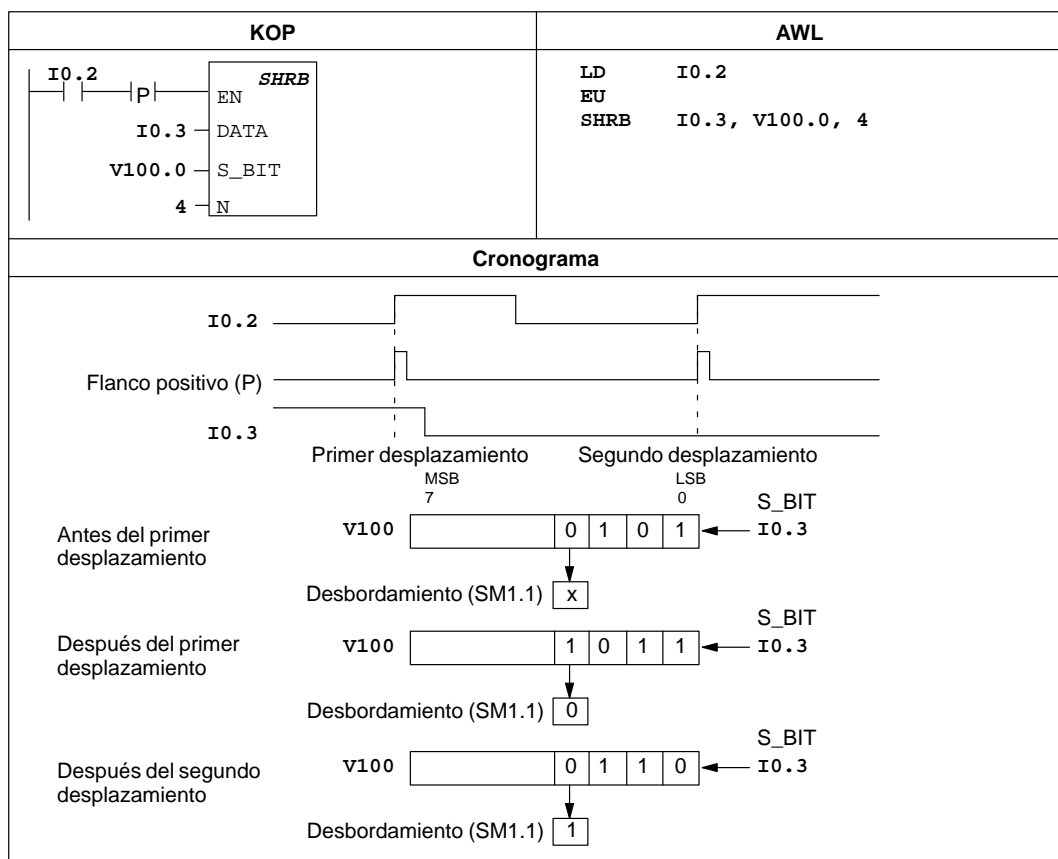
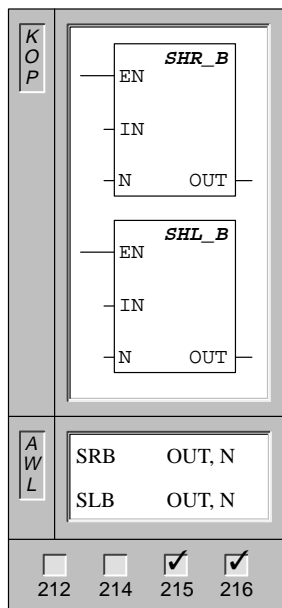


Figura 10-32 Ejemplo de la operación Registro de desplazamiento en KOP y AWL

### 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).

Operandos: IN: VB, IB, QB, MB, SMB, SB, AC, \*VD, \*AC  
 N: VB, IB, QB, MB, SMB, SB, AC, constante, \*VD, \*AC  
 OUT: VB, IB, QB, MB, SMB, SB, AC, \*VD, \*AC

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 adoptará el valor del último bit desplazado hacia afuera.

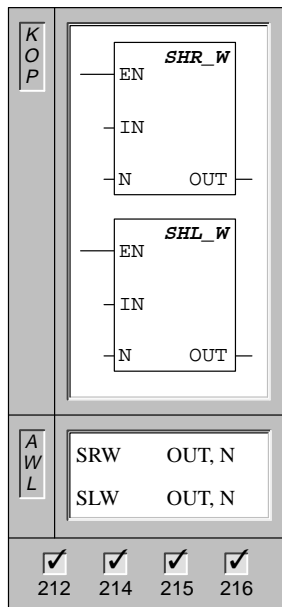
Las operaciones de desplazamiento de bytes no llevan signo.

Nota: Al programar en KOP se puede reducir el espacio de memoria necesario indicando que IN sea igual a OUT.

Estas operaciones afectan a las siguientes marcas especiales:

SM1.0 (cero); SM1.1 (desbordamiento)

### 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 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).

Operandos: IN: VW, T, C, IW, MW, SMW, AC, QW, AIW, constante, \*VD, \*AC, SW  
 N: VB, IB, QB, MB, SMB, AC, constante, \*VD, \*AC, SB  
 OUT: VW, T, C, IW, QW, MW, SMW, AC, \*VD, \*AC, SW

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 adoptará el valor del último bit desplazado hacia afuera.

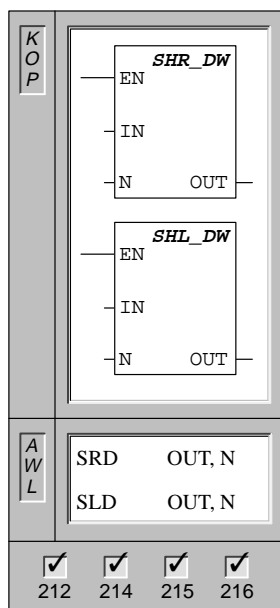
Las operaciones de desplazamiento de palabras no llevan signo.

Nota: Al programar en KOP se puede reducir el espacio de memoria necesario indicando que IN sea igual a OUT.

Estas operaciones afectan a las siguientes marcas especiales:

SM1.0 (cero); SM1.1 (desbordamiento)

## 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 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).

Operandos: IN: VD, ID, QD, MD, SMD, AC, HC, constante, \*VD, \*AC, SD  
 N: VB, IB, QB, MB, SMB, AC, constante, \*VD, \*AC, SB  
 OUT: VD, ID, QD, MD, SMD, AC, \*VD, \*AC, SD

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 adoptará el valor del último bit desplazado hacia afuera.

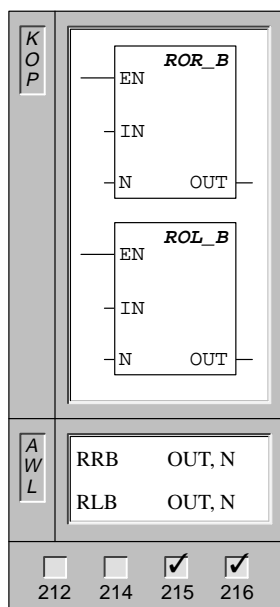
Las operaciones de desplazamiento de palabras dobles no llevan signo.

Nota: Al programar en KOP se puede reducir el espacio de memoria necesario indicando que IN sea igual a OUT.

Estas operaciones afectan a las siguientes marcas especiales:

SM1.0 (cero); SM1.1 (desbordamiento)

## 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).

Operandos: IN: VB, IB, QB, MB, SMB, SB, AC, \*VD, \*AC, SB  
 N: VB, IB, QB, MB, SMB, SB, AC, constante, \*VD, \*AC, SB  
 OUT: VB, IB, QB, MB, SMB, SB, AC, \*VD, \*AC, SB

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.

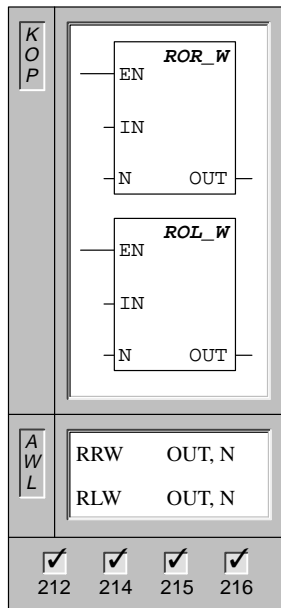
Las operaciones de rotación de bytes no llevan signo.

Nota: Al programar en KOP se puede reducir el espacio de memoria necesario indicando que IN sea igual a OUT.

Estas operaciones afectan a las siguientes marcas especiales:

SM1.0 (cero); SM1.1 (desbordamiento)

### 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 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).

Operandos: IN: VW, T, C, IW, MW, SMW, AC, QW, AIW, constante, \*VD, \*AC, SW  
 N: VB, IB, QB, MB, SMB, AC, constante, \*VD, \*AC, SB  
 OUT: VW, T, C, IW, QW, MW, SMW, AC, \*VD, \*AC, SW

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.

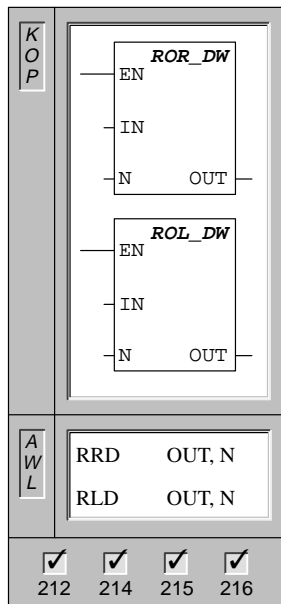
Las operaciones de rotación de palabras no llevan signo.

Nota: Al programar en KOP se puede reducir el espacio de memoria necesario indicando que IN sea igual a OUT.

Estas operaciones afectan a las siguientes marcas especiales:

SM1.0 (cero); SM1.1 (desbordamiento)

### 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 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).

Operandos: IN: VD, ID, QD, MD, SMD, AC, HC, constante, \*VD, \*AC, SD  
 N: VB, IB, QB, MB, SMB, AC, constante, \*VD, \*AC, SB  
 OUT: VD, ID, QD, MD, SMD, AC, \*VD, \*AC, SD

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.

Las operaciones de rotación de palabras dobles no llevan signo.

Nota: Al programar en KOP se puede reducir el espacio de memoria necesario indicando que IN sea igual a OUT.

Estas operaciones afectan a las siguientes marcas especiales:

SM1.0 (cero); SM1.1 (desbordamiento)

# Ejemplos de operaciones de rotación y desplazamiento

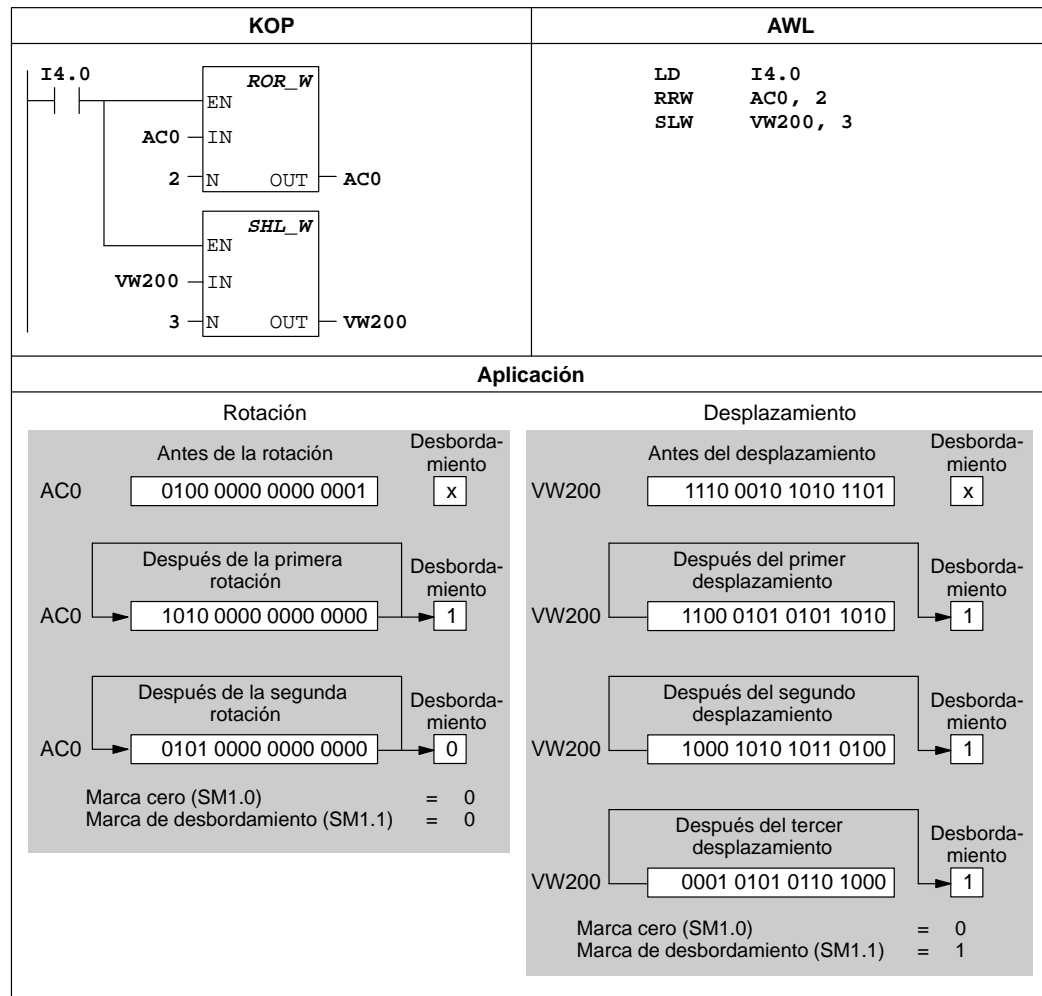
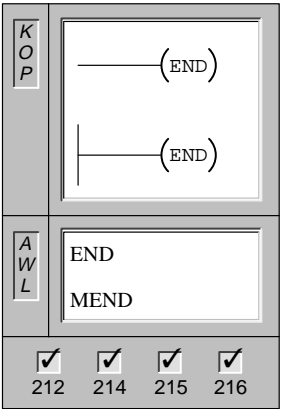


Figura 10-33 Ejemplo de operaciones de desplazamiento y rotación en KOP y AWL

## 10.10 Operaciones de control del programa

### END



La operación condicional **Finalizar programa principal** (END) finaliza el programa en función de la combinación lógica precedente.

La bobina absoluta **Finalizar programa principal** (END) se debe utilizar para finalizar el programa principal de usuario.

En AWL, la operación absoluta Finalizar programa principal se representa con la operación **MEND**.

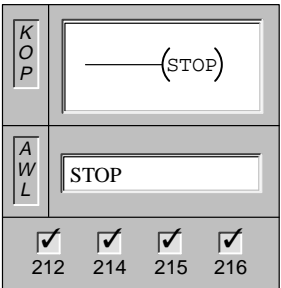
Operandos: ninguno

Todos los programas de usuario se deben finalizar con la operación absoluta END. La operación condicional END permite finalizar la ejecución del programa antes de la operación absoluta END.

#### Nota

Las operaciones END condicional y END absoluta se pueden utilizar en el programa principal, pero no en subrutinas ni en rutinas de interrupción.

### STOP

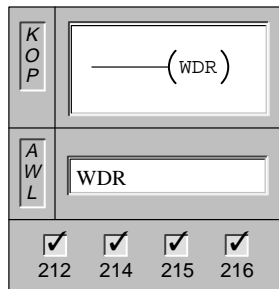


La operación **STOP** finaliza inmediatamente la ejecución del programa haciendo que la CPU cambie de RUN a STOP.

Operandos: ninguno

Si la operación STOP se ejecuta en una rutina de interrupción, ésta se finalizará inmediatamente ignorando las interrupciones pendientes. El resto del programa se sigue procesando y 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 terminar 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 dure más de 300 ms o que la actividad de interrupción aumente de modo que el ciclo principal quede interrumpido más de 300 ms, es preciso utilizar la operación WDR para redisparar el temporizador de vigilancia.

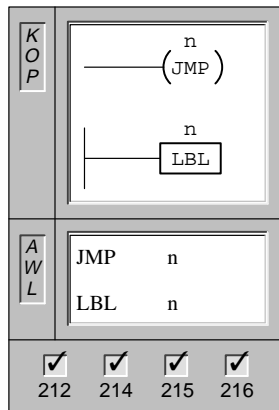
Cambiando el selector a la posición STOP, la CPU pasará a modo STOP en 1,4 segundos.

Ejemplo de las operaciones STOP, WDR y END

KOP		AWL
Network 1		Network LD SM5.0 STOP .
Network 15		Network LD M5.6 WDR .
Network 78		Network MEND

Figura 10-34 Ejemplo de las operaciones STOP, WDR y END en KOP y AWL

## Saltar a meta, Definir meta



La operación **Saltar a meta** (JMP) 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** (LBL) indica la meta a la que se salta.

Operandos:    n:        0 a 255

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.

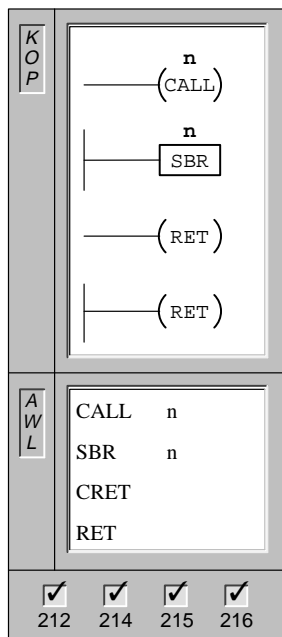
La figura 10-35 muestra un ejemplo de las operaciones Saltar a meta y Definir meta.

### Ejemplo de la operación Saltar a meta

KOP		AWL
<p>Network 14</p> <p>Si no se han perdido datos remanentes, saltar a LBL 4.</p> <p>•</p> <p>•</p> <p>•</p>		<p><b>Network</b></p> <p><b>LDN</b> SM0.2</p> <p><b>JMP</b> 4</p> <p>•</p> <p>•</p> <p>•</p>
<p>Network 33</p> <p>La operación Salto a meta se puede utilizar en el programa principal, en las subrutinas o en las rutinas de interrupción. La operación de salto y la meta correspondiente deben encontrarse siempre en el mismo segmento lógico (es decir, bien sea en el programa principal, en la subrutina o en la rutina de interrupción).</p>		<p><b>Network</b></p> <p><b>LBL</b> 4</p>

Figura 10-35 Ejemplo de las operaciones Saltar a meta y Definir meta en KOP y AWL

## Llamar subrutina, Comenzar subrutina y Retorno de subrutina



La operación **Llamar subrutina** (CALL) transfiere el control a la subrutina (n).

La operación **Comenzar subrutina** (SBR) marca el comienzo de la subrutina (n).

La operación **Retorno condicional de subrutina** se puede utilizar para finalizar una subrutina en función de la combinación lógica precedente.

Todas las subrutinas tienen que terminar con la operación **Retorno absoluto de subrutina**.

Operandos: n: 0 a 63

Una vez ejecutada la subrutina, el control vuelve a la operación que sigue a la llamada de la subrutina (CALL).

Se pueden anidar (situar una llamada a subrutina en otra) hasta ocho subrutinas (profundidad de anidamiento = 8 niveles). Si bien la recursión (la subrutina se llama a sí misma) está permitida, hay que utilizarla cuidadosamente.

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.

Asimismo, cuando se llama a una subrutina, el primer valor de la pila es siempre un "1" lógico. Por lo tanto es posible conectar salidas o cuadros directamente a la barra izquierda del segmento que sigue a la operación Comenzar subrutina (SBR). En AWL puede omitirse la operación de carga que sigue a la operación SBR.

Los acumuladores se transfieren entre el programa principal y las subrutinas. Los acumuladores no se almacenan ni se restablecen si se utilizan con subrutinas.

La figura 10-36 muestra ejemplos de las operaciones Llamar subrutina, Comenzar subrutina y Retorno de subrutina.

### Uso restringido

Al utilizar subrutinas, deberá tener en cuenta los siguientes puntos:

- Sitúe todas las subrutinas después del final del programa principal KOP.
- En una subrutina no se pueden utilizar las operaciones LSCR, SCRE, SCRT y END.
- Finalice todas las subrutinas con la operación Retorno absoluto desde subrutina (RET).

### Ejemplo de la operación Llamar subrutina



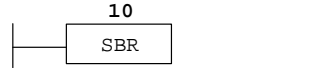


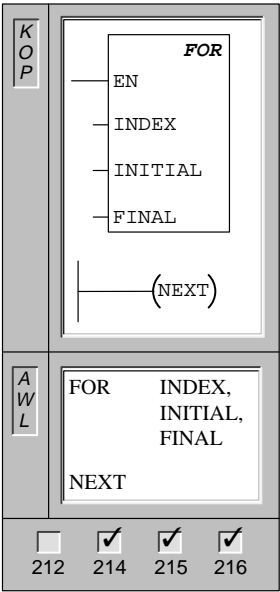
KOP		AWL
Network 1		En el primer ciclo: Llamar SBR10 para inicializar.
Network 39		Colocar todas las subrutinas tras la operación END.
Network 50		Comienzo de la subrutina 10.
Network 65		Se puede utilizar un retorno condicional (RET) desde la subrutina 10.
Network 68		Todas las subrutinas se deben finalizar con un retorno absoluto (RET). Aquí se finaliza la subrutina 10.

Figura 10-36 Ejemplo de operaciones de subrutinas en KOP y AWL

FOR, NEXT



La operación **FOR** ejecuta las operaciones que se encuentren entre FOR y NEXT. Se debe indicar el valor actual de conteo del bucle (INDEX), el valor inicial (INITIAL) y el valor final (FINAL).

La operación **NEXT** marca el final del bucle FOR y pone a "1" el primer valor de la pila.

Operandos: INDEX: VW, T, C, IW, QW, MW, SMW, AC, \*VD, \*AC, SW

INITIAL: VW, T, C, IW, QW, MW, SMW, AC, AIW, constante, \*VD, \*AC, SW

FINAL: VW, T, C, IW, QW, MW, SMW, AC, AIW, constante, \*VD, \*AC, SW

Ejemplo: si el valor de INITIAL 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 INDEX 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 INDEX y el resultado se compara con el valor final. Si INDEX es mayor que el valor final, se finaliza el bucle.

Las operaciones FOR/NEXT repiten un bucle del programa un número determinado de veces. Cada operación FOR exige una operación NEXT. Los bucles FOR/NEXT pueden anidarse (situar un bucle FOR/NEXT dentro de otro) hasta una profundidad de ocho niveles.

La figura 10-37 muestra un ejemplo de las operaciones FOR/NEXT.

### Ejemplo de las operaciones FOR/NEXT

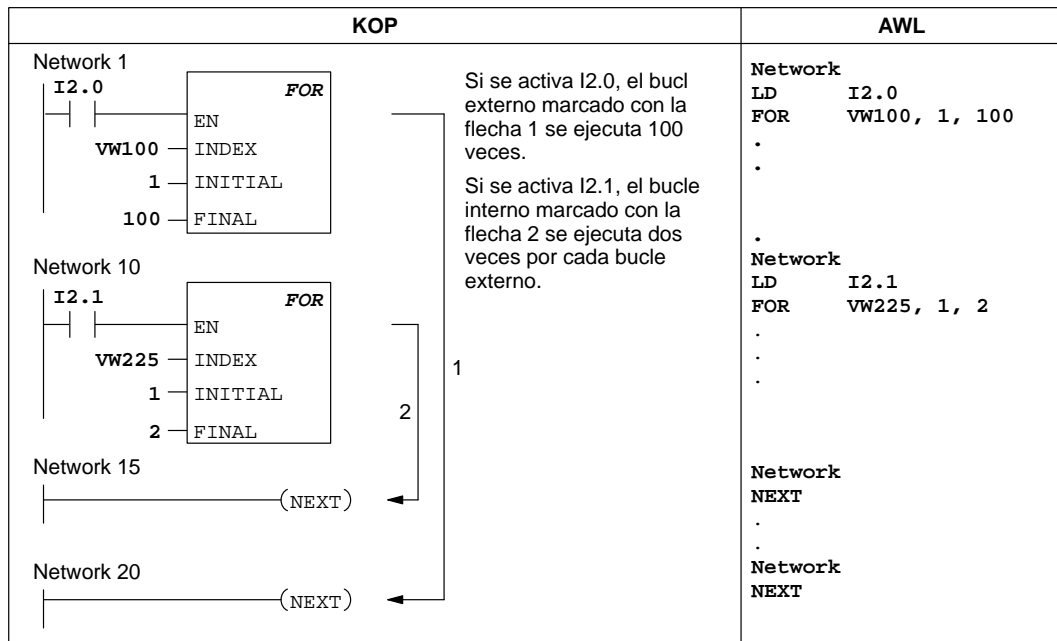
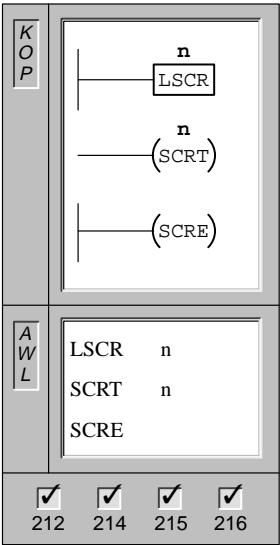


Figura 10-37 Ejemplo de las operaciones FOR/NEXT en KOP y AWL

Operaciones del relé de control secuencial



La operación **Cargar relé de control secuencial** (LSCR) 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** (SCRT) identifica el bit SCR que se debe habilitar (el siguiente bit S a activar). Cuando la corriente fluye hasta la bobina, el bit S indicado 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** (SCRE) indica el fin de un segmento SCR.

Operandos:    n:            S

Descripción de las operaciones del relé de control secuencial

En KOP y AWL, los relés de control secuencial (SCRs) se utilizan para estructurar las instalaciones o las etapas en segmentos equivalentes del programa. Los SCR permiten segmentar lógicamente el programa de usuario.

La operación LSCR carga el valor del bit S indicado por la operación en la pila del relé de control secuencial (pila SCR) y 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 unir directamente los cuadros y las bobinas a la barra de alimentación izquierda sin necesidad de interconectar un contacto. La figura 10-38 muestra la pila SCR y la pila lógica, así como los efectos de la operación LSCR.

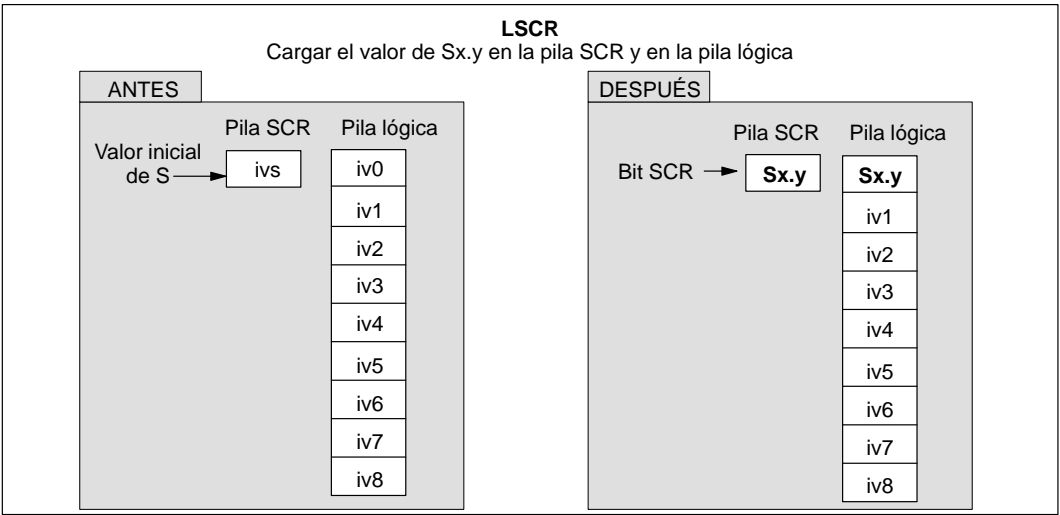


Figura 10-38 Efectos de la operación LSCR en la pila lógica

Tenga en cuenta las siguientes observaciones respecto a las operaciones del relé de control secuencial:

- Todas las operaciones que se encuentren entre la operación LSCR y la operación SCRE conforman 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, deberá tener en cuenta los siguientes puntos:

- Los relés de control secuencial (SCRs) se pueden utilizar en el programa principal, mas no en las subrutinas o en las rutinas de interrupción.
- En un segmento SCR no se pueden usar las operaciones Saltar a meta (JMP) ni Definir meta (LBL). Por tanto, no se pueden utilizar para saltar hacia adentro, ni hacia afuera del segmento SCR, ni tampoco dentro del mismo. No obstante, las operaciones de salto y de meta se pueden emplear para saltar por encima de segmentos SCR.
- En un segmento SCR no se pueden utilizar las operaciones FOR, NEXT ni END.

### Ejemplo de una operación SCR

La figura 10-39 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 causa una transición a la etapa 2. Dicha transición desactiva el segmento SCR (S0.1) de la primera etapa y activa el segmento SCR (S0.2) de la segunda etapa.

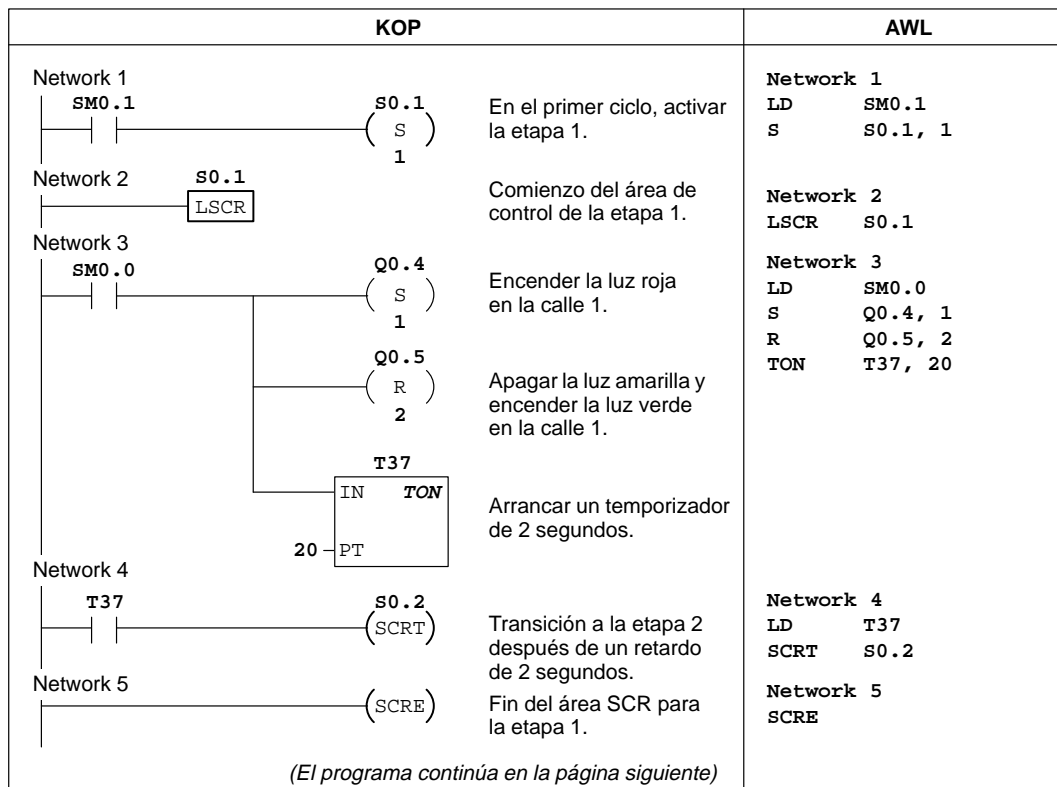


Figura 10-39 Ejemplo de un relé de control secuencial (SCR)

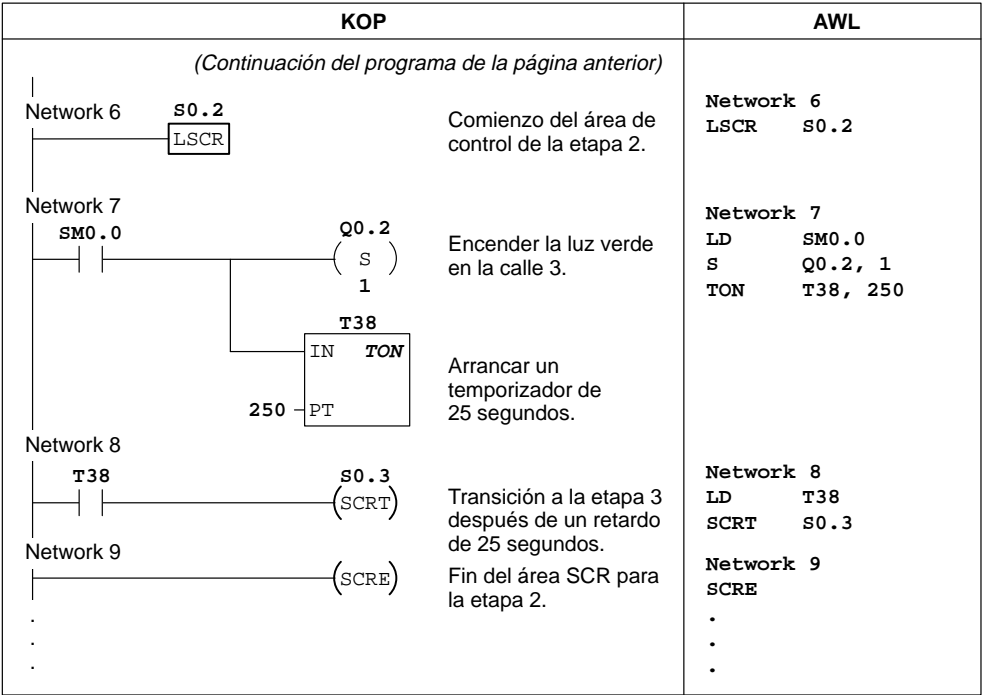


Figura 10-39 Ejemplo de un relé de control secuencial (SCR), continuación

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 10-40.

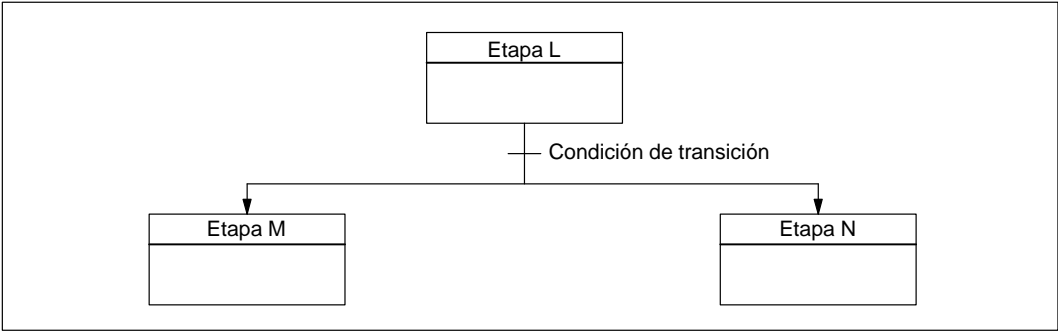


Figura 10-40 División de una cadena secuencial

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 10-41.

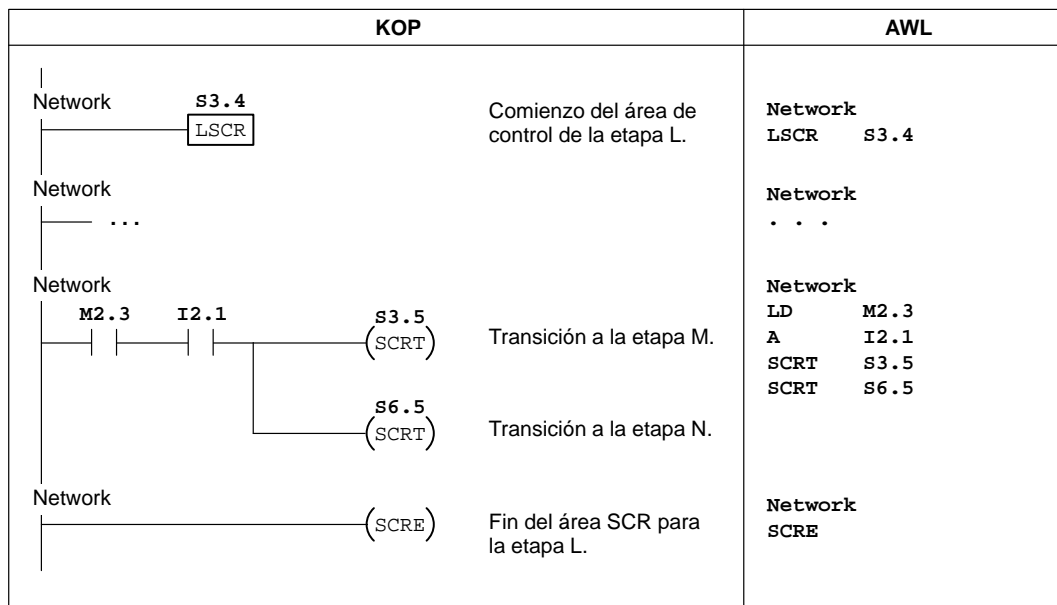


Figura 10-41 Ejemplo de la división de una cadena secuencial

### Convergir cadenas secuenciales

Al ser preciso convergir dos o más cadenas secuenciales para crear una cadena, se presenta una situación similar. Todas las cadenas secuenciales se deben terminar antes de poder ejecutar la siguiente etapa. La figura 10-42 muestra la convergencia de dos cadenas secuenciales.

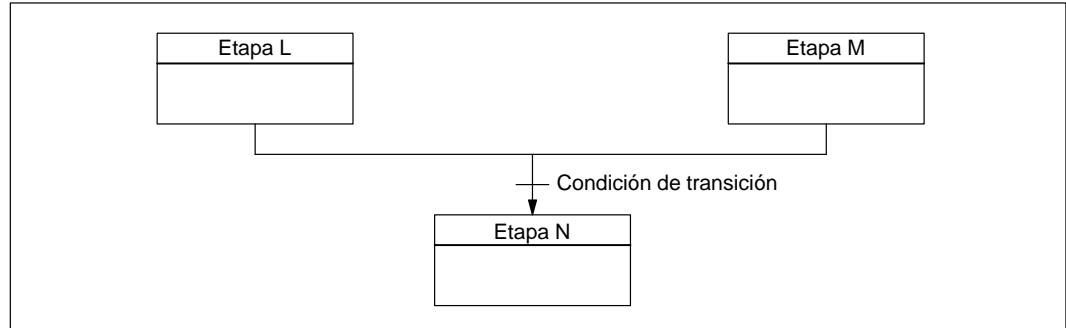


Figura 10-42 Convergencia de cadenas secuenciales

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 se muestra a continuación.

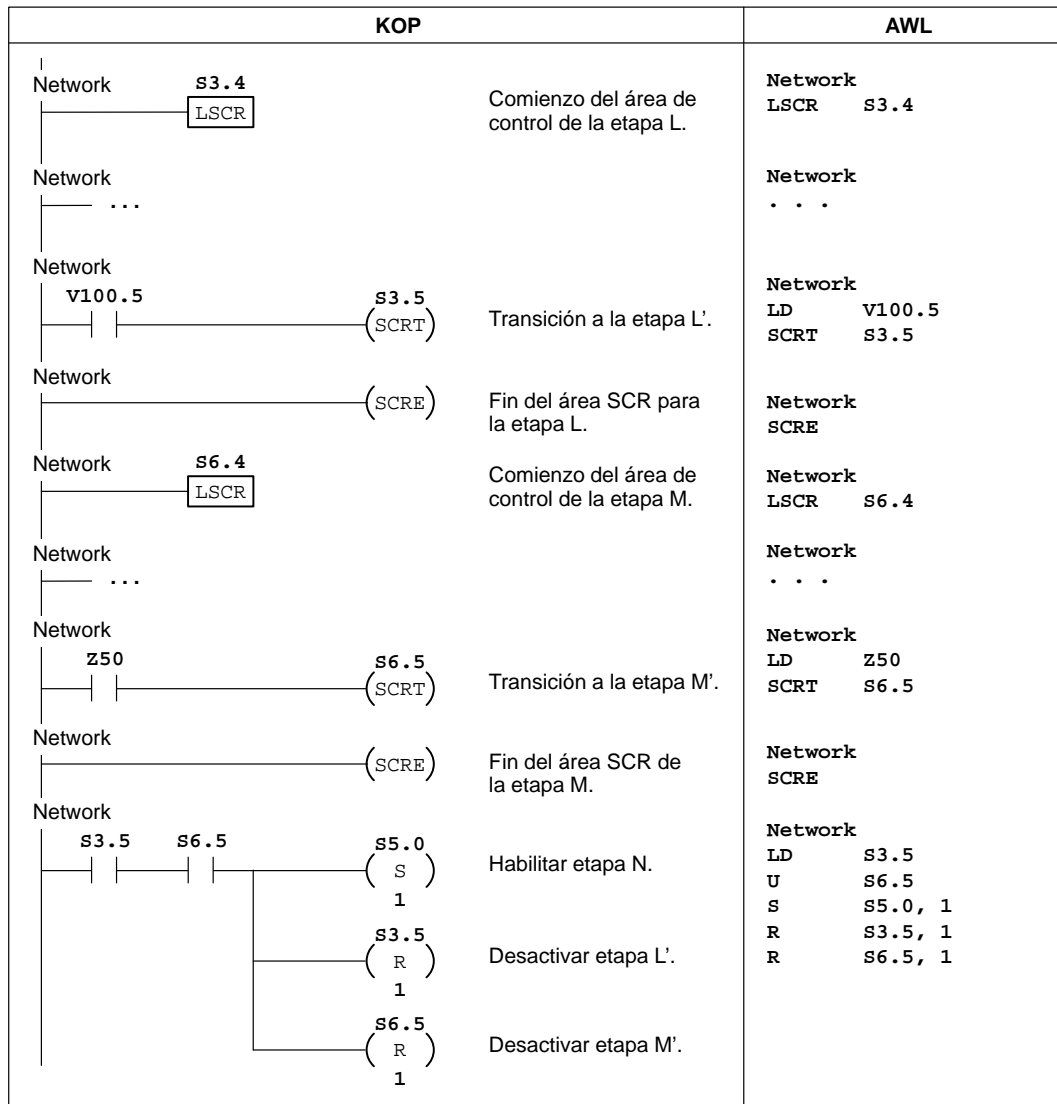


Figura 10-43 Ejemplo de la convergencia de cadenas secuenciales

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 10-44 muestra dicha situación.

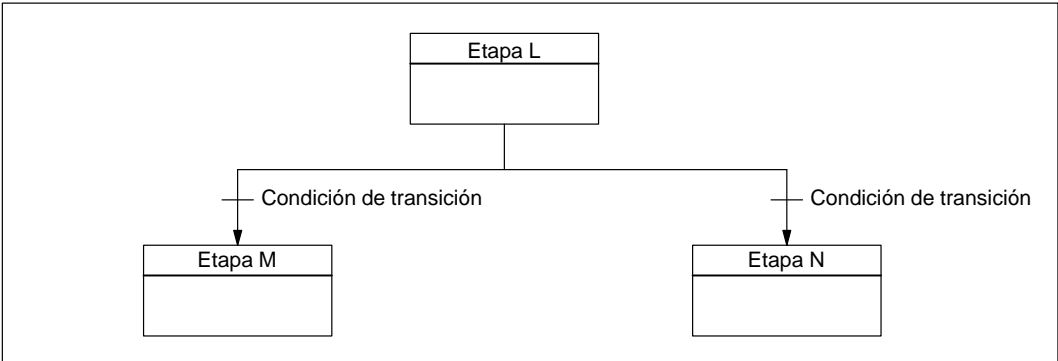


Figura 10-44 Dirigir una cadena secuencial a otra, dependiendo de la condición de transición

La figura 10-45 muestra el correspondiente programa SCR.

KOP			AWL
Network	S3.4 LSCR	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 10-45 Ejemplo de transiciones condicionales

## 10.11 Operaciones lógicas de pilas

### Combinar primer y segundo valor mediante Y

A	ALD			
W				
L				
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
212	214	215	216	

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

A	OLD			
W				
L				
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
212	214	215	216	

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

A	LPS			
W				
L				
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
212	214	215	216	

La operación **Duplicar primer valor** (LPS) duplica el primer valor y lo desplaza dentro de la pila. El último valor de la pila se expulsa y se pierde.

Operandos: ninguno

### Copiar segundo valor

A	LRD			
W				
L				
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
212	214	215	216	

La operación **Copiar segundo valor** (LRD) copia el segundo valor de la pila en el nivel superior. 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

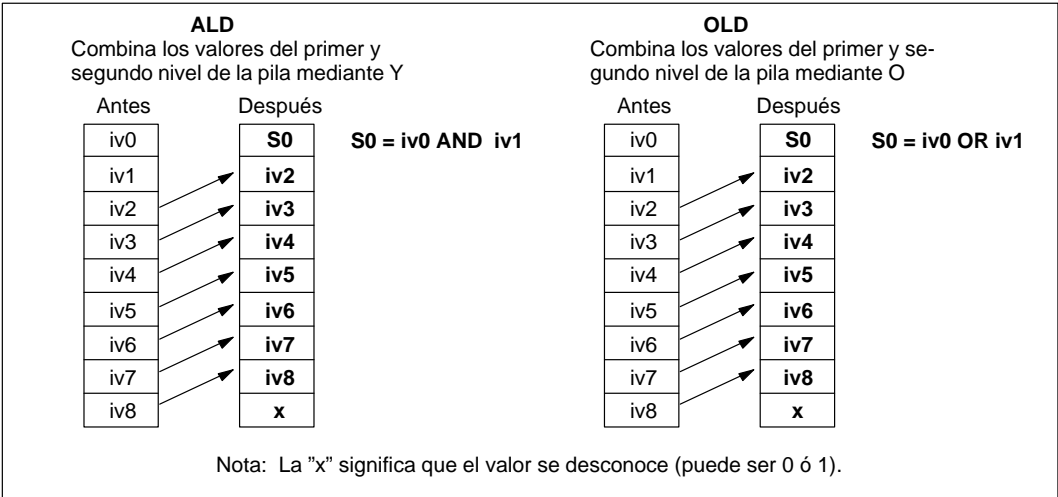
A	LPP			
W				
L				
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
212	214	215	216	

La operación **Sacar primer valor** (LPP) desplaza el primer valor fuera de la pila. El segundo valor se convierte entonces en el primer nivel de la pila.

Operandos: ninguno

Operaciones lógicas de pilas

La figura 10-46 muestra cómo funcionan las operaciones Combinar primer y segundo valor mediante Y y mediante O.



### Ejemplo de una operación lógica de pila

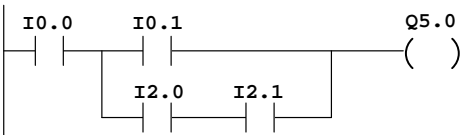
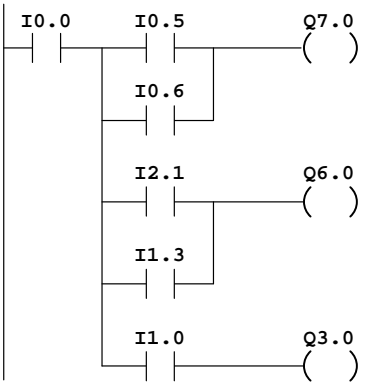
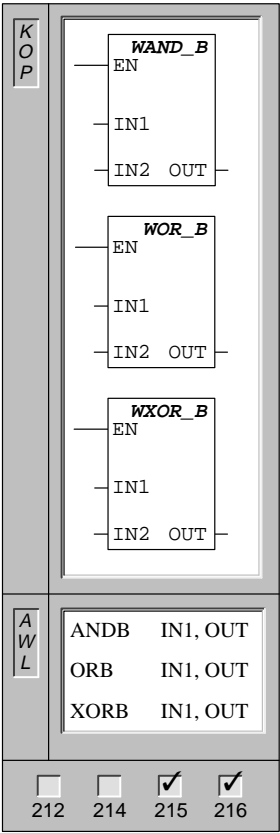
KOP	AWL
<p>Network 1</p>  <p>Network 2</p> 	<pre> NETWORK LD    I0.0 LD    I0.1 LD    I2.0 A     I2.1 OLD ALD =     Q5.0  NETWORK LD    I0.0 LPS LD    I0.5 O     I0.6 ALD =     Q7.0 LRD LD    I2.1 O     I1.3 ALD =     Q6.0 LPP A     I1.0 =     Q3.0 </pre>

Figura 10-48 Ejemplo de una operación lógica de pila en KOP y AWL

### 10.12 Operaciones lógicas

**Combinación Y con bytes, Combinación O con bytes, 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.

Operandos: IN1, IN2: VB, IB, QB, MB, SMB, AC, constante, \*VD, \*AC, SB

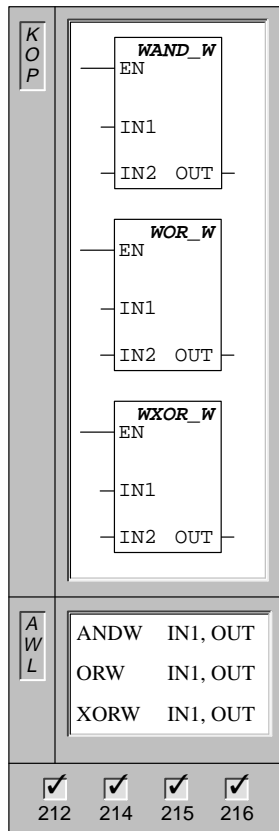
OUT: VB, IB, QB, MB, SMB, SB, AC, \*VD, \*AC, SB

Nota: Al programar en KOP se puede reducir el espacio de memoria necesario indicando que IN1 sea igual a OUT.

Estas operaciones afectan a la siguiente marca especial:

SM1.0 (cero)

**Combinación Y con palabras, Combinación O con palabras, 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.

Operandos: IN1, IN2: VW, T, C, IW, QW, MW, SMW, AC, AIW, constante, \*VD, \*AC, SW

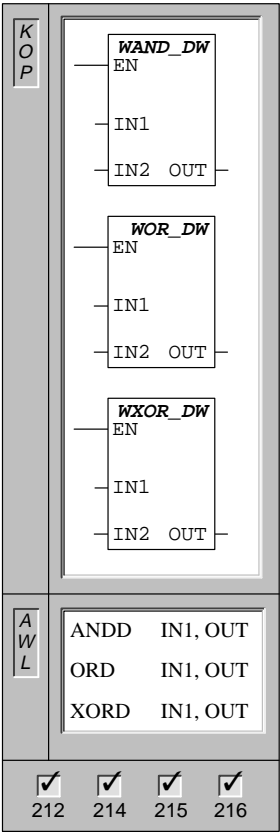
OUT: VW, T, C, IW, QW, MW, SMW, AC, \*VD, \*AC, SW

Nota: Al programar en KOP se puede reducir el espacio de memoria necesario indicando que IN1 sea igual a OUT.

Estas operaciones afectan a la siguiente marca especial:

SM1.0 (cero)

Combinación Y con palabras dobles, Combinación O con palabras dobles, 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.

Operandos: IN1, IN2: VD, ID, QD, MD, SMD, AC, HC, constante, \*VD, \*AC, SD  
OUT: VD, ID, QD, MD, SMD, AC, \*VD, \*AC, SD

Nota: Al programar en KOP se puede reducir el espacio de memoria necesario indicando que IN1 sea igual a OUT.

Estas operaciones afectan a la siguiente marca especial:

SM1.0 (cero)

# Ejemplos de las operaciones de combinación con Y, O y O-exclusiva

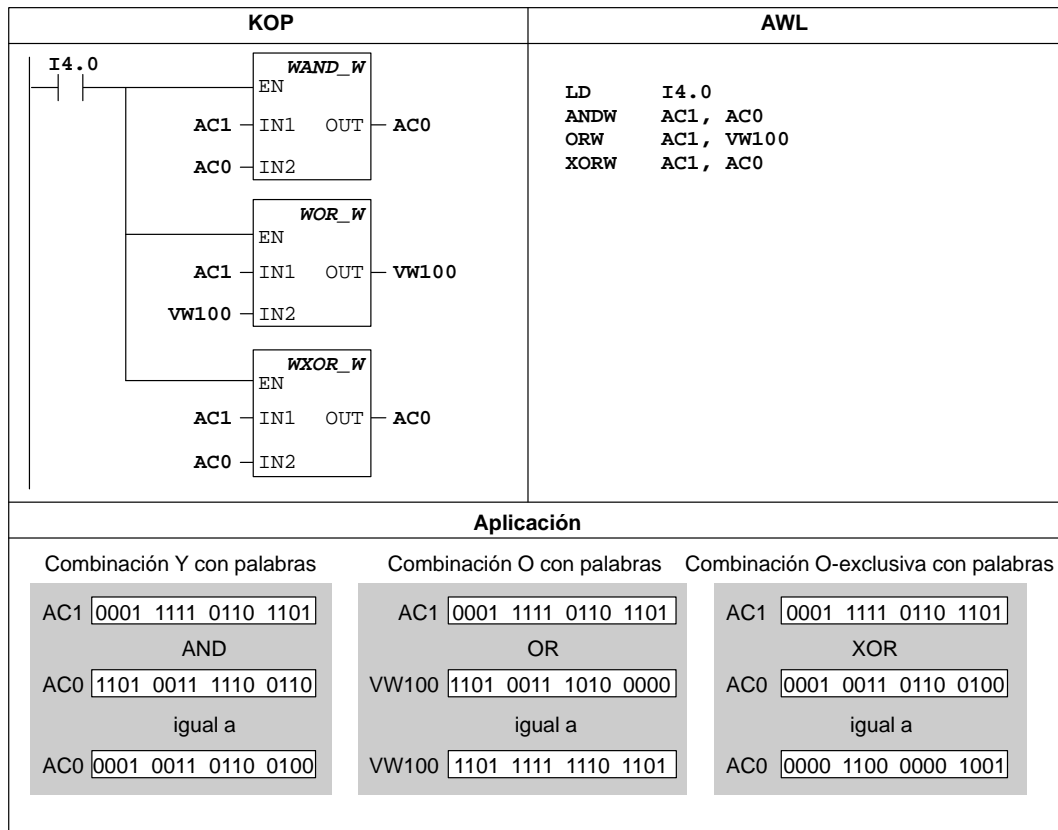
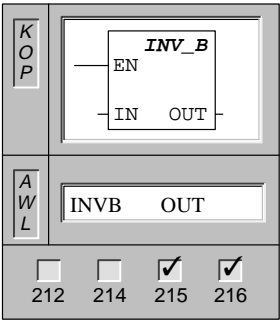


Figura 10-49 Ejemplo de las operaciones lógicas en KOP y AWL

**Invertir byte**



La operación **Invertir byte** forma el complemento a 1 del valor del byte de entrada (IN) y carga el resultado en un valor de byte (OUT).

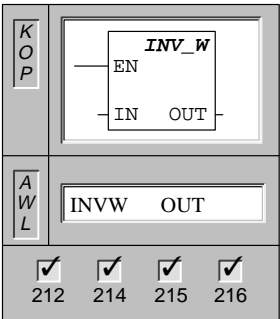
Operandos: IN: VB, IB, QB, MB, SMB, SB, AC, \*VD, \*AC, SB  
OUT: VB, IB, QB, MB, SMB, SB, AC, \*VD, \*AC, SB

Nota: Al programar en KOP se puede reducir el espacio de memoria necesario indicando que IN sea igual a OUT.

Esta operación afecta a la siguiente marca especial:

SM1.0 (cero)

**Invertir palabra**



La operación **Invertir palabra** forma el complemento a 1 del valor de la palabra de entrada (IN) y carga el resultado en un valor de palabra (OUT).

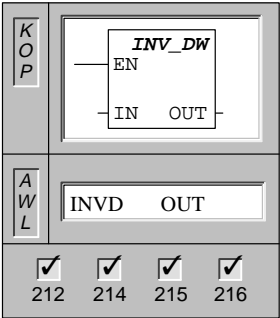
Operandos: IN: VW, T, C, IW, QW, MW, SMW, AC, AIW, constante, \*VD, \*AC, SW  
OUT: VW, T, C, IW, QW, MW, SMW, AC, \*VD, \*AC, SW

Nota: Al programar en KOP se puede reducir el espacio de memoria necesario indicando que IN sea igual a OUT.

Esta operación afecta a la siguiente marca especial:

SM1.0 (cero)

**Invertir palabra doble**



La operación **Invertir palabra doble** forma el complemento a 1 del valor de la palabra doble de entrada (IN) y carga el resultado en un valor de palabra doble (OUT).

Operandos: IN: VD, ID, QD, MD, SMD, AC, HC, constante, \*VD, \*AC, SD  
OUT: VD, ID, QD, MD, SMD, AC, \*VD, \*AC, SD

Nota: Al programar en KOP se puede reducir el espacio de memoria necesario indicando que IN sea igual a OUT.

Esta operación afecta a la siguiente marca especial:

SM1.0 (cero)

### Ejemplo de la operación Invertir

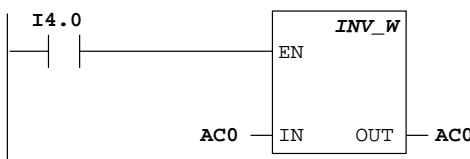
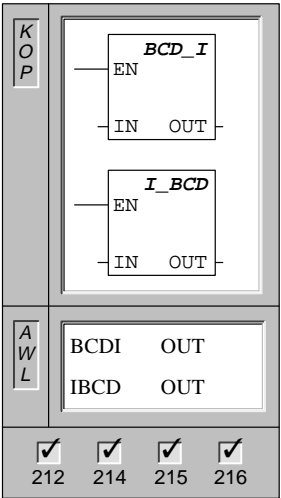
KOP		AWL	
		<pre>LD    I4.0 INVW  AC0</pre>	
Aplicación			
Invertir palabra			
AC0	<div>1101 0111 1001 0101</div>		
	complemento		
AC0	<div>0010 1000 0110 1010</div>		

Figura 10-50 Ejemplo de una operación Invertir en KOP y AWL

### 10.13 Operaciones de conversión

#### Convertir de BCD a entero, Convertir de entero a BCD



La operación **Convertir de BCD a entero** convierte el valor BCD (decimal codificado en binario) de entrada y carga el resultado en OUT.

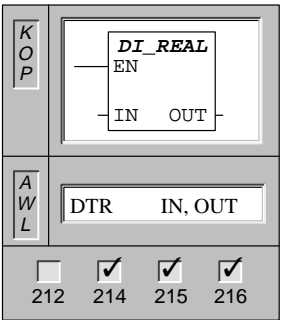
La operación **Convertir de entero a BCD** convierte el valor entero de entrada en un valor BCD y carga el resultado en OUT.

Operandos: IN: VW, T, C, IW, QW, MW, SMW, AC, AIW, constante, \*VD, \*AC, SW  
OUT: VW, T, C, IW, QW, MW, SMW, AC, \*VD, \*AC, SW

Nota: Al programar en KOP se puede reducir el espacio de memoria necesario indicando que IN sea igual a OUT.

Estas operaciones afectan a la siguiente marca especial:  
SM1.6 (BCD no válido)

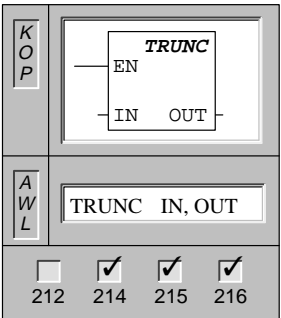
#### Convertir de entero de palabra doble a real



La operación **Convertir de entero de palabra doble a real** convierte un entero de 32 bits con signo (IN) en un número real de 32 bits (OUT).

Operandos: IN: VD, ID, QD, MD, SMD, AC, HC, constante, \*VD, \*AC, SD  
OUT: VD, ID, QD, MD, SMD, AC, \*VD, \*AC, SD

#### Truncar



La operación **Truncar** convierte un número real de 32 bits (IN) en un entero de 32 bits con signo (OUT). Sólo se convierte la parte entera del número real (redondear a cero).

Operandos: IN: VD, ID, QD, MD, SMD, AC, constante, \*VD, \*AC, SD  
OUT: VD, ID, QD, MD, SMD, AC, \*VD, \*AC, SD

Estas operaciones afectan a la siguiente marca especial:  
SM1.1 (desbordamiento)

## Ejemplos de convertir y truncar

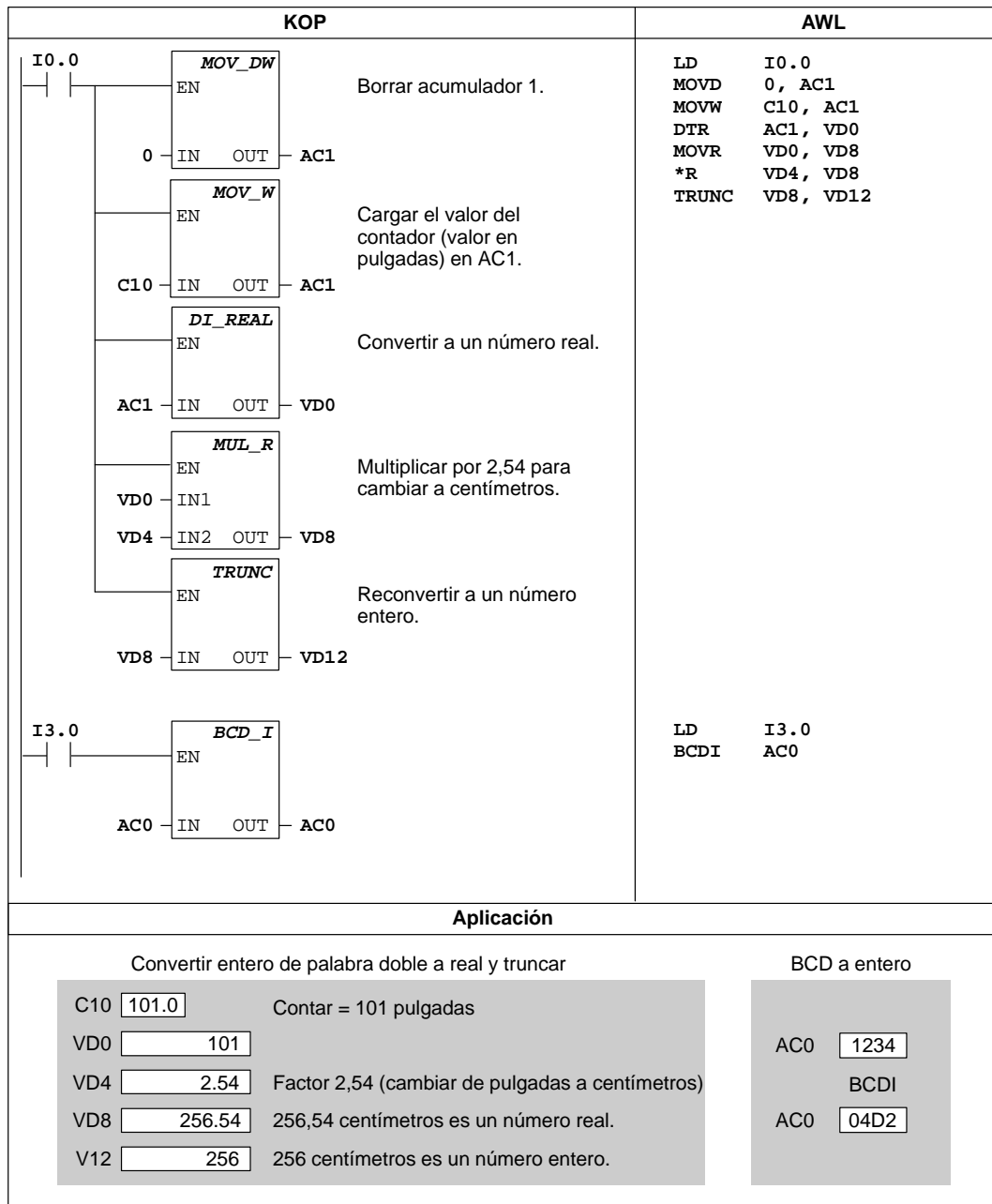
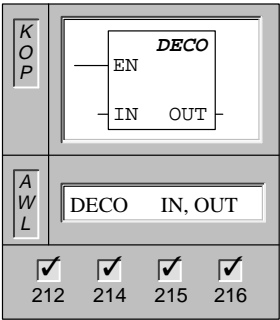


Figura 10-51 Ejemplo de una operación de conversión de un número real

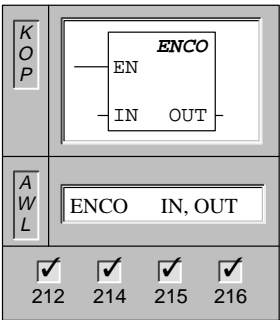
Decodificar



La operación **Decodificar** activa el bit de la palabra de salida (OUT). Dicho 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.

Operandos: IN: VB, IB, QB, MB, SMB, AC, constante, \*VD, \*AC, SB  
OUT: VW, T, C, IW, QW, MW, SMW, AC, AQW, \*VD, \*AC, SW

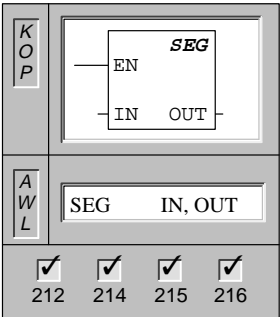
Codificar



La operación **Codificar** escribe el número del bit (bit #) menos significativo de la palabra de entrada (IN) en el medio byte menos significativo (4 bits) del byte de salida (OUT).

Operandos: IN: VW, T, C, IW, QW, MW, SMW, AC, AIW, constante, \*VD, \*AC, SW  
OUT: VB, IB, QB, MB, SMB, AC, \*VD, \*AC, SB

Segmento



La operación **Segmento** (SEG) genera 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).

Operandos: IN: VB, IB, QB, MB, SMB, AC, constante, \*VD, \*AC, SB  
OUT: VB, IB, QB, MB, SMB, AC, \*VD, \*AC, SB

La figura 10-52 muestra la codificación del indicador de siete segmentos utilizado por la operación Segmento.

(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 10-52 Codificación del indicador de siete segmentos

### Ejemplos de las operaciones Decodificar y Codificar

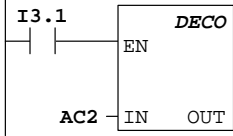
KOP	AWL
 <p>Activar el bit que corresponde al código de error en AC2.</p>	<pre>LD    I3.1 DECO  AC2, VW40</pre>
Aplicación	
<p>AC2 contiene el código de error 3. La operación DECO activa el bit en VW40 que corresponde a este código de error.</p>	
<div> <div>AC2</div> <div>3</div> </div> <div> <div>DECO</div> <div>15 3 0</div> <div>VW40</div> <div>0000 0000 0000 1000</div> </div>	

Figura 10-53 Activar un bit de error con la operación Decodificar (ejemplo)

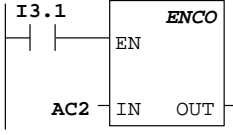
KOP	AWL
 <p>Convertir el bit de error de AC2 en el código de error de VB40.</p>	<pre>LD    I3.1 ENCO  AC2, VB40</pre>
Aplicación	
<p>AC2 contiene el bit de error. La operación ENCO convierte el bit menos significativo en un código de error que se almacena en VB40.</p>	
<div> <div>AC2</div> <div>15 9 0</div> <div>1001 0000 0000 0000</div> </div> <div> <div>ENCO</div> <div>VB40</div> <div>9</div> </div>	

Figura 10-54 Convertir el bit de error en un código de error con la operación Codificar (ejemplo)

### Ejemplo de la operación Segmento

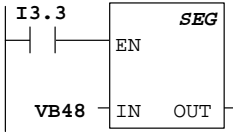
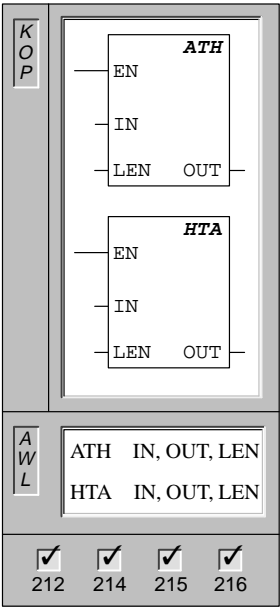
KOP	AWL
	<pre>LD    I3.3 SEG   VB48, AC1</pre>
Aplicación	
<div> <div>VB48</div> <div>05</div> </div> <div> <div>SEG</div> <div>AC1</div> <div>6D</div> <div>□ (carácter indicado)</div> </div>	

Figura 10-55 Ejemplo de la operación Segmento

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 la dirección 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 IN en una cadena ASCII, comenzando en la dirección OUT. El número de dígitos hexadecimales a convertir viene indicado por la longitud (LEN). Es posible convertir 255 dígitos hexadecimales como máximo.

Operandos: IN, OUT: VB, IB, QB, MB, SMB, \*VD, \*AC, SB  
LEN: VB, IB, QB, MB, SMB, AC, constante, \*VD, \*AC, SB

Los caracteres ASCII admisibles son los valores hexadecimales 30 a 39 y 41 a 46.

Estas operaciones afectan a la siguiente marca especial:  
SM1.7 (ASCII no válido)

### Ejemplo de la operación Convertir de ASCII a hexadecimal

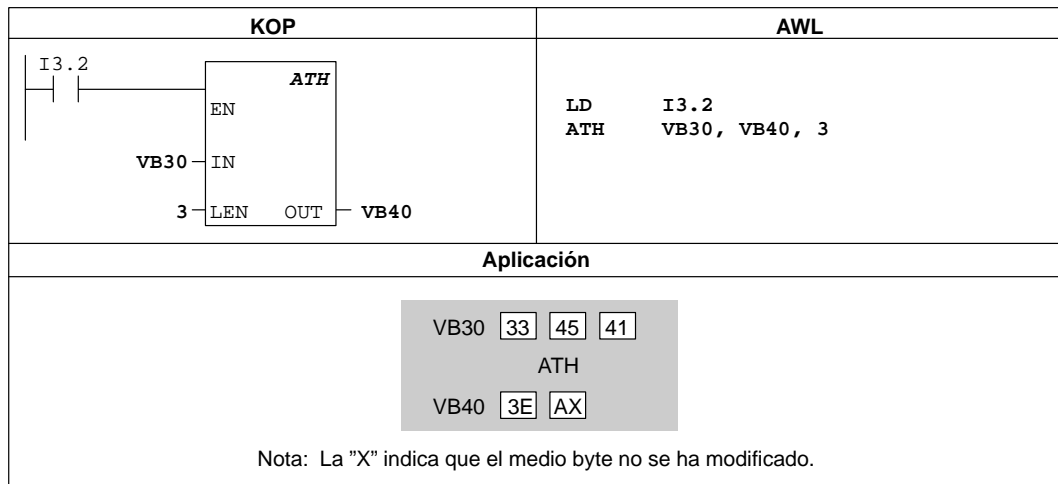
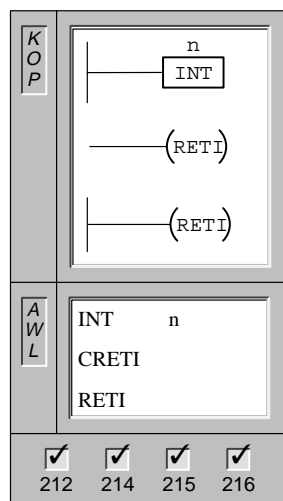


Figura 10-56 Ejemplo de la operación Convertir de ASCII a hexadecimal

## 10.14 Operaciones de interrupción y comunicación

### Comenzar rutina de interrupción, Retorno desde rutina de interrupción



La operación **Comenzar rutina de interrupción** marca el comienzo de la rutina de interrupción (n).

La operación **Retorno condicional desde rutina de interrupción** finaliza una rutina en función de la combinación lógica precedente.

La operación **Retorno absoluto desde rutina de interrupción** se debe utilizar para finalizar todas las rutinas de interrupción.

Operandos: n: 0 a 127

### Rutinas de interrupción

Toda rutina de interrupción se puede identificar con una marca de interrupción que indica el comienzo de la rutina. Ésta comprende las operaciones que se colocan entre dicha marca y la operación Retorno absoluto desde rutina de interrupción. La rutina de interrupción se ejecuta como respuesta a un evento interno o externo asociado. Para salir de la rutina (y devolver así el control al programa principal) se puede ejecutar la operación Retorno absoluto desde rutina de interrupción (RETI), o bien la operación Retorno condicional desde rutina de interrupción. Toda rutina de interrupción se debe finalizar con la operación Retorno absoluto.

### 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".

### Uso restringido

Al utilizar rutinas de interrupción, deberá tener en cuenta los siguientes puntos:

- Posicione todas las rutinas de interrupción después del final del programa principal KOP.
- No utilice las operaciones DISI, ENI, CALL, HDEF, FOR/NEXT, LSCR, SCRE, SCRT y END en las rutinas de interrupción.
- Finalice todas las rutinas de interrupción con la operación Retorno absoluto desde rutina de interrupción (RETI).

### Asistencia del sistema al producirse interrupciones

Las interrupciones pueden afectar a los contactos, bobinas y acumuladores. Por lo tanto, 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. Ello evita que se perturbe el programa principal debido a derivaciones a o desde rutinas de interrupción.

### 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 proporcionar datos a ser utilizados en 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 surgir de las acciones debidas a 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. Dichas 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 (MOV\_B, MOV\_W, MOV\_DW, MOV\_R). 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 múltiples 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 el lugar del programa principal donde se disponen las operaciones que acceden a las direcciones compartidas. Una vez ejecutadas todas las operaciones que utilicen dichas direcciones compartidas, se deberán habilitar de nuevo las interrupciones. Mientras esté inhibida la interrupción, la correspondiente rutina no se podrá ejecutar. 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**

K O P	—(ENI)			
	—(DISI)			
A W L	ENI			
	DISI			
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	212	214	215	216

La operación **Habilitar todos los eventos de interrupción** (ENI) habilita la ejecución de todos los eventos.

La operación **Inhibir todos los eventos de interrupción** (DISI) inhibe la ejecución de todos los eventos.

Operandos: ninguno

Cambiando a modo RUN se inhiben las interrupciones. En cuanto la CPU pasa a 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.

**Asociar interrupción, Desasociar interrupción**

K O P	—EN <b>ATCH</b>			
	—INT			
A W L	—EVENT			
	—EN <b>DTCH</b>			
A W L	—EVENT			
	ATCH INT, EVENT			
A W L	DTCH EVENT			
	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	212	214	215	216

La operación **Asociar interrupción** (ATCH) asocia el número de una rutina de interrupción (INT) a un evento de interrupción (EVENT), habilitando así éste último.

La operación **Desasociar interrupción** (DTCH) desasocia un evento de interrupción (EVENT) de todas las rutinas de interrupción, deshabilitando así el evento.

Operandos: INT : 0 a 127  
EVENT: 0 a 26

**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 a tal efecto 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 10-13 muestra los diferentes tipos de eventos de interrupción.

Tabla 10-13 Descripción de los eventos de interrupción

Nº de evento	Descripción de la interrupción	212	214	215	216
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
3	Flanco negativo, I0.1		Sí	Sí	Sí
4	Flanco positivo, I0.2		Sí	Sí	Sí
5	Flanco negativo, I0.2		Sí	Sí	Sí
6	Flanco positivo, I0.3		Sí	Sí	Sí
7	Flanco negativo, I0.3		Sí	Sí	Sí
8	Interface 0: Recibir carácter	Sí	Sí	Sí	Sí
9	Interface 0: Transmisión finalizada	Sí	Sí	Sí	Sí
10	Interrupción temporizada 0, SMB34	Sí	Sí	Sí	Sí
11	Interrupción temporizada 1, SMB35		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	Sí
14	HSC1, cambio de sentido		Sí	Sí	Sí
15	HSC1, puesto a 0 externamente		Sí	Sí	Sí
16	HSC2 CV=PV (valor actual = valor predeterminado)		Sí	Sí	Sí
17	HSC2, cambio de sentido		Sí	Sí	Sí
18	HSC2, puesto a 0 externamente		Sí	Sí	Sí
19	Interrupción Valor de conteo de impulsos PLS0		Sí	Sí	Sí
20	Interrupción Valor de conteo de impulsos PLS1		Sí	Sí	Sí
21	Interrupción Temporizador T32 CT = PT			Sí	Sí
22	Interrupción Temporizador T96 CT = PT			Sí	Sí
23	Interface 0: Recepción de mensajes finalizada			Sí	Sí
24	Interface 1: Recepción de mensajes finalizada				Sí
25	Interface 1: Recibir carácter				Sí
26	Interface 1: Transmisión finalizada				Sí
* Si el evento 12 (HSC0, CV = PV) está asociado a una interrupción, los eventos 0 y 1 no se podrán asociar a interrupciones. Del mismo modo, si los eventos 0 ó 1 están asociados a una interrupción, el evento 12 no se podrá asociar a una interrupción.					

## Interrupciones de comunicación

El interface serie de comunicación del sistema de automatización se puede controlar mediante un programa KOP o AWL. La comunicación a través de dicho interface 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 el apartado "Operaciones de comunicación".

## 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 10-14 figuran las entradas disponibles para las interrupciones. Los eventos Flanco positivo y Flanco negativo se pueden captar para cada una de dichas entradas. Estos eventos también se pueden utilizar para indicar una condición de error que requiera atención inmediata en cuanto se produzca el evento.

Tabla 10-14 Interrupciones asistidas en los flancos positivos y/o negativos

Interrupciones E/S	CPU 212	CPU 214	CPU 215	CPU 216
Entradas y salidas	I0.0	I0.0 a I0.3	I0.0 a I0.3	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. Cada uno de estos eventos de los contadores rápidos permite reaccionar ante eventos que no se puedan controlar durante el tiempo de ciclo del sistema de automatización.

Las interrupciones de salida de impulsos avisan inmediatamente cuándo 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 asistir una o más interrupciones temporizadas (v. tabla 10-15). 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 5 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.

Tabla 10-15 Interrupciones temporizadas asistidas

Interrupciones temporizadas	CPU 212	CPU 214	CPU 215	CPU 216
Nº de interrupciones temporizadas asistidas	1	2	2	2

Cada vez que transcurre la temporización, el evento de interrupción temporizado transfiere el control a la rutina de interrupción correspondiente. Por lo general, estos eventos de interrupción se utilizan para consultar periódicamente las entradas analógicas.

Al asociar un evento de interrupción temporizado a una rutina de interrupción, se habilita el evento e inmediatamente empieza a transcurrir la temporización. Durante ese proceso, el sistema capta el 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 reasociarse 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 dicha cola). La figura 10-58 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. Dichas interrupciones se asisten 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 ejecuta cuando el valor actual del temporizador activo sea igual a su valor predeterminado al actualizar la CPU el temporizador de 1 ms (v. apt. 10.5). 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 se guía según el siguiente esquema fijo:

- 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 a la vez. Cuando se comienza a ejecutar una rutina de interrupción, se procesa hasta el final. No se podrá interrumpir por otra, ni siquiera por una rutina de mayor prioridad. Las interrupciones que se presenten mientras se está ejecutando otra interrupción se ponen en cola de espera para ser procesadas posteriormente.

La tabla 10-16 muestra las tres colas de espera y el número máximo de interrupciones que pueden acoger.

Tabla 10-16 Colas de espera y número máximo de interrupciones que pueden acoger

Cola de espera	CPU 212	CPU 214	CPU 215	CPU 216
Interrupciones de comunicación	4	4	4	8
Interrupciones E/S	4	16	16	16
Interrupciones temporizadas	2	4	8	8

Potencialmente 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 10-17 muestra dichas 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 10-17 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 10-18 muestra las interrupciones, sus prioridades y los números de los eventos asociados.

Tabla 10-18 Descripción de los eventos de interrupción

Nº de evento	Descripción de la interrupción	Prioridad	Prioridad de grupo
8	Interface 0: Recibir carácter	Comunicación (más alta)	0
9	Interface 0: Transmisión finalizada		0*
23	Interface 0: Recepción de mensajes finalizada		0*
24	Interface 1: Recepción de mensajes finalizada		1
25	Interface 1: Recibir carácter		1*
26	Interface 1: Transmisión finalizada		1*
0	Flanco positivo, I0.0**	E/S (media)	0
2	Flanco positivo, I0.1		1
4	Flanco positivo, I0.2		2
6	Flanco positivo, I0.3		3
1	Flanco negativo, I0.0**		4
3	Flanco negativo, I0.1		5
5	Flanco negativo, I0.2		6
7	Flanco negativo, I0.3		7
12	HSC0 CV=PV (valor actual = valor predeterminado)**		0
13	HSC1 CV=PV (valor actual = valor predeterminado)		8
14	HSC1, cambio de sentido		9
15	HSC1, puesto a 0 externamente		10
16	HSC2 CV=PV (valor actual = valor predeterminado)		11
17	HSC2, cambio de sentido		12
18	HSC2, puesto a 0 externamente		13
19	Interrupción Valor de conteaje de impulsos PLS0		14
20	Interrupción Valor de conteaje de impulsos PLS1		15
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
* Puesto que la comunicación es semidúplex por naturaleza, las interrupciones de transmisión y recepción tienen la misma prioridad.			
** Si el evento 12 (HSC0, CV = PV) está asociado a una interrupción, los eventos 0 y 1 no se podrán asociar a interrupciones. Del mismo modo, si los eventos 0 ó 1 están asociados a una interrupción, el evento 12 no se podrá asociar a una interrupción.			

Ejemplo de interrupciones

La figura 10-57 muestra un ejemplo de operaciones con rutinas de interrupción.

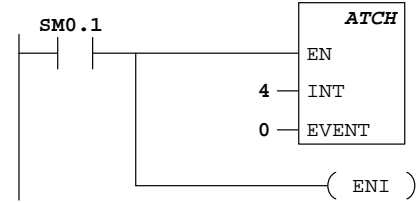
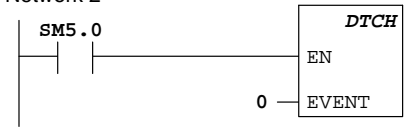

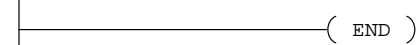



KOP		AWL
Network 1		En el primer ciclo: Definir rutina de interrupción 4 como Flanco positivo en I0.0.
Network 2		Habilitar todos los eventos de interrupción.
Network 3		Si se detecta un error E/S, inhibir la interrupción Flanco positivo en I0.0. (Esta línea de corriente es opcional).
Network 50		Inhibir todas las interrupciones si se activa M5.0.
Network 60		Fin programa principal KOP.
Network 65		Subrutina de interrupción Flanco positivo en E/S.
Network 66		Retorno condicional debido a un error de E/S.
		Fin de la rutina de interrupción Flanco positivo en I0.0.

Figura 10-57 Ejemplo de las operaciones de interrupción

La figura 10-58 muestra cómo leer el valor de una entrada analógica mediante una interrupción temporizada.

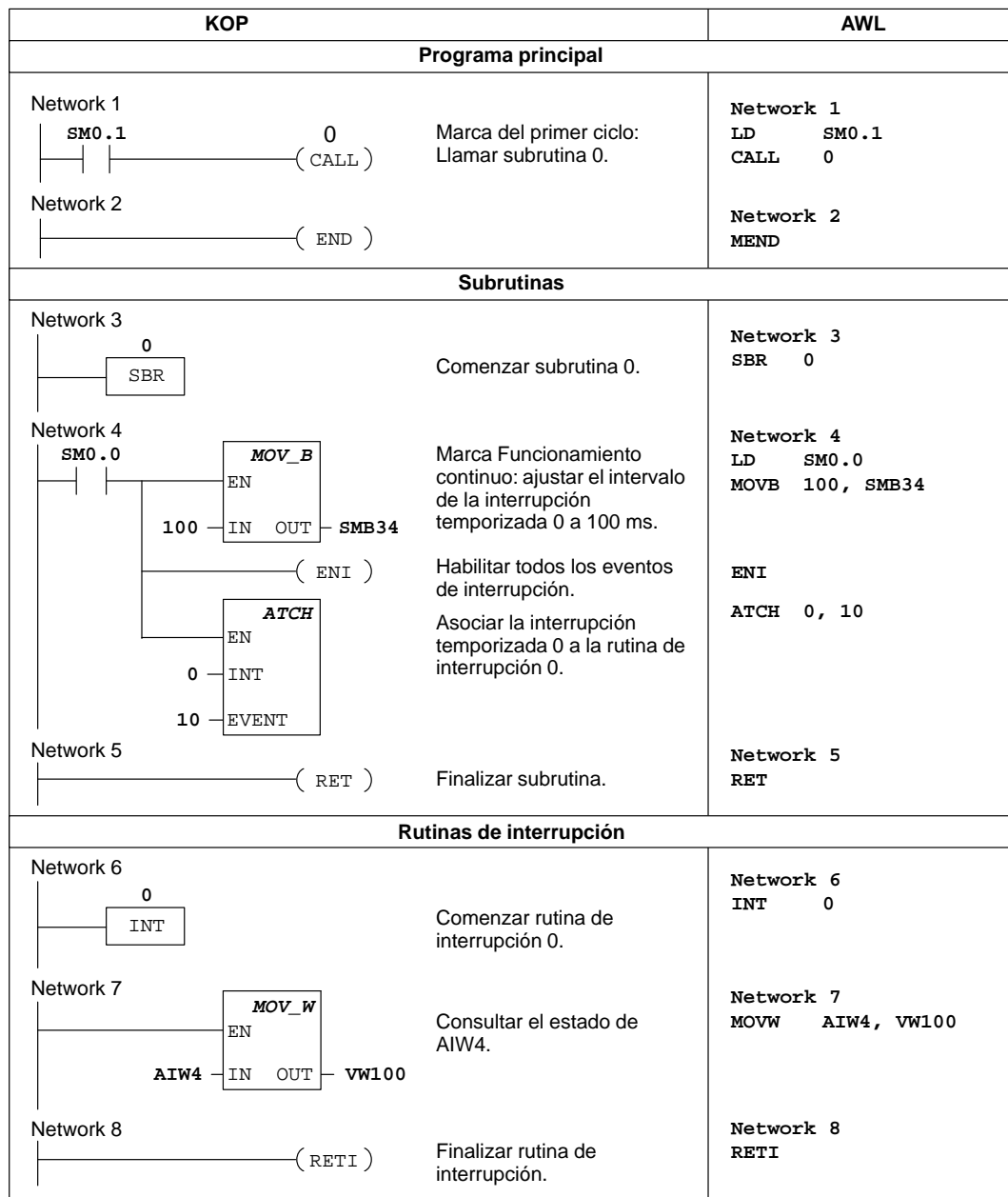


Figura 10-58 Ejemplo de lectura de una entrada analógica mediante una interrupción temporizada

Transmitir mensaje, Recibir mensaje

K O P	<b>XMT</b>	
	— EN	
	— TABLE	
	— PORT	
A W L	XMT TABLE, PORT	
	<input checked="" type="checkbox"/> 212	<input checked="" type="checkbox"/> 214
	<input checked="" type="checkbox"/> 215	<input checked="" type="checkbox"/> 216

La operación **Transmitir mensaje** (XMT) transmite el mensaje contenido en el búfer de datos (TABLE). La primera entrada del búfer indica cuántos bytes se han de transmitir. PORT indica el interface de programación por donde se va a transmitir.

Operandos: TABLE: VB, IB, QB, MB, SMB, \*VD, \*AC, SB  
PORT: 0 a 1

La operación XMT se utiliza en modo Freeport para transmitir datos por el (los) interface(s) de comunicación.

K O P	<b>RCV</b>	
	— EN	
	— TABLE	
	— PORT	
A W L	RCV TABLE, PORT	
	<input type="checkbox"/> 212	<input type="checkbox"/> 214
	<input checked="" type="checkbox"/> 215	<input checked="" type="checkbox"/> 216

La operación **Recibir mensaje** (RCV) llama a los cambios de los ajustes que inician o finalizan 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 interface indicado (PORT) se almacenan en el búfer de datos (TABLE). La primera entrada del búfer indica el número de bytes que se han recibido.

Operandos: TABLE: VB, IB, QB, MB, SMB, \*VD, \*AC, SB  
PORT: 0 a 1

La operación RCV se utiliza en modo Freeport para recibir datos por el (los) interface(s) de comunicación.

Modo Freeport

El programa de usuario puede controlar el interface serie de la CPU. La comunicación a través de dicho interface se denomina modo Freeport (comunicación programable por el usuario). Al elegirse el modo Freeport, el programa KOP controla el interface 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 por completo el protocolo de comunicación. Las marcas especiales SMB30 (para el interface 0) y SMB130 (para el interface 1, si la CPU dispone de dos interfaces) se utilizan para elegir la velocidad de transferencia y la paridad.

El modo Freeport se inhibe y la comunicación se restablece (p.ej., acceso a través de la unidad de programación) cuando la CPU pasa a modo STOP.

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 asistir 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 (interface 0) o de SMB130 (interface 1). Estando en modo Freeport, la CPU no se puede comunicar con la unidad de programación.

---

**Nota**

El paso 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 interfaces de comunicación 0 y 1, respectivamente, permitiendo elegir la velocidad de transferencia, la paridad y el número de bits por carácter. La tabla 10-19 muestra los bytes de control del modo Freeport.

Tabla 10-19 Bytes de marcas especiales SMB30 y SMB130

Interface 0	Interface 1	Descripción
Formato de SMB30	Formato de SMB130	<div style="display: flex; justify-content: space-between; align-items: center;"> <div style="text-align: center;"> <small>MSB</small> 7 </div> <div style="text-align: center;"> <small>LSB</small> 0 </div> </div> <div style="display: flex; align-items: center; margin-top: 5px;"> <div style="border: 1px solid black; padding: 2px; margin: 0 2px;">p</div> <div style="border: 1px solid black; padding: 2px; margin: 0 2px;">p</div> <div style="border: 1px solid black; padding: 2px; margin: 0 2px;">d</div> <div style="border: 1px solid black; padding: 2px; margin: 0 2px;">b</div> <div style="border: 1px solid black; padding: 2px; margin: 0 2px;">b</div> <div style="border: 1px solid black; padding: 2px; margin: 0 2px;">b</div> <div style="border: 1px solid black; padding: 2px; margin: 0 2px;">m</div> <div style="border: 1px solid black; padding: 2px; margin: 0 2px;">m</div> </div> <div style="margin-left: 10px;">Byte de control del modo Freeport</div>
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 (para la CPU 212: = 19.200 bits/s) 001 = 19.200 bits/s 010 = 9.600 bits/s 011 = 4.800 bits/s 100 = 2.400 bits/s 101 = 1.200 bits/s 110 = 600 bits/s 111 = 300 bits/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: En el caso del interface 0 se genera un bit de parada para todas las configuraciones, con excepción de los 7 bits por carácter (sin paridad), donde se generan dos bits de parada. En el caso del interface 1 se genera un bit de parada para todas las configuraciones.		

#### Utilizar la operación XMT para transmitir datos

La operación XMT facilita la transferencia de datos. Con dicha operación 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 interface 0 y evento de interrupción 26 para el interface 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.

#### Utilizar la operación RCV para recibir datos

La operación RCV facilita la recepción de mensajes. Con dicha operación 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 interface 0 y evento de interrupción 24 para el interface 1), si una rutina de interrupción se ha asociado al evento Recepción de mensajes finalizada. También es posible recibir mensajes sin utilizar interrupciones, vigilando a tal efecto la marca especial SMB86.

SMB86 (o SMB186) no será igual a cero al estar desactivado el cuadro RCV. 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. La tabla 10-20 (SMB86 a SMB94 para el interface 0 y SMB186 a SMB194 para el interface 1) describe dichas condiciones.

---

#### 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).

---

Interface 0	Interface 1	Descripción
SMB86	SMB186	<div><div><div>MSB 7</div><div><div>n</div><div>r</div><div>e</div><div>0</div><div>0</div><div>t</div><div>c</div><div>p</div></div><div>LSB 0</div></div><div>Byte de estado de recepción de mensajes</div></div> <p>n: 1 = El usuario ha inhibido la recepción de mensajes</p> <p>r: 1 = Se finalizó la recepción de mensajes: error en parámetros de entrada o falta condición inicial o final</p> <p>e: 1 = Carácter final recibido</p> <p>t: 1 = Se finalizó la recepción de mensajes: ha transcurrido la temporización</p> <p>c: 1 = Se finalizó la recepción de mensajes: se ha excedido el número máximo de caracteres</p> <p>p: 1 = Se finalizó la recepción de mensajes debido a un error de paridad</p>
SMB87	SMB187	<div><div><div>MSB 7</div><div><div>n</div><div>x</div><div>y</div><div>z</div><div>m</div><div>t</div><div>0</div><div>0</div></div><div>LSB 0</div></div><div>Byte de control de recepción de mensajes</div></div> <p>n: 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.</p> <p>x: 0 = Ignorar SMB88 o SMB188. 1 = Utilizar el valor de SMB88 o de SMB188 para detectar el comienzo del mensaje.</p> <p>y: 0 = Ignorar SMB89 o SMB189. 1 = Utilizar el valor de SMB89 o de SMB189 para detectar el fin del mensaje.</p> <p>z: 0 = Ignorar SMW90 o SMB190. 1 = Utilizar el valor de SMW90 para detectar una condición de inactividad.</p> <p>m: 0 = Utilizar el temporizador como temporizador entre caracteres. 1 = Utilizar el temporizador como temporizador de mensajes.</p> <p>t: 0 = Ignorar SMW92 o SMW192. 1 = Finalizar la recepción si se excede el período de tiempo indicado en SMW92 o SMW192.</p> <p>Estos bits definen los criterios para identificar el mensaje (incluyendo los criterios para el comienzo y el fin del mensaje). Para determinar el comienzo de un mensaje, los criterios habilitados a tal efecto se combinan mediante Y, debiendo presentarse en forma de secuencia (línea de inactividad seguida de un carácter inicial). Para determinar el fin de un mensaje, los criterios habilitados a tal efecto se combinan mediante O.</p> <p>Ecuaciones de los criterios para el comienzo y el fin de un mensaje: Comienzo del mensaje = z * x Fin del mensaje = y + t + número máximo de caracteres alcanzados</p> <p>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).</p>
SMB88	SMB188	Carácter de comienzo del mensaje.
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.

Tabla 10-20 Marcas especiales SMB86 a SMB94 y SMB186 a SMB194, continuación

Interface 0	Interface 1	Descripción
SMB92 SMB93	SMB192 SMB193	Vigilancia de tiempo 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 terminación de mensajes por el conteo de caracteres.

### Recibir datos mediante interrupciones de caracteres

Para disponer de una mayor flexibilidad en los protocolos asistidos, 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 interfaces 0 y 1. Si debido a la recepción de un carácter por el interface 0 se ejecuta la rutina de interrupción asociada a ese evento (evento de interrupción 8), SMB2 contendrá el carácter recibido por el interface 0, en tanto que SMB3 contendrá la paridad de dicho carácter. Si debido a la recepción de un carácter por el interface 1 se ejecuta la rutina de interrupción asociada a ese evento (evento de interrupción 25), SMB2 contendrá el carácter recibido por el interface 1, en tanto que SMB3 contendrá la paridad de dicho carácter.

Ejemplo de las operaciones Recibir mensaje y Transmitir mensaje

Este programa de ejemplo 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.

KOP			AWL
Network 1			Network 1
SM0.1	MOV_B	En el primer ciclo: – Inicializar modo Freeport – Elegir 9600 bit/s – Elegir 8 bits de datos – Elegir sin paridad	LD SM0.1
16#9	IN OUT SMB30		MOVB 16#9, SMB30
	MOV_B	Inicializar byte de control RCV mensajes – RCV habilitada – Carácter para detectar fin de mensaje – Detectar condición de inactividad como condición inicial del mensaje	MOVB 16#B0, SMB87
16#B0	IN OUT SMB87		MOVB 16#0A, SMB89
	MOV_B	Ajustar carácter final del mensaje en hex 0A (cambio de línea)	MOVW +5, SMW90
16#A	IN OUT SMB89		MOVB 100, SMB94
	MOV_W	Ajustar timeout de línea vacía en 5 ms	ATCH 0, 23
+5	IN OUT SMW90		ATCH 1, 9
	MOV_B	Ajustar número máx. de caracteres en 100	ENI
100	IN OUT SMB94		RCV VB100, 0
	ATCH	Asociar interrupción a evento Recepción finalizada	
0	INT		
23	EVENT		
	ATCH	Asociar interrupción a evento Transmisión finalizada	
1	INT		
9	EVENT		
	( ENI )	Habilitar las interrupciones de usuario.	
	RCV	Habilitar cuadro RCV con búfer en VB100 para interface 0	
VB100	TABLE		
0	PORT		

Figura 10-59 Ejemplo de la operación Transmitir mensaje

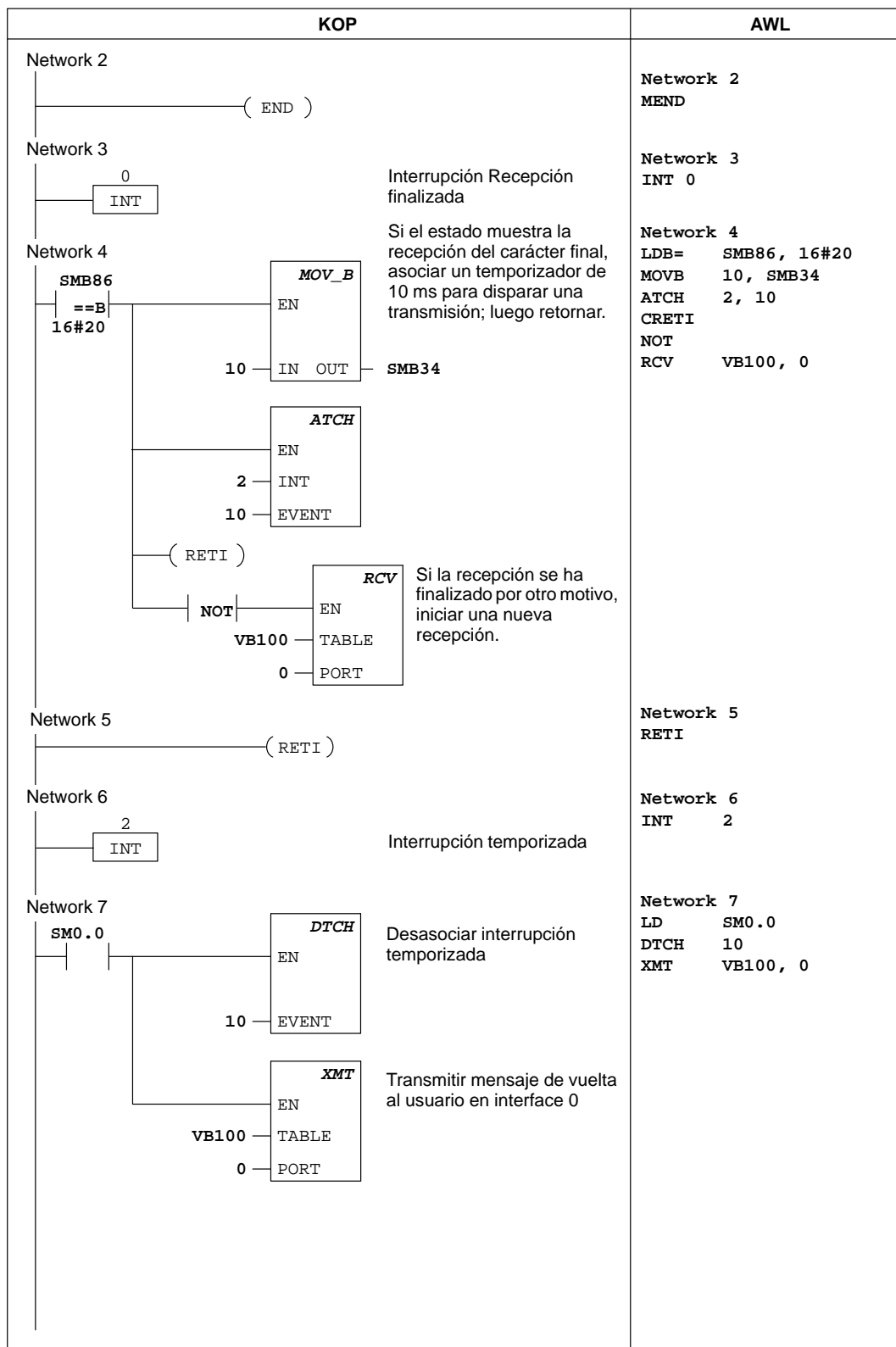


Figura 10-59 Ejemplo de la operación Transmitir mensaje (continuación)

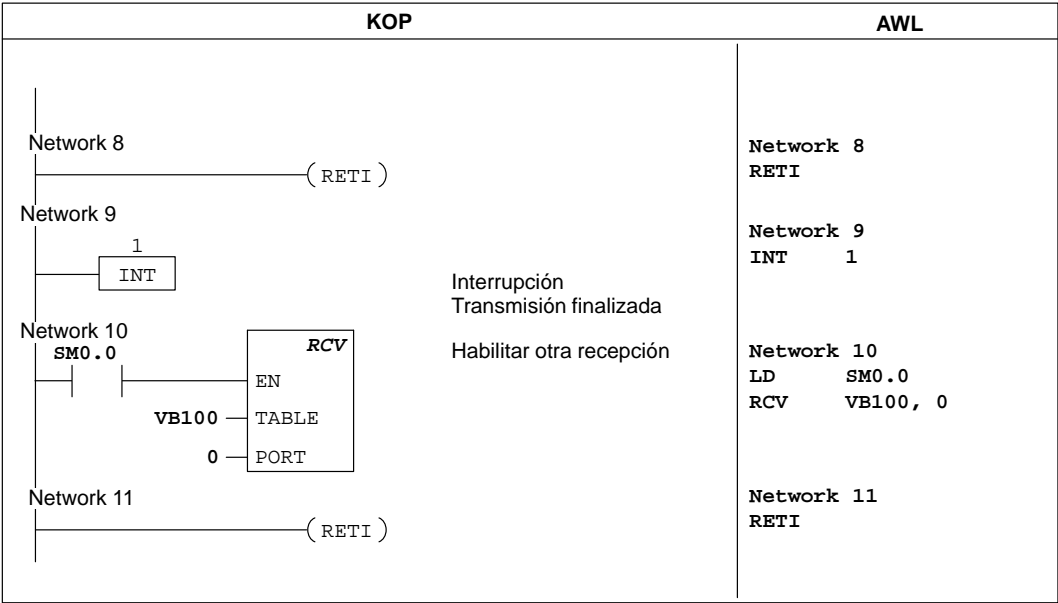
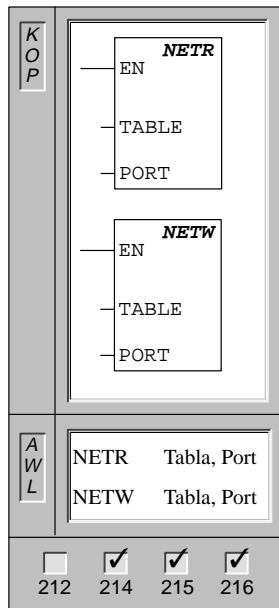


Figura 10-59 Ejemplo de la operación Transmitir mensaje (continuación)

## Leer de la red, Escribir en la red



La operación **Leer de la red** (NETR) inicia una comunicación para registrar datos de una estación remota a través del interface indicado (PORT), según define la tabla de descripción (TABLE).

La operación **Escribir en la red** (NETW) inicia una comunicación para escribir datos en una estación remota a través del interface indicado (PORT), según define la tabla de descripción (TABLE).

Operandos: TABLE: VB, MB, \*VD, \*AC

PORT: 0 a 1

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. Es posible activar simultáneamente ocho operaciones NETR y NETW como máximo. 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 10-60 muestra la tabla a la que hace referencia el parámetro TABLE en las operaciones NETR y NETW.

		D Concluida (operación ejecutada): 0 = no ejecutada 1 = ejecutada		
		A Activa (operación en cola de espera): 0 = no activa 1 = activa		
		E Error (operación incluye un error): 0 = sin error 1 = error		
Desplaz. de byte	7	0		
0	D	A	E	Código de error
1	Dirección de la estación remota			
2	Puntero al área de datos			
3	en la			
4	estación remota			
5	(I, Q, M, S o V)			
6	Longitud de datos			
7	Byte de datos 0			
8	Byte de datos			
	↑			
22	Byte de datos 15			

**Dirección de la estación remota:** dirección del PLC a cuyos datos se desea acceder.

**Puntero al área de datos en la estación remota:** puntero indirecto a los datos que se desea acceder.

**Longitud de datos:** número de bytes de datos a acceder en la estación remota (1 a 16 bytes)

**Área de datos a recibir o transferir:** 1 a 16 bytes reservados para los datos, como se describe a continuación:

Para NETR, esta área de datos es donde se almacenarán los valores leídos de la estación remota después de la operación NETR.

Para NETW, esta área de datos es donde se almacenarán los valores a enviar a la estación remota después de la operación NETW.

Código de error	Definición
0	Sin error
1	Error de timeout; no responde la estación remota
2	Error de recepción; error de paridad, de ajuste o de suma de verificación en la respuesta
3	Error offline; colisiones causadas por direcciones de estación repetidas o fallo del hardware
4	Error de desbordamiento en la cola de espera; se han activado más de ocho cuadros NETR/NETW
5	Violación de protocolo; se intentó ejecutar NETR/NETW sin habilitar PPI+ en SMB30
6	Parámetro no válido; la tabla NETR/NETW contiene un valor no válido
7	Sin recursos; la estación remota está ocupada (secuencia de cargar en PG o de cargar en CPU en curso)
8	Error de capa 7; violación de aplicación de protocolo
9	Error de mensaje; dirección de datos errónea o longitud de datos incorrecta
A-F	No utilizado; (reservado para futuras aplicaciones)

Figura 10-60 Definición de TABLE para NETR y NETW

Ejemplo de las operaciones Leer de la red y Escribir en la red

La figura 10-61 muestra un ejemplo para ilustrar la utilidad de las operaciones NETR (Leer de la red) y NETW (Escribir en la red). En este ejemplo se considera una línea de producción donde se están llenando botes de mantequilla que se envían a una de las cuatro máquinas empaquetadoras. La empaquetadora embala ocho botes de mantequilla en cada caja. Una máquina distribuidora controla el flujo de los botes de mantequilla hacia cada una de las empaquetadoras. Se utilizan cuatro CPUs 212 para controlar las empaquetadoras y una CPU 214 equipada con un visualizador de textos TD 200 para controlar a la distribuidora. La figura 10-61 muestra también la configuración de la red.

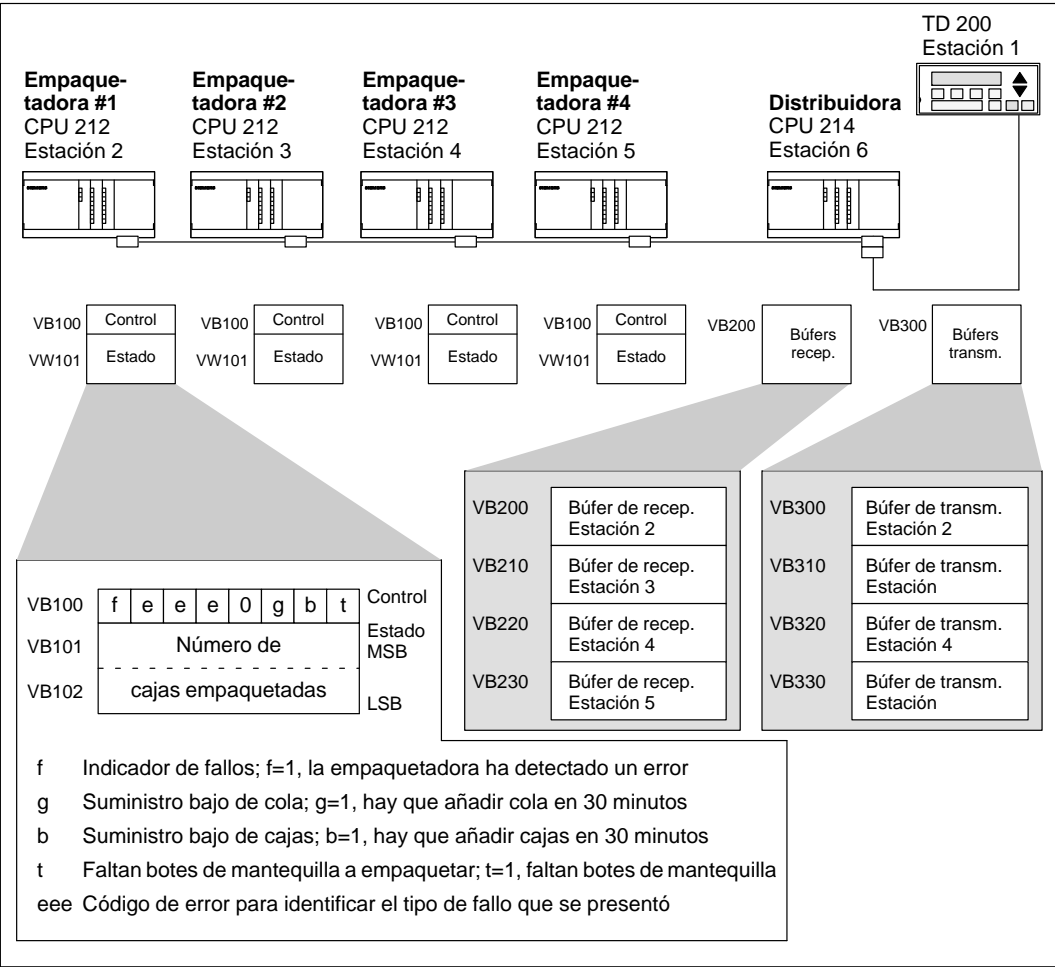


Figura 10-61 Ejemplo de las operaciones NETR y NETW

La figura 10-62 muestra detalladamente los búfers de recepción y transmisión para acceder a los datos de la estación 2 (en VB200 y VB300, respectivamente).

La CPU 214 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 10-63 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 10-62 Ejemplo de TABLE para NETR y NETW

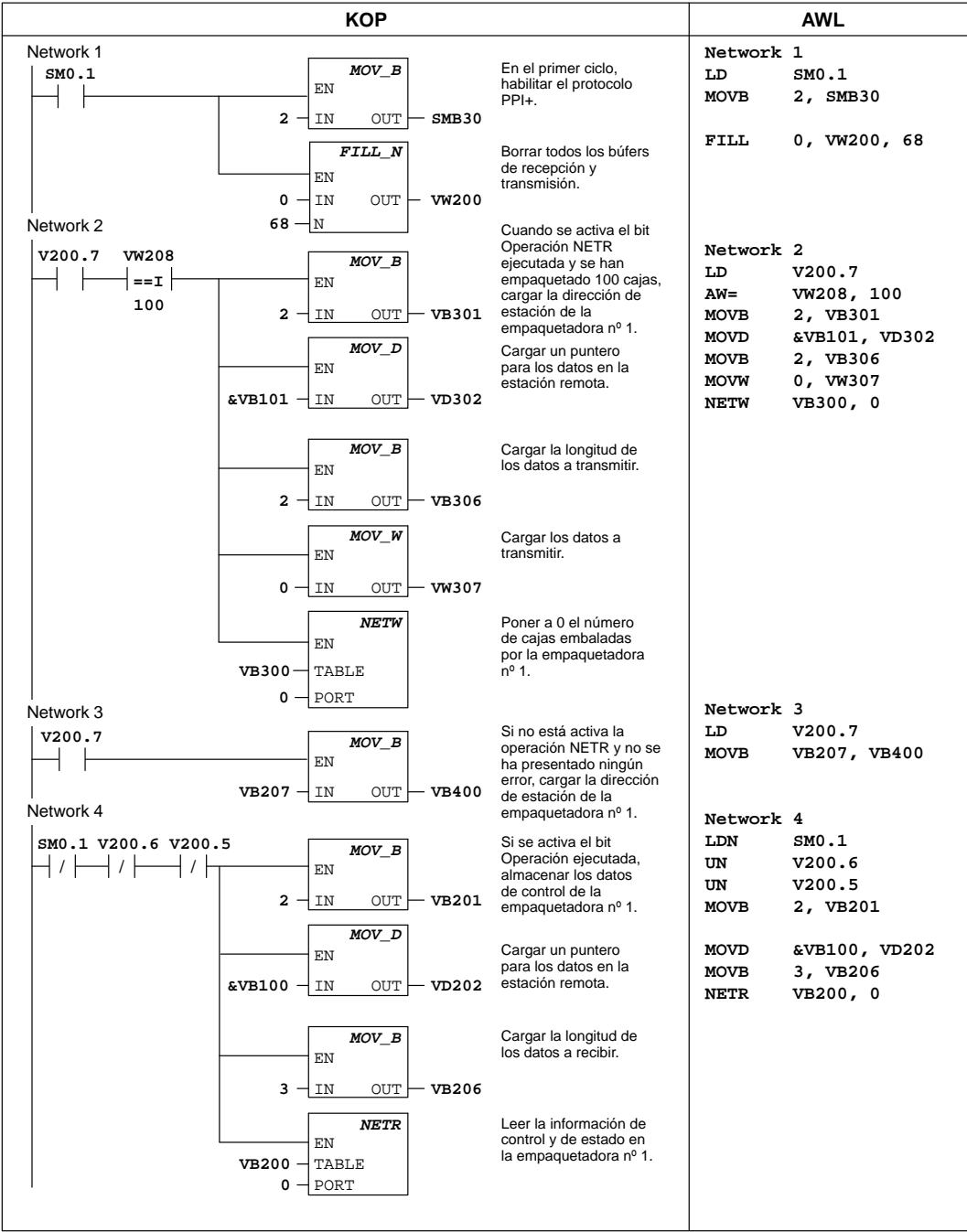


Figura 10-63 Ejemplo de las operaciones NETR y NETW

# Datos técnicos

# A

## Índice del capítulo

Apartado	Descripción	Página
A.1	Datos técnicos generales	A-3
A.2	CPU 212 alimentación DC, entradas DC, salidas DC	A-6
A.3	CPU 212 alimentación AC, entradas DC, salidas de relé	A-8
A.4	CPU 212 alimentación AC 24 V, entradas DC 24 V, salidas de relé	A-10
A.5	CPU 212 alimentación AC, entradas AC, salidas AC	A-12
A.6	CPU 212 alimentación AC, entradas DC tipo fuente, salidas de relé	A-14
A.7	CPU 212 alimentación AC, entradas AC 24 V, salidas AC	A-16
A.8	CPU 212 alimentación AC, entradas AC, salidas de relé	A-18
A.9	CPU 214 alimentación DC, entradas DC, salidas DC	A-20
A.10	CPU 214 alimentación AC, entradas DC, salidas de relé	A-22
A.11	CPU 214 alimentación AC, entradas AC, salidas AC	A-24
A.12	CPU 214 alimentación AC, entradas tipo fuente DC, salidas de relé	A-26
A.13	CPU 214 alimentación AC, entradas AC 24 V, salidas AC	A-28
A.14	CPU 214 alimentación AC, entradas AC, salidas de relé	A-30
A.15	CPU 215 alimentación DC, entradas DC, salidas DC	A-32
A.16	CPU 215 alimentación AC, entradas DC, salidas de relé	A-34
A.17	CPU 216 alimentación DC, entradas DC, salidas DC	A-36
A.18	CPU 216 alimentación AC, entradas DC, salidas de relé	A-38
A.19	Módulo de ampliación EM 221, 8 entradas digitales DC 24 V	A-40
A.20	Módulo de ampliación EM 221, 8 entradas digitales AC 120 V	A-41
A.21	Módulo de ampliación EM 221, 8 entradas digitales tipo fuente DC 24 V	A-42
A.22	Módulo de ampliación EM 221, 8 entradas digitales AC 24 V	A-43
A.23	Módulo de ampliación EM 222, 8 salidas digitales DC 24 V	A-44
A.24	Módulo de ampliación EM 222, 8 salidas de relé	A-45
A.25	Módulo de ampliación EM 222, 8 salidas digitales AC 120/230 V	A-46
A.26	Módulo de ampliación EM 223, 4 entradas digitales DC 24 V / 4 salidas digitales DC 24 V	A-48
A.27	Módulo de ampliación EM 223, 8 entradas digitales DC 24 V / 8 salidas digitales DC 24 V	A-50
A.28	Módulo de ampliación EM 223, 16 entradas digitales DC 24 V / 16 salidas digitales DC 24 V	A-52
A.29	Módulo de ampliación EM 223, 4 entradas digitales DC 24 V / 4 salidas digitales de relé	A-54

Apartado	Descripción	Página
A.30	Módulo de ampliación EM 223, 4 entradas digitales AC 120 V / 4 salidas digitales AC 120/230 V	A-55
A.31	Módulo de ampliación EM 223, 8 entradas digitales DC 24 V / 8 salidas digitales de relé	A-56
A.32	Módulo de ampliación EM 223, 16 entradas digitales DC 24 V / 16 salidas digitales de relé	A-58
A.33	Módulo de ampliación EM 231, 3 entradas analógicas AI de 12 bits	A-60
A.34	Módulo de ampliación EM 232, 2 salidas analógicas AQ de 12 bits	A-66
A.35	Módulo de ampliación EM 235, 3 entradas analógicas AI / 1 salida analógica AQ de 12 bits	A-69
A.36	Cartucho de memoria 8K x 8	A-78
A.37	Cartucho de memoria 16K x 8	A-79
A.38	Cartucho de pila	A-80
A.39	Cable de módulo de ampliación	A-81
A.40	Cable PC/PPI	A-82
A.41	Simulador de entradas DC para la CPU 212	A-84
A.42	Simulador de entradas DC para la CPU 214	A-85
A.43	Simulador de entradas DC para las CPUs 215/216	A-86

## A.1 Datos técnicos generales

### Homologaciones nacionales e internacionales

Las homologaciones nacionales e internacionales que se indican a continuación fueron empleadas para determinar las características apropiadas de funcionamiento y comprobación de la gama S7-200. En la tabla A-1 se indica la conformidad específica con dichas 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
- 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

Las CPUs S7-200 y todos los módulos de ampliación S7-200 cumplen con los datos técnicos indicados en la tabla A-1.

Tabla A-1 Datos técnicos de la gama S7-200

<b>Condiciones ambientales – Transporte y almacenamiento</b>	
IEC 68-2-2, Test Bb, calor seco y IEC 68-2-1, Test Ab, Frío	-40° C a +70° C
IEC 68-2-30, Test 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
<b>Condiciones ambientales – Funcionamiento</b>	
Temperatura de funcionamiento	0° C a 55° C, 95% de humedad máxima no condensante
IEC 68-2-14 Test 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	0,35 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.
<b>Compatibilidad electromagnética – Inmunidad<sup>1</sup> según EN50082-2<sup>1</sup></b>	
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 EN50204	26 MHz a 1 GHz 10 V/m, 80% modulación con señal de 1 kHz  900 MHz ± 5 MHz, 10 V/m, 50% ciclo de trabajo, frecuencia de repetición 200 Hz
EN 61000-4-4 (IEC 801-4) Transitorios 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) Resistencia a sobrecargas	2 kV asimétrico, 1 kV simétrico 5 impulsos positivos / 5 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 2 según EN50081 -1 y -2 <sup>2</sup>	
EN 55011, clase A, grupo 1, conducida <sup>1</sup> 0,15 MHz a 0,5 MHz 0,5 MHz 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 <sup>1</sup> 30 MHz a 230 kHz 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 A, grupo 1, conducida <sup>3</sup> 0,15 a 0,5 MHz  0,5 MHz a 5 MHz 5 MHz a 30 MHz	< 66 dB (mV) decremento casi cresta con frecuencia logarítmica a 56 dB (μV) < 56 dB (mV) 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 A, grupo 1, radiada <sup>3</sup> 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/5 V circuitos 115 V circuitos hasta 24 V/5 V 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 Aplicable a todos los equipos que lleven la marca CE (Comunidad Europea).
- 3 La unidad deberá montarse en una caja metálica puesta a tierra. La línea de alimentación de corriente alterna se deberá equipar con un filtro Schaffner FN 680-2.5/06 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 proporcionados por el comercio especializado. El rendimiento real puede variar dependiendo de la aplicación.

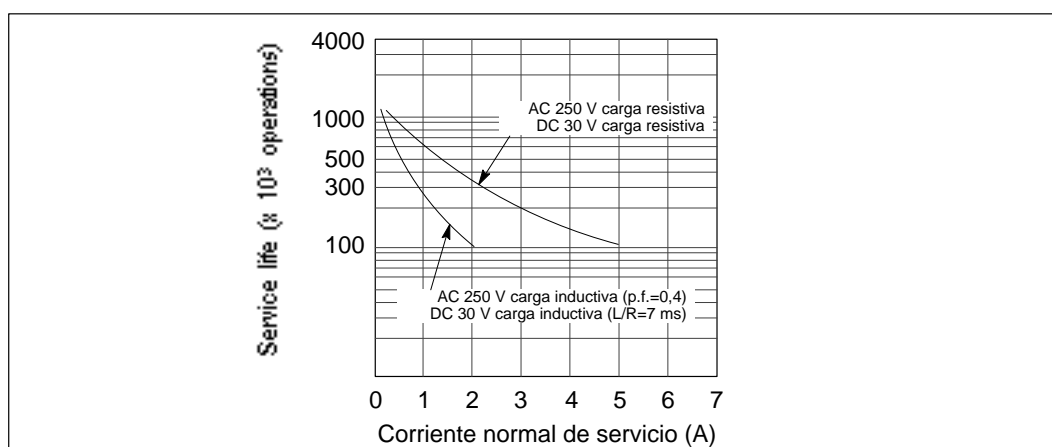


Figura A-1 Vida útil de los relés

## A.2 CPU 212 alimentación DC, entradas DC, salidas DC

Nº de referencia: 6ES7 212-1AA01-0XB0

Características generales		Salidas (continuación)	
Dimensiones (l x a x p)	160 x 80 x 62 mm	Retardo de conmutación	25 µs ON, 120 µs OFF
Peso	0,3 kg	Sobrecorriente momentánea	4 A, 100 ms
Disipación	5 W a 1,75 A de carga	Caída de tensión	1,8 V máx. con corriente máxima
Tamaño programa de usuario / memoria	512 palabras/EEPROM	Separación galvánica	AC 500 V, 1 minuto
Tamaño datos usuario/memoria	512 palabras/RAM	Protección contra cortocircuitos	ninguna
Retención de datos	típ. 50 h (8 h mín. a 40° C)	<b>Entradas</b>	
E/S locales <sup>1</sup>	8 entradas/6 salidas	Tipo de entrada (IEC 1131-2)	Tipo 1, con sumidero de corriente
Número máximo de módulos de ampliación	2	Área en ON	DC 15 V a 30 V, mín. 4 mA DC 35 V, 500 ms sobretensión transitoria
E/S digitales asistidas	64 entradas/64 salidas	Tensión nominal en ON	DC 24 V, 7 mA
E/S analógicas asistidas	16 entradas/16 salidas	Máximo en OFF	DC 5 V, 1 mA
Velocidad de ejecución booleana	1,2 µs/operación	Tiempo de respuesta I0.0 a I0.7	máx. 0,3 ms
Marcas internas	128	Separación galvánica	AC 500 V, 1 minuto
Temporizadores	64 temporizadores	<b>Alimentación</b>	
Contadores	64 contadores	Margen de tensión	DC 20,4 a 28,8 V
Contadores rápidos	1 software (máx. 2 KHz)	Corriente de entrada	típ. 60 mA, sólo CPU 500 mA carga máx.
Potenciómetros analógicos	1	Medición UL/CSA	50 VA
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según VDE 0160 según CE	Tiempo de retardo	mín. 10 ms de DC 24 V
<b>Salidas</b>		Extra-corriente de cierre	10 A pico a DC 28,8 V
Tipo de salida	Transistor, tipo fuente	Protección con fusibles (no reemplazable)	1 A, 125 V, de acción lenta
Margen de tensión	20,4 VDC a 28,8 VDC	Corriente disponible DC 5 V	260 mA para CPU 340 mA para módulo de ampliación
Corriente de carga máxima por cada salida	0 a 40 °C 0,75 A 55° C <sup>2</sup> 0,50 A	Aislamiento	no
por 2 salidas adyacentes	1,00 A	<b>Alimentación para sensores DC</b>	
todas las salidas en total	2,25 A	Margen de tensión	DC 16,4 V a 28,8 V
Carga inductiva, apriete	(por hilo neutro)	Rizado/corriente parásita (<10 MHz)	igual a la tensión alimentada
Impulso individual	2A L/R = 10 ms 1A L/R = 100 ms	Corriente disponible DC 24 V	180 mA
Repetición	1 W disipación de energía (1/2 Li <sup>2</sup> x frecuencia de conmutación < 1W)	Limitación de corriente de cortocircuito	< 600 mA
Corriente de derivación	100 µA	Aislamiento	no

<sup>1</sup> En la CPU están previstas 8 entradas y 8 salidas en la imagen del proceso para E/S locales.

<sup>2</sup> Disminución lineal de potencia 40 a 55° C. Disminución de potencia montaje vertical 10° C.

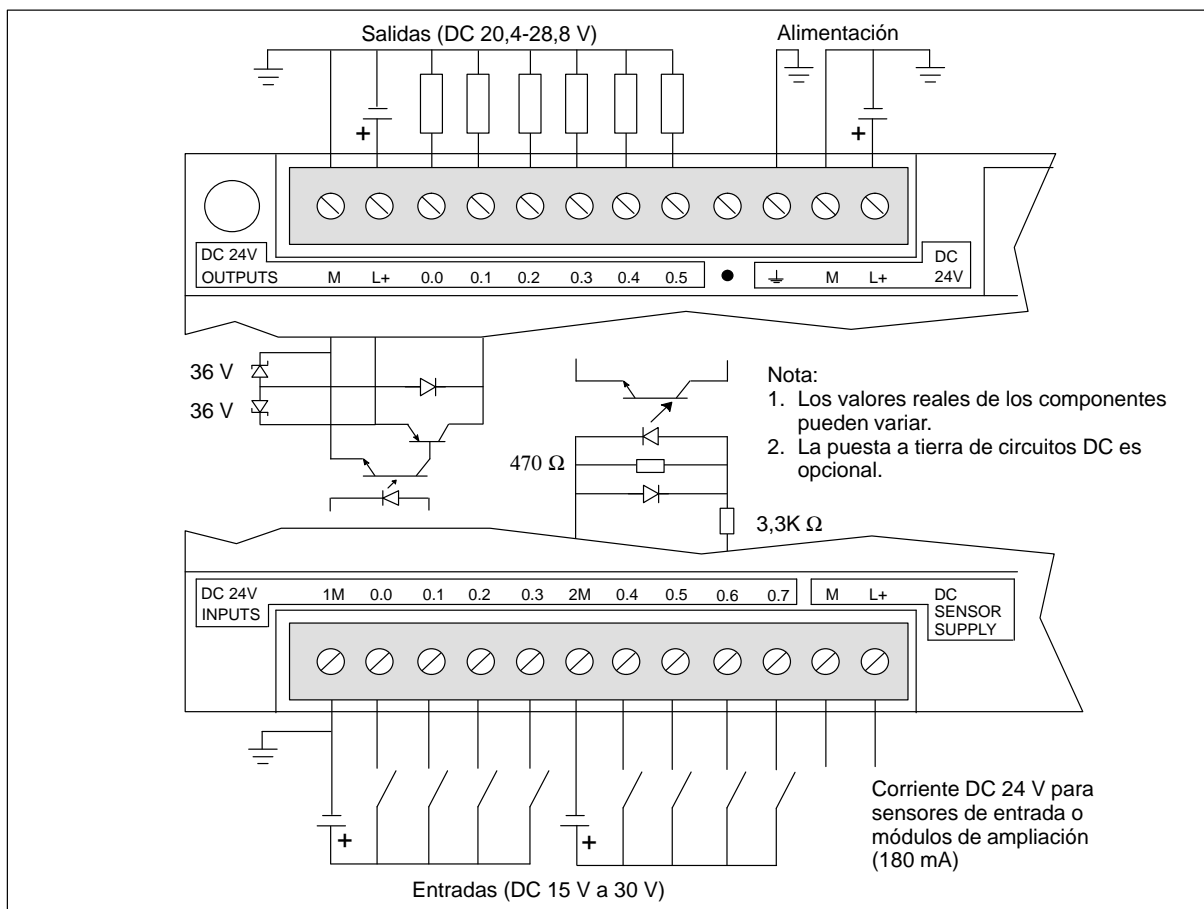


Figura A-2 Identificación de terminales de conexión para la CPU 212 DC/DC/DC

### A.3 CPU 212 alimentación AC, entradas DC, salidas de relé

Nº de referencia: 6ES7 212-1BA01-0XB0

Características generales	
Dimensiones (l x a x p)	160 x 80 x 62 mm
Peso	0,4 kg
Disipación	6 W
Tamaño programa de usuario / memoria	512 palabras/EEPROM
Tamaño datos usuario/memoria	512 palabras/RAM
Retención de datos	típ. 50 h (8 h mín. a 40° C)
E/S locales <sup>1</sup>	8 entradas/6 salidas
Número máximo de módulos de ampliación	2
E/S digitales asistidas	64 entradas/64 salidas
E/S analógicas asistidas	16 entradas/16 salidas
Velocidad de ejecución booleana	1,2 µs/operación
Marcas internas	128
Temporizadores	64 temporizadores
Contadores	64 contadores
Contadores rápidos	1 software (máx. 2 KHz)
Potenciómetros analógicos	1
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según VDE 0160 según CE
Salidas	
Tipo de salida	Relé, contacto de baja potencia
Margen de tensión	DC 5 V a 30 V / AC 250 V
Corriente de carga máxima	2 A/salida, 6 A/hilo neutro
Sobretensión transitoria	7A al estar cerrados los contactos
Resistencia de aislamiento	mín. 100 MΩ (nuevo)
Retardo de conmutación	máx. 10 ms
Vida útil	10 000 000 mecánico 100 000 con carga normal
Resistencia de contacto	máx. 200 mΩ (nuevo)
Aislamiento	
bobina a contacto	AC 1500 V, 1 minuto
contacto a contacto (entre contactos abiertos)	AC 750 V, 1 minuto
Protección contra cortocircuitos	ninguna

Entradas	
Tipo de entrada (IEC 1131-2)	Tipo 1, con sumidero de corriente
Área en ON	DC 15 V a 30 V, mín. 4 mA DC 35 V, 500 ms sobretensión transitoria
Tensión nominal en ON	DC 24 V, 7 mA
Máximo en OFF	DC 5 V, 1 mA
Tiempo de respuesta I0.0 a I0.7	máx. 0,3 ms
Separación galvánica	AC 500 V, 1 minuto
Alimentación	
Margen de tensión/ de frecuencia	AC 85 V a 264 V a 47 a 63 Hz
Corriente de entrada	típ. 4 VA, sólo CPU 50 VA carga máx.
Tiempo de retardo	mín. 20 ms de AC 110 V
Extra-corriente de cierre	20 A pico a AC 264 V
Protección con fusibles (no reemplazable)	2 A, 250 V, de acción lenta
Corriente disponible DC 5 V	260 mA para CPU 340 mA para módulo de ampliación
Aislamiento	Sí. Transformador, AC 1500 V, 1 minuto
Alimentación para sensores DC	
Margen de tensión	DC 20,4 a 28,8 V
Rizado/corriente parásita (<10 MHz)	máx. 1 V de pico a pico
Corriente disponible DC 24 V	180 mA
Limitación de corriente de cortocircuito	< 600 mA
Aislamiento	no

1 En la CPU están previstas 8 entradas y 8 salidas en la imagen del proceso para E/S locales.

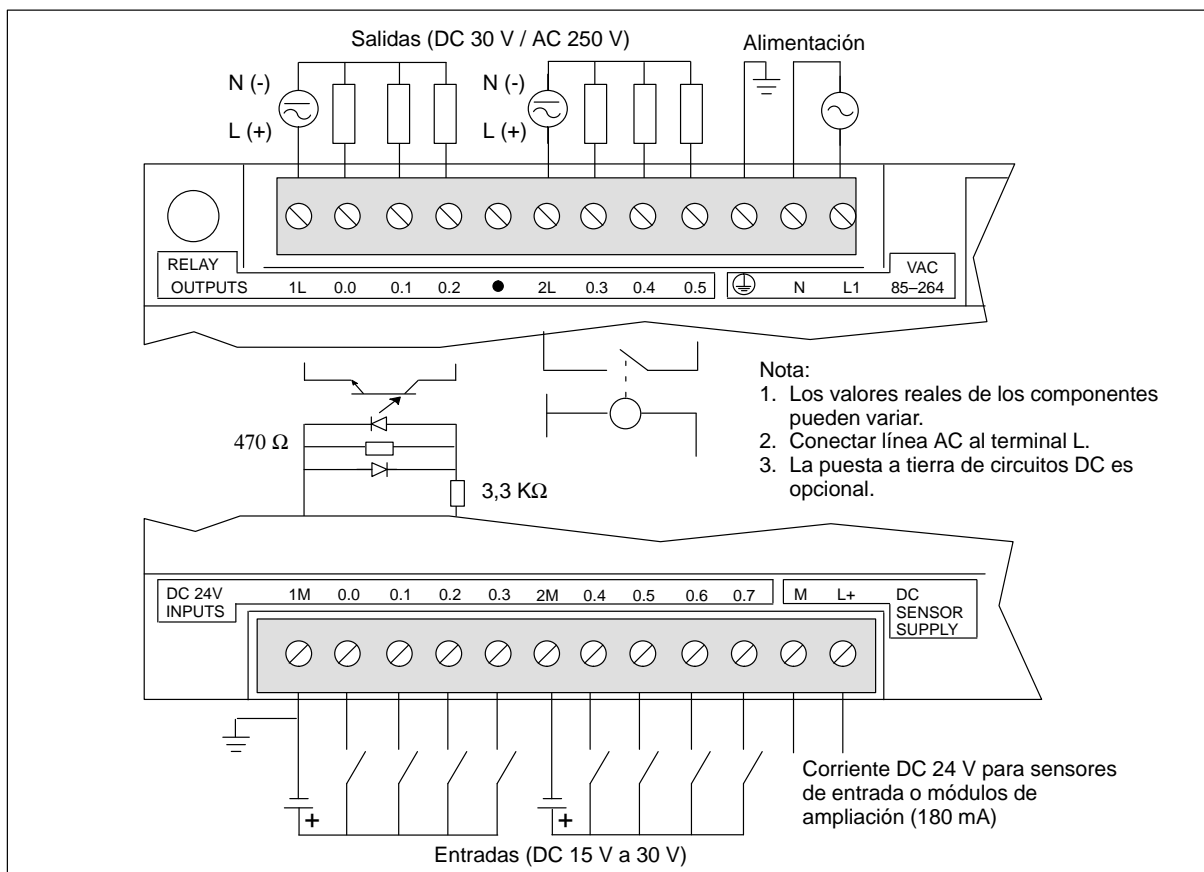


Figura A-3 Identificación de terminales de conexión para la CPU 212 AC/DC/relé

## A.4 CPU 212 alimentación AC 24 V, entradas DC, salidas de relé

Nº de referencia: 6ES7 212-1FA01-0XB0

Características generales	
Dimensiones (l x a x p)	160 x 80 x 62 mm
Peso	0,4 kg
Disipación	6 W
Tamaño programa de usuario / memoria	512 palabras/EEPROM
Tamaño datos usuario/memoria	512 palabras/RAM
Retención de datos	típ. 50 h (8 h mín. a 40° C)
E/S locales <sup>1</sup>	8 entradas/6 salidas
Número máximo de módulos de ampliación	2
E/S digitales asistidas	64 entradas/64 salidas
E/S analógicas asistidas	16 entradas/16 salidas
Velocidad de ejecución booleana	1,2 µs/operación
Marcas internas	128
Temporizadores	64 temporizadores
Contadores	64 contadores
Contadores rápidos	1 software (máx. 2 KHz)
Potenciómetros analógicos	1
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según VDE 0160 según CE
Salidas	
Tipo de salida	Relé, contacto de baja potencia
Margen de tensión	DC 5 V a 30 V / AC 250 V
Corriente de carga máxima	2 A/salida, 6 A/hilo neutro
Sobretensión transitoria	7A al estar cerrados los contactos
Resistencia de aislamiento	mín. 100 MΩ (nuevo)
Retardo de conmutación	máx. 10 ms
Vida útil	10 000 000 mecánico 100 000 con carga normal
Resistencia de contacto	máx. 200 mΩ (nuevo)
Aislamiento	
bobina a contacto	AC 1500 V, 1 minuto
contacto a contacto (entre contactos abiertos)	AC 750 V, 1 minuto
Protección contra cortocircuitos	ninguna

Entradas	
Tipo de entrada (IEC 1131-2)	Tipo 1, con sumidero de corriente
Área en ON	DC 15 V a 30 V, mín. 4 mA DC 35 V, 500 ms sobretensión transitoria
Tensión nominal en ON	DC 24 V, 7 mA
Máximo en OFF	DC 5 V, 1 mA
Tiempo de respuesta I0.0 a I0.7	máx. 0,3 ms
Separación galvánica	AC 500 V, 1 minuto
Alimentación	
Margen de tensión/ de frecuencia	AC 20 V a 29 V, 47 a 63 Hz
Corriente de entrada	típ. 4 VA, sólo CPU 50 VA carga máx.
Tiempo de retardo	mín. 20 ms de AC 24 V
Extra-corriente de cierre	20 A pico a AC 29 V
Protección con fusibles (no reemplazable)	2 A, 250 V, de acción lenta
Corriente disponible DC 5 V	260 mA para CPU 340 mA para módulo de ampliación
Aislamiento	Sí. Transformador, AC 500 V, 1 minuto
Alimentación para sensores DC	
Margen de tensión	DC 20,4 a 28,8 V
Rizado/corriente parásita (<10 MHz)	máx. 1 V de pico a pico
Corriente disponible DC 24 V	180 mA
Limitación de corriente de cortocircuito	< 600 mA
Aislamiento	no

1 En la CPU están previstas 8 entradas y 8 salidas en la imagen del proceso para E/S locales.

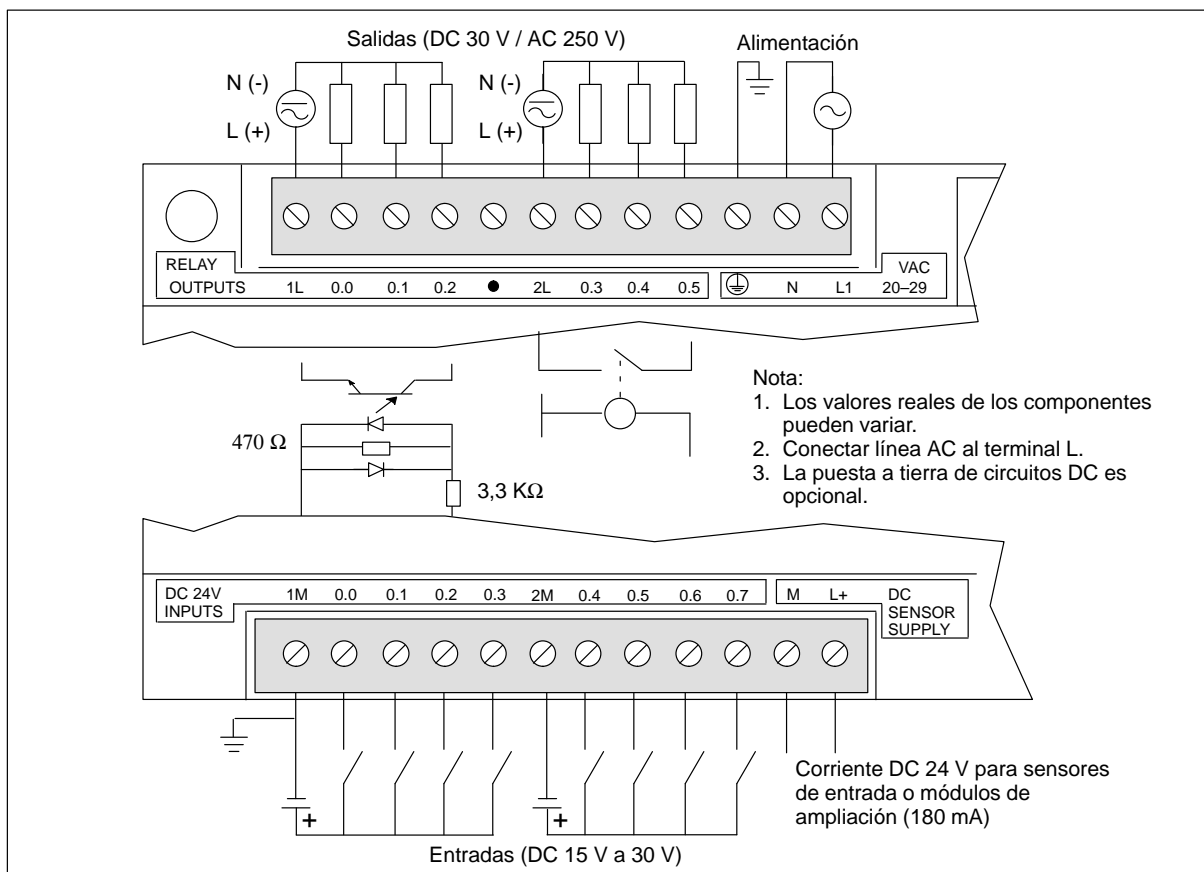


Figura A-4 Identificación de terminales de conexión para la CPU 212 24 VAC/DC/relé

## A.5 CPU 212 alimentación AC, entradas AC, salidas AC

Nº de referencia: 6ES7 212-1CA01-0XB0

Características generales		Salidas (continuación)	
Dimensiones (l x a x p)	160 x 80 x 62 mm	Retardo de conmutación	1/2 ciclo
Peso	0,4 kg	Sobrecorriente momentánea	30 A pico, 1 ciclo / 10 A pico, 5 ciclos
Disipación	7 W a 2,5 A de carga	Caída de tensión	máx. 1,5 V con corriente máx.
Tamaño programa de usuario / memoria	512 palabras/EEPROM	Separación galvánica	AC 1500 V, 1 minuto
Tamaño datos usuario/memoria	512 palabras/RAM	Protección contra cortocircuitos	ninguna
Retención de datos	típ. 50 h (8 h mín. a 40° C)	<b>Entradas</b>	
E/S locales <sup>1</sup>	8 entradas/6 salidas	Tipo de entrada (IEC 1131-2)	Tipo 1, con sumidero de corriente
Número máximo de módulos de ampliación	2	Área en ON	AC 79 V a 135 V, 47 a 63 Hz, mín. 4 mA
E/S digitales asistidas	64 entradas/64 salidas	Tensión nominal en ON	AC 120 V, 60 Hz, 7 mA
E/S analógicas asistidas	16 entradas/16 salidas	Máximo en OFF	AC 20 V, 1 mA
Velocidad de ejecución booleana	1,2 µs/operación	Tiempo de respuesta	típ. 10 ms, máx. 15 ms
Marcas internas	128	Separación galvánica	AC 1500 V, 1 minuto
Temporizadores	64 temporizadores	<b>Alimentación</b>	
Contadores	64 contadores	Margen de tensión/ de frecuencia	AC 85 V a 264 V a 47 a 63 Hz
Contadores rápidos	1 software (máx. 50 Hz)	Corriente de entrada	típ. 4 VA, sólo CPU 50 VA carga máx.
Potenciómetros analógicos	1	Tiempo de retardo	mín. 20 ms de AC 110 V
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según CE	Extra-corriente de cierre	20 A pico a AC 264 V
<b>Salidas</b>		Protección con fusibles (no reemplazable)	2 A, 250 V, de acción lenta
Tipo de salida	Triac, pasando por cero	Corriente disponible DC 5 V	320 mA para CPU 280 mA para módulo de ampliación
Margen de tensión/ de frecuencia	AC 20 V a 264 V, 47 a 63 Hz	Aislamiento	Sí. Transformador, AC 1500 V, 1 minuto
Factor de potencia del circuito de carga	0,3 a 1,0	<b>Alimentación para sensores DC</b>	
Carga inductiva, apriete	MOV 275 V tensión de trabajo	Margen de tensión	DC 20,4 a 28,8 V
Corriente de carga máxima por cada salida	0 a 40 °C    55° C <sup>2</sup> 1,20 A    1,00 A	Rizado/corriente parásita (<10 MHz)	máx. 1 V de pico a pico
por 2 salidas adyacentes	1,50 A    1,25 A	Corriente disponible DC 24 V	180 mA
todas las salidas en total*	3,50 A    2,50 A	Limitación de corriente de cortocircuito	< 600 mA
Corriente de carga mínima	30 mA	Aislamiento	no
Corriente de derivación	1,5 mA, AC 120 V/2,0 mA, AC 240 V		

<sup>1</sup> En la CPU están previstas 8 entradas y 8 salidas en la imagen del proceso para E/S locales.

<sup>2</sup> Disminución lineal de potencia 40 a 55° C. Disminución de potencia montaje vertical 10° C.

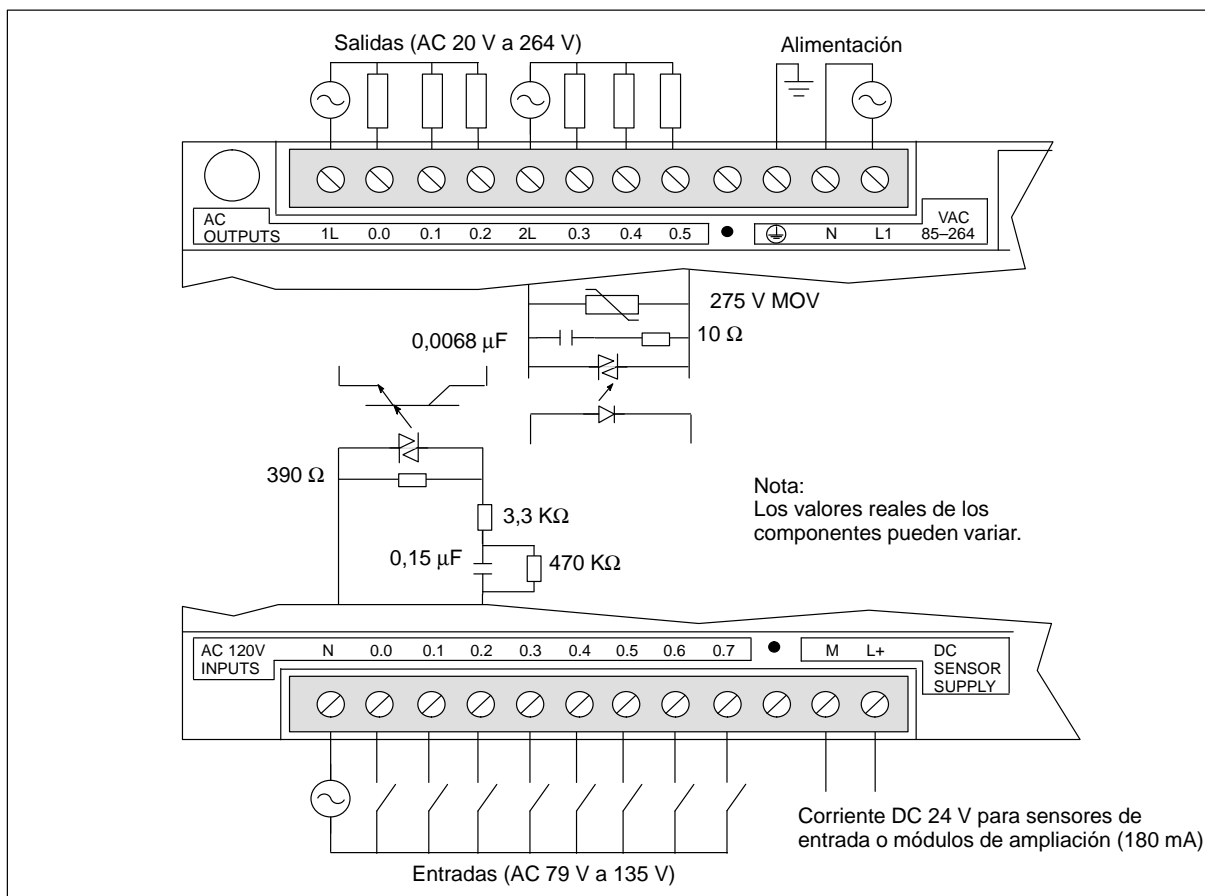


Figura A-5 Identificación de terminales de conexión para la CPU 212 AC/AC/AC

## A.6 CPU 212 alimentación AC, entradas DC tipo fuente, salidas de relé

Nº de referencia: 6ES7 212-1BA10-0XB0

Características generales		Entradas	
Dimensiones (l x a x p)	160 x 80 x 62 mm	Tipo de entrada	fuelle
Peso	0,4 kg	Margen de tensión de entrada	DC 15 V a 30 V, DC 35V, 500 ms
Disipación	6 W	Tensión nominal en ON	mín. 4 mA
Tamaño programa de usuario / memoria	512 palabras/EEPROM	Máximo en OFF	máx. 1 mA
Tamaño datos usuario/memoria	512 palabras/RAM típ. 50 h	Tiempo de respuesta I0.0 a I0.7	máx. 0,3 ms
Retención de datos (8 h mín. a 40° C)		Separación galvánica	AC 500 V, 1 minuto
E/S locales <sup>1</sup>	8 entradas/6 salidas		
Número máximo de módulos de ampliación	2		
E/S digitales asistidas	64 entradas/64 salidas		
E/S analógicas asistidas	16 entradas/16 salidas		
Velocidad de ejecución booleana	1,2 µs/operación		
Marcas internas	128		
Temporizadores	64 temporizadores		
Contadores	64 contadores		
Contadores rápidos	1 software (máx. 2 KHz)		
Potenciómetros analógicos	1		
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según VDE 0160 según CE		
Salidas		Alimentación	
Tipo de salida	Relé, contacto de baja potencia	Margen de tensión/ de frecuencia	AC 85 V a 264 V a 47 a 63 Hz
Margen de tensión	DC 5 V a 30 V / AC 250 V	Corriente de entrada	típ. 4 VA, sólo CPU 50 VA carga máx.
Corriente de carga máxima	2 A/salida, 6 A/hilo neutro	Tiempo de retardo	mín. 20 ms de AC 110 V
Sobretensión transitoria	7A al estar cerrados los contactos	Extra-corriente de cierre	20 A pico a AC 264 V
Resistencia de aislamiento	mín. 100 MΩ (nuevo)	Protección con fusibles (no reemplazable)	2 A, 250 V, de acción lenta
Retardo de conmutación	máx. 10 ms	Corriente disponible DC 5 V	260 mA para CPU 340 mA para módulo de ampliación
Vida útil	10 000 000 mecánico 100 000 con carga normal	Aislamiento	Sí. Transformador, AC 1500 V, 1 minuto
Resistencia de contacto	máx. 200 mΩ (nuevo)		
Aislamiento bobina a contacto	AC 1500 V, 1 minuto		
contacto a contacto (entre contactos abiertos)	AC 750 V, 1 minuto		
Protección contra cortocircuitos	ninguna		
		Alimentación para sensores DC	
		Margen de tensión	DC 20,4 a 28,8 V
		Rizado/corriente parásita (<10 MHz)	máx. 1 V de pico a pico
		Corriente disponible DC 24 V	180 mA
		Limitación de corriente de cortocircuito	< 600 mA
		Aislamiento	no

<sup>1</sup> En la CPU están previstas 8 entradas y 8 salidas en la imagen del proceso para E/S locales.

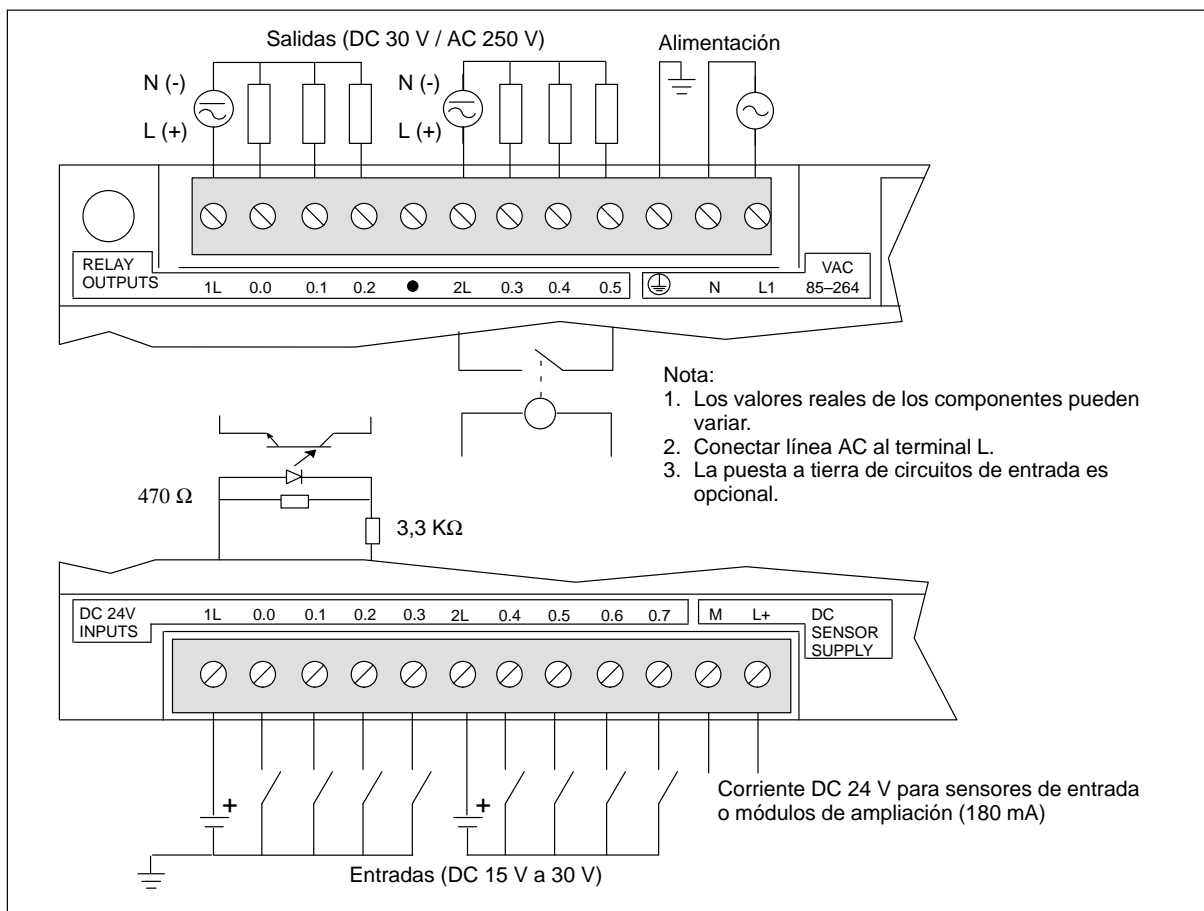


Figura A-6 Identificación de terminales de conexión para la CPU 212 AC/DC tipo fuente/relé

## A.7 CPU 212 alimentación AC, entradas AC 24 V, salidas AC

Nº de referencia: 6ES7 212-1DA01-0XB0

Características generales		Salidas (continuación)	
Dimensiones (l x a x p)	160 x 80 x 62 mm	Retardo de conmutación	1/2 ciclo
Peso	0,4 kg	Sobrecorriente momentánea	30 A pico, 1 ciclo / 10 A pico, 5 ciclos
Disipación	7 W a 2,5 A de carga	Caída de tensión	máx. 1,5 V con corriente máx.
Tamaño programa de usuario / memoria	512 palabras/EEPROM	Separación galvánica	AC 1500 V, 1 minuto
Tamaño datos usuario/memoria	512 palabras/RAM	Protección contra cortocircuitos	ninguna
Retención de datos	típ. 50 h (8 h mín. a 40° C)	<b>Entradas</b>	
E/S locales <sup>1</sup>	8 entradas/6 salidas	Tipo de entrada (IEC 1131-2)	Tipo 1, con sumidero de corriente
Número máximo de módulos de ampliación	2	Área en ON	AC 15 V a 30 V, 47 a 63 Hz, mínimo 4 mA
E/S digitales asistidas	64 entradas/64 salidas	Tensión nominal en ON	AC 24 V, 60 Hz, 7 mA
E/S analógicas asistidas	16 entradas/16 salidas	Máximo en OFF	AC 5 V, 1 mA
Velocidad de ejecución booleana	1,2 µs/operación	Tiempo de respuesta	típ. 10 ms, máx. 15 ms
Marcas internas	128	Separación galvánica	AC 1500 V, 1 minuto
Temporizadores	64 temporizadores	<b>Alimentación</b>	
Contadores	64 contadores	Margen de tensión/ de frecuencia	AC 85 V a 264 V a 47 a 63 Hz
Contadores rápidos	1 software (máx. 50 Hz)	Corriente de entrada	típ. 4 VA, sólo CPU 50 VA carga máx.
Potenciómetros analógicos	1	Tiempo de retardo	mín. 20 ms de AC 110 V
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según CE	Extra-corriente de cierre	20 A pico a AC 264 V
<b>Salidas</b>		Protección con fusibles (no reemplazable)	2 A, 250 V, de acción lenta
Tipo de salida	Triac, pasando por cero	Corriente disponible DC 5 V	320 mA para CPU 280 mA para módulo de ampliación
Margen de tensión/ de frecuencia	AC 20 V a 264 V, 47 a 63 Hz	Aislamiento	Sí. Transformador, AC 1500 V, 1 minuto
Factor de potencia del circuito de carga	0,3 a 1,0	<b>Alimentación para sensores DC</b>	
Carga inductiva, apriete	MOV 275 V tensión de trabajo	Margen de tensión	DC 20,4 a 28,8 V
Corriente de carga máxima por cada salida	0 a 40 °C    55° C <sup>2</sup> 1,20 A    1,00 A	Rizado/corriente parásita (<10 MHz)	máx. 1 V de pico a pico
por 2 salidas adyacentes	1,50 A    1,25 A	Corriente disponible DC 24 V	180 mA
todas las salidas en total	3,50 A    2,50 A	Limitación de corriente de cortocircuito	< 600 mA
Corriente de carga mínima	30 mA	Aislamiento	no
Corriente de derivación	1,5 mA, AC 120 V/2,0 mA, AC 240 V		

<sup>1</sup> En la CPU están previstas 8 entradas y 8 salidas en la imagen del proceso para E/S locales.

<sup>2</sup> Disminución lineal de potencia 40 a 55° C. Disminución de potencia montaje vertical 10° C

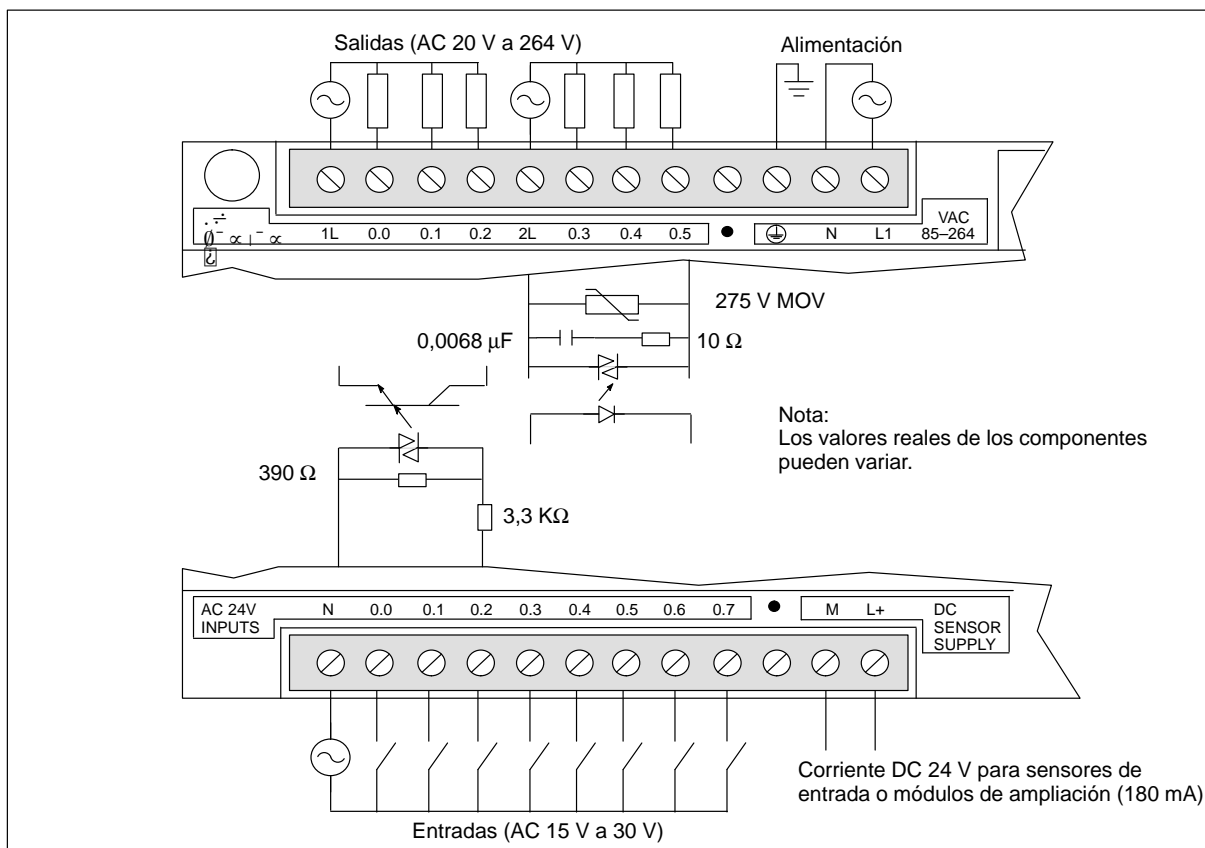


Figura A-7 Identificación de terminales de conexión para la CPU 212 AC/AC/AC

## A.8 CPU 212 alimentación AC, entradas AC, salidas de relé

Nº de referencia: 6ES7 212-1GA01-0XB0

Características generales	
Dimensiones (l x a x p)	160 x 80 x 62 mm
Peso	0,4 kg
Disipación	6 W
Tamaño programa de usuario / memoria	512 palabras/EEPROM
Tamaño datos usuario/memoria	512 palabras/RAM
Retención de datos	típ. 50 h (mín. 8 h a 40° C)
E/S locales <sup>1</sup>	8 entradas/6 salidas
Número máximo de módulos de ampliación	2
E/S digitales asistidas	64 entradas/64 salidas
E/S analógicas asistidas	16 entradas/16 salidas
Tiempo de ejecución por operación	1,2 µs/operación
Marcas internas	128
Temporizadores	64 temporizadores
Contadores	64 contadores
Contadores rápidos	1 software (máx. 2 KHz)
Potenciómetros analógicos	1
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según VDE 0160 según CE
Salidas	
Tipo de salida	Relé, contacto de baja potencia
Margen de tensión	DC 5 V a 30 V / AC 250 V
Corriente de carga máxima	2 A/salida
Sobretensión transitoria	7A al estar cerrados los contactos
Resistencia de aislamiento	mín. 100 MΩ (nuevo)
Retardo de conmutación	máx. 10 ms
Vida útil	10 000 000 mecánico 100 000 con carga normal
Resistencia de contacto	máx. 200 mΩ (nuevo)
Aislamiento	
bobina a contacto	AC 1500 V, 1 minuto
contacto a contacto	AC 1000 V, 1 minuto
Protección contra cortocircuitos	ninguna

Entradas	
Tipo de entrada (IEC 1131-2)	Tipo 1, con sumidero de corriente
Área en ON	AC 79 a 135 V, 47 a 63 Hz. mín. 4 mA
Tensión nominal en ON	AC 120 V, 60 Hz, 7 mA
Máximo en OFF	AC 20 V, 1 mA
Tiempo de respuesta	típ. 10 ms, máx. 15 ms
Separación galvánica	AC 1500 V, 1 minuto
Alimentación	
Margen de tensión/ de frecuencia	AC 85 V a 264 V a 47 a 63 Hz
Corriente de entrada	típ. 4 VA, sólo CPU 50 VA carga máx.
Tiempo de retardo	mín. 20 ms de AC 110 V
Extra-corriente de cierre	20 A pico a AC 264 V
Protección con fusibles (no reemplazable)	2 A, 250 V, de acción lenta
Corriente disponible DC 5 V	260 mA para CPU 340 mA para módulo de ampliación
Aislamiento	Sí. Transformador, AC 1500 V, 1 minuto
Alimentación para sensores DC	
Margen de tensión	DC 20,4 a 28,8 V
Rizado/corriente parásita (<10 MHz)	máx. 1 V de pico a pico
Corriente disponible DC 24 V	180 mA
Limitación de corriente de cortocircuito	< 600 mA
Aislamiento	no

<sup>1</sup> En la CPU están previstas 8 entradas y 8 salidas en la imagen del proceso para E/S locales.

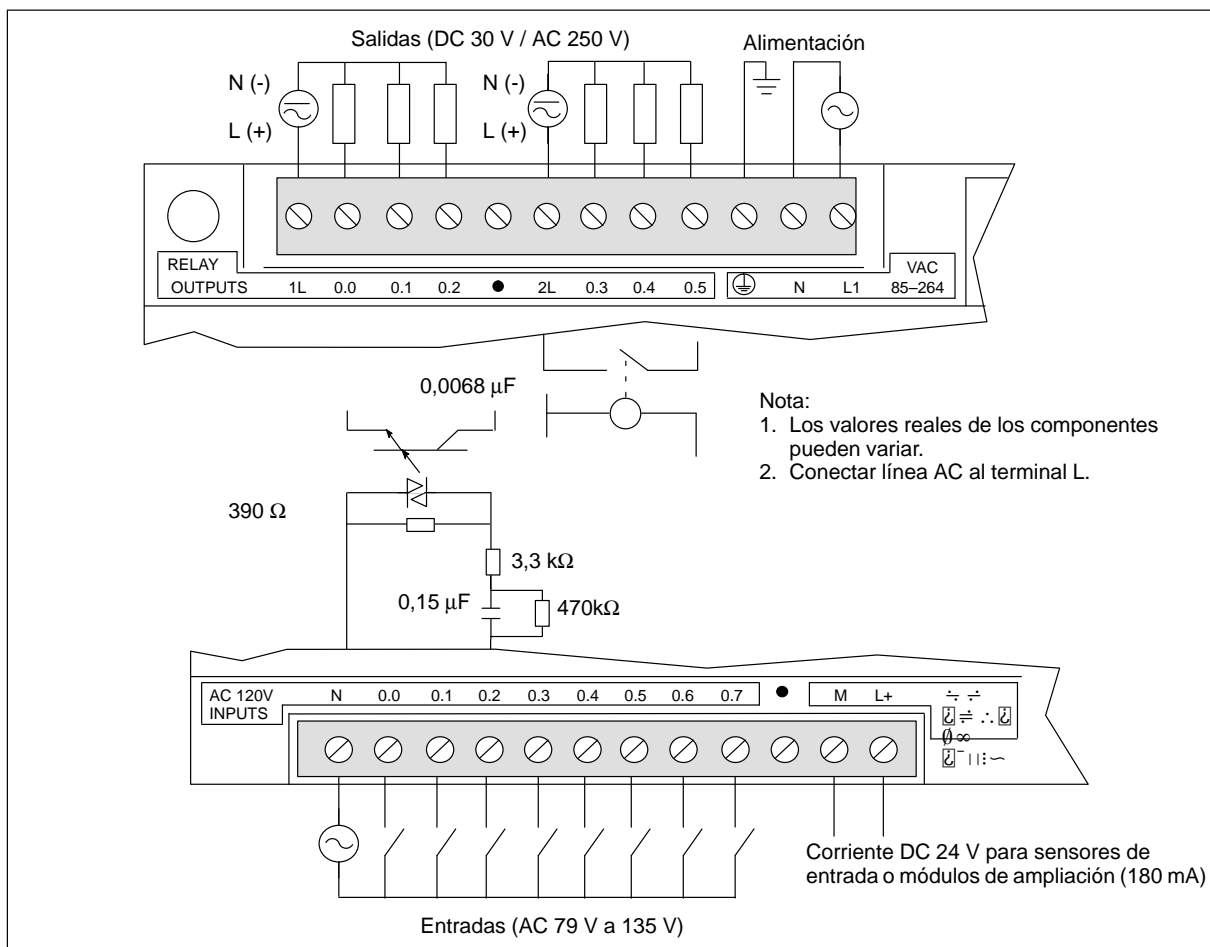


Figura A-8 Identificación de terminales de conexión para la CPU 212 AC/AC/relé

## A.9 CPU 214 alimentación DC, entradas DC, salidas DC

Nº de referencia: 6ES7 214-1AC01-0XB0

Características generales		Separación galvánica	AC 500 V, 1 minuto
Dimensiones (l x a x p)	197 x 80 x 62 mm	<b>Salidas</b>	
Peso	0,4 kg	Tipo de salida	Transistor, tipo fuente
Disipación	8 W a 3 A de carga	Margen de tensión	DC 20,4 a 28,8 V
Tamaño programa de usuario/ memoria	2 Kpalabras/EEPROM	Corriente de carga máxima por cada salida	0 a 40 °C 55° C <sup>2</sup> 0,75 A 0,50 A
Tamaño datos usuario/ memoria	2 Kpalabras/RAM	por 2 salidas adyacentes	1,00 A 0,75 A
Ret. de datos y del tiempo real		todas las salidas en total	4,00 A 3,00 A
Condensador de alta potencia	típ. 190 h (mín. 120 h a 40° C)	Carga inductiva, apriete	(por hilo neutro)
Pila opcional	200 días de uso continuo	Impulso individual	2A L/R = 10 ms
E/S locales <sup>1</sup>	14 entradas/10 salidas	Repetición	1A L/R = 100 ms
Número máximo de módulos de ampliación	7		1 W disipación de energía (1/2 Li <sup>2</sup> x frecuencia de conmutación < 1W)
E/S digitales asistidas	64 entradas/64 salidas	Corriente de derivación	100 µA
E/S analógicas asistidas	16 entradas/16 salidas	Retardo de conmutación	25 µs ON, 120 µs OFF
Velocidad de ejecución booleana	0,8 µs/operación	Sobrecorriente momentánea	4 A, 100 ms
Marcas internas	256	Caída de tensión	1,8 V máx. con corriente máxima
Temporizadores	128 temporizadores	Separación galvánica	AC 500 V, 1 minuto
Contadores	128 contadores	Protección contra cortocircuitos	ninguna
Contadores rápidos	1 software (máx. 2 KHz) 2 hardware (máx. 7 KHz c/u)	<b>Alimentación</b>	
Tolerancia del reloj de tiempo real (TOD)	6 minutos/mes	Margen de tensión	DC 20,4 a 28,8 V
Salidas de impulsos	2 (máx. 4 kHz c/u)	Corriente de entrada	típ. 85 mA, sólo CPU 900 mA carga máx.
Potenciómetros analógicos	2	Medición UL/CSA	50 VA
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según VDE 0160 según CE	Tiempo de retardo	mín. 10 ms de DC 24 V
<b>Entradas</b>		Extra-corriente de cierre	10 A pico a DC 28,8 V
Tipo de entrada (IEC 1131-2)	Tipo 1, con sumidero de corriente	Protección con fusibles (no reemplazable)	1 A, 125 V, de acción lenta
Área en ON	DC 15 V a 30 V, mín. 4 mA DC 35 V, 500 ms sobretensión transitoria	Corriente disponible DC 5 V	340 mA para CPU; 660 mA para módulo de ampliación
Tensión nominal en ON	DC 24 V, 7 mA	Aislamiento	no
Máximo en OFF	DC 5 V, 1 mA	<b>Alimentación para sensores DC</b>	
Tiempo de respuesta máxima I0.0 a I1.5	0,2 ms a 8,7 ms seleccionable 0,2 ms predeterminado	Margen de tensión	DC 16,4 V a 28,8 V
I0.6 a I1.5 como se usa en HSC1 y HSC2	típ. 30 µs/máx. 70 µs	Rizado/corriente parásita (<10 MHz)	igual a la tensión alimentada
		Corriente disponible DC 24 V	280mA
		Limitación de corriente de cortocircuito	< 600 mA
		Aislamiento	no

<sup>1</sup> En la CPU están previstas 16 entradas y 16 salidas en la imagen del proceso para E/S locales.

<sup>2</sup> Disminución lineal de potencia 40 a 55° C. Disminución de potencia montaje vertical 10° C

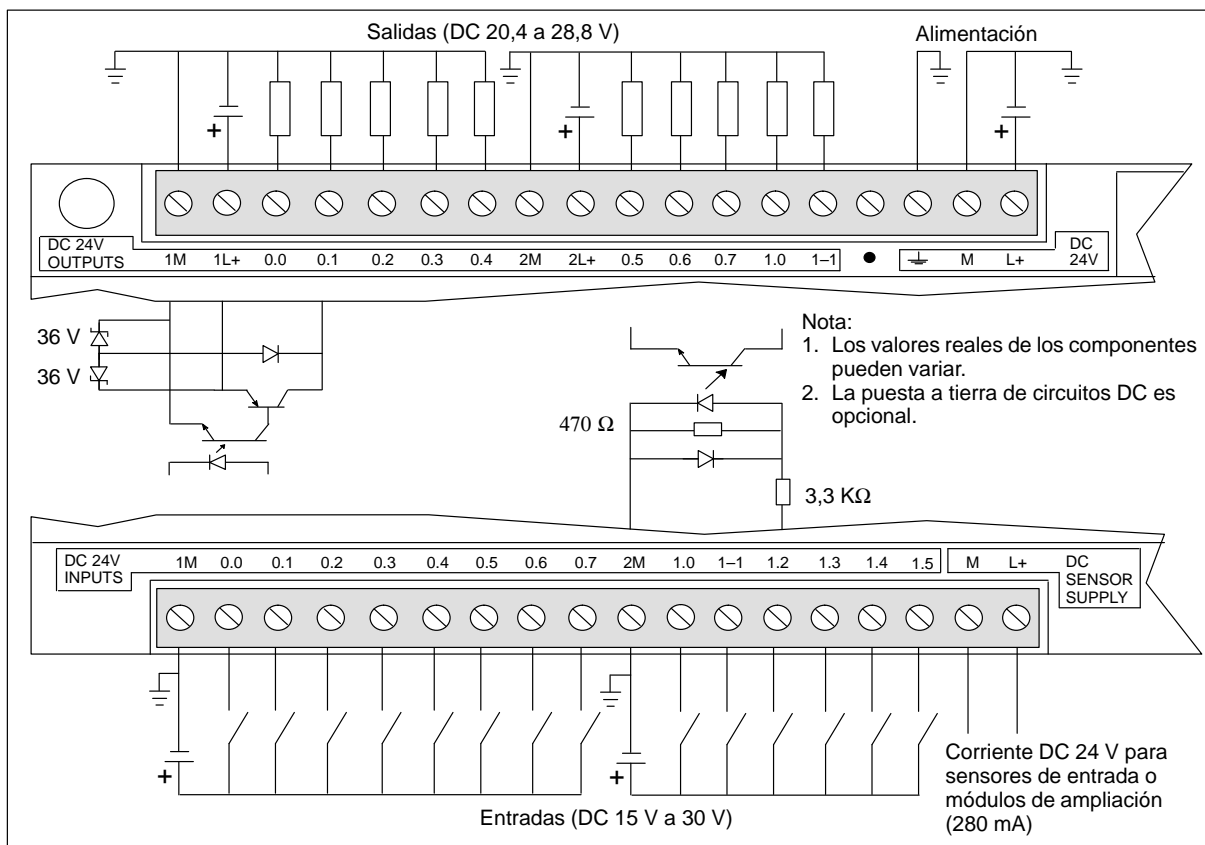


Figura A-9 Identificación de terminales de conexión para la CPU 214 DC/DC/DC

## A.10 CPU 214 alimentación AC, entradas DC, salidas de relé

Nº de referencia: 6ES7 214-1BC01-0XB0

Características generales		Salidas	
Dimensiones (l x a x p)	197 x 80 x 62 mm	Tipo de salida	Relé, contacto de baja potencia
Peso	0,5 kg	Margen de tensión	DC 5 V a 30 V / AC 250 V
Disipación	9 W	Corriente de carga máxima	2 A/salida, 8 A/hilo neutro
Tamaño programa de usuario/ memoria	2 Kpalabras/EEPROM	Sobretensión transitoria	7A al estar cerrados los contactos
Tamaño datos usuario/ memoria	2 Kpalabras/RAM	Resistencia de aislamiento	mín. 100 MΩ (nuevo)
Ret. de datos y del tiempo real		Retardo de conmutación	máx. 10 ms
Condensador de alta potencia	típ. 190 h (mín. 120 h a 40° C)	Vida útil	10 000 000 mecánico 100 000 con carga normal
Pila opcional	200 días de uso continuo	Resistencia de contacto	máx. 200 mΩ (nuevo)
E/S locales <sup>1</sup>	14 entradas/10 salidas	Aislamiento	
Número máximo de módulos de ampliación	7	bobina a contacto	AC 1500 V, 1 minuto
E/S digitales asistidas	64 entradas/64 salidas	contacto a contacto	AC 750 V, 1 minuto
E/S analógicas asistidas	16 entradas/16 salidas	(entre contactos abiertos)	
Velocidad de ejecución booleana	0,8 μs/operación	Protección contra cortocircuitos	ninguno
Marcas internas	256	Alimentación	
Temporizadores	128 temporizadores	Margen de tensión/ de frecuencia	AC 85 V a 264 V a 47 a 63 Hz
Contadores	128 contadores	Corriente de entrada	típ. 4,5 VA, sólo CPU 50 VA carga máx.
Contadores rápidos	1 software (máx. 2 KHz) 2 hardware (máx. 7 KHz c/u)	Tiempo de retardo	mín. 20 ms de AC 110 V,
Tolerancia del reloj de tiempo real (TOD)	6 minutos/mes	Extra-corriente de cierre	20 A pico a AC 264 V
Salidas de impulsos	no recomendadas	Protección con fusibles	2 A, 250 V, de acción lenta (no reemplazable)
Potenciómetros analógicos	2	Corriente disponible DC 5 V	340 mA para CPU; 660 mA para módulo de ampliación
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según VDE 0160 según CE	Aislamiento	Sí. Transformador, AC 1500 V, 1 minuto
Entradas		Alimentación para sensores DC	
Tipo de entrada (IEC 1131-2)	Tipo 1, con sumidero de corriente	Margen de tensión	DC 20,4 a 28,8 V
Área en ON	DC 15 V a 30 V, mín. 4 mA DC 35 V, 500 ms sobretensión transitoria	Rizado/corriente parásita (<10 MHz)	máx. 1 V de pico a pico
Tensión nominal en ON	DC 24 V, 7 mA	Corriente disponible DC 24 V	280mA
Máximo en OFF	DC 5 V, 1 mA	Limitación de corriente de cortocircuito	< 600 mA
Tiempo de respuesta máxima I0.0 a I1.5	0,2 ms a 8,7 ms seleccionable 0,2 ms predeterminado	Aislamiento	no
I0.6 a I1.5 como se usa en HSC1 y HSC2	típ. 30 μs/máx. 70 μs		
Separación galvánica	AC 500 V, 1 minuto		

<sup>1</sup> En la CPU están previstas 16 entradas y 16 salidas en la imagen del proceso para E/S locales.

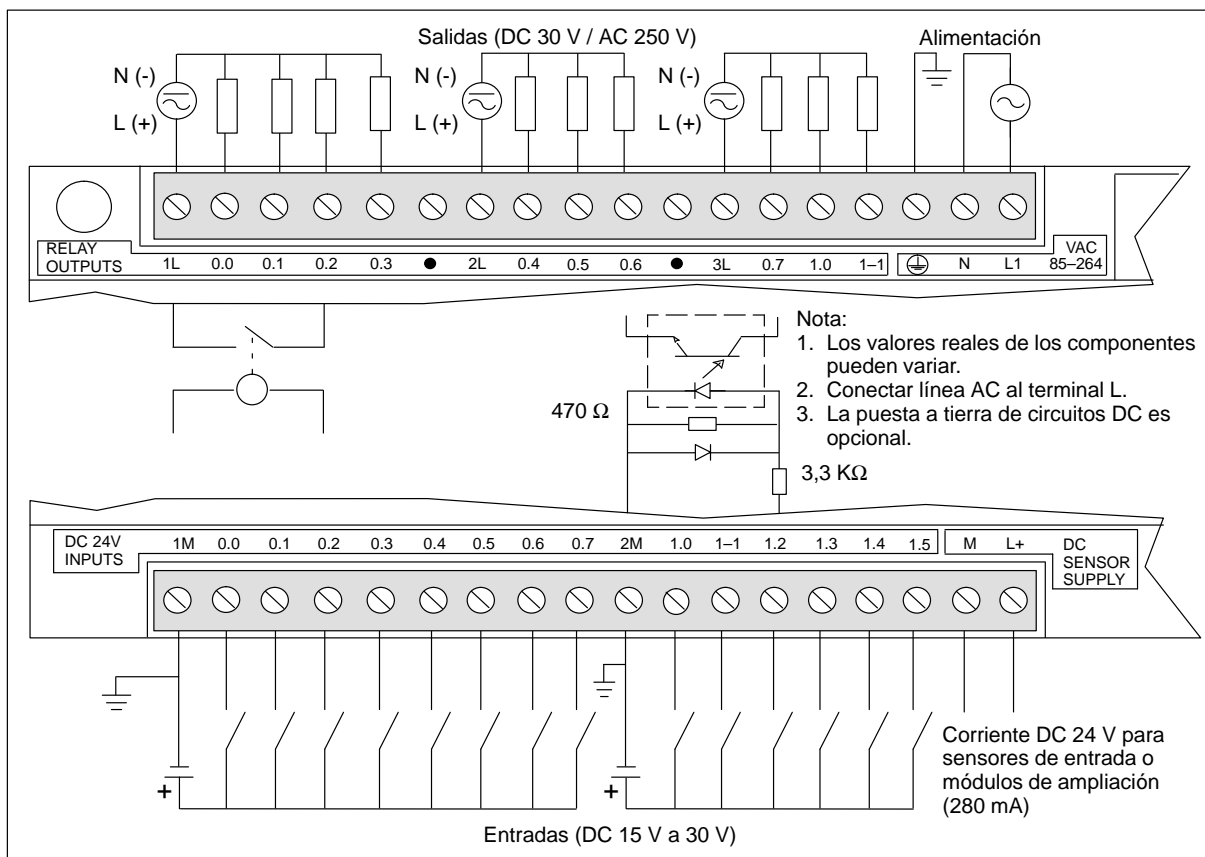


Figura A-10 Identificación de terminales de conexión para la CPU 214 AC/DC/relé

## A.11 CPU 214 alimentación AC, entradas AC, salidas AC

Nº de referencia: 6ES7 214-1CC01-0XB0

Características generales		Salidas	
Dimensiones (l x a x p)	197 x 80 x 62 mm	Tipo de salida	Triac, pasando por cero
Peso	0,5 kg	Margen de tensión/ de frecuencia	AC 20 V a 264 V, 47 a 63 Hz
Disipación	11 W a 4,25 A de carga	Factor de potencia del circuito de carga	0,3 a 1,0
Tamaño programa de usuario/ memoria	2 Kpalabras/EEPROM	Carga inductiva, apriete	MOV 275 V tensión de trabajo
Tamaño datos usuario/memoria	2 Kpalabras/RAM	Corriente de carga máxima por cada salida	0 a 40 °C 55° C <sup>2</sup> 1,20 A 1,00 A
Ret. de datos y del tiempo real	típ. 190 h (mín. 120 h a 40° C)	por 2 salidas adyacentes	1,50 A 1,25 A
Condensador de alta potencia	200 días de uso continuo	todas las salidas en total	6,00 A 4,25 A
E/S locales <sup>1</sup>	14 entradas/10 salidas	Corriente de carga mínima	30 mA
Número máximo de módulos de ampliación	7	Corriente de derivación	1,5 mA, AC 120 V/2,0 mA, AC 240 V
E/S digitales asistidas	64 entradas/64 salidas	Retardo de conmutación	1/2 ciclo
E/S analógicas asistidas	16 entradas/16 salidas	Sobrecorriente momentánea	30 A pico, 1 ciclo / 10 A pico, 5 ciclos
Velocidad de ejecución booleana	0,8 µs/operación	Caída de tensión	máx. 1,5 V con corriente máx.
Marcas internas	256	Separación galvánica	AC 1500 V, 1 minuto
Temporizadores	128 temporizadores	Protección contra cortocircuitos	ninguna
Contadores	128 contadores	<b>Alimentación</b>	
Contadores rápidos	1 software (50 Hz) 2 hardware (50 Hz c/u)	Margen de tensión/ de frecuencia	AC 85 V a 264 V a 47 a 63 Hz
Tolerancia del reloj de tiempo real (TOD)	6 minutos/mes	Corriente de entrada	típ. 4,5 VA, sólo CPU 50 VA carga máxima
Salidas de impulsos	2 (máx. 100 Hz c/u)	Tiempo de retardo	mín. 20 ms de AC 110 V,
Potenciómetros analógicos	2	Extra-corriente de cierre	20 A pico a AC 264 V
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según CE	Protección con fusibles (no reemplazable)	2 A, 250 V, de acción lenta
<b>Entradas</b>		Corriente disponible DC 5 V	440 mA para CPU; 560 mA para módulo de ampliación
Tipo de entrada (IEC 1131-2)	Tipo 1, con sumidero de corriente	Aislamiento	Sí. Transformador, AC 1500 V, 1 minuto
Área en ON	AC 79 V a 135 V, 47 a 63 Hz, mín. 4 mA	<b>Alimentación para sensores DC</b>	
Tensión nominal en ON	AC 120 V, 60 Hz, 7 mA	Margen de tensión	DC 20,4 a 28,8 V
Máximo en OFF	AC 20 V, 1 mA	Rizado/corriente parásita (<10 MHz)	máx. <1 V de pico a pico
Tiempo de respuesta máxima	0,2 ms a 8,7 ms seleccionable, más 15,0 ms con filtro fijo 15,2 ms predeterminado	Corriente disponible DC 24 V	280mA
Separación galvánica	AC 1500 V, 1 minuto	Limitación de corriente de cortocircuito	< 600 mA
		Aislamiento	no

<sup>1</sup> En la CPU están previstas 16 entradas y 16 salidas en la imagen del proceso para E/S locales.

<sup>2</sup> Disminución lineal de potencia 40 a 55° C. Disminución de potencia montaje vertical 10° C

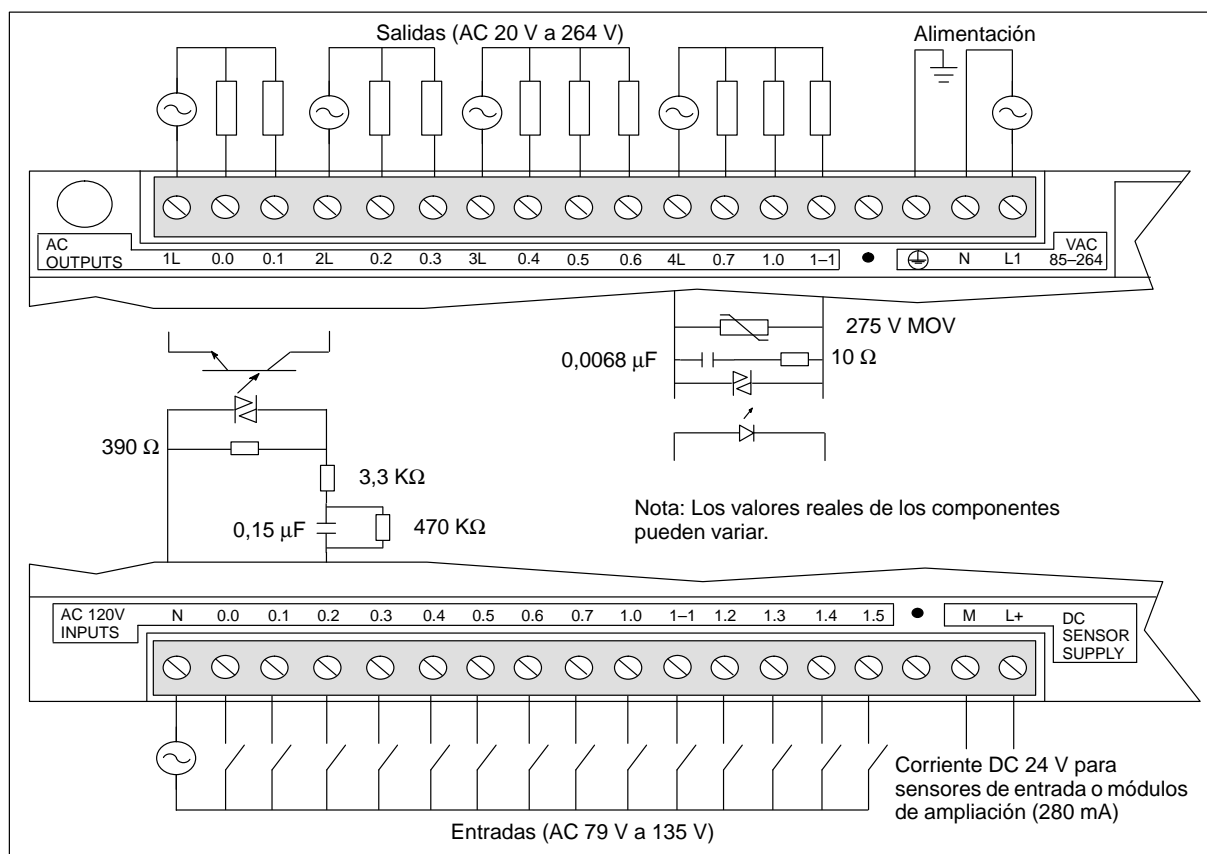


Figura A-11 Identificación de terminales de conexión para la CPU 214 AC/AC/AC

## A.12 CPU 214 alimentación AC, entradas tipo fuente DC, salidas de relé

Nº de referencia: 6ES7 214-1BC10-0XB0

Características generales		Salidas	
Dimensiones (l x a x p)	197 x 80 x 62 mm	Tipo de salida	Relé, contacto de baja potencia
Peso	0,5 kg	Margen de tensión	DC 5 V a 30 V / AC 250 V
Disipación	9 W	Corriente de carga máxima	2 A/salida, 8 A/hilo neutro
Tamaño programa de usuario/ memoria	2 Kpalabras/EEPROM	Sobretensión transitoria	7A al estar cerrados los contactos
Tamaño datos usuario/memoria	2 Kpalabras/RAM	Resistencia de aislamiento	mín. 100 MΩ (nuevo)
Ret. de datos y del tiempo real		Retardo de conmutación	máx. 10 ms
Condensador de alta potencia	típ. 190 h (mín. 120 h a 40° C)	Vida útil	10 000 000 mecánico 100 000 con carga normal
Pila opcional	200 días de uso continuo	Resistencia de contacto	máx. 200 mΩ (nuevo)
E/S locales <sup>1</sup>	14 entradas/10 salidas	Aislamiento	
Número máximo de módulos de ampliación	7	bobina a contacto	AC 1500 V, 1 minuto
E/S digitales asistidas	64 entradas/64 salidas	contacto a contacto	AC 750 V, 1 minuto
E/S analógicas asistidas	16 entradas/16 salidas	(entre contactos abiertos)	
Velocidad de ejecución booleana	0,8 µs/operación	Protección contra cortocircuitos	ninguna
Marcas internas	256	Alimentación	
Temporizadores	128 temporizadores	Margen de tensión/ de frecuencia	AC 85 V a 264 V a 47 a 63 Hz
Contadores	128 contadores	Corriente de entrada	típ. 4,5 VA, sólo CPU 50 VA carga máx.
Contadores rápidos	1 software (máx. 2 KHz) 2 hardware (máx. 7 KHz c/u)	Tiempo de retardo	mín. 20 ms de AC 110 V,
Tolerancia del reloj de tiempo real (TOD)	6 minutos/mes	Extra-corriente de cierre	20 A pico a AC 264 V
Salidas de impulsos	no recomendadas	Protección con fusibles (no reemplazable)	2 A, 250 V, de acción lenta
Potenciómetros analógicos	2	Corriente disponible DC 5 V	340 mA para CPU; 660 mA para módulo de ampliación
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según VDE 0160 según CE	Aislamiento	Sí. Transformador, AC 1500 V, 1 minuto
Entradas		Alimentación para sensores DC	
Tipo de entrada	fuelle	Margen de tensión	DC 20,4 a 28,8 V
Margen de tensión de entrada	DC 15 V a 30 V, DC 35V, 500 ms	Rizado/corriente parásita (<10 MHz)	máx. 1 V de pico a pico
Tensión nominal en ON	mín. 4 mA	Corriente disponible DC 24 V	280mA
Máximo en OFF	máx. 1 mA	Limitación de corriente de cortocircuito	< 600 mA
Tiempo de respuesta máxima I0.0 a I1.5	0,2 ms a 8,7 ms seleccionable 0,2 ms predeterminado	Aislamiento	no
I0.6 a I1.5 como se usa en HSC1 y HSC2	típ. 30 µs/máx. 70 µs		
Separación galvánica	AC 500 V, 1 minuto		

<sup>1</sup> En la CPU están previstas 16 entradas y 16 salidas en la imagen del proceso para E/S locales.

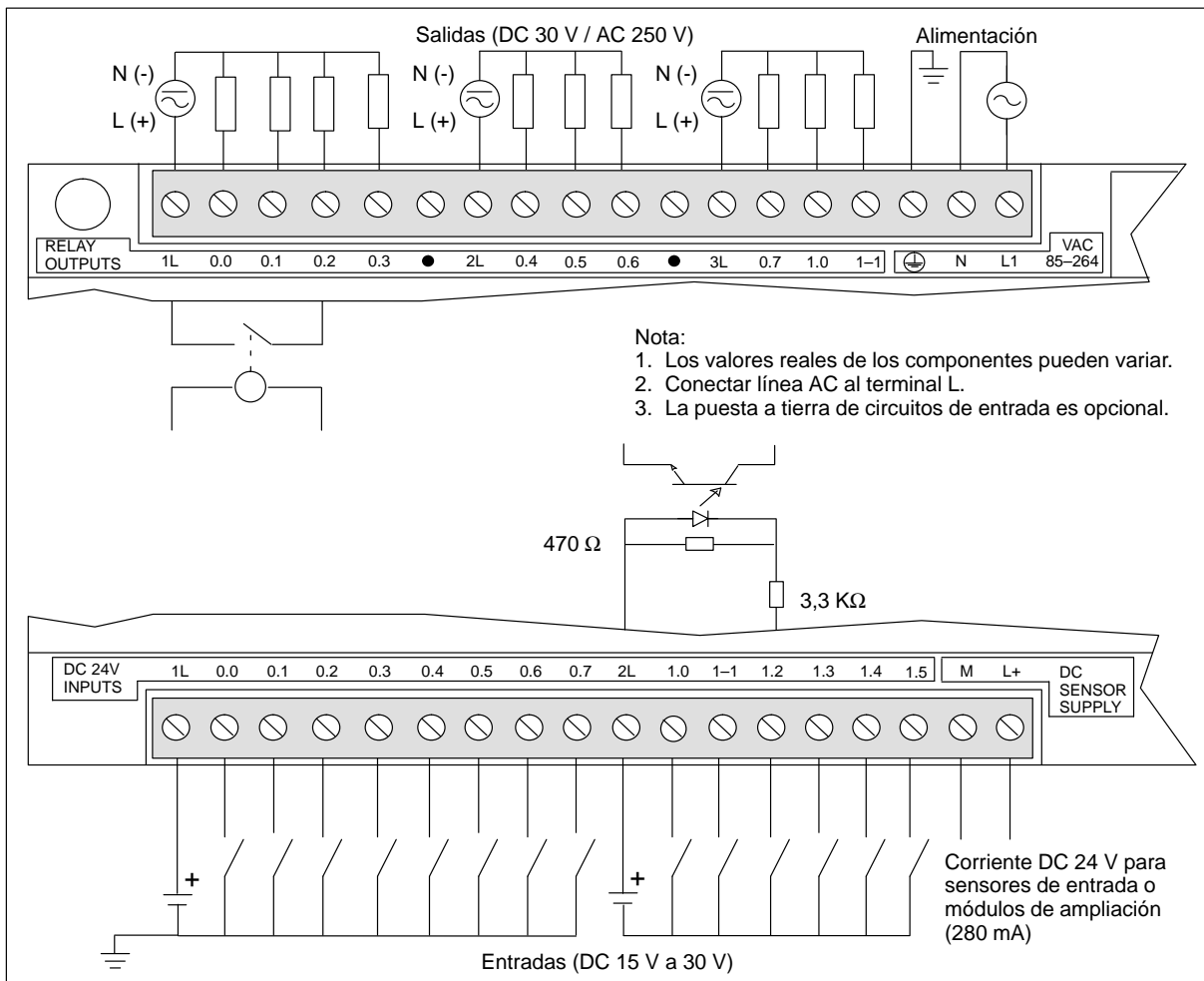


Figura A-12 Identificación de terminales de conexión para la CPU 214 AC/DC tipo fuente/relé

## A.13 CPU 214 alimentación AC, entradas AC 24 V, salidas AC

Nº de referencia: 6ES7 214-1DC01-0XB0

Características generales		Salidas	
Dimensiones (l x a x p)	197 x 80 x 62 mm	Tipo de salida	Triac, pasando por cero
Peso	0,5 kg	Margen de tensión/ de frecuencia	AC 20 V a 264 V, 47 a 63 Hz
Disipación	11 W a 4,25 A de carga	Factor de potencia del circuito de carga	0,3 a 1,0
Tamaño programa de usuario/ memoria	2 Kpalabras/EEPROM	Carga inductiva, apriete	MOV 275 V tensión de trabajo
Tamaño datos usuario/memoria	2 Kpalabras/RAM	Corriente de carga máxima por cada salida	0 a 40 °C 55° C <sup>2</sup> 1,20 A 1,00 A
Ret. de datos y del tiempo real	típ. 190 h (mín. 120 h a 40° C)	por 2 salidas adyacentes	1,50 A 1,25 A
Condensador de alta potencia	200 días de uso continuo	todas las salidas en total	6,00 A 4,25 A
E/S locales <sup>1</sup>	14 entradas/10 salidas	Corriente de carga mínima	30 mA
Número máximo de módulos de ampliación	7	Corriente de derivación	1,5 mA, AC 120 V/2,0 mA, AC 240 V
E/S digitales asistidas	64 entradas/64 salidas	Retardo de conmutación	1/2 ciclo
E/S analógicas asistidas	16 entradas/16 salidas	Sobrecorriente momentánea	30 A pico, 1 ciclo / 10 A pico, 5 ciclos
Velocidad de ejecución booleana	0,8 µs/operación	Caída de tensión	máx. 1,5 V con corriente máx.
Marcas internas	256	Separación galvánica	AC 1500 V, 1 minuto
Temporizadores	128 temporizadores	Protección contra cortocircuitos	ninguna
Contadores	128 contadores	<b>Alimentación</b>	
Contadores rápidos	1 software (50 Hz) 2 hardware (50 Hz c/u)	Margen de tensión/ de frecuencia	AC 85 V a 264 V a 47 a 63 Hz
Tolerancia del reloj de tiempo real (TOD)	6 minutos/mes	Corriente de entrada	típ. 4,5 VA, sólo CPU 50 VA carga máxima
Salidas de impulsos	2 (máx. 100 Hz c/u)	Tiempo de retardo	mín. 20 ms de AC 110 V,
Potenciómetros analógicos	2	Extra-corriente de cierre	20 A pico a AC 264 V
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según CE	Protección con fusibles (no reemplazable)	2 A, 250 V, de acción lenta
<b>Entradas</b>		Corriente disponible DC 5 V	440 mA para CPU; 560 mA para módulo de ampliación
Tipo de entrada (IEC 1131-2)	Tipo 1, con sumidero de corriente	Aislamiento	Sí. Transformador, AC 1500 V, 1 minuto
Área en ON	AC 15 V a 30 V, 47 a 63 Hz, mínimo 4 mA	<b>Alimentación para sensores DC</b>	
Tensión nominal en ON	AC 24 V, 60 Hz, 7 mA	Margen de tensión	DC 20,4 a 28,8 V
Máximo en OFF	AC 5 V, 1 mA	Rizado/corriente parásita (<10 MHz)	máx. <1 V de pico a pico
Tiempo de respuesta máxima	0,2 ms a 8,7 ms seleccionable, más 15,0 ms con filtro fijo 15,2 ms predeterminado	Corriente disponible DC 24 V	280mA
Separación galvánica	AC 1500 V, 1 minuto	Limitación de corriente de cortocircuito	< 600 mA
		Aislamiento	no

<sup>1</sup> En la CPU están previstas 16 entradas y 16 salidas en la imagen del proceso para E/S locales.

<sup>2</sup> Disminución lineal de potencia 40 a 55° C. Disminución de potencia montaje vertical 10° C

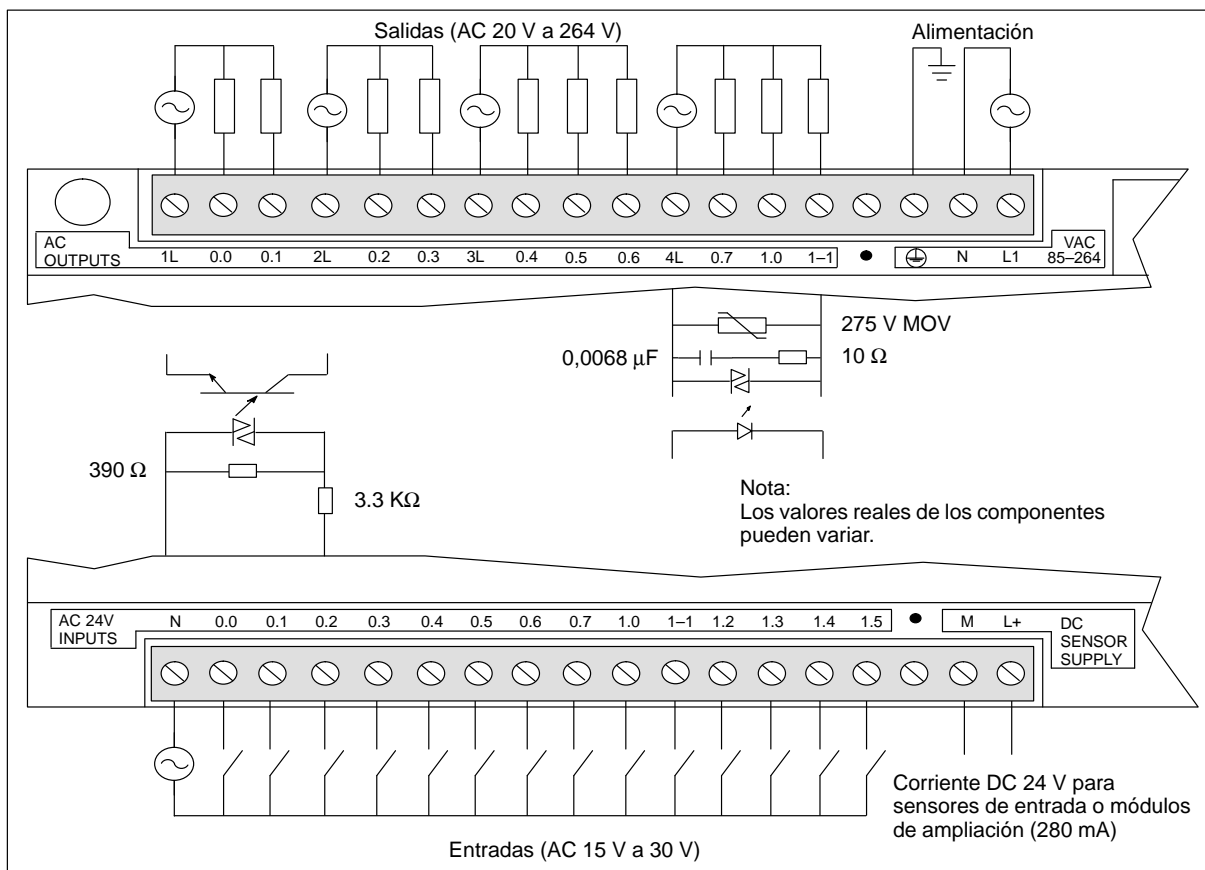


Figura A-13 Identificación de terminales de conexión para la CPU 214 AC/AC/AC

## A.14 CPU 214 alimentación AC, entradas AC, salidas de relé

Nº de referencia: 6ES7 214-1GC01-0XB0

Características generales		Salidas	
Dimensiones (l x a x p)	197 x 80 x 62 mm	Tipo de salida	Relé, contacto de baja potencia
Peso	0,5 kg	Margen de tensión	DC 5 V a 30 V / AC 250 V
Disipación	9 W	Corriente de carga máxima	2 A/salida
Tamaño programa de usuario/ memoria	2K palabras/EEPROM	Sobretensión transitoria	7A al estar cerrados los contactos
Tamaño datos usuario/memoria	2K palabras/RAM	Resistencia de aislamiento	mín. 100 MΩ (nuevo)
Ret. de datos y del tiempo real		Retardo de conmutación	máx. 10 ms
Condensador de alta potencia	típ. 190 h (mín. 120 h a 40° C)	Vida útil	10 000 000 mecánico 100 000 con carga normal
Pila opcional	200 días de uso continuo	Resistencia de contacto	máx. 200 mΩ (nuevo)
E/S locales <sup>1</sup>	14 entradas/10 salidas	Aislamiento	
Número máximo de módulos de ampliación	7	bobina a contacto	AC 1500 V, 1 minuto
E/S digitales asistidas	64 entradas/64 salidas	contacto a contacto	AC 1000 V, 1 minuto
E/S analógicas asistidas	16 entradas/16 salidas	Protección contra cortocircuitos	ninguna
Tiempo de ejecución por operación	0,8 μs/operación	Alimentación	
Marcas internas	256	Margen de tensión/ de frecuencia	AC 85 V a 264 V a 47 a 63 Hz
Temporizadores	128 temporizadores	Corriente de entrada	típ. 4,5 VA, sólo CPU 50 VA carga máx.
Contadores	128 contadores	Tiempo de retardo	mín. 20 ms de AC 110 V,
Contadores rápidos	1 software (máx. 2 kHz) 2 hardware (máx. 7 kHz, c/u)	Extra-corriente de cierre	20 A pico a AC 264 V
Tolerancia del reloj de tiempo real (TOD)	6 minutos/mes	Protección con fusibles (no reemplazable)	2 A, 250 V, de acción lenta
Salidas de impulsos	no recomendadas	Corriente disponible DC 5 V	340 mA para CPU; 660 mA para módulo de ampliación
Potenciómetros analógicos	2	Aislamiento	Sí. Transformador, AC 1500 V, 1 minuto
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según VDE 0160 según CE	Alimentación para sensores DC	
Entradas		Margen de tensión	DC 20,4 a 28,8 V
Tipo de entrada (IEC 1131-2)	Tipo 1, con sumidero de corriente	Rizado/corriente parásita (<10 MHz)	máx. 1 V de pico a pico
Área en ON	AC 79 V a 135 V, 47 a 63 Hz mín. 4 mA	Corriente disponible DC 24 V	280mA
Tensión nominal en ON	AC120 V, 60 Hz. 7 mA	Limitación de corriente de cortocircuito	< 600 mA
Máximo en OFF	AC 20 V, 1 mA	Aislamiento	no
Tiempo de respuesta máxima	0,2 ms a 8,7 ms seleccionable, más 15,0 ms con filtro fijo 15,2 ms predeterminado		
Separación galvánica	AC 1500 V, 1 minuto		

<sup>1</sup> En la CPU están previstas 16 entradas y 16 salidas en la imagen del proceso para E/S locales.

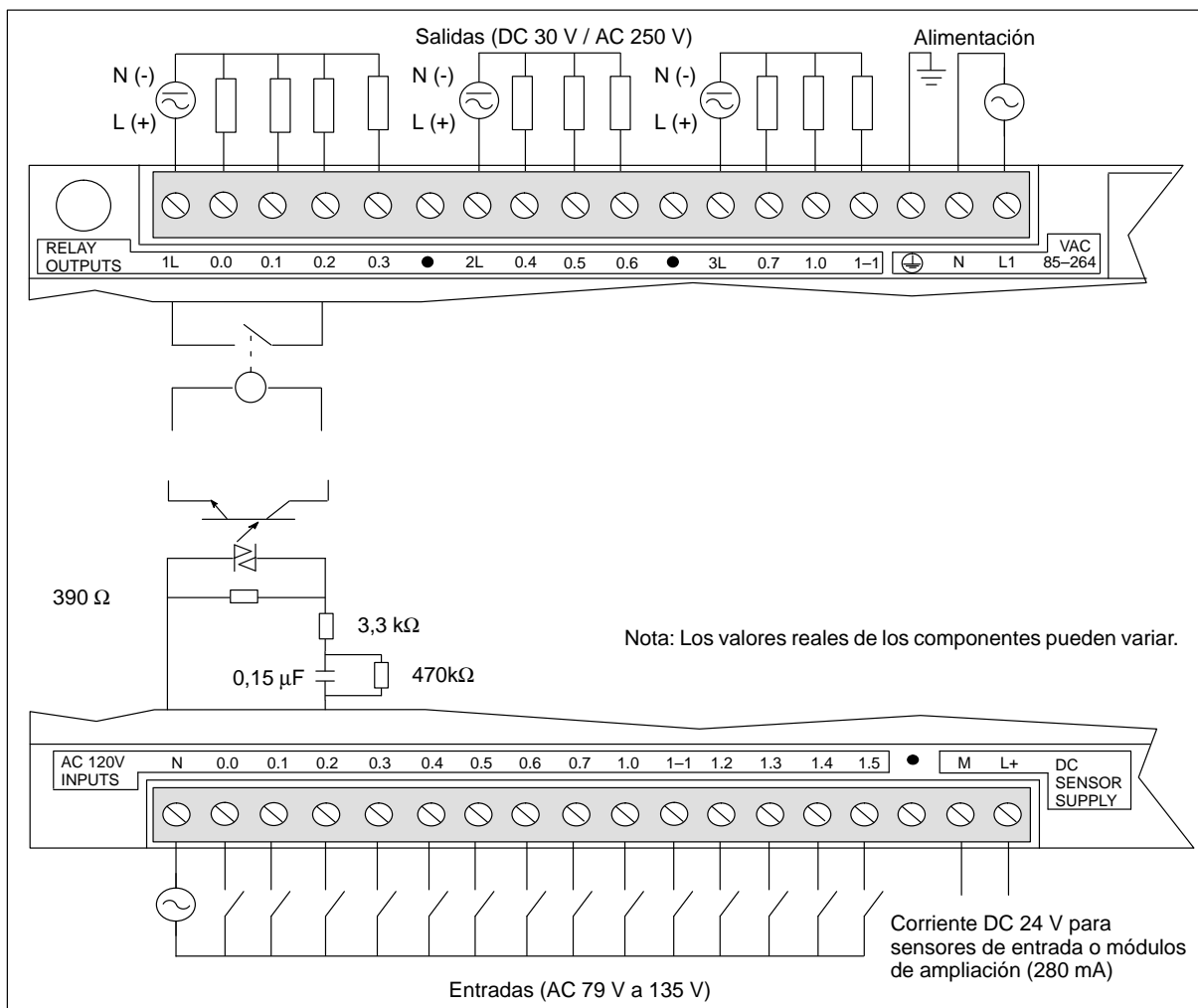


Figura A-14 Identificación de terminales de conexión para la CPU 214 AC/AC/relé

## A.15 CPU 215 alimentación DC, entradas DC, salidas DC

Nº de referencia: 6ES7 215-2AD00-0XB0

Características generales		Salidas	
Dimensiones (l x a x p)	217,3 x 80 x 62 mm	Tipo de salida	MOSFET, fuente
Peso	0,5 kg	Margen de tensión	DC 20,4 a 28,8 V
Disipación	8 W	Corriente de carga máxima	0 a 55 ° C
Tamaño programa de usuario/ memoria	4 Kpalabras/EEPROM	Q0.0 a Q0.7	0,5 A/salida
Tamaño datos usuario/ memoria	2,5 Kpalabras/RAM	Q1.0, Q1.1	1,0 A/salida
Ret. de datos y del tiempo real		Las salidas se pueden conectar en paralelo para corrientes superiores.	
Condensador de alta potencia	típ. 190 h (mín. 120 h a 40° C)	Corriente de derivación	
Pila opcional	200 días de uso continuo	Q0.0 a Q0.7	200 µA
E/S locales <sup>1</sup>	14 entradas/10 salidas	Q1.0, Q1.1	400 µA
Número máximo de módulos de ampliación	7	Retardo de conmutación	
E/S digitales asistidas	64 entradas/64 salidas	Q0.0, 0.1	100 µs, ON/OFF
E/S analógicas asistidas	16 entradas/16 salidas	Demás salidas	150 µs ON, 400 µs OFF
Velocidad de ejecución booleana	0,8 µs/operación	Resistencia ON	máx. 400 mΩ
Marcas internas	256	Protección contra cortocircuitos	
Temporizadores	256 temporizadores	Q0.0 a Q0.7	0,7 a 1,5 A/canal
Contadores	256 contadores	Q1.0, Q1.1	1,5 a 3 A/canal
Contadores rápidos	1 software (máx. 2 KHz) 2 hardware (máx. 20 KHz c/u)	Separación galvánica	AC 500 V, 1 minuto
Tolerancia del reloj de tiempo real (TOD)	6 minutos/mes	<b>Alimentación</b>	
Salidas de impulsos	2 (máx. 4 kHz c/u)	Margen de tensión	DC 20,4 a 28,8 V
Potenciómetros analógicos	2	Corriente de entrada	típ. 120 mA, sólo CPU 1,4 A carga máx.
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según VDE 0160 según CE	Medición UL/CSA	50 VA
<b>Entradas</b>		Tiempo de retardo	mín. 10 ms de DC 24 V
Tipo de entrada	fuelle Tipo 1 IEC con sumidero de corriente	Extra-corriente de cierre	10 A pico a DC 28,8 V
Área en ON	DC 15 V a 30 V, mín. 4 mA DC 35 V, 500 ms sobretensión transitoria	Protección con fusibles (no reemplazable)	2 A, de acción lenta
Tensión nominal en ON	DC 24 V, 7 mA	Corriente disponible DC 5 V	1000 mA para módulo de ampliación
Máximo en OFF	DC 5 V, 1 mA	Aislamiento	no
Tiempo de respuesta máxima I0.0 a I1.5	0,2 ms a 8,7 ms seleccionable 0,2 ms predeterminado	<b>Alimentación para sensores DC</b>	
I0.6 a I1.5 como se usa en HSC1 y HSC2	6 µs ON, 30 µs OFF	Margen de tensión	DC 16,4 V a 28,8 V
Separación galvánica	AC 500 V, 1 minuto	Rizado/corriente parásita (<10 MHz)	igual a la tensión alimentada
		Corriente disponible DC 24 V	400 mA
		Limitación de corriente de cortocircuito	< 600 mA
		Aislamiento	no
		<b>Alimentación para comunicación 5 V DP</b>	
		Corriente disponible DC 5 V:	90 mA, disponible en interface DP, pin 6-5, para repetidor DP
		Aislamiento	Transformador, AC 500 V, 1 minuto

<sup>1</sup> En la CPU están previstas 16 entradas y 16 salidas en la imagen del proceso para E/S locales.

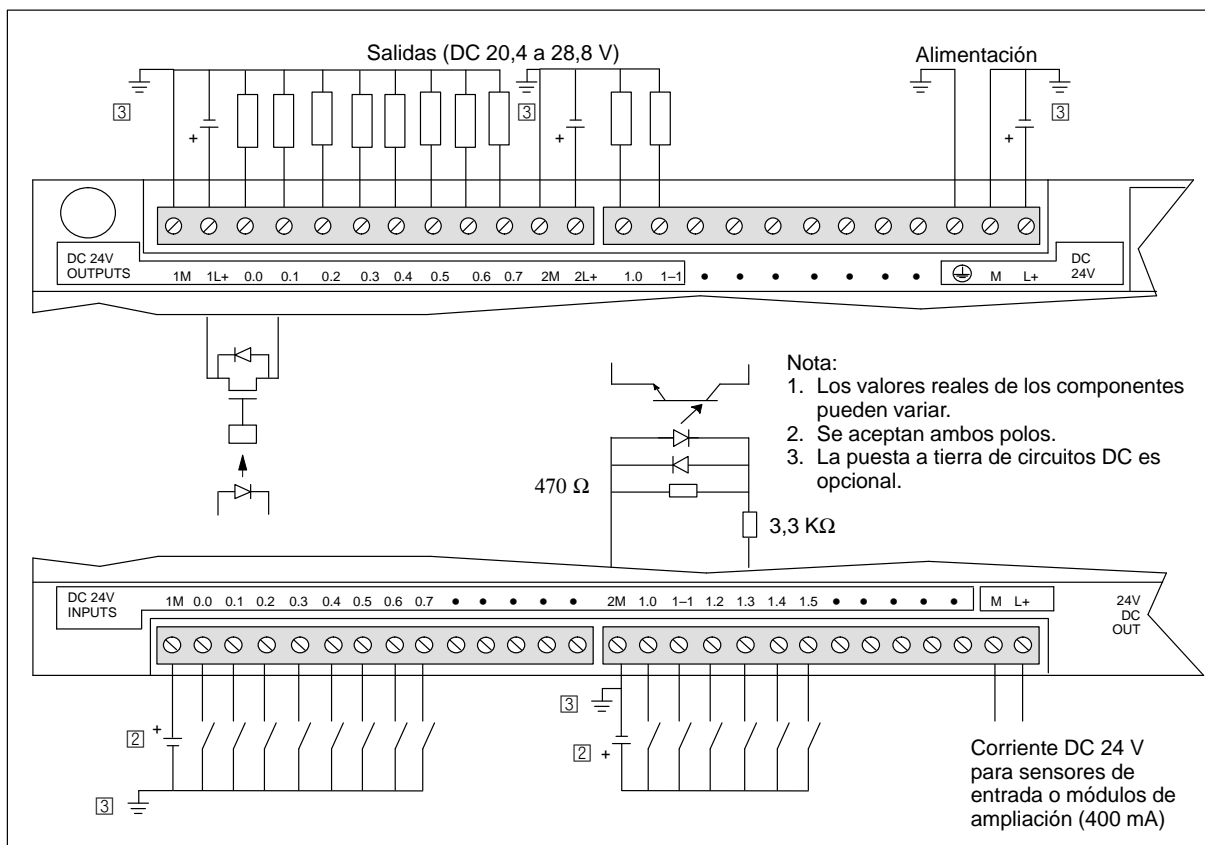


Figura A-15 Identificación de terminales de conexión para la CPU 215 DC/DC/DC

## A.16 CPU 215 alimentación AC, entradas DC, salidas de relé

Nº de referencia: 6ES7 215-2BD00-0XB0

Características generales		Salidas	
Dimensiones (l x a x p)	217,3 x 80 x 62 mm	Tipo de salida	Relé, contacto de baja potencia
Peso	0,6 kg	Margen de tensión	DC 5 V a 30 V / AC 250 V
Disipación	9 W	Corriente de carga máxima	2 A/salida, 6 A/hilo neutro
Tamaño programa de usuario/ memoria	4 Kpalabras/EEPROM	Sobretensión transitoria	7A al estar cerrados los contactos
Tamaño datos usuario/memoria	2,5 Kpalabras/RAM	Resistencia de aislamiento	mín. 100 MΩ (nuevo)
Ret. de datos y del tiempo real		Retardo de conmutación	máx. 10 ms
Condensador de alta potencia	típ. 190 h (mín. 120 h a 40°C)	Vida útil	10 000 000 mecánico 100 000 con carga normal
Pila opcional	200 días de uso continuo	Resistencia de contacto	máx. 200 mΩ (nuevo)
E/S locales <sup>1</sup>	14 entradas/10 salidas	Aislamiento	
Número máximo de módulos de ampliación	7	bobina a contacto	AC 1500 V, 1 minuto
E/S digitales asistidas	64 entradas/64 salidas	contacto a contacto	AC 750 V, 1 minuto
E/S analógicas asistidas	16 entradas/16 salidas	(entre contactos abiertos)	
Velocidad de ejecución booleana	0,8 μs/operación	Protección contra cortocircuitos	ninguna
Marcas internas	256	Alimentación	
Temporizadores	256 temporizadores	Margen de tensión/ de frecuencia	AC 85 V a 264 V a 47 a 63 Hz
Contadores	256 contadores	Corriente de entrada	típ. 6 VA, sólo CPU 50 VA carga máx.
Contadores rápidos	1 software (máx. 2 KHz) 2 hardware (máx. 20 KHz c/u)	Tiempo de retardo	mín. 20 ms de AC 110 V,
Tolerancia del reloj de tiempo real (TOD)	6 minutos/mes	Extra-corriente de cierre	20 A pico a AC 264 V
Salidas de impulsos	no recomendadas	Protección con fusibles	2 A, 250 V, de acción lenta (no reemplazable)
Potenciómetros analógicos	2	Corriente disponible DC 5 V	1000 mA para módulo de ampliación
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según VDE 0160 según CE	Aislamiento	Sí. Transformador, AC 1500 V, 1 minuto
Entradas		Alimentación para sensores DC	
Tipo de entrada	Fuente Tipo I IEC 1131 con sumidero de corriente	Margen de tensión	DC 19,2 V a 28,8 V
Área en ON	DC 15 V a 30 V, mín. 4 mA DC 35 V, 500 ms sobretensión transitoria	Rizado/corriente parásita (<10 MHz)	máx. 1 V de pico a pico
Tensión nominal en ON	DC 24 V, 7 mA	Corriente disponible DC 24 V	400 mA
Máximo en OFF	DC 5 V, 1 mA	Limitación de corriente de cortocircuito	< 600 mA
Tiempo de respuesta máxima I0.0 a I1.5	0,2 ms a 8,7 ms seleccionable 0,2 ms predeterminado	Aislamiento	no
I0.6 a I1.5 como se usa en HSC1 y HSC2	6 μs ON, 30 μs OFF	Alimentación para comunicación 5V DP	
		Corriente disponible DC 5 V:	90 mA, disponible en interface DP, pin 6-5, para repetidor DP
		Aislamiento	Transformador, AC 500 V, 1 minuto

<sup>1</sup> En la CPU están previstas 16 entradas y 16 salidas en la imagen del proceso para E/S locales.

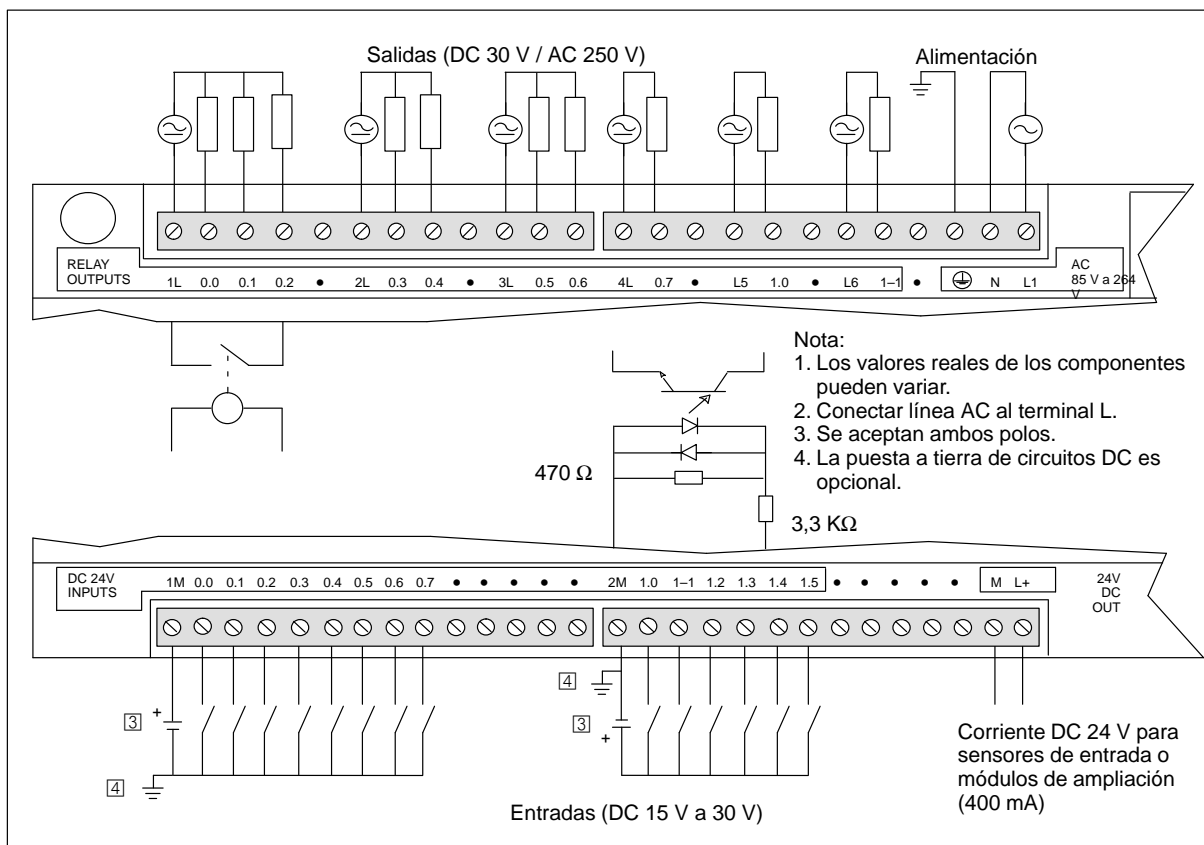


Figura A-16 Identificación de terminales de conexión para la CPU 215 AC/DC/relé

## A.17 CPU 216 alimentación DC, entradas DC, salidas DC

Nº de referencia: 6ES7 216-2AD00-0XB0

Características generales		Salidas	
Dimensiones (l x a x p)	217,3 x 80 x 62 mm	Tipo de salida	MOSFET, fuente
Peso	0,5 kg	Margen de tensión	DC 20,4 V a 28,8 V
Disipación	8 W	Corriente de carga máxima	0 a 55 ° C
Tamaño programa de usuario/ memoria	4 Kpalabras/EEPROM	Las salidas se pueden conectar en paralelo para corrientes superiores.	0,5 A/salida
Tamaño datos usuario/memoria	2,5 Kpalabras/RAM	Corriente de derivación	200 µA
Ret. de datos y del tiempo real		Retardo de conmutación	
Condensador de alta potencia	típ. 190 h (mín. 120 h a 40° C)	Q0.0, 0.1	100 µs, ON/OFF
Pila opcional	200 días de uso continuo	Demás salidas	150 µs ON, 400 µs OFF
E/S locales <sup>1</sup>	24 entradas/16 salidas	Resistencia ON	máx. 400 mW
Número máximo de módulos de ampliación	7	Protección contra cortocircuitos	0,7 a 1,5 A/canal
E/S digitales asistidas	64 entradas/64 salidas	Separación galvánica	AC 500 V, 1 minuto
E/S analógicas asistidas	16 entradas/16 salidas	Alimentación	
Velocidad de ejecución booleana	0,8 µs/operación	Margen de tensión	DC 20,4 V a 28,8 V
Marcas internas	256	Corriente de entrada	típ. 100 mA, sólo CPU 1,2 A carga máx.
Temporizadores	256 temporizadores	Medición UL/CSA	50 VA
Contadores	256 contadores	Tiempo de retardo	mín. 10 ms de DC 24 V
Contadores rápidos	1 software (máx. 2 KHz) 2 hardware (máx. 20 KHz c/u)	Extra-corriente de cierre	10 A pico a DC 28,8 V
Tolerancia del reloj de tiempo real (TOD)	6 minutos/mes	Protección con fusibles (no reemplazable)	2 A, de acción lenta
Salidas de impulsos	2 (máx. 4 kHz c/u)	Corriente disponible DC 5 V	1000 mA para módulo de ampliación
Potenciómetros analógicos	2	Aislamiento	no
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según VDE 0160 según CE	Alimentación para sensores DC	
Entradas		Margen de tensión	DC 16,4 V a 28,8 V
Tipo de entrada	Fuente Tipo 1 IEC 1131 con sumidero de corriente	Rizado/corriente parásita (<10 MHz)	igual a la tensión alimentada
Área en ON	DC 15 V a 30 V, mín. 4 mA DC 35 V, 500 ms sobretensión transitoria	Corriente disponible DC 24 V	400 mA
Tensión nominal en ON	DC 24 V, 7 mA	Limitación de corriente de cortocircuito	< 600 mA
Máximo en OFF	DC 5 V, 1 mA	Aislamiento	no
Tiempo de respuesta máxima			
I0.0 a I1.5	0,2 ms a 8,7 ms seleccionable 0,2 ms predeterminado		
I0.6 a I1.5 como se usa en HSC1 y HSC2	6 µs ON, 30 µs OFF		
I1.6 a I2.7	máx. 4 ms		
Separación galvánica	AC 500 V, 1 minuto		

<sup>1</sup> En la CPU están previstas 24 entradas y 16 salidas en la imagen del proceso para E/S locales.

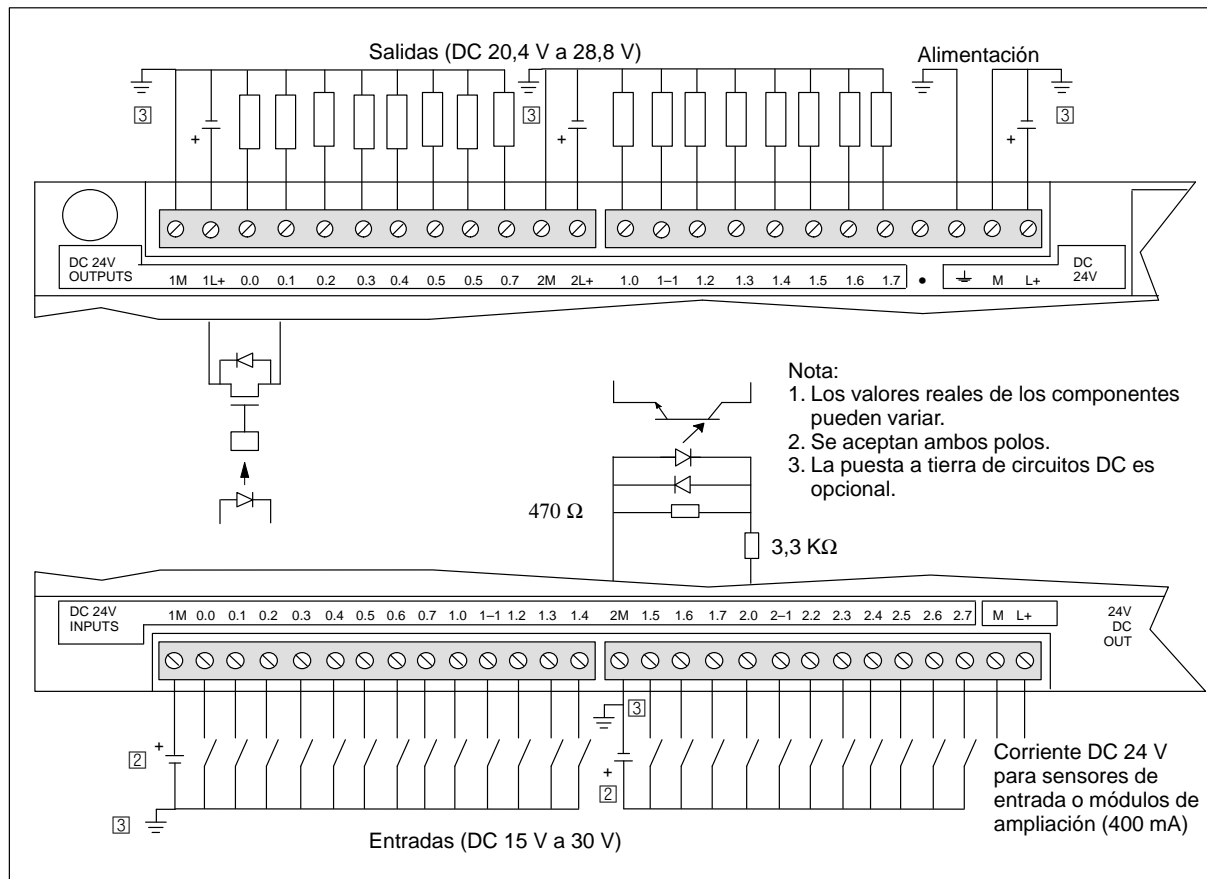


Figura A-17 Identificación de terminales de conexión para la CPU 216 DC/DC/DC

## A.18 CPU 216 alimentación AC, entradas DC, salidas de relé

Nº de referencia: 6ES7 216-2BD00-0XB0

Características generales		Salidas	
Dimensiones (l x a x p)	217,3 x 80 x 62 mm	Tipo de salida	Relé, contacto de baja potencia
Peso	0,6 kg	Margen de tensión	DC 5 V a 30 V / AC 250 V
Disipación	9 W	Corriente de carga máxima	2 A/salida, 10 A/hilo neutro
Tamaño programa de usuario/memoria	4 Kpalabras/EEPROM	Sobretensión transitoria	7A al estar cerrados los contactos
Tamaño datos usuario/memoria	2,5 Kpalabras/RAM	Resistencia de aislamiento	mín. 100 MΩ (nuevo)
Ret. de datos y del tiempo real		Retardo de conmutación	máx. 10 ms
Condensador de alta potencia	típ. 190 h	Vida útil	10 000 000 mecánico 100 000 con carga normal
Pila opcional	(mín. 120 h a 405C) 200 días de uso continuo	Resistencia de contacto	máx. 200 mΩ (nuevo)
E/S locales <sup>1</sup>	24 entradas/16 salidas	Aislamiento	
Número máximo de módulos de ampliación	7	bobina a contacto	AC 1500 V, 1 minuto
E/S digitales asistidas	64 entradas/64 salidas	contacto a contacto	AC 750 V, 1 minuto
E/S analógicas asistidas	16 entradas/16 salidas	(entre contactos abiertos)	
Velocidad de ejecución booleana	0,8 μs/operación	Protección contra cortocircuitos	ninguna
Marcas internas	256	<b>Alimentación</b>	
Temporizadores	256 temporizadores	Margen de tensión/ de frecuencia	AC 85 V a 264 V a 47 a 63 Hz
Contadores	256 contadores	Corriente de entrada	típ. 6 VA, sólo CPU 50 VA carga máx.
Contadores rápidos	1 software (máx. 2 KHz) 2 hardware (máx. 20 KHz c/u)	Tiempo de retardo	mín. 20 ms de AC 110 V,
Tolerancia del reloj de tiempo real (TOD)	6 minutos/mes	Extra-corriente de cierre	20 A pico a AC 264 V
Salidas de impulsos	no recomendadas	Protección con fusibles	2 A, 250 V, de acción lenta
Potenciómetros analógicos	2	(no reemplazable)	
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según VDE 0160 según CE	Corriente disponible DC 5 V	1000 mA para módulo de ampliación
<b>Entradas</b>		Aislamiento	Sí. Transformador, AC 1500 V, 1 minuto
Tipo de entrada	fuelle Tipo IEC 1131 con sumidero de corriente	<b>Alimentación para sensores DC</b>	
Área en ON	DC 15 V a 30 V, mín. 4 mA DC 35 V, 500 ms sobretensión transitoria	Margen de tensión	DC 19,2 V a 28,8 V
Tensión nominal en ON	DC 24 V, 7 mA	Rizado/corriente parásita (<10 MHz)	máx. 1 V de pico a pico
Máximo en OFF	DC 5 V, 1 mA	Corriente disponible DC 24 V	400 mA
Tiempo de respuesta máxima I0.0 a I1.5	0,2 ms a 8,7 ms seleccionable 0,2 ms predeterminado	Limitación de corriente de cortocircuito	< 600 mA
I0.6 a I1.5 como se usa en HSC1 y HSC2	6 μs ON, 30 μs OFF	Aislamiento	no
I1.6 a I2.7	máx. 4 ms		
Separación galvánica	AC 500 V, 1 minuto		

<sup>1</sup> En la CPU están previstas 24 entradas y 16 salidas en la imagen del proceso para E/S locales.

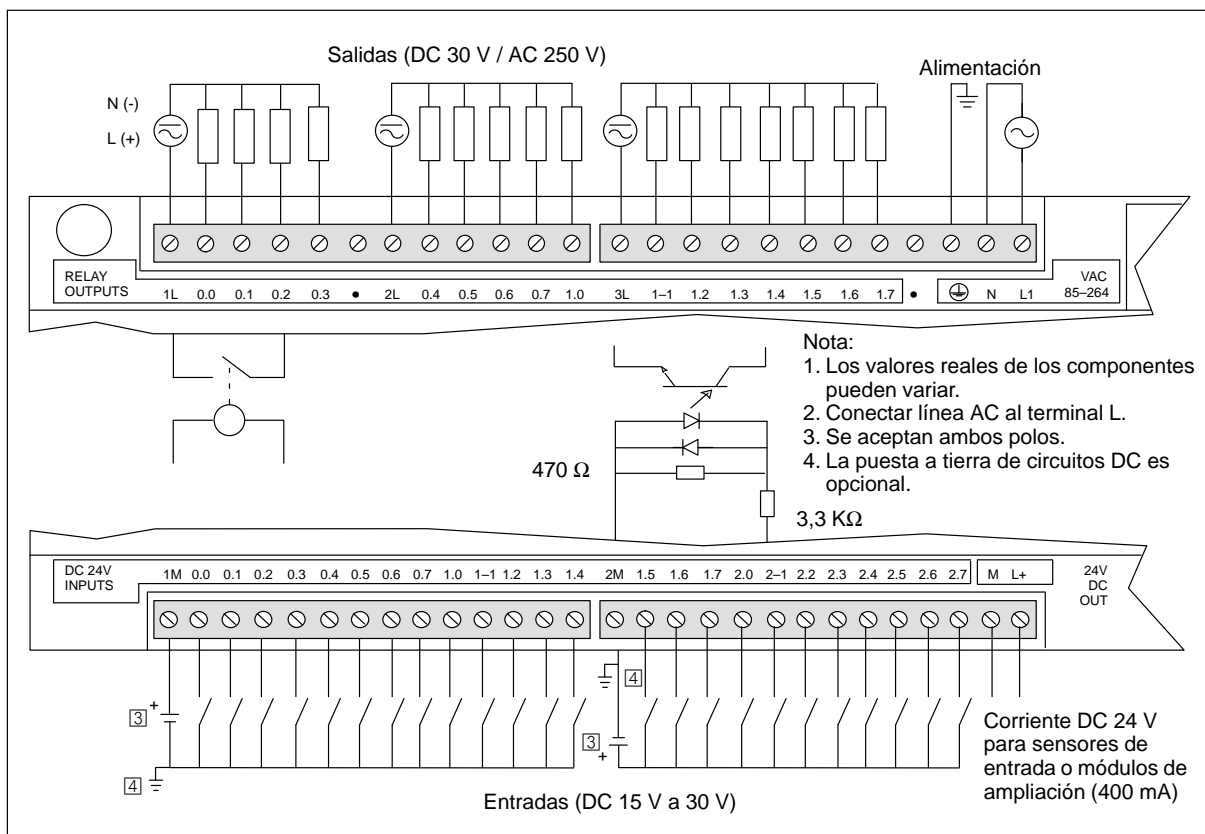


Figura A-18 Identificación de terminales de conexión para la CPU 216 AC/DC/relé

A.19 Módulo de ampliación EM 221, 8 entradas digitales DC 24 V

Nº de referencia: 6ES7 221-1BF00-0XA0

Características generales		Entradas	
Dimensiones (l x a x p)	90 x 80 x 62 mm	Tipo de entrada	Tipo 1, con sumidero de corriente según IEC 1131-2
Peso	0,2 kg	Área en ON	DC 15 V a 30 V, mín. 4 mA DC 35 V, 500 ms sobretensión transitoria
Disipación	2 W	Tensión nominal en ON	DC 24 V, 7 mA
E/S <sup>1</sup>	8 entradas digitales	Máximo en OFF	DC 5 V, 1 mA
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según VDE 0160 según CE	Tiempo de respuesta	típ. 3,5 ms/máx. 4,5 ms
		Separación galvánica	AC 500 V, 1 minuto
		Consumo	
		Corriente disponible DC 5 V	60 mA del aparato central
		Corriente de sensor DC 24 V	60 mA del aparato central o de una fuente de alimentación externa

<sup>1</sup> En la CPU están previstas 8 entradas en la imagen del proceso para este módulo.

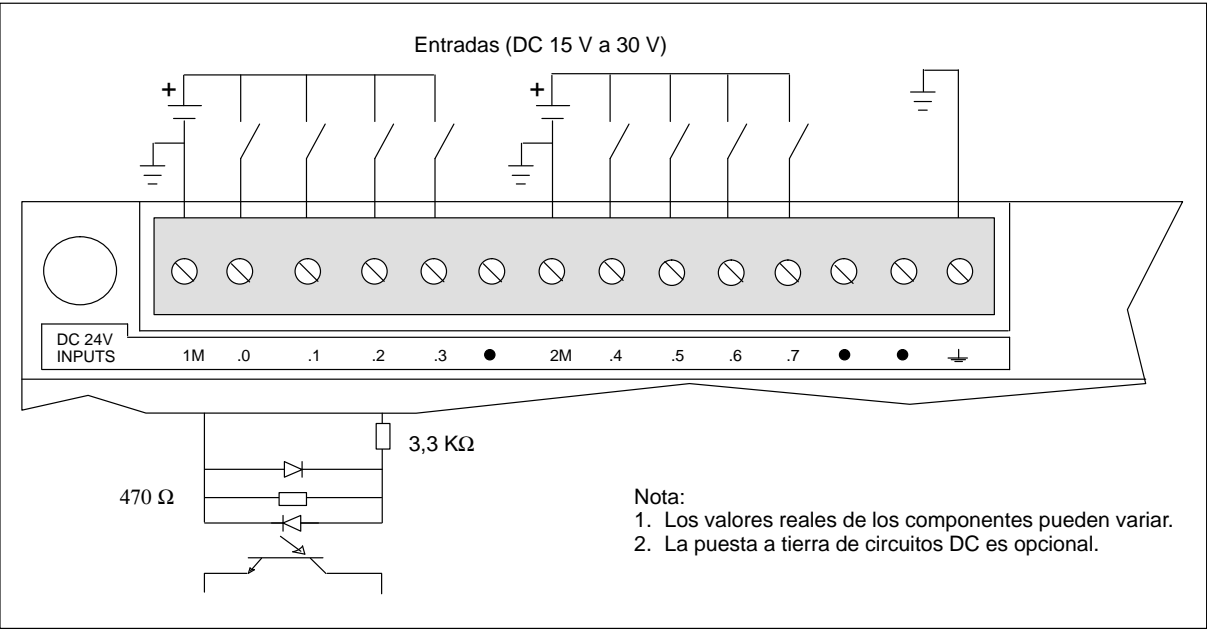


Figura A-19 Identificación de terminales de conexión para el EM221, 8 entradas digitales DC 24 V

A.20 Módulo de ampliación EM 221, 8 entradas digitales AC 120 V

Nº de referencia: 6ES7 221-1EF00-0XA0

Características generales		Entradas	
Dimensiones (l x a x p)	90 x 80 x 62 mm	Tipo de entrada	Tipo 1, con sumidero de corriente por IEC 1131-2
Peso	0,2 kg	Área en ON	AC 79 V a 135 V, 47 a 63 Hz, mín. 4 mA
Disipación	2 W	Tensión nominal en ON	AC 120 V, 60 Hz, 7 mA
E/S <sup>1</sup>	8 entradas digitales	Máximo en OFF	AC 20 V, 1 mA
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según CE	Tiempo de respuesta	máx. 15 ms
		Separación galvánica	AC 1500 V, 1 minuto
		Consumo	
		Corriente disponible DC 5 V	70 mA del aparato central

<sup>1</sup> En la CPU están previstas 8 entradas en la imagen del proceso para este módulo.

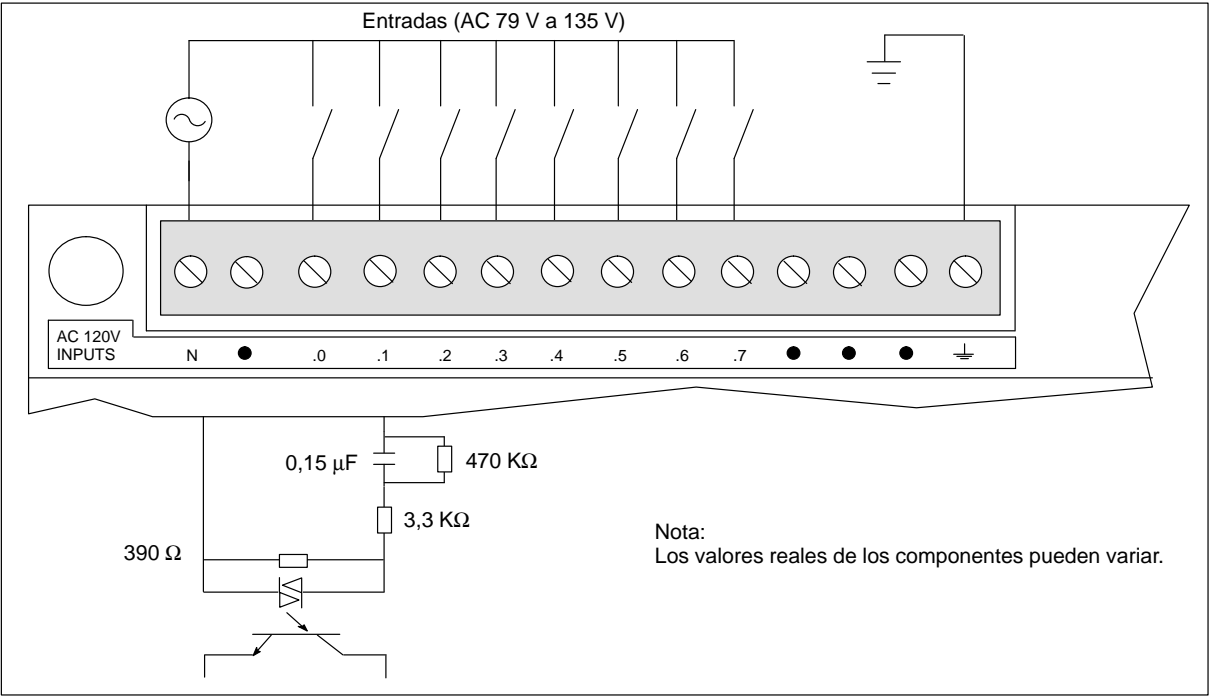


Figura A-20 Identificación de terminales de conexión para el EM221, 8 entradas digitales AC 120 V

A.21 Módulo de ampliación EM 221, 8 entradas digitales tipo fuente DC 24 V

Nº de referencia: 6ES7 221-1BF10-0XA0

Características generales		Entradas	
Dimensiones (l x a x p)	90 x 80 x 62 mm	Tipo de entrada	fuelle
Peso	0,2 kg	Margen de tensión de entrada	DC 15 V a 30 V, DC 35V, 500 ms
Disipación	2 W	Tensión nominal en ON	mín. 4 mA
E/S <sup>1</sup>	8 entradas digitales	Máximo en OFF	máx. 1 mA
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según VDE 0160 según CE	Tiempo de respuesta	típ. 3,5 ms/máx. 4,5 ms
		Separación galvánica	AC 500 V, 1 minuto
		Consumo	
		Corriente disponible DC 5 V	60 mA del aparato central
		Corriente de sensor DC 24 V	60 mA del aparato central o de una fuente de alimentación externa

<sup>1</sup> En la CPU están previstas 8 entradas en la imagen del proceso para este módulo.

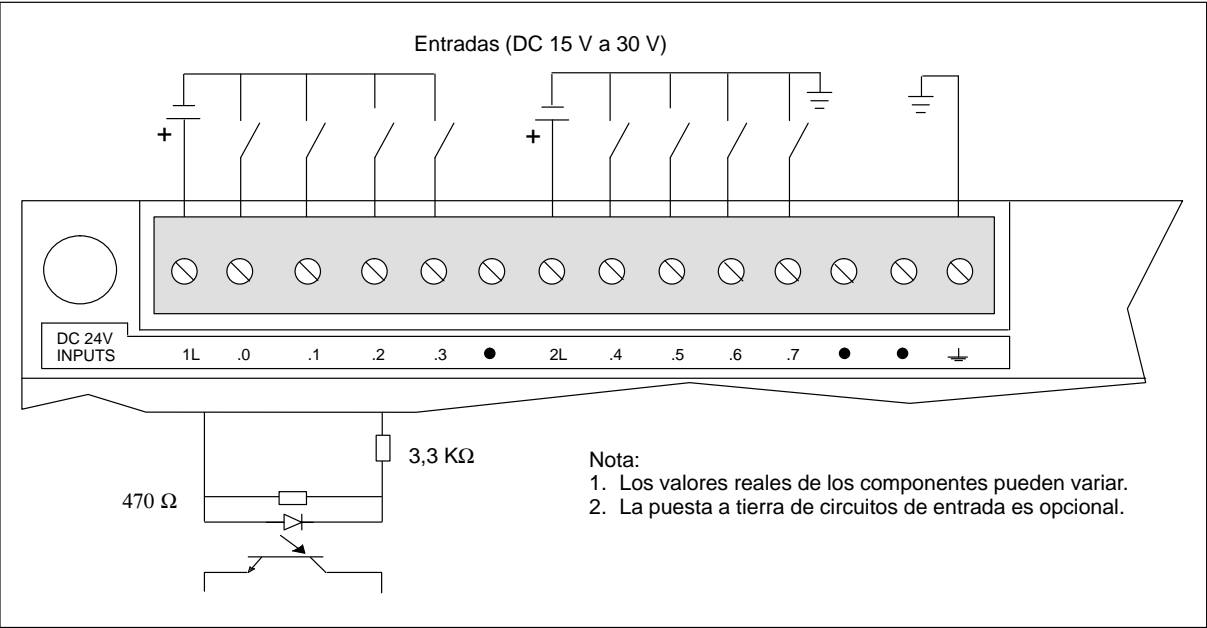


Figura A-21 Identificación de terminales de conexión para el EM221, 8 entradas digitales tipo fuente DC 24 V

A.22 Módulo de ampliación EM221, 8 entradas digitales AC 24 V

Nº de referencia: 6ES7 221-1JF00-0XA0

Características generales		Entradas	
Dimensiones (l x a x p)	90 x 80 x 62 mm	Tipo de entrada	Tipo 1, con sumidero de corriente por IEC 1131-2
Peso	0,2 kg	Área en ON	AC 15 V a 30 V, 47 a 63 Hz, mín. 4 mA
Disipación	2 W	Tensión nominal en ON	AC 24 V, 60 Hz, 7 mA
E/S <sup>1</sup>	8 entradas digitales	Máximo en OFF	AC 5 V, 1 mA
Homologaciones (pendientes)	UL 508 CSA C22.2 142 FM clase I, categoría 2 según CE	Tiempo de respuesta	máx. 15 ms
		Separación galvánica	AC 1500 V, 1 minuto
		Consumo	
		Corriente disponible DC 5 V	70 mA del aparato central

<sup>1</sup> En la CPU están previstas 8 entradas en la imagen del proceso para este módulo.

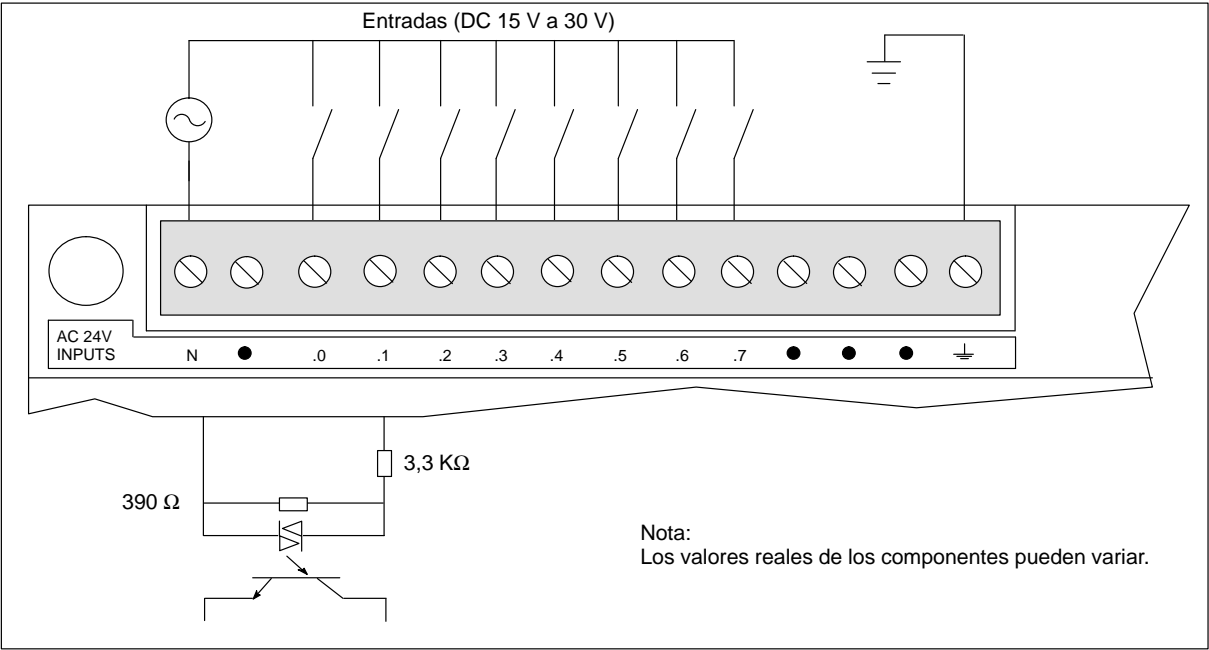


Figura A-22 Identificación de terminales de conexión para el EM221, 8 entradas digitales AC 24 V

A.23 Módulo de ampliación EM222, 8 salidas digitales DC 24 V

Nº de referencia: 6ES7 222-1BF00-0XA0

Características generales		Salidas (continuación)	
Dimensiones (l x a x p)	90 x 80 x 62 mm	Carga inductiva, apriete	(por hilo neutro)
Peso	0,2 kg	Impulso individual	2A L/R = 10 ms
Disipación	4W a 3 A de carga	Repetición	1A L/R = 100 ms
E/S <sup>1</sup>	8 salidas digitales		1 W disipación de energía
Homologaciones	UL 508 CSA C22.2 142		(1/2 Li² x frecuencia de conmutación < 1W)
	FM clase I, categoría 2 según VDE 0160 según CE	Corriente de derivación	100 µA
Salidas		Retardo de conmutación	50 µs ON, 200 µs OFF
Tipo de salida	Transistor, tipo fuente	Sobrecorriente momentánea	4 A, 100 ms
Margen de tensión	DC 20,4 V a 28,8 V	Caída de tensión	1,8 V máx. con corriente máxima
Corriente de carga máxima por cada salida	0 a 40 °C 0,75 A	Separación galvánica	AC 500 V, 1 minuto
por 2 salidas adyacentes	1,00 A	Protección contra cortocircuitos	ninguna
todas las salidas en total	4,00 A		
	55° C <sup>2</sup> 0,50 A	Consumo	
		Corriente disponible DC 5 V	80 mA del aparato central
		Corriente para salidas	Suministrada por el usuario a través del hilo neutro del módulo

<sup>1</sup> En la CPU están previstas 8 salidas en la imagen del proceso para este módulo.

<sup>2</sup> Disminución lineal de potencia 40 a 55° C. Disminución de potencia montaje vertical 10° C

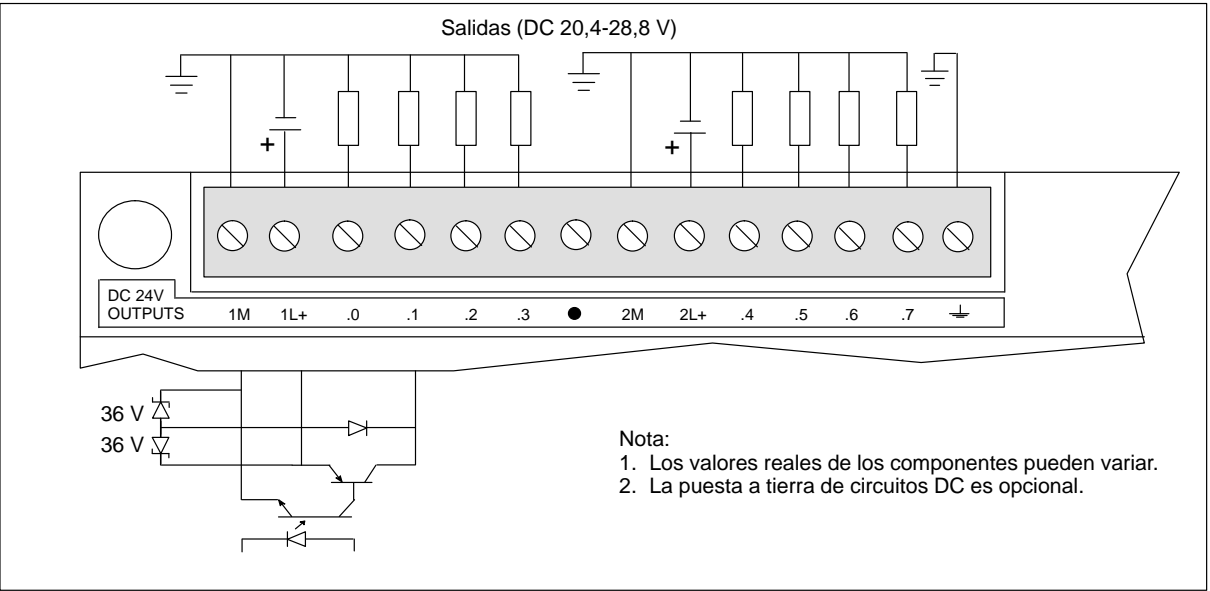


Figura A-23 Identificación de terminales de conexión para el EM222, 8 salidas digitales DC 24 V

A.24 Módulo de ampliación EM 222, 8 salidas digitales de relé

Nº de referencia: 6ES7 222-1HF00-0XA0

Características generales		Salidas (continuación)	
Dimensiones (l x a x p)	90 x 80 x 62 mm	Retardo de conmutación	máx. 10 ms
Peso	0,2 kg	Vida útil	10 000 000 mecánico 100 000 con carga normal
Disipación	3 W	Resistencia de contacto	máx. 200 mΩ (nuevo)
E/S <sup>1</sup>	8 salidas digitales de relé	Aislamiento	
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según VDE 0160 según CE	bobina a contacto	AC 1500 V, 1 minuto
		contacto a contacto	AC 750 V, 1 minuto
		(entre contactos abiertos)	
		Protección contra cortocircuitos	ninguna
Salidas		Consumo	
Tipo de salida	Relé, contacto de baja potencia	Corriente disponible DC 5 V	80 mA del aparato central
Margen de tensión	DC 5 V a 30 V / AC 250 V	Corriente de bobina DC 24 V	85 mA del aparato central o de una fuente de alimentación externa
Corriente de carga máxima	2 A/salida, 8 A/hilo neutro	Corriente para salidas	Suministrada por el usuario a través del hilo neutro del módulo
Sobretensión transitoria	7A al estar cerrados los contactos		
Resistencia de aislamiento	mín. 100 MΩ (nuevo)		

<sup>1</sup> En la CPU están previstas 8 salidas en la imagen del proceso para este módulo.

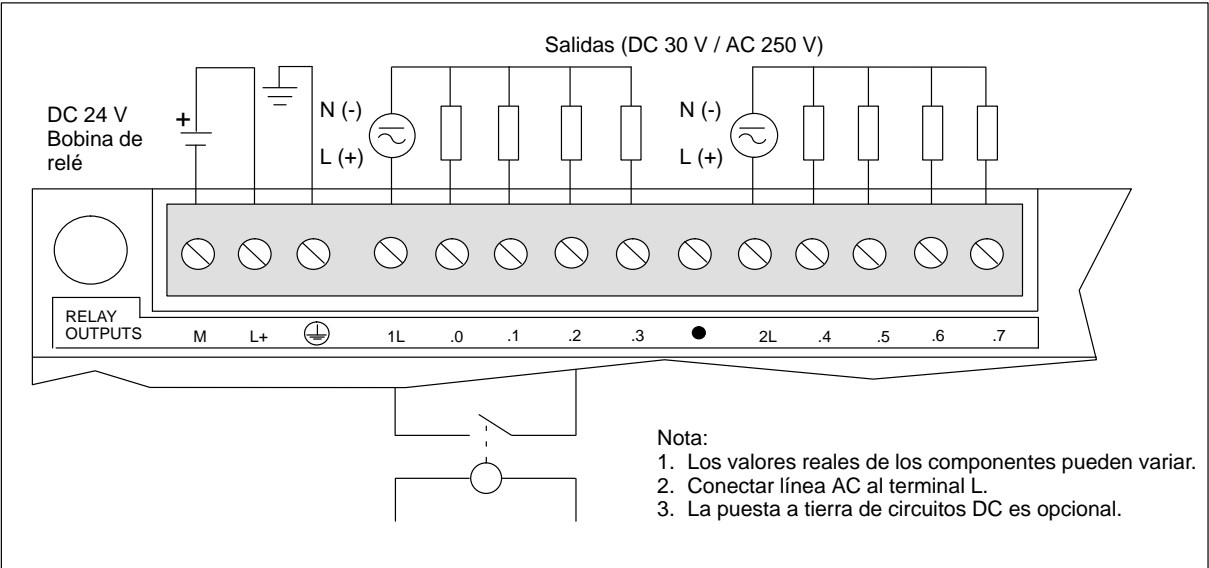


Figura A-24 Identificación de terminales de conexión para el EM222, 8 salidas de relé

## A.25 Módulo de ampliación EM222, 8 salidas digitales AC 120/230 V

Nº de referencia: 6ES7 222-1EF00-0XA0

Características generales		Salidas (continuación)	
Dimensiones (l x a x p)	90 x 80 x 62 mm	Corriente de carga mínima	30 mA
Peso	0,2 kg	Corriente de derivación	1,5 mA, AC 120 V/2,0 mA, AC 240 V
Disipación	5 W a 3,5 A de carga	Retardo de conmutación	1/2 ciclo
E/S <sup>1</sup>	8 salidas digitales	Sobrecorriente momentánea	30 A pico, 1 ciclo 10 A pico, 5 ciclos
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según CE	Caída de tensión	máx. 1,5 V con corriente máx.
Salidas		Separación galvánica	AC 1500 V, 1 minuto
Tipo de salida	Triac, conectar pasando por cero	Protección contra cortocircuitos	ninguna
Margen de tensión/ de frecuencia	AC 20 V a 264 V, 47 a 63 Hz	Consumo	
Factor de potencia del circuito de carga	0,3 a 1,0	Corriente disponible DC 5 V	120 mA del aparato central
Corriente de carga máxima por cada salida	0 a 40 °C 1,20 A 55° C <sup>2</sup> 1,00 A	Corriente para salidas	Suministrada por el usuario a través del hilo neutro del módulo
por 2 salidas adyacentes	1,50 A 1,25 A		
todas las salidas en total	4,75 A 3,50 A		

<sup>1</sup> En la CPU están previstas 8 salidas en la imagen del proceso para este módulo.

<sup>2</sup> Disminución lineal de potencia 40 a 55° C. Disminución de potencia montaje vertical 10° C

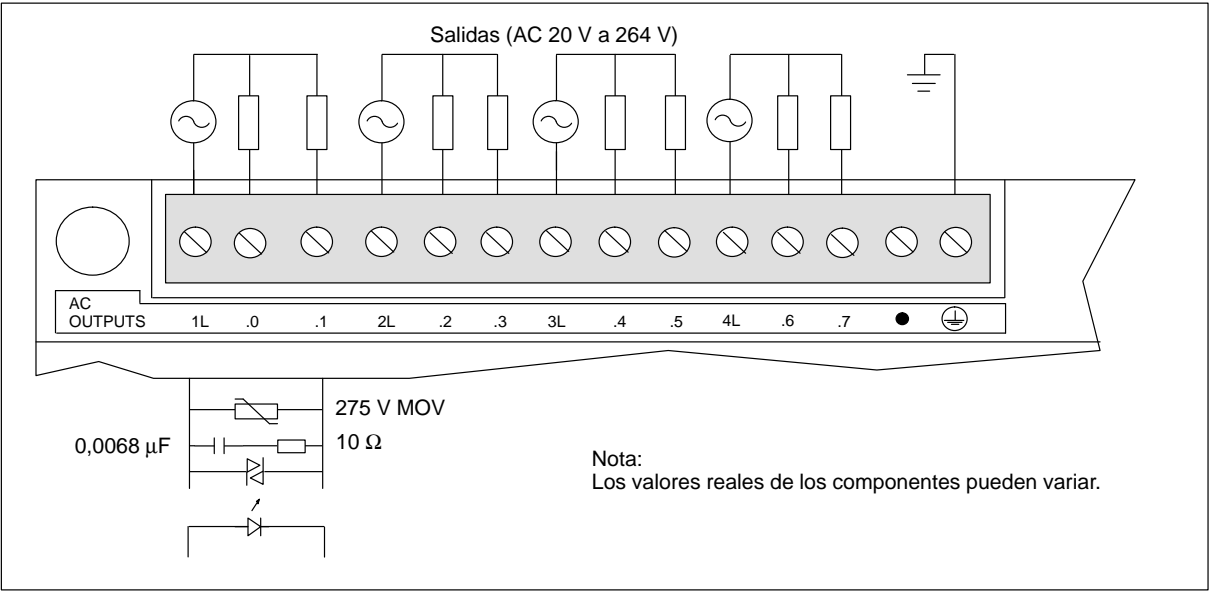


Figura A-25 Identificación de terminales de conexión para el EM222, 8 salidas digitales AC 120/230 V

## A.26 Módulo de ampliación EM 223, 4 entradas digitales DC 24 V / 4 salidas digitales DC 24 V

Nº de referencia: 6ES7 223-1BF00-0XA0

Características generales		Salidas (continuación)	
Dimensiones (l x a x p)	90 x 80 x 62 mm	Corriente de derivación	máx. 1 µA
Peso	0,2 kg	Retardo de conmutación	máx. 25 µs ON, 120 µs OFF
Disipación	3,5 W a 3 A de carga	Sobrecorriente momentánea	7 A, 100 ms
E/S <sup>1</sup>	4 entradas digitales 4 salidas digitales	Separación galvánica	AC 500 V, 1 minuto
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según VDE 0160 según CE	Protección contra cortocircuitos	ninguna
Salidas		Entradas	
Tipo de salida	Transistor, tipo fuente (canal P MOSFET)	Tipo de entrada	Tipo 1, con sumidero de corriente según IEC 1131-2
Margen de tensión	DC 20,4 a 28,8 V	Área en ON	DC 15 V a 30 V, mín. 4 mA DC 35 V, 500 ms sobretensión transitoria
Resistencia para ON	máx. 400 mΩ	Tensión nominal en ON	DC 24 V, 7 mA
Corriente de carga máxima por cada salida	0 a 40 °C 2,50 A 55 °C* 2,00 A	Máximo en OFF	DC 5 V, 1 mA
todas las salidas en total	4,00 A 3,00 A	Tiempo de respuesta	típ. 3,5 ms / máx. 4,5 ms
*Disminución lineal de potencia 40 a 55° C		Separación galvánica	AC 500 V, 1 minuto
Disminución de potencia montaje vertical 10° C (En caso de cargas fuertes de corriente se pueden conectar dos salidas en paralelo).		Consumo	
Carga inductiva, apriete	(por hilo neutro)	Corriente disponible DC 5 V	80 mA del aparato central
Impulso individual	2A L/R = 10 ms 1A L/R = 100 ms	Corriente de sensor DC 24 V	30 mA del aparato central o de una fuente de alimentación externa
Repetición	1 W disipación de energía (1/2 Li <sup>2</sup> x frecuencia de conmutación < 1W)	Corriente para salidas	Suministrada por el usuario a través del hilo neutro del módulo

<sup>1</sup> En la CPU están previstas 8 entradas y 8 salidas en la imagen del proceso para este módulo.

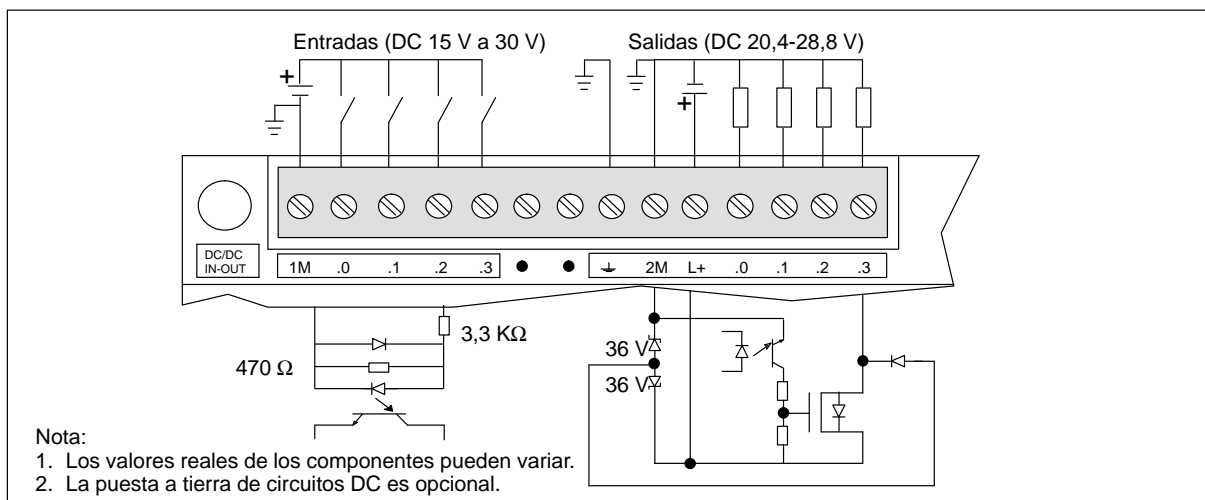


Figura A-26 Identificación de terminales de conexión para el EM223, 4 entradas digitales DC 24 V / 4 salidas digitales DC 24 V

## A.27 Módulo de ampliación EM 223, 8 entradas digitales DC 24 V / 8 salidas digitales DC 24 V

Nº de referencia: 6ES7 223-1BH00-0XA0

Características generales		Entradas	
Dimensiones (l x a x p)	90 x 80 x 62 mm	Tipo de entrada	fuelle Tipo 1 con sumidero de corriente según IEC 1131-2
Peso	0,2 kg	Área en ON	DC 15 V a 30 V, mín. 4 mA DC 35 V, 500 ms sobretensión transitoria
Disipación	3,0 W	Tensión nominal en ON	DC 24 V, 7 mA
E/S <sup>1</sup>	8 entradas digitales 8 salidas digitales	Máximo en OFF	DC 5 V, 1 mA
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según VDE 0160 según CE	Tiempo de respuesta	máx. 4,0 ms
Salidas		Separación galvánica	AC 500 V, 1 minuto
Tipo de salida	MOSFET, fuente	Consumo	
Margen de tensión	DC 20,4 V a 28,8 V	Corriente disponible DC 5 V	120 mA del aparato central
Corriente de carga máxima Las salidas se pueden conectar en paralelo para corrientes superiores.	0 a 55° C 0,5 A/salida	Corriente de sensor DC 24 V	60 mA del aparato central o de una fuente de alimentación externa
Corriente de derivación	200 µA	Corriente para salidas	Suministrada por el usuario a través del hilo neutro del módulo
Retardo de conmutación	150 µs ON, 400 µs OFF		
Resistencia ON	máx. 400 mΩ		
Protección contra cortocircuitos	0,7 a 1,5 A/canal		
Separación galvánica	AC 500 V, 1 minuto		

<sup>1</sup> En la CPU están previstas 8 entradas y 8 salidas en la imagen del proceso para este módulo.

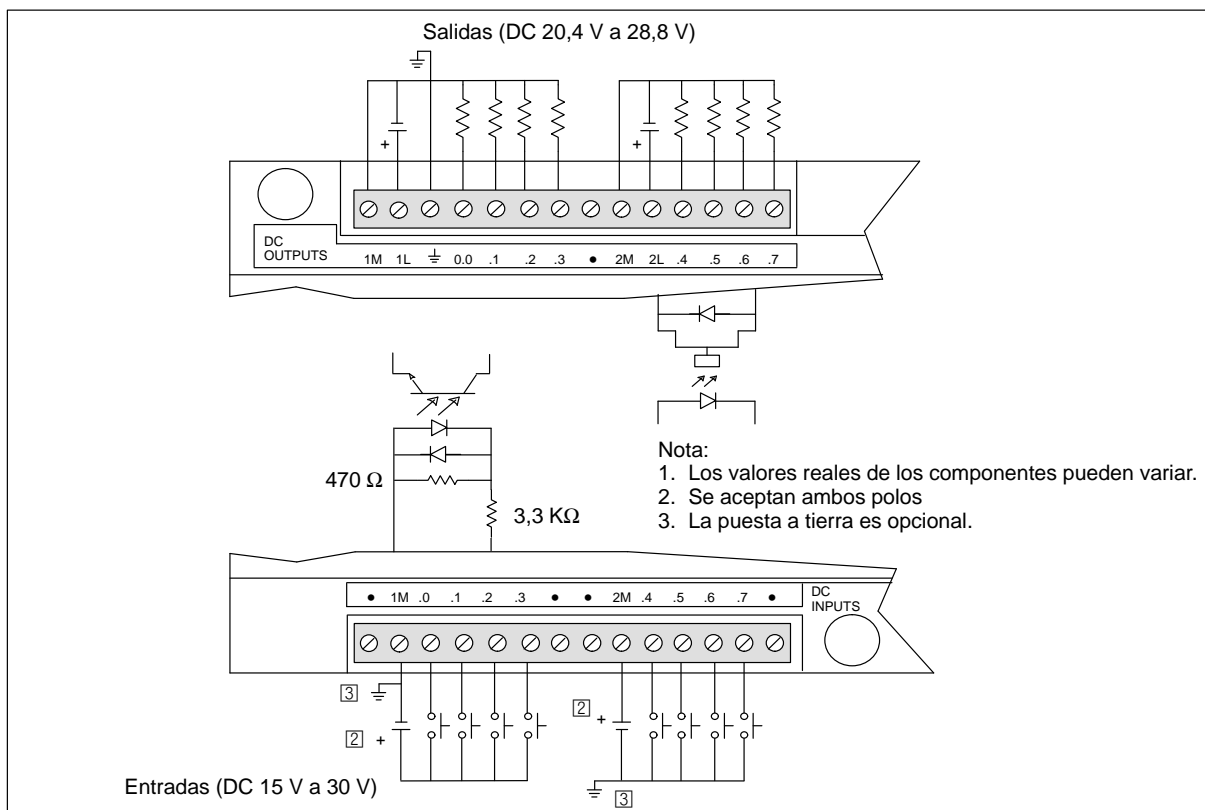


Figura A-27 Identificación de terminales de conexión para el EM223, 8 entradas digitales DC 24 V / 8 salidas digitales DC 24 V

## A.28 Módulo de ampliación EM 223, 16 entradas digitales DC 24 V / 16 salidas digitales DC 24 V

Nº de referencia: 6ES7 223-1BL00-0XA0

Características generales		Entradas	
Dimensiones (l x a x p)	160 x 80 x 62 mm	Tipo de entrada	fuelle Tipo 1 con sumidero de corriente según IEC 1131-2
Peso	0,4 kg	Área en ON	DC 15 a 30 V, mín. 4 mA DC 35 V, 500 ms sobretensión transitoria
Disipación	5,5 W	Tensión nominal en ON	DC 24 V, 7 mA
E/S <sup>1</sup>	16 entradas digitales 16 salidas digitales	Máximo en OFF	DC 5 V, 1 mA
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según VDE 0160 según CE	Tiempo de respuesta	máx. 4,0 ms
Salidas		Separación galvánica	AC 500 V, 1 minuto
Tipo de salida	MOSFET, fuente	Consumo	
Margen de tensión	DC 20,4 V a 28,8 V	Corriente disponible DC 5 V	210 mA del aparato central
Corriente de carga máxima Las salidas se pueden conectar en paralelo para corrientes superiores.	0 a 55° C 0,5 A/salida	Corriente de sensor DC 24 V	120 mA del aparato central o de la fuente de alimentación externa
Corriente de derivación	200 µA	Corriente para salidas	Suministrada por el usuario a través del hilo neutro del módulo
Retardo de conmutación	150 µs ON, 400 µs OFF		
Resistencia ON	máx. 400 mΩ		
Protección contra cortocircuitos	0,7 a 1,5 A/canal		
Separación galvánica	AC 500 V, 1 minuto		

<sup>1</sup> En la CPU están previstas 16 entradas y 16 salidas en la imagen del proceso para este módulo.

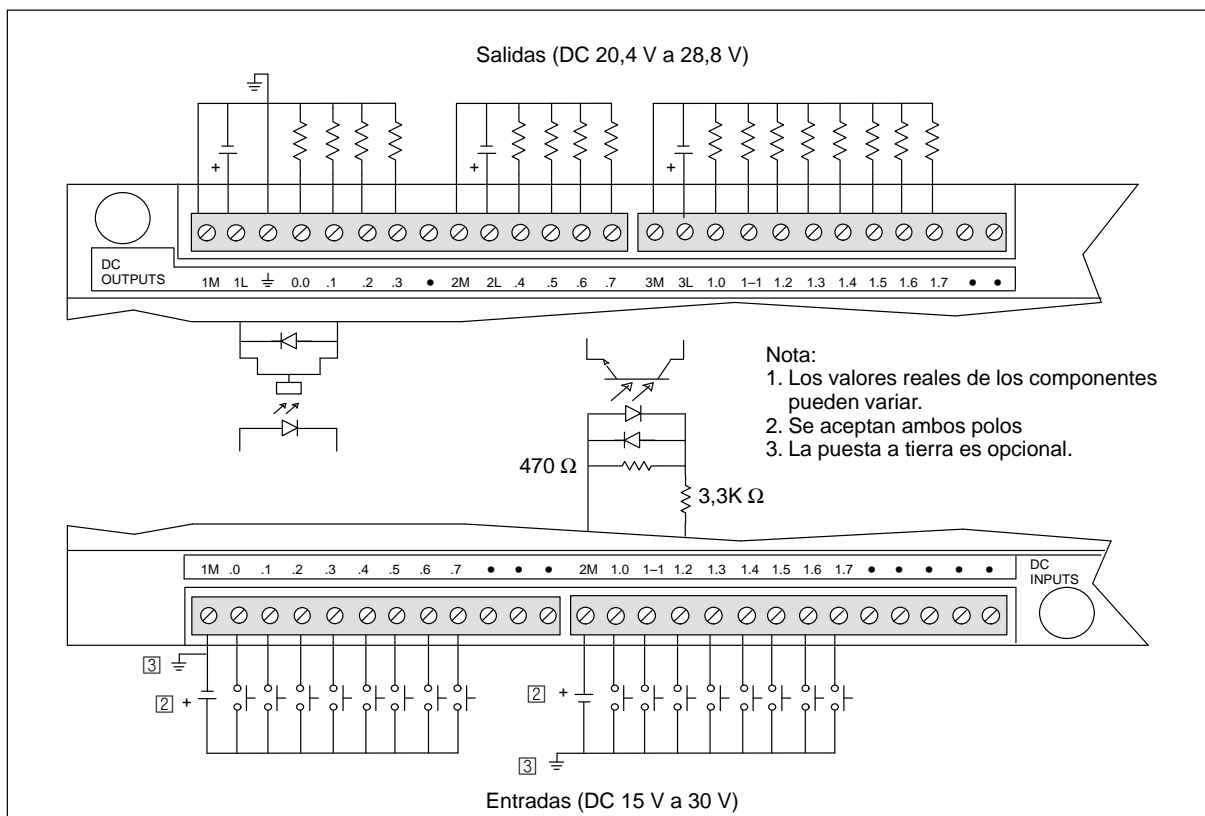


Figura A-28 Identificación de terminales de conexión para el EM223 , 16 entradas digitales DC 24 V / 16 salidas digitales DC 24 V

A.29 Módulo de ampliación EM 223, 4 entradas digitales 4 DC V / 4 salidas digitales de relé

Nº de referencia: 6ES7 223-1HF00-0XA0

Características generales		Salidas (continuación)	
Dimensiones (l x a x p)	90 x 80 x 62 mm	Resistencia de contacto	máx. 200 mΩ (nuevo)
Peso	0,2 kg	Protección contra cortocircuitos	ninguna
Disipación	2 W	Entradas	
E/S <sup>1</sup>	4 entradas digitales 4 salidas digitales de relé	Tipo de entrada	Tipo 1, con sumidero de corriente según IEC 1131-2
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según VDE 0160 según CE	Área en ON	DC 15 V a 30 V, mín. 4 mA DC 35 V, 500 ms sobretensión transitoria
Salidas		Tensión nominal en ON	DC 24 V, 7 mA
Tipo de salida	Relé, contacto de baja potencia	Máximo en OFF	DC 5 V, 1 mA
Margen de tensión	DC 5 V a 30 V / AC 250 V	Tiempo de respuesta	típ. 3,5 ms / máx. 4,5 ms
Corriente de carga máxima	2 A / salida	Separación galvánica	AC 500 V, 1 minuto
Resistencia de aislamiento	máx. 100 MΩ (nuevo)	Consumo	
Retardo de conmutación	máx. 10 ms	Corriente disponible DC 5 V	80 mA del aparato central
Vida útil	10 000 000 mecánico 100 000 con carga normal	Corriente de sensor DC 24 V	30 mA del aparato central o de una fuente de alimentación externa
Aislamiento		Corriente de bobina DC 24 V	35 mA del aparato central o de una fuente de alimentación externa
bobina a contacto	AC 100 V, 1 minuto	Corriente para salidas	Suministrada por el usuario a través del hilo neutro del módulo
contacto a contacto (entre contactos abiertos)	AC 750 V, 1 minuto		

<sup>1</sup> En la CPU están previstas 8 entradas y 8 salidas en la imagen del proceso para este módulo.

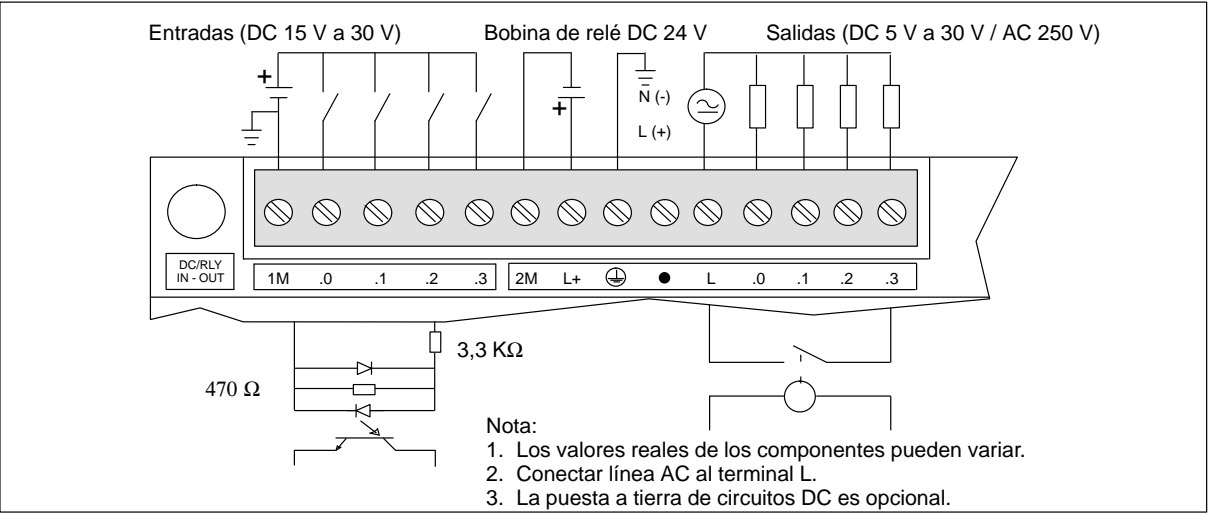


Figura A-29 Identificación de terminales de conexión para el EM223, 4 entradas digitales DC 24 V / 4 salidas digitales de relé

A.30 Módulo de ampliación EM 223, 4 entradas digitales AC 120 V / 4 salidas digitales AC 120/230 V

Nº de referencia: 6ES7 223-1EF00-0XA0

Características generales		Salidas (continuación)	
Dimensiones (l x a x p)	90 x 80 x 62 mm	Sobrecorriente momentánea	50 A pico, 1 ciclo 15 A pico, 5 ciclos
Peso	0,2 kg	Caída de tensión	1,8 V máx. con corriente máxima
Disipación	5,5 W a 3 A de carga	Separación galvánica	AC 1500 V, 1 minuto
E/S <sup>1</sup>	4 entradas digitales 4 salidas digitales	Protección contra cortocircuitos	ninguna
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según CE	<b>Entradas</b>	
<b>Salidas</b>		Tipo de entrada	Tipo 1, con sumidero de corriente según IEC 1131-2
Tipo de salida	Triac, conectar pasando por cero	Área en ON	AC 79 V a 135 V, 47 a 63 Hz mín. 4 mA
Margen de tensión/de frecuencia	AC 70 V a 264 V, 47 a 63 Hz	Tensión nominal en ON	AC 120 V, 60 Hz, 7 mA
Factor de potencia del circuito de carga	0,3 a 1,0	Máximo en OFF	AC 20 V, 1 mA
Corriente de carga máxima por cada salida	0 a 40 °C 2,40 A 55° C <sup>2</sup> 2,00 A	Tiempo de respuesta	máx. 15 ms
todas las salidas en total	4,00 A 3,00 A	Separación galvánica	AC 1500 V, 1 minuto
Corriente de carga mínima	10 mA	<b>Consumo</b>	
Corriente de derivación	2,5 mA, 120 V 4,0 mA, 230 V	Corriente disponible DC 5 V	100 mA del aparato central
Retardo de conmutación	1/2 ciclo	Corriente para salidas	Suministrada por el usuario a través del hilo neutro del módulo

- <sup>1</sup> En la CPU están previstas 8 entradas y 8 salidas en la imagen del proceso para este módulo.  
<sup>2</sup> Disminución lineal de potencia 40 a 55° C. Disminución de potencia montaje vertical 10° C

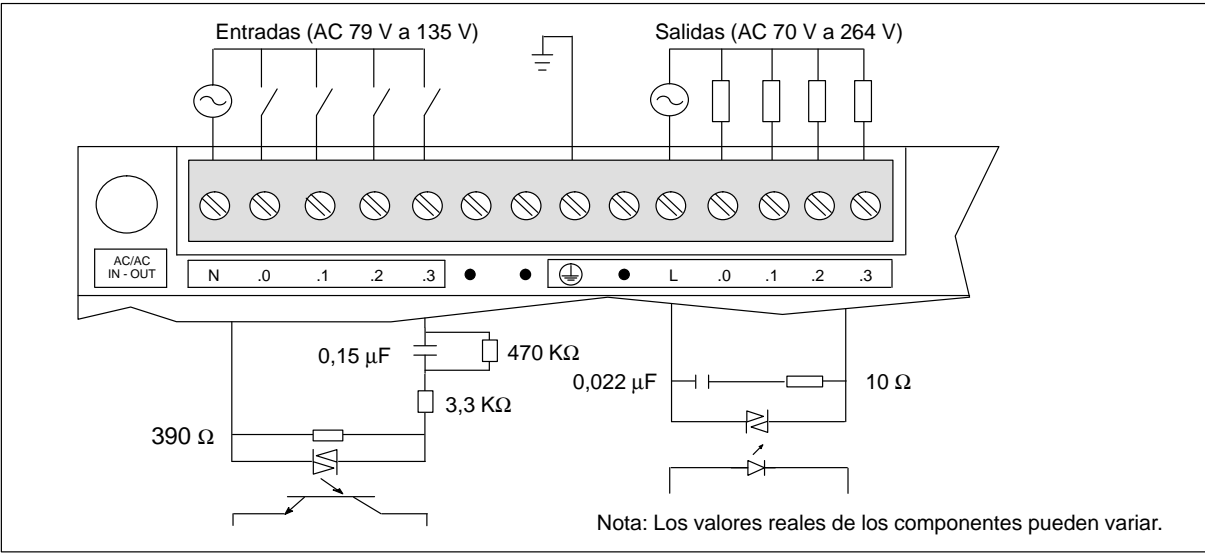


Figura A-30 Identificación de terminales de conexión para el EM223, 4 entradas digitales AC 120 V / 4 salidas digitales AC 120/230 V

## A.31 Módulo de ampliación EM 223, 8 entradas digitales DC 24 V / 8 salidas digitales de relé

Nº de referencia: 6ES7 223-1PH00-0XA0

Características generales		Entradas	
Dimensiones (l x a x p)	90 x 80 x 62 mm	Tipo de entrada	fuelle Tipo 1 con sumidero de corriente según IEC 1131-2
Peso	0,3 kg	Área en ON	DC 15 V a 30 V, mín. 4 mA DC 35 V, 500 ms sobretensión transitoria
Disipación	2,5 W	Tensión nominal en ON	DC 24 V, 7 mA
E/S <sup>1</sup>	8 entradas digitales 8 salidas digitales	Máximo en OFF	DC 5 V, 1 mA
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según VDE 0160 según CE	Tiempo de respuesta	máx. 4,0 ms
Salidas		Separación galvánica	AC 500 V, 1 minuto
Tipo de salida	Relé, contacto de baja potencia	Consumo	
Margen de tensión	DC 5 V a 30 V / AC 250 V	Corriente disponible DC 5 V	100 mA del aparato central
Corriente de carga máxima	2 A/salida, 8 A/hilo neutro	Corriente de sensor DC 24 V	90 mA del aparato central o de una fuente de alimentación externa
Resistencia de aislamiento	máx. 100 MΩ (nuevo)	Corriente para salidas	Suministrada por el usuario a través del hilo neutro del módulo
Retardo de conmutación	máx. 10 ms		
Vida útil	10 000 000 mecánico 100 000 con carga normal		
Resistencia de contacto	máx. 200 mW (nuevo)		
Aislamiento			
bobina a contacto	AC 1500 V, 1 minuto		
contacto a contacto (entre contactos abiertos)	AC 750 V, 1 minuto		
Protección contra cortocircuitos	ninguna		

<sup>1</sup> En la CPU están previstas 8 entradas y 8 salidas en la imagen del proceso para este módulo.

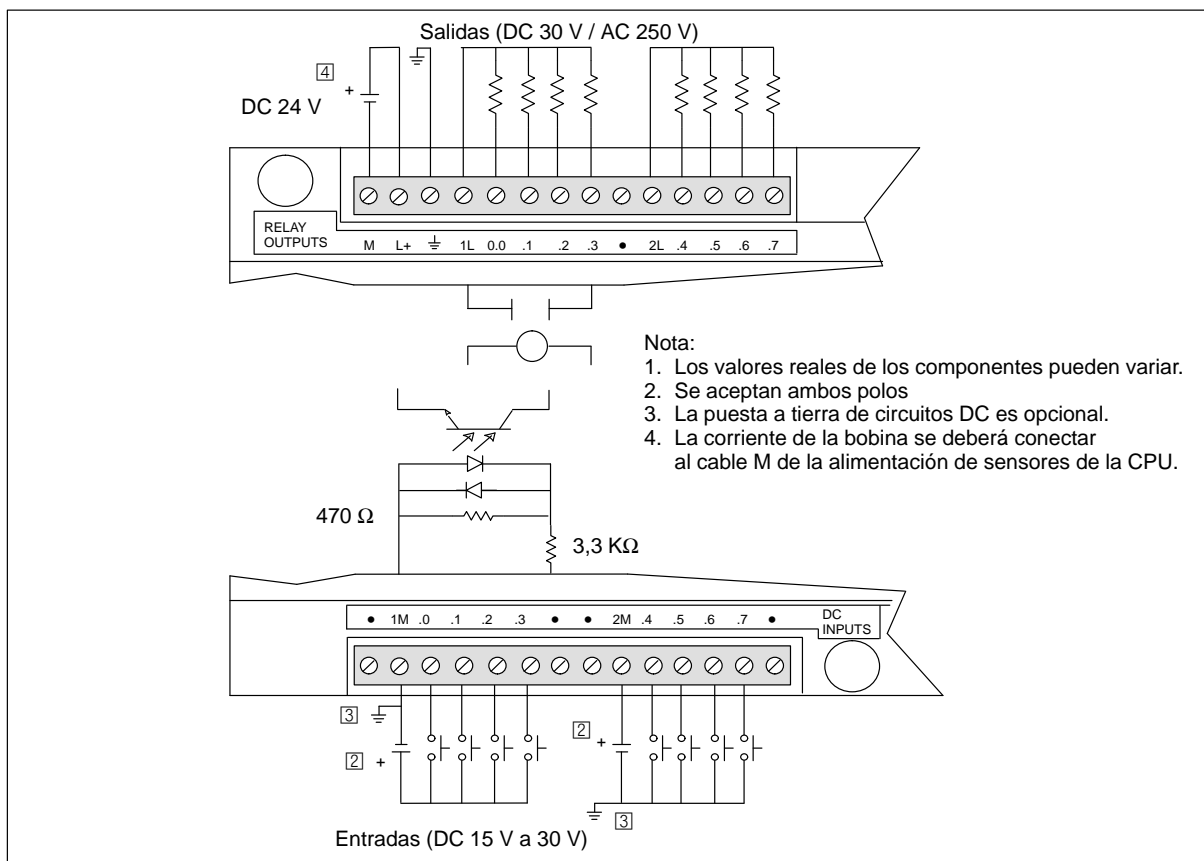


Figura A-31 Identificación de terminales de conexión para el EM223, 8 entradas digitales DC 24 V / 8 salidas digitales de relé

## A.32 Módulo de ampliación EM 223, 16 entradas digitales DC 24 V / 16 salidas digitales de relé

Nº de referencia: 6ES7 223-1PL00-0XA0

Características generales		Entradas	
Dimensiones (l x a x p)	160 x 80 x 62 mm	Tipo de entrada	fuelle Tipo 1 con sumidero de corriente según IEC 1131-2
Peso	0,45 kg	Área en ON	DC 15 V a 30 V, mín. 4 mA DC 35 V, 500 ms sobretensión transitoria
Disipación	7 W	Tensión nominal en ON	DC 24 V, 7 mA
E/S <sup>1</sup>	16 entradas digitales 16 salidas digitales de relé	Máximo en OFF	DC 5 V, 1 mA
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según VDE 0160 según CE	Tiempo de respuesta	típ. 3,5 ms / máx. 4,5 ms
Salidas		Separación galvánica	AC 500 V, 1 minuto
Tipo de salida	Relé, contacto de baja potencia	Consumo	
Margen de tensión	DC 5 V a 30 V / AC 250 V	Corriente disponible DC 5 V	160 mA del aparato central
Corriente de carga máxima	2 A/salida, 8 A/hilo neutro	Corriente de sensor DC 24 V	120 mA del aparato central o de la fuente de alimentación externa
Resistencia de aislamiento	máx. 100 MΩ (nuevo)	Corriente de bobina DC 24 V <sup>2</sup>	130 mA del aparato central o de la fuente de alimentación externa
Retardo de conmutación	máx. 10 ms	Corriente para salidas	Suministrada por el usuario a través del hilo neutro del módulo
Vida útil	10 000 000 mecánico 100 000 con carga normal		
Resistencia de contacto	máx. 200 mW (nuevo)		
Aislamiento			
bobina a contacto	AC 1500 V, 1 minuto		
contacto a contacto	AC 750 V, 1 minuto		
(entre contactos abiertos)			
Protección contra cortocircuitos	ninguna		

<sup>1</sup> En la CPU están previstas 16 entradas y 16 salidas en la imagen del proceso para este módulo.

<sup>2</sup> La corriente de la bobina se deberá conectar al cable M de la alimentación de sensores de la CPU.

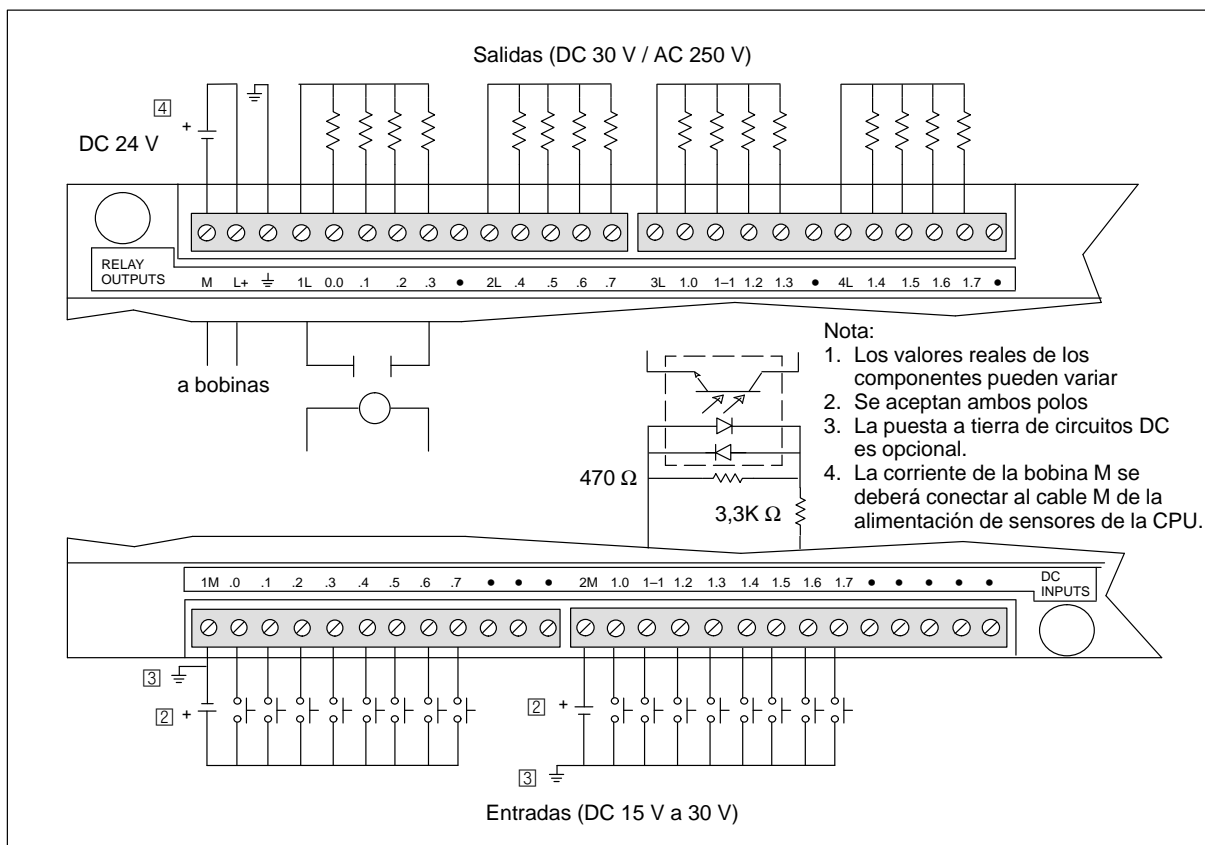


Figura A-32 Identificación de terminales de conexión para el EM223, 16 entradas digitales DC 24 V / 16 salidas digitales de relé

A.33 Módulo de ampliación EM 231, 3 entradas analógicas AI de 12 bits

Nº de referencia: 6ES7 231-0HC00-0XA0

Características generales		Entradas (continuación)	
Dimensiones (l x a x p)	90 x 80 x 62 mm	Tiempo de conversión analógica/digital	< 250 µs
Peso	0,2 kg	Respuesta de salto analógica	1,5 ms a 95%
Disipación	2 W	Rechazo de modo común	40 dB, DC a 60 Hz
E/S <sup>1</sup>	3 entradas analógicas	Tensión en modo común	Tensión de señal más tensión en modo común menor o igual a 12 V
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según VDE 0160 según CE	Formato de palabra de datos <sup>2</sup> Unipolar, margen total	0 a 32000
Entradas		Consumo	
Tipo de entrada	Diferencial	Corriente disponible DC 5 V	70 mA del aparato central
Impedancia de entrada	≥ 10 MW	Alimentación externa	60 mA del aparato central o de una fuente de alimentación externa (DC 24 V tensión nominal, clase 2 o alimentación por sensor DC)
Atenuación filtro de entrada	-3 db @ 3,1 kHz	Indicador LED, EXTf	
Tensión de entrada máxima	30 V	Fallo de tensión	
Corriente de entrada máxima	32 mA	Baja tensión, en DC 24 V externa	
Definición	Convertidor A/D de 12 bits		
Aislamiento	no		

<sup>1</sup> En la CPU están previstas 4 entradas analógicas para este módulo.  
<sup>2</sup> Incrementos de palabra de datos en pasos de 8, valores justificados a la izquierda (v. fig. A-35).

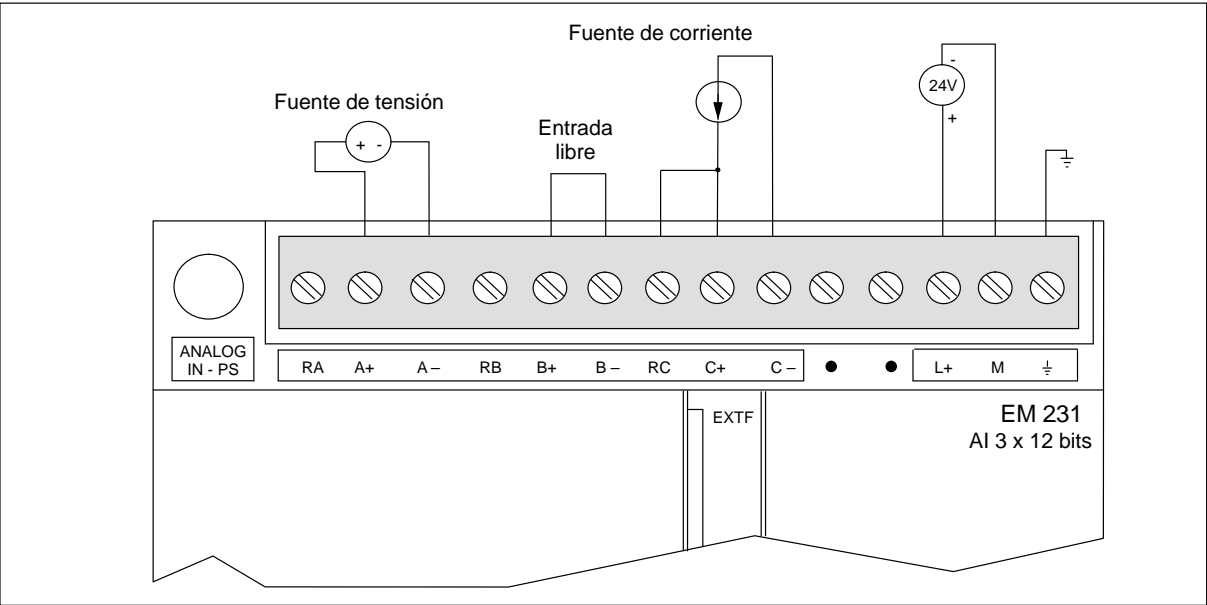


Figura A-33 Identificación de terminales de conexión para el EM231, 3 entradas analógicas AI de 12 bits

Calibración y configuración

Al potenciómetro de calibración y a los interruptores DIP de configuración se puede acceder a través de las rejillas de ventilación del módulo, como muestra la figura A-34.

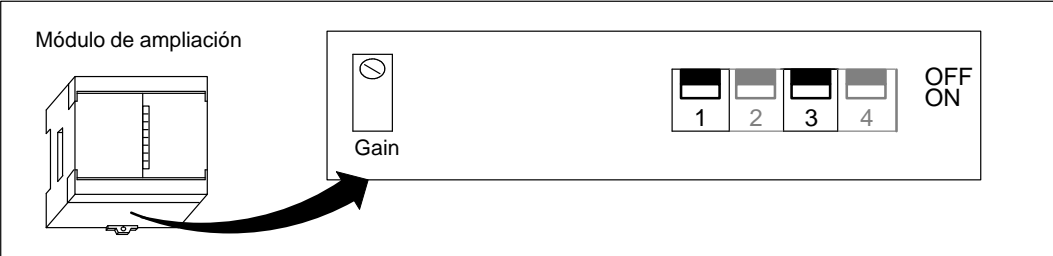


Figura A-34 Potenciómetro de calibración e interruptores DIP de configuración

Configuración

La tabla A-2 muestra cómo configurar el módulo utilizando los interruptores DIP. El margen de las entradas analógicas se selecciona con los interruptores 1 y 3. Todas las entradas analógicas se activan en un mismo margen.

Tabla A-2 Tabla de interruptores DIP de configuración del EM231, 3 entradas analógicas

Interruptor DIP		Margen de tensión	Definición
1	3		
ON	OFF	0 a 5 V	1,25 mV
ON	OFF	0 a 20 mA <sup>1</sup>	5 µA
OFF	ON	0 a 10 V	2,5 mV

<sup>1</sup> 0 a 20 mA según medición con una resistencia interna de 250 Ω conectada en el sentido de la corriente.

Calibrar entradas

- Al calibrar un módulo sólo se pueden corregir los errores de ganancia del margen máximo. Los errores de desplazamiento no se compensan. La calibración afecta a los tres canales de entrada. Después de la calibración puede suceder que difieran los valores leídos entre los canales.
- Con objeto de calibrar el módulo correctamente es preciso utilizar un programa diseñado para crear un promedio de los valores leídos del módulo. Utilice el Asistente para filtrar entradas analógicas de STEP 7-Micro/WIN con objeto de crear dicho programa (v. apt. 5.3). Use 64 o más muestreos para calcular el valor promedio.
- Para calibrar una entrada:
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. La lectura de un valor de cero indica la magnitud del error de desplazamiento. Este error no se puede corregir mediante la calibración.
  5. Aplique una señal de margen máximo a una entrada. Lea el valor que ha recibido la CPU.
  6. Con el potenciómetro GAIN, seleccione el valor 32.000 u otro valor digital deseado.

Formato de la palabra de datos

La figura A-35 muestra la disposición del valor de datos de 12 bits dentro de la palabra de entrada analógica de la CPU.

Si la repetibilidad diverge en sólo  $\pm 0,45\%$  del margen máximo puede producirse una variación de  $\pm 144$  contajes en el valor leído de la entrada analógica.

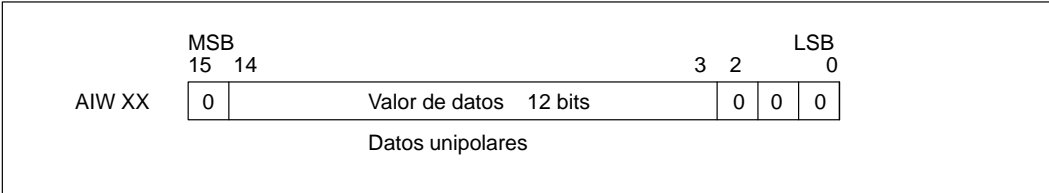


Figura A-35 Formato de la palabra de datos

**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. Los tres ceros a la derecha modifican el valor de la palabra de datos en incrementos de 8 por cada cambio del valor ADC.

### Esquema de conexiones de las entradas

La figura A-36 muestra el esquema de conexiones de las entradas del módulo de ampliación EM231.

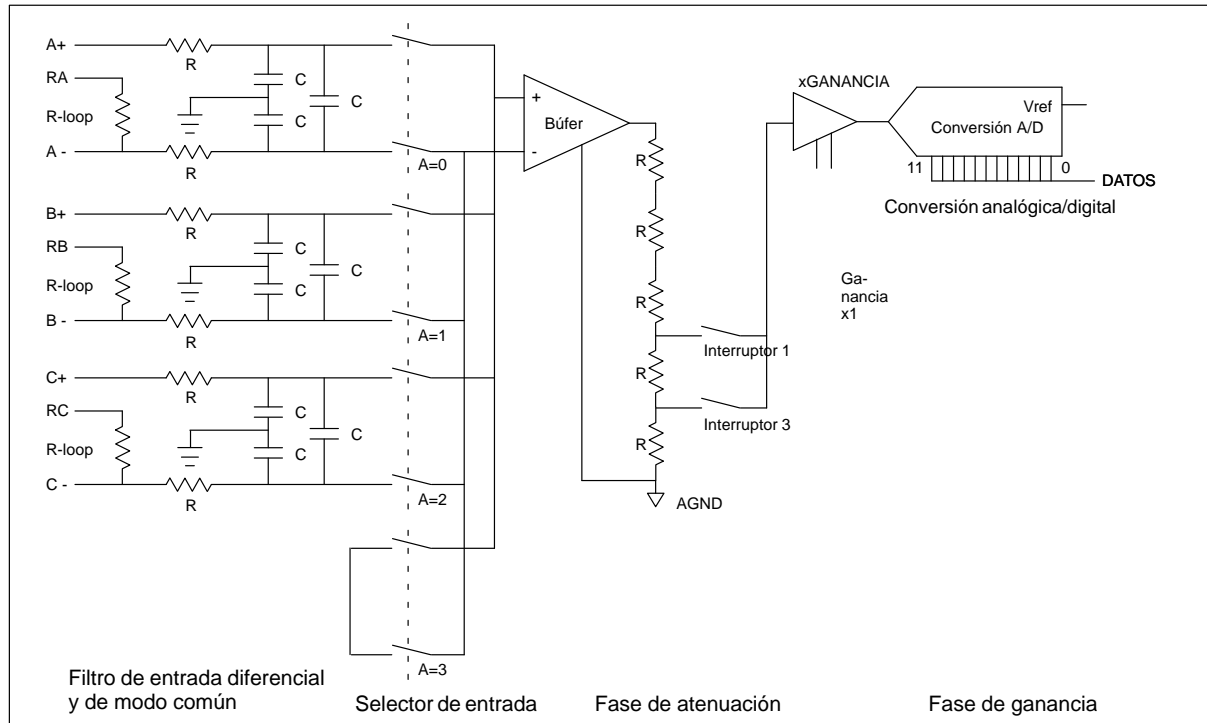


Figura A-36 Esquema de conexiones de las entradas del EM 231

### Reglas de instalación del módulo de ampliación EM231

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.
- Calibre el módulo.
- 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-33.
- Evite doblar excesivamente los cables.
- Conduzca los cables a través de canales.
- Verifique que las señales de entrada estén exentas de potencial o sean líneas de referencia del neutro externo de 24 V del módulo analógico.

### Descripción y uso del módulo de entradas analógicas: precisión y repetibilidad

El EM231 es un módulo de entradas analógicas de 12 bits rápido y económico. Dicho módulo puede convertir una entrada analógica en su correspondiente valor digital en 171  $\mu$ s para la CPU 212 y en 139  $\mu$ s para las demás CPUs S7-200. 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.

El EM231 proporciona 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 un módulo rápido, 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.

Utilice el Asistente para filtrar entradas analógicas de STEP 7-Micro/WIN (v. apt. 5.3) para agregar a su programa una rutina que calcule dicho promedio. Recuerde que un valor promedio calculado mediante una cantidad elevada de muestreos estabilizará el resultado, ralentizando simultáneamente su respuesta a cambios en la señal de entrada. En el caso de señales de entrada analógicas que cambien lentamente se recomiendan 64 o más muestreos para crear el promedio.

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-37. Dicha 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-3 se indican los datos relativos a la repetibilidad y la precisión media con respecto a los márgenes configurables.

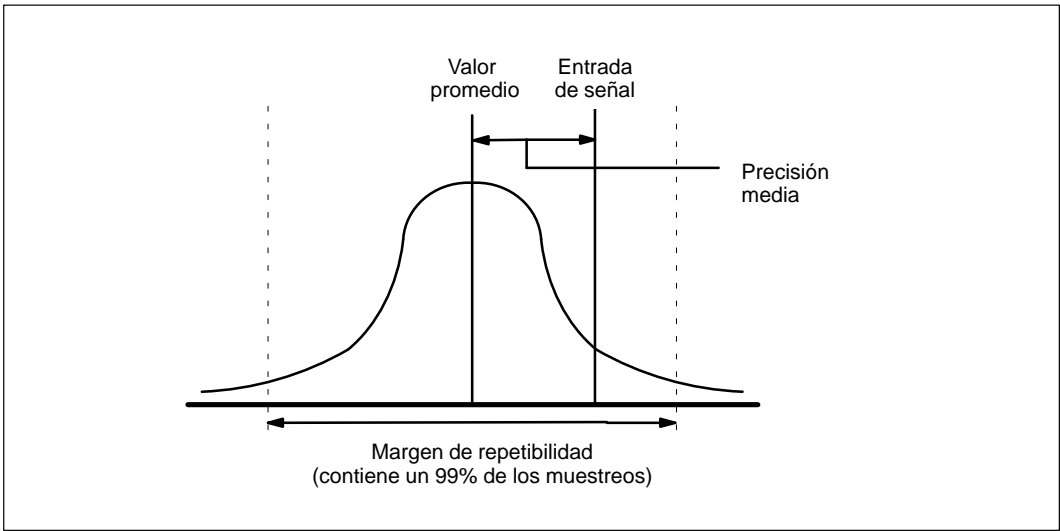


Figura A-37 Definición de la precisión

Tabla A-3 Repetibilidad y precisión media de las CPUs S7-200 con alimentación DC y AC

Margen máximo	Repetibilidad <sup>1</sup>		Precisión media <sup>1, 2, 3, 4</sup>	
	% del margen máximo	Contajes	% del margen máximo	Contajes
CPUs S7-200 con alimentación DC				
0 a 5 V	± 0.075%	± 24	± 0.1%	± 32
0 a 20 mA				
0 a 10 V				
CPUs S7-200 con alimentación AC				
0 a 5 V	± 0.15%	± 48	± 0.1%	± 64
0 a 20 mA				
0 a 10 V				

<sup>1</sup> Mediciones realizadas después de haber calibrado el margen de entrada seleccionado.  
<sup>2</sup> 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.  
<sup>3</sup> 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.  
<sup>4</sup> La precisión media incluye los efectos de la falta de linealidad y de la deriva de 0 a 55° C.

## A.34 Módulo de ampliación EM 232, 2 salidas analógicas AQ de 12 bits

Nº de referencia: 6ES7 232-0HB00-0XA0

Características generales		Precisión	
Dimensiones (l x a x p)	90 x 80 x 62 mm	Caso más desfavorable, 0 a 55° C	
Peso	0,2 kg	Salida de tensión	± 2% del margen máximo
Disipación	2 W	Salida de corriente	+ 2% del margen máximo
E/S <sup>1</sup>	2 salidas analógicas	Típ. 25° C	
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según VDE 0160 según CE	Salida de tensión	± 0,5% del margen máximo
Salidas		Salida de corriente	+ 0,5% del margen máximo
Margen de señal		Tiempo de estabilización	
Salida de tensión	± 10 V	Salida de tensión	100 µs
Salida de corriente	0 a 20 mA	Salida de corriente	2 ms
Resolución, margen máximo		Excitación máxima con una alimentación de 24 V	
Tensión	12 bits	Salida de tensión	mín. 5000 Ω
Corriente	11 bits	Salida de corriente	máx. 500 Ω
Resolución, margen máximo		Consumo	
Tensión, bipolar	1 en 2000 impulsos de conteo, 0,5% del margen máximo	Corriente disponible DC 5 V	70 mA del aparato central
Corriente, unipolar	1 en 2000 impulsos de conteo, 0,5% del margen máximo	Alimentación externa	60 mA, más 40 mA de corriente de salida suministrada por el aparato central o por una fuente de corriente externa (DC 24 V de tensión nominal, clase 2 o alimentación por sensor DC)
Formato palabra de datos		Indicador LED, EXTF	
Margen máximo		Error de alimentación	Baja tensión, fuera del margen permitido
Tensión, bipolar	-32768 a + 32752		
Corriente, unipolar	0 a +32752		
Margen máximo			
Bipolar	-32000 a +32000		
Unipolar	0 a + 32000		

<sup>1</sup> En la CPU están previstas 2 salidas analógicas para este módulo.

La figura A-38 muestra la identificación de terminales de conexión para el módulo de ampliación EM232 que incorpora 2 salidas analógicas de 12 bits.

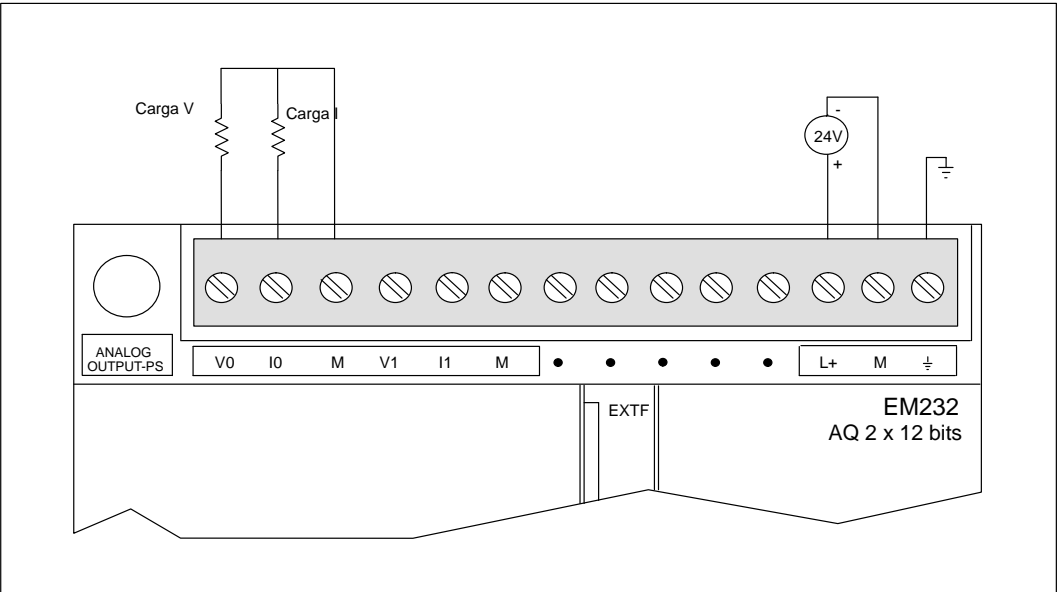


Figura A-38 Identificación de terminales de conexión para el módulo de ampliación EM232, 2 entradas analógicas AQ de 12 bits

Formato de la palabra de datos de salida

La figura A-39 muestra la disposición del valor de 12 bits dentro de la palabra de salida analógica de la CPU.

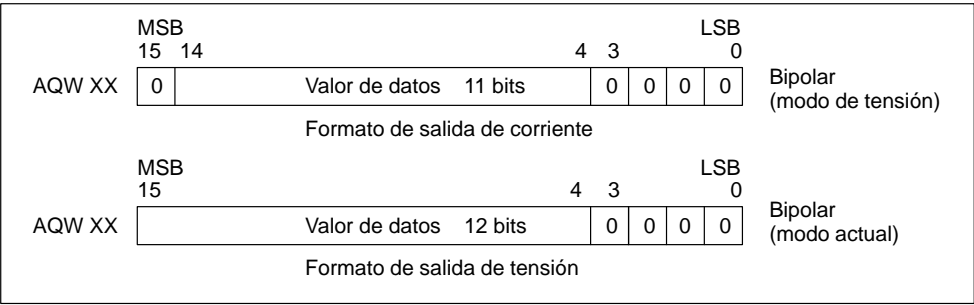


Figura A-39 Formato de la palabra de datos de salida

**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.

### Esquema de conexiones de las salidas

La figura A-40 muestra el esquema de conexiones de las salidas del EM232.

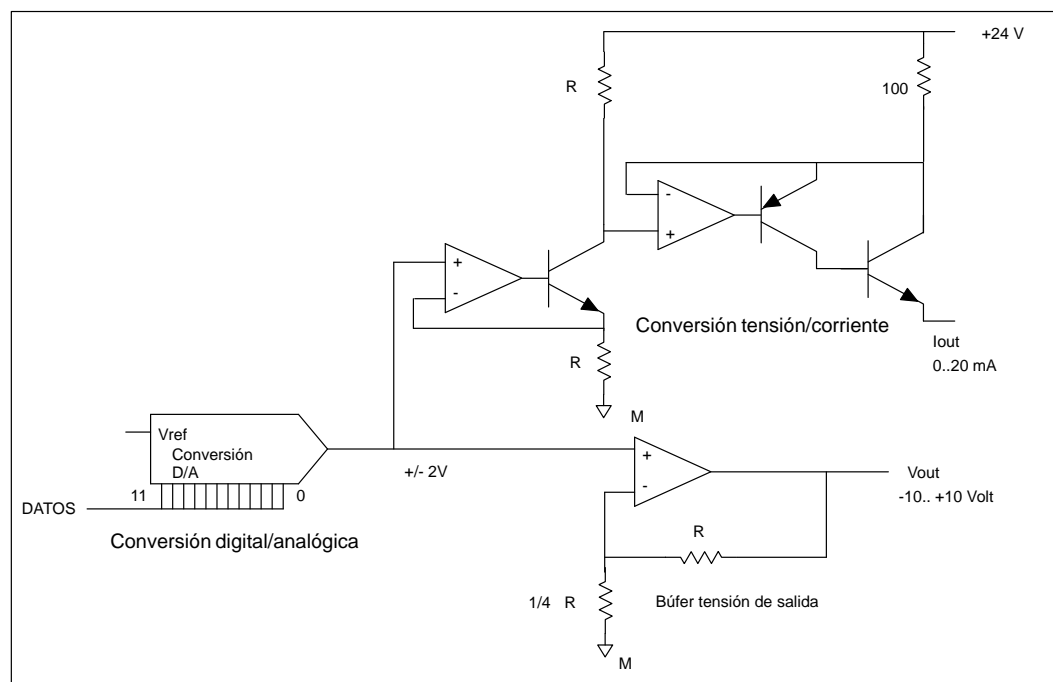


Figura A-40 Esquema de conexiones de las salidas del EM 232

### Reglas de instalación del módulo EM 232

Tenga en cuenta las siguientes reglas para asegurar la precisión:

- 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 la alimentación de sensores.
- 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.

### Definir los datos analógicos

- Precisión: desviación del valor previsto en una E/S determinada.
- Resolución: efecto de un cambio de LSB reflejado en la salida.

## A.35 Módulo de ampliación EM 235, 3 entradas analógicas AI / 1 salida analógica AQ de 12 bits

Nº de referencia: 6ES7 235-0KD00-0XA0

Características generales		Entradas	
Dimensiones (l x a x p)	90 x 80 x 62 mm	Tipo de entrada	Diferencial
Peso	0,2 kg	Impedancia de entrada	$\geq 10 \text{ MW}$
Disipación	2 W	Atenuación filtro de entrada	-3db @ 3.1 kHz
E/S <sup>1</sup>	3 entradas analógicas 1 salida analógica	Tensión de entrada máxima	30 V
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según VDE 0160 según CE	Corriente de entrada máxima	32 mA
<b>Salidas</b>		Definición	Convertidor A/D de 12 bits
Margen de señal		Aislamiento	no
Salida de tensión	$\pm 10 \text{ V}$	Tiempo de conversión analógica/digital	$< 250 \text{ } \mu\text{s}$
Salida de corriente	0 a 20 mA	Respuesta de salto analógica	1,5 ms a 95%
Resolución, margen máximo		Tensión en modo común	Tensión de señal más tensión en modo común menor o igual a 12 V
Tensión	12 bits	Rechazo de modo común	40 dB, DC a 60 Hz
Corriente	11 bits	Formato de la palabra de datos <sup>2</sup>	
Formato palabra de datos <sup>2</sup>		Margen bipolar <sup>3</sup>	-32000 a +32000
Margen bipolar <sup>3</sup>	-32000 a +32000	Margen unipolar <sup>2</sup>	0 a + 32000
Margen unipolar <sup>2</sup>	0 a + 32000	<b>Consumo</b>	
Precisión		Corriente disponible DC 5 V	70 mA del aparato central
En el peor de los casos, 0 a 60° c		Alimentación externa	60 mA más 20 mA de corriente de salida suministrada por el aparato central o por una fuente de corriente externa (DC 24 V de tensión nominal, clase 2 o alimentación por sensor DC)
Salida de tensión	$\pm 2\%$ del margen máximo	<b>Indicador LED, EXTF</b>	
Salida de corriente	+ 2% del margen máximo	Fallo de tensión	Baja tensión, en DC 24 V externa
Típ. 25° c			
Salida de tensión	$\pm 0,5\%$ del margen máximo		
Salida de corriente	+ 0,5% del margen máximo		
Tiempo de estabilización			
Salida de tensión	100 $\mu\text{s}$		
Salida de corriente	2 ms		
Excitación máxima con una alimentación de 24 V			
Salida de tensión	mín. 5000 $\Omega$		
Salida de corriente	máx. 500 $\Omega$		

<sup>1</sup> En la CPU están previstas 4 entradas analógicas y 2 salidas analógicas en la imagen del proceso para este módulo.

<sup>2</sup> Incrementos de la palabra de datos en pasos de 16, valores ADC justificados a la izquierda (v. figuras A-43 y A-45).

<sup>3</sup> Incrementos de la palabra de datos en pasos de 8, valores ADC justificados a la izquierda (v. fig. A-43).

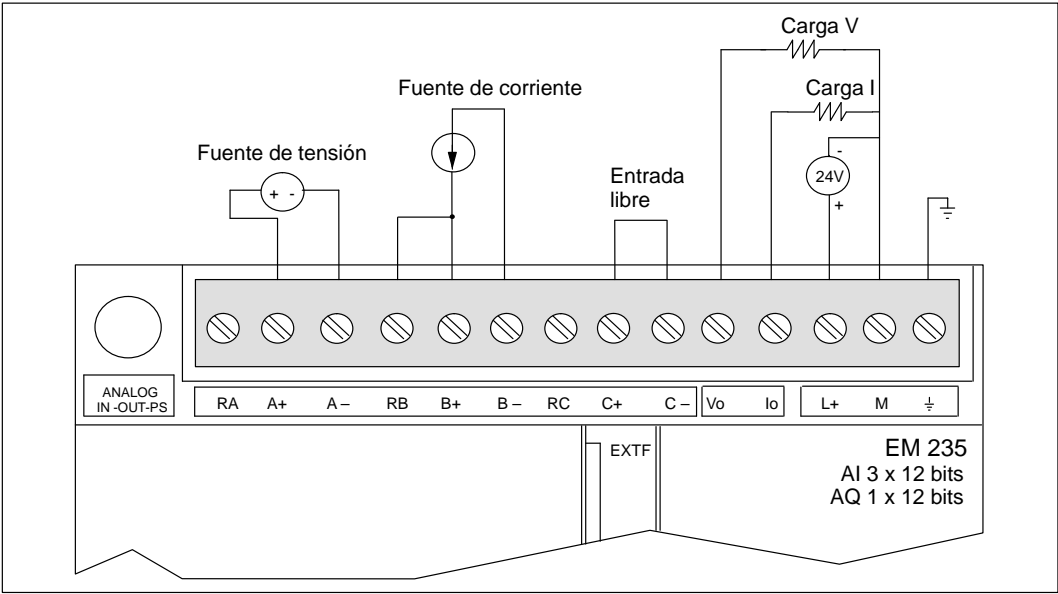


Figura A-41 Identificación de terminales de conexión para el módulo de ampliación EM235 de 3 entradas analógicas AI / 1 salida analógica AQ de 12 bits

Calibración y configuración

A los potenciómetros de calibración y a los interruptores DIP de configuración se puede acceder a través de las rejillas de ventilación del módulo, como muestra la figura A-42.

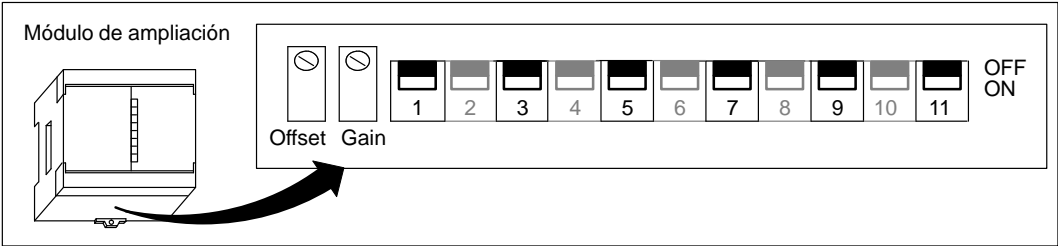


Figura A-42 Potenciómetros de calibración e interruptores DIP de configuración

## Configuración

La tabla A-4 muestra cómo configurar el módulo utilizando los interruptores DIP. El margen de las entradas analógicas se selecciona con los interruptores 1, 3, 5, 7, 9 y 11. Todas las entradas se activan en un mismo margen y formato.

Tabla A-4 Interruptores de configuración para el módulo EM 235

Interruptor de configuración						Margen de tensión	Definición
1 <sup>1</sup>	3	5	7	9	11		
ON	ON	OFF	ON	OFF	OFF	0 a 50 mV	12,5 $\mu$ V
ON	ON	OFF	OFF	ON	OFF	0 a 100 mV	25 $\mu$ V
ON	OFF	ON	ON	OFF	OFF	0 a 500 mV	125 $\mu$ V
ON	OFF	ON	OFF	ON	OFF	0 a 1 V	250 $\mu$ V
ON	OFF	OFF	ON	OFF	OFF	0 a 5 V	1,25 mV
ON	OFF	OFF	ON	OFF	OFF	0 a 20 mA <sup>2</sup>	5 $\mu$ A
ON	OFF	OFF	OFF	ON	OFF	0 a 10 V	2,5 mV
OFF	ON	OFF	ON	OFF	OFF	$\pm 25$ mV	12,5 $\mu$ V
OFF	ON	OFF	OFF	ON	OFF	$\pm 50$ mV	25 $\mu$ V
OFF	ON	OFF	OFF	OFF	ON	$\pm 100$ mV	50 $\mu$ V
OFF	OFF	ON	ON	OFF	OFF	$\pm 250$ mV	125 $\mu$ V
OFF	OFF	ON	OFF	ON	OFF	$\pm 500$ mV	250 $\mu$ V
OFF	OFF	ON	OFF	OFF	ON	$\pm 1$ V	500 $\mu$ V
OFF	OFF	OFF	ON	OFF	OFF	$\pm 2,5$ V	1,25 mV
OFF	OFF	OFF	OFF	ON	OFF	$\pm 5$ V	2,5 mV
OFF	OFF	OFF	OFF	OFF	ON	$\pm 10$ V	5 mV

<sup>1</sup> El interruptor 1 permite la selección de la polaridad: ON para unipolar, OFF para bipolar. Conexión de la CPU a la red necesaria al conmutar entre formato unipolar y formato bipolar. Los interruptores 3, 5, 7, 9 y 11 permiten seleccionar el margen de tensión.

<sup>2</sup> 0 a 20 mA según medición con una resistencia interna de 250 ohmios conectada en el sentido de la corriente.

## Calibración de entradas

La calibración afecta a los tres canales de entrada. Después de la misma puede suceder que difieran los valores leídos entre los canales.

Con objeto de calibrar el módulo correctamente es preciso utilizar un programa diseñado para crear un promedio de los valores leídos del módulo. Utilice el Asistente para filtrar entradas analógicas de STEP 7-Micro/WIN con objeto de crear dicho programa (v. apt. 5.3). Use 64 o más muestreos para calcular el valor promedio.

Para calibrar una entrada:

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).

## Formato de la palabra de datos de entrada

La figura A-43 muestra la disposición del valor de datos de 12 bits dentro de la palabra de entrada analógica de la CPU.

Si la repetibilidad diverge en sólo  $\pm 0,50\%$  del margen máximo puede producirse una variación de  $\pm 160$  contajes sobre el valor leído de la entrada analógica.

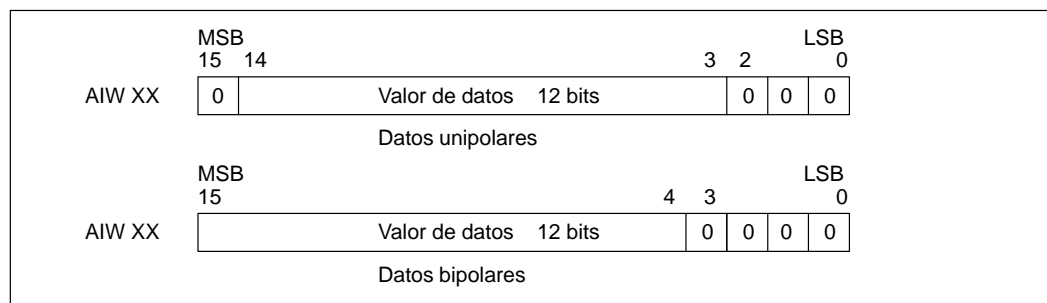


Figura A-43 Formato de la palabra de datos

### 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.

Esquema de conexiones de las entradas

La figura A-44 muestra el esquema de conexiones de las entradas del EM235.

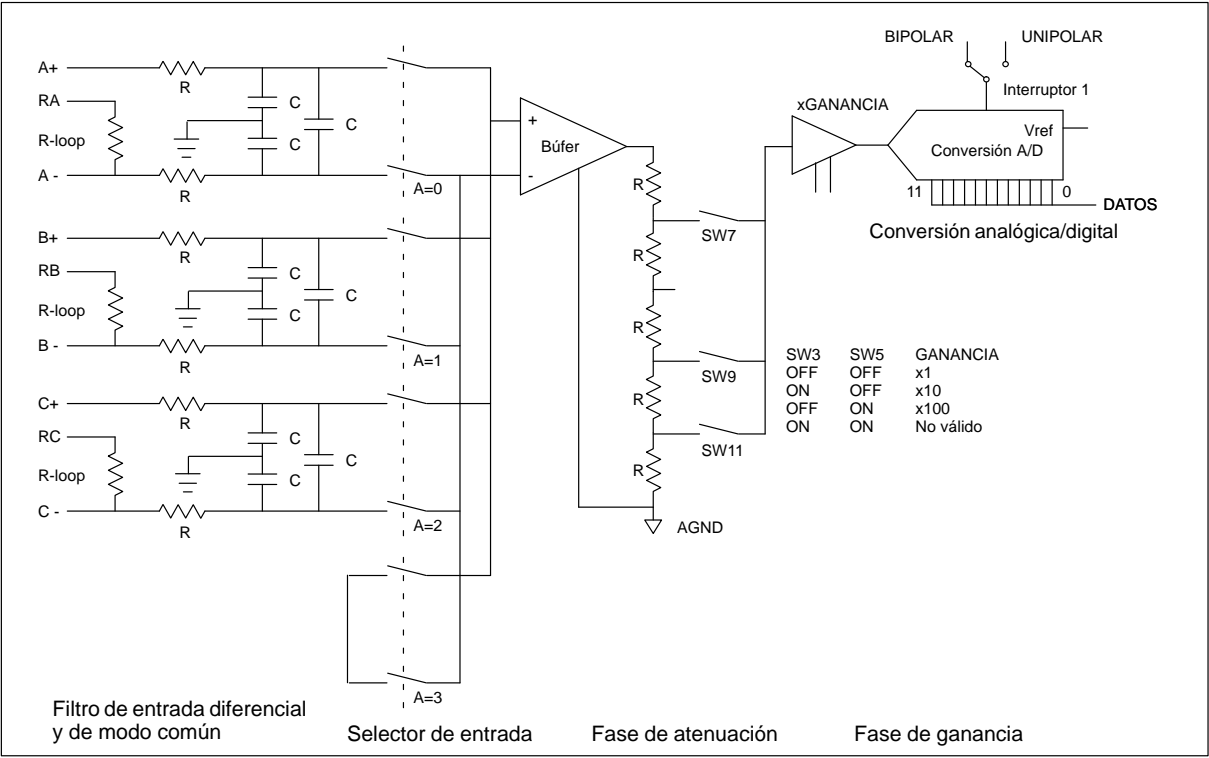


Figura A-44 Esquema de conexiones de las entradas del EM 235

Formato de la palabra de datos de salida

La figura A-45 muestra la disposición del valor de 12 bits dentro de la palabra de salida analógica de la CPU. La figura A-46 muestra el esquema de conexiones de las salidas del EM235.

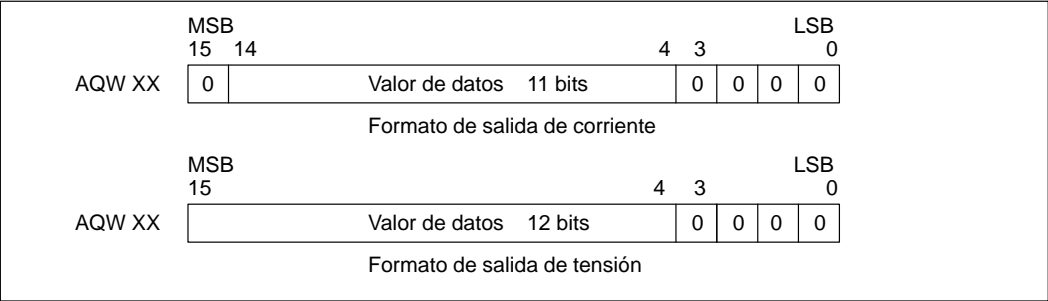


Figura A-45 Formato de la palabra de datos de salida

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.

Esquema de conexiones de las salidas

La figura A-46 muestra el esquema de conexiones de las salidas del EM235.

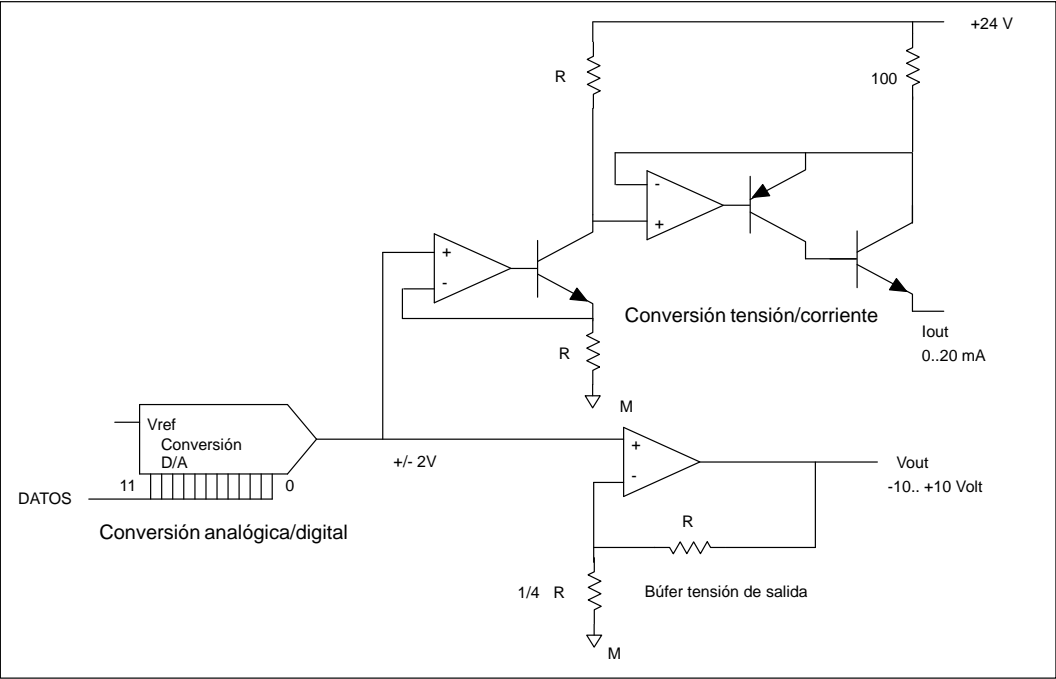


Figura A-46 Esquema de conexiones de las salidas del EM 235

### Reglas de instalación del EM235

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.
- Calibre el módulo.
- 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-41.
- 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 estén exentas de potencial o sean líneas de referencia del neutro externo de 24 V del módulo analógico.

---

#### **Nota**

No es recomendable utilizar este módulo de ampliación con termopares.

---

### Descripción y uso del módulo de entradas analógicas: precisión y repetibilidad

El EM235 es un módulo rápido de E/S analógicas de 12 bits rápido y económico. El módulo puede convertir una entrada analógica en su correspondiente valor digital en 171  $\mu$ s para la CPU 212 y en 139  $\mu$ s para las demás CPUs S7-200. 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.

El EM235 proporciona 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 un módulo rápido, 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.

Utilice el Asistente para filtrar entradas analógicas de STEP 7-Micro/WIN para agregar a su programa una rutina que calcule dicho promedio. Recuerde que un valor promedio calculado mediante una cantidad elevada de muestreos estabilizará el resultado, ralentizando simultáneamente la respuesta a cambios en la señal de entrada. En el caso de señales de entrada analógicas que cambien lentamente se recomiendan 64 o más muestreos para crear el promedio.

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-47. Dicha 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-5 se indican los datos relativos a la repetibilidad y la precisión media con respecto a los márgenes configurables.

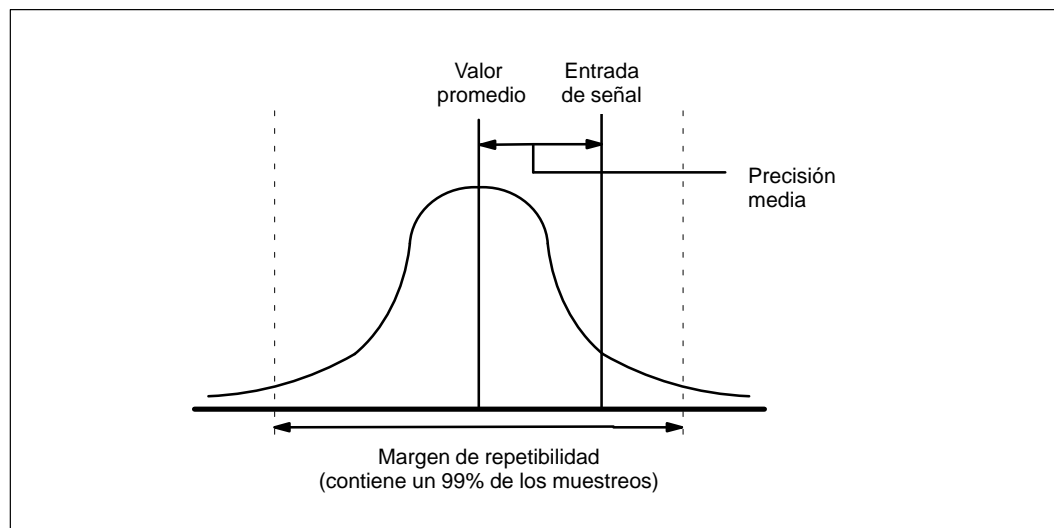


Figura A-47 Definición de la precisión

Tabla A-5 Repetibilidad y precisión media de las CPUs S7-200 con alimentación DC y AC

Margen máximo	Repetibilidad <sup>1</sup>		Precisión media <sup>1, 2, 3, 4</sup>	
	% del margen máximo	Contajes	% del margen máximo	Contajes
CPUs S7-200 con alimentación DC				
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				
±25 mV	± 0.075%	± 48	± 0.25%	± 160
±50 mV			± 0.2%	±128
±100 mV			± 0.1%	± 64
±250 mV			± 0.05%	±32
±500 mV				
±1 V				
±2,5 V				
±5 V				
±10 V				
CPUs S7-200 con alimentación AC				
0 a 50 mV	± 0.15%	± 48	± 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				
±25 mV	± 0.15%	±96	± 0.25%	± 160
±50 mV			± 0.2%	±128
±100 mV			± 0.1%	± 64
±250 mV			± 0.05%	±32
±500 mV				
±1 V				
±2,5 V				
±5 V				
±10 V				

<sup>1</sup> Mediciones realizadas después de haber calibrado el margen de entrada seleccionado.

<sup>2</sup> 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.

<sup>3</sup> 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.

<sup>4</sup> La precisión media incluye los efectos de la falta de linealidad y de la deriva de 0 a 55° C.

A.36 Cartucho de memoria 8K x 8

Nº de referencia: 6ES7 291-8GC00-0XA0

Características generales	
Dimensiones (l x a x p)	28 x 10 x 16 mm
Peso	3,6 g
Disipación	0,5 mW
Tipo de memoria	EEPROM
Memoria de usuario	4096 bytes para programa de usuario + 1024 bytes de datos útiles + datos internos del sistema
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según VDE 0160 según CE

Nota

El cartucho de memoria de 8K se ofrece en versiones de 4 y 5 pines. Dichas versiones son completamente compatibles.

El cartucho de memoria de 8K se puede utilizar con cualquier modelo de CPU S7-200, pero no almacenará el programa de tamaño máximo de la CPU 215 o de la CPU 216. Para evitar problemas en relación con el tamaño del programa, se recomienda utilizar el cartucho de memoria de 8K sólo con la CPU 214 o con la PDS 210.

Los cartuchos de memoria sólo se pueden utilizar para transportar programas entre CPUs de un mismo tipo. (Por ejemplo, un cartucho de memoria programado con una CPU 214 sólo se podrá utilizar en otra CPU 214).

Dimensiones del cartucho de memoria

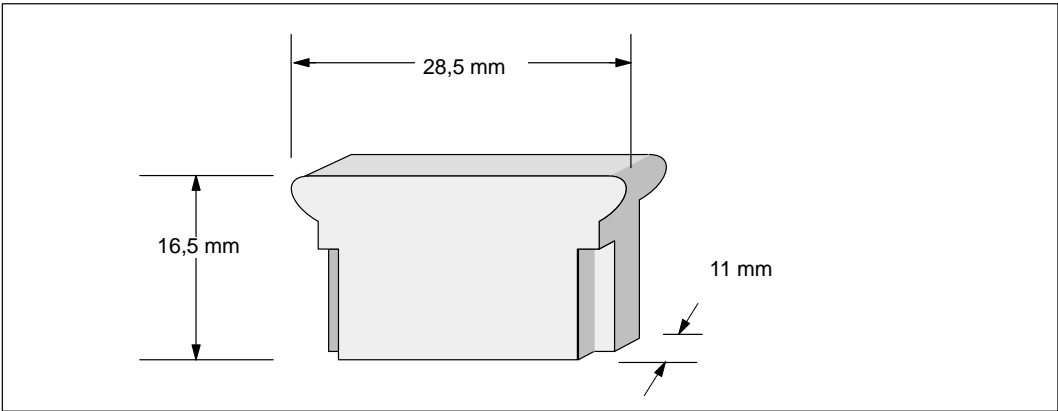


Figura A-48 Dimensiones del cartucho de memoria de 8K x 8

A.37 Cartucho de memoria 16K x 8

Nº de referencia: 6ES7 291-8GD00-0XA0

Características generales	
Dimensiones (l x a x p)	28 x 10 x 16 mm
Peso	3,6 g
Disipación	0,5 mW
Tipo de memoria	EEPROM
Memoria de usuario	8192 bytes para programa de usuario + 5120 bytes de datos útiles + datos internos del sistema
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según VDE 0160 según CE

Nota

El cartucho de memoria de 16K se puede utilizar en la PDS 210, así como en las CPUs 214, 215 y 216.

Los cartuchos de memoria sólo se pueden utilizar para transportar programas entre CPUs de un mismo tipo. (Por ejemplo, un cartucho de memoria programado con una CPU 214 sólo se podrá utilizar en otra CPU 214).

Dimensiones del cartucho de memoria

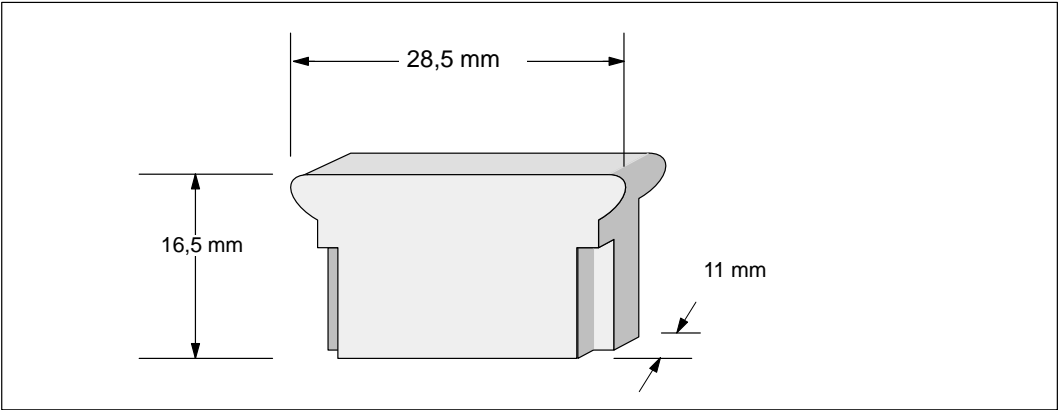


Figura A-49 Dimensiones del cartucho de memoria 16K x 8

A.38 Cartucho de pila

Nº de referencia: 6ES7 291-8BA00-0XA0

Características generales	
Dimensiones (l x a x p)	28 x 10 x 16 mm
Peso	3,6 g
Tamaño	
pila (diám. x alt).Pila	9,9 x 2,5 mm
Tamaño (diámetro x altura)	litio (< 0,6 gramos)
Tipo	10 años
Vida útil de almacenaje	200 días de uso continuo*
Vida útil típica	3 V 30 mA/h (Renata CR 1025)
Sustitución	Se recomienda cambiar la pila en intervalos de 1 año.
Homologaciones	UL 508    CSA C22.2 142 FM clase I, categoría 2 según VDE 0160 según CE
*La pila sólo podrá funcionar después de haberse descargado el condensador de alta potencia de la CPU. Los cortes de corriente de duración inferior al tiempo de retención de datos del condensador de alta potencia no disminuyen la vida útil de la pila.	

Dimensiones del cartucho de pila

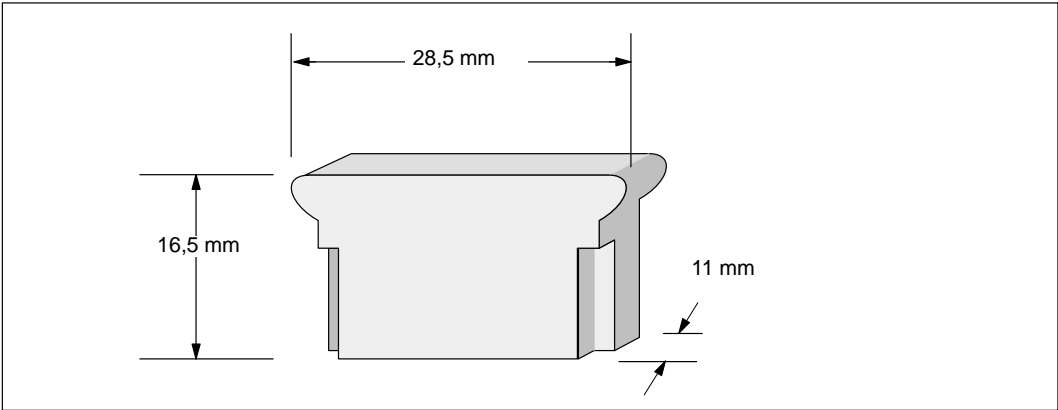


Figura A-50 Dimensiones del cartucho de pila

A.39 Cable de E/S de ampliación

Nº de referencia: 6ES7 290-6BC50-0XA0

Características generales	
Longitud del cable	0,8 m
Peso	0,2 kg
Tipo de conector	Conector para tarjeta

Instalación típica del cable de conexión de E/S de ampliación

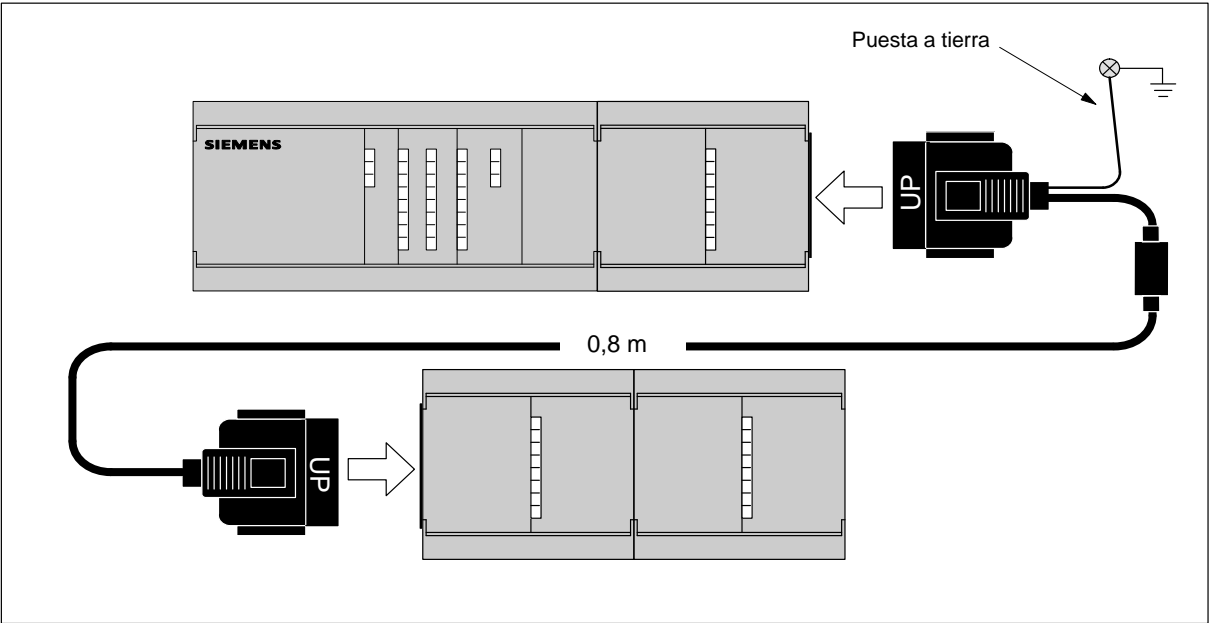


Figura A-51 Instalación típica del cable de conexión de E/S de ampliación



**Cuidado**

Si no se instala correctamente el cable de conexión para los módulos de ampliación, pueden producirse daños en los equipos.

En este caso, a través del cable pueden circular corrientes indeseadas que deteriorarán el módulo de ampliación.

Instalar siempre la conexión de tal manera que, al conectar el módulo de ampliación, la parte superior del cable ("UP") indique hacia arriba (v. fig. A-51).

## A.40 Cable PC/PPI

Nº de referencia: 6ES7 901-3BF00-0XA0

Características generales															
Longitud del cable	5 m														
Peso	0,3 kg														
Disipación	0,5 W														
Tipo de conector PC PLC	subminiatura (serie D) de 9 pines (enchufe) subminiatura (serie D) de 9 pines (pines)														
Tipo de cable	RS232 a RS485, no aislado														
Tiempo de espera recepción/transmisión	Tiempo 2 caracteres														
Velocidad de transferencia asistida (seleccionable mediante interruptores DIP)	<table> <tr> <th colspan="2">Interruptor</th></tr> <tr> <td>38,4 k</td><td>0000</td></tr> <tr> <td>19,2 k</td><td>0010</td></tr> <tr> <td>9,6 k</td><td>0100</td></tr> <tr> <td>2,4 k</td><td>1000</td></tr> <tr> <td>1,2 k</td><td>1010</td></tr> <tr> <td>600</td><td>1100</td></tr> </table>	Interruptor		38,4 k	0000	19,2 k	0010	9,6 k	0100	2,4 k	1000	1,2 k	1010	600	1100
Interruptor															
38,4 k	0000														
19,2 k	0010														
9,6 k	0100														
2,4 k	1000														
1,2 k	1010														
600	1100														
Homologaciones	UL 508 CSA C22.2 142 FM clase I, categoría 2 según VDE 0160 según CE														

Tabla A-6 Asignación de pines del cable PC/PPI

Pin RS-232	Función en el PC	Pin RS-485	Función en la CPU S7-200
2	Datos recibidos (PC recibe)	8	Señal A
3	Datos transmitidos (PC envía)	3	Señal B
5	Masa de señal	7	+24 V
		2	Retorno +24 V (masa lógica PLC)
		1	Blindaje (masa lógica PLC)



### Cuidado

Al interconectar equipos con potenciales de referencia diferentes pueden circular corrientes indeseadas por el cable de enlace.

Dichas 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.

### Dimensiones del cable PC/PPI

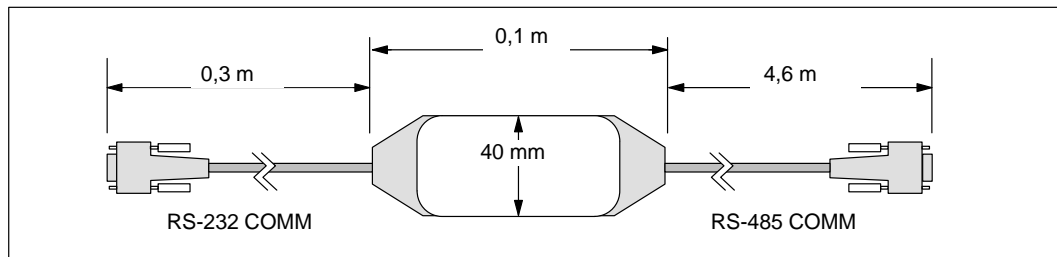


Figura A-52 Dimensiones del cable PC/PPI

A.41 Simulador de entradas DC para la CPU 212

Nº de referencia: 6ES7 274-1XF00-0XA0

Características generales	
Dimensiones (l x a x p)	61 x 36 x 22 mm
Peso	0,02 kg
E/S	8

Instalación

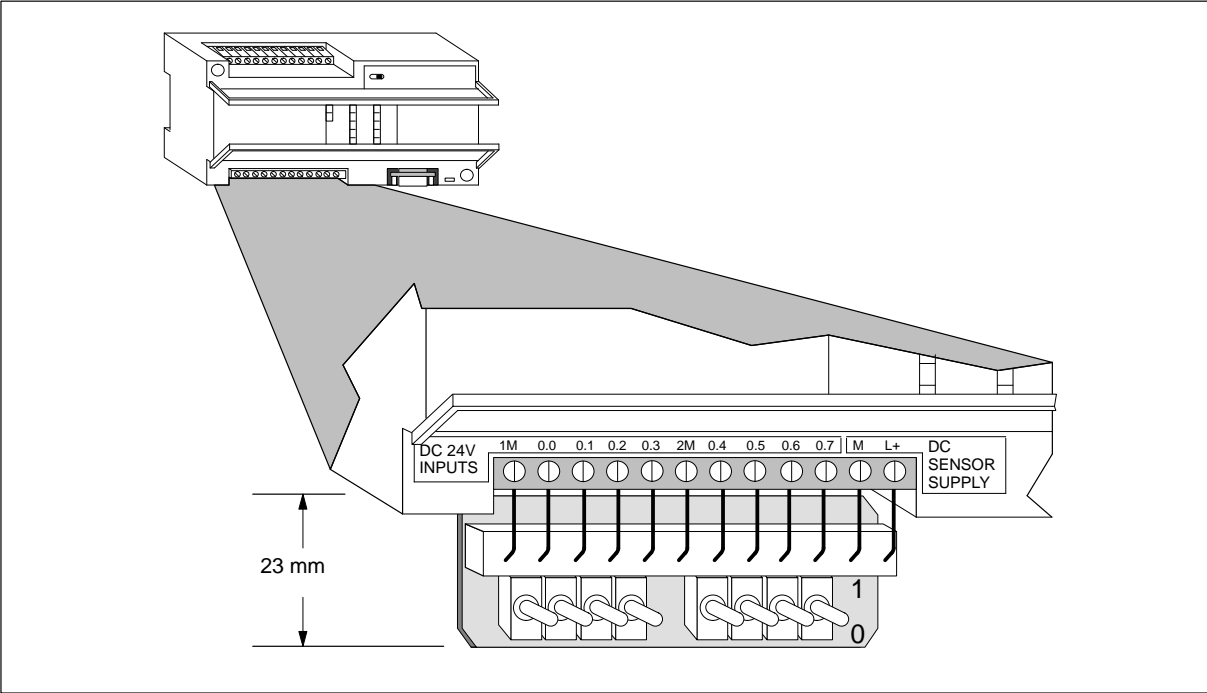


Figura A-53 Instalación del simulador de entradas DC para la CPU 212

A.42 Simulador de entradas DC para la CPU 214

Nº de referencia: 6ES7 274-1XH00-0XA0

Características generales	
Dimensiones (l x a x p)	91 x 36 x 22 mm
Peso	0,03 kg
E/S	14

Instalación

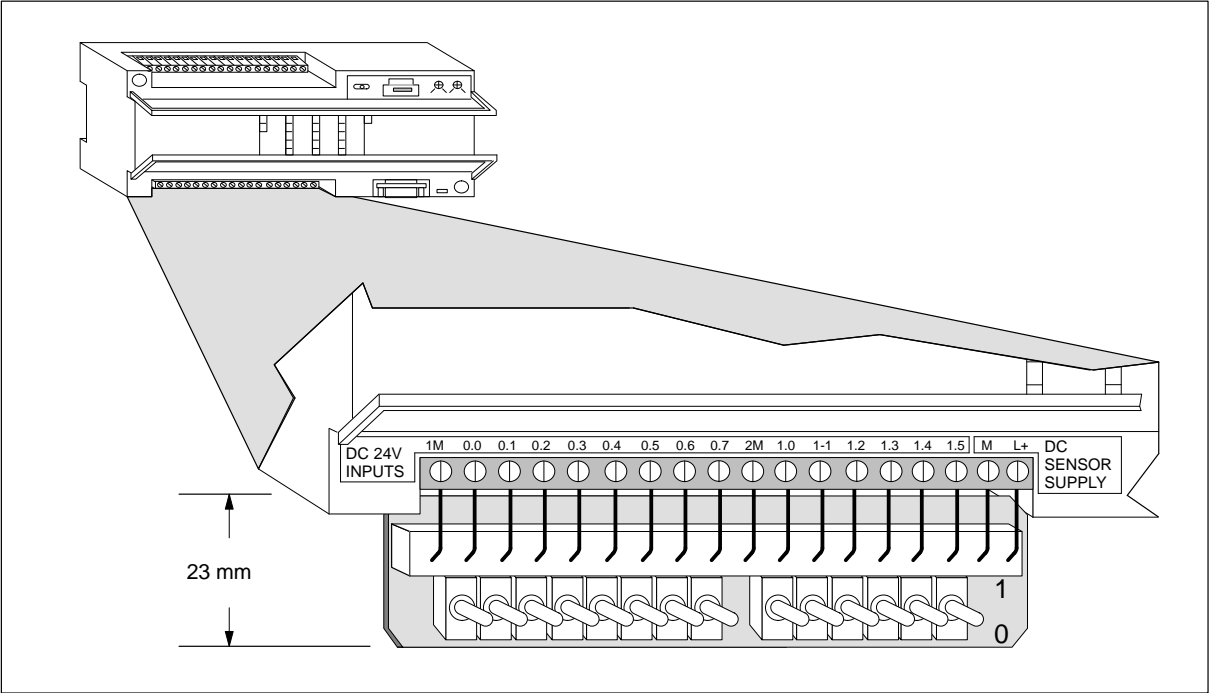


Figura A-54 Instalación del simulador de entradas DC para la CPU 214

A.43 Simulador de entradas DC para las CPUs 215/216

Nº de referencia: 6ES7 274-1XK00-0XA0

Características generales	
Dimensiones (l x a x p)	147 x 36 x 25 mm
Peso	0,04 kg
E/S	24

Instalación

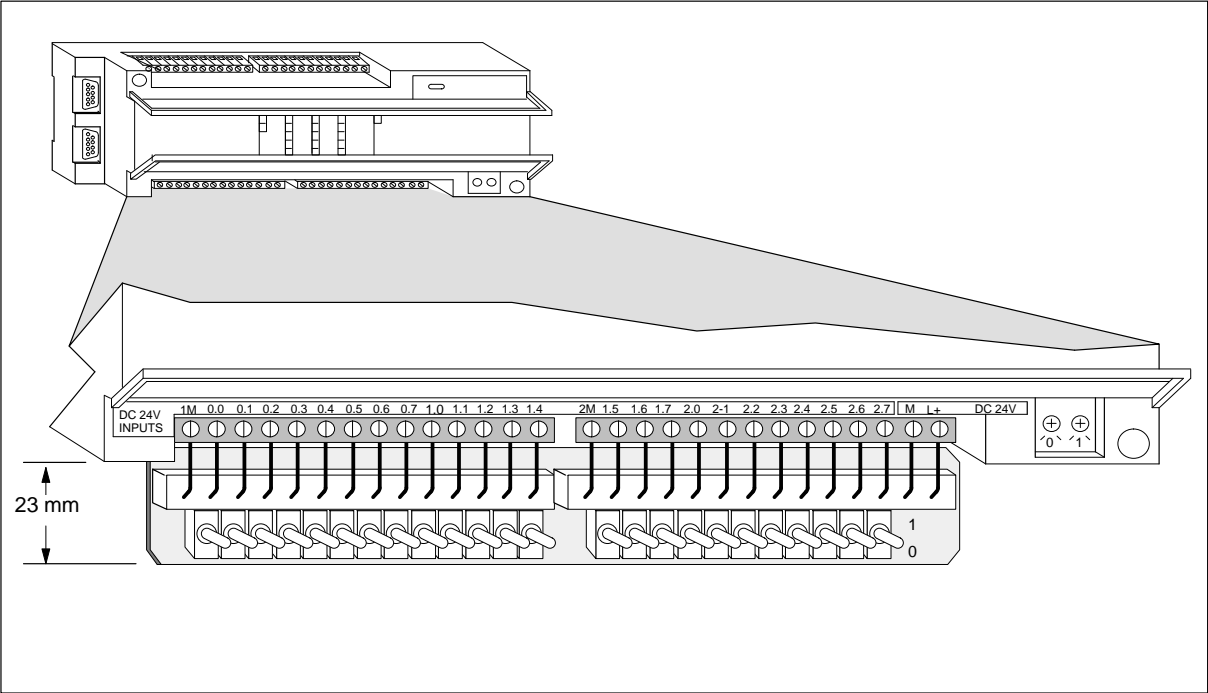


Figura A-55 Instalación del simulador de entradas DC para las CPUs 215/216

# B

## Tabla de consumo de corriente

Cada CPU S7-200 (unidad central) suministra corriente continua de 5 V y 24 V a los módulos de ampliación.

- La corriente continua de 5 V se suministra automáticamente a los módulos de ampliación a través del bus de ampliación.
- Cada CPU incorpora una fuente de alimentación para sensores de DC 24 V para las entradas o las bobinas de relé de los módulos de ampliación. La alimentación de DC 24 V se debe conectar manualmente a dichas entradas o bobinas de relé.

Utilice esta tabla para determinar cuánta corriente puede suministrar la CPU para su configuración. En el Anexo A se indican la alimentación de la CPU y la corriente necesaria para los módulos de ampliación. En el apartado 2.5 se incluye un ejemplo para calcular el consumo de corriente.

Alimentación	DC 5 V	DC 24 V

*menos*

Consumo del sistema	DC 5 V	DC 24 V
	Unidad central	
Consumo total		

*igual a*

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



# Códigos de error

# C

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
C.1	Códigos de errores fatales y mensajes	C-2
C.2	Errores de programación del tiempo de ejecución	C-3
C.3	Violación de reglas de compilación	C-4

## C.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 C-1 muestra una lista con las descripciones de los códigos de errores fatales que se pueden leer de la CPU.

Tabla C-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	Error: valor no válido del contacto de comparación
0013	Cartucho de memoria vacío o programa no apto para esta CPU

## C.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 C-2 muestra una lista con las descripciones de los errores no fatales.

Tabla C-2 Errores de programación del tiempo de ejecución

Código de error	Error de programación del tiempo de ejecución (no fatal)
0000	No se presentó ningún 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 HSC (conflicto)
0003	Entradas asignadas a un HSC que ya está asociado a una interrupción de entrada (conflicto)
0004	Se ha intentado ejecutar una operación ENI, DISI o HDEF en una rutina de interrupción
0005	Antes de finalizar el primer HSC se ha intentado ejecutar un segundo HSC con el mismo número (HSC de la rutina de interrupción en conflicto con HSC del programa principal)
0006	Error de direccionamiento indirecto
0007	Error en datos para operación TODW (Escribir en reloj de tiempo real)
0008	Excedida la profundidad máxima de anidado para subrutina de usuario
0009	Ejecución de una operación XMT o RCV mientras se está ejecutando otra operación XMT o RCV
000A	Se ha intentado redefinir un HSC ejecutando otra operación HDEF para el mismo HSC
0091	Error de margen (con información sobre direcciones): verificar las áreas de operandos
0092	Error en el campo de conteo de una operación (con información sobre el conteo): verificar el valor máximo de conteo
0094	Error de margen al escribir en la memoria no volátil (con información sobre direcciones)

### C.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 C-3 se describen los códigos de error generados al violarse las reglas de compilación.

Tabla C-3 Violación de reglas de compilación

Código de error	Error de compilación (no fatal)
0080	Programa demasiado extenso para la compilación: reducir 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 conteo de una operación (con información sobre el conteo): verificar el valor máximo de conteo
0093	Excedida la profundidad de anidado FOR/NEXT
0095	Falta la operación LSCR (cargar SCR)
0096	Falta la operación SCORE (fin de SCR) u operación no admisible antes de la operación SCORE

## 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 D-1, SMB0 contiene ocho bits de estado que la CPU S7-200 actualiza al final de cada ciclo.

Tabla D-1 Byte de marcas SMB0 (SM0.0 a SM0.7)

Bits de marcas	Descripción
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 D-2, SMB1 contiene varios indicadores potenciales de error. Estos bits se activan y se desactivan durante el tiempo de ejecución.

Tabla D-2 Byte de marcas SMB1 (SM1.0 a SM1.7)

Bits de marcas	Descripción
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 D-3, cada carácter recibido en dicho modo se deposita en este búfer, accesible fácilmente desde el programa KOP

Tabla D-3 Byte de marcas SMB2

Byte de marcas	Descripción
SMB2	Este byte contiene todos los caracteres recibidos de los interfaces 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 D-4, SM3.0 se activa si se detecta un error de paridad. Utilice esta marca para rechazar el mensaje.

Tabla D-4 Byte de marcas SMB3 (SM3.0 a SM3.7)

Bits de marcas	Descripción
SM3.0	Error de paridad del interface 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 D-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 D-5 Byte de marcas SMB4 (SM4.0 a SM4.7)

Bits de marcas	Descripción
SM4.0 <sup>1</sup>	Este bit se activa si se desborda la cola de espera para las interrupciones de comunicación
SM4.1 <sup>1</sup>	Este bit se activa si se desborda la cola de espera para las interrupciones de E/S
SM4.2 <sup>1</sup>	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 (interface 0)
SM4.6	Este bit se activa si el transmisor está en vacío (interface 1)
SM4.7	Reservado

<sup>1</sup> Utilice los bits de estado 4.0, 4.1 y 4.2 sólo en rutinas de interrupción. Dichos 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 D-6, SMB5 contiene los bits de estado acerca de las condiciones de error detectadas en las entradas y salidas (E/S). Dichos bits contienen una panorámica de los errores de E/S detectados.

Tabla D-6 Byte de marcas SMB5 (SM5.0 a SM5.7)

Bits de marcas	Descripción
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 a SM5.7	Reservados

**SMB6: Identificador de la CPU**

Como muestra la tabla D-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 su uso futuro.

Tabla D-7 Byte de marcas SMB6

Bits de marcas	Descripción
Formato	<div><div><div>MSB</div><div>7</div><div><div><div>x</div><div>x</div><div>x</div><div>x</div><div>r</div><div>r</div><div>r</div><div>r</div></div><div>0</div><div>LSB</div></div><div>Identificador de la CPU</div></div></div>
SM6.4 a SM6.7	<div><div>xxxx = 0000 = CPU 212</div><div>0010 = CPU 214</div><div>1000 = CPU 215</div><div>1001 = CPU 216</div></div>
SM6.0 a SM6.3	Reservados

**SMB7: Reservado**

SMB7 está reservado para su 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 D-8, el byte de número par de cada pareja de bytes constituye el registro del identificador de módulo. Dicho 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. Dicho byte indica los errores de configuración y de alimentación de las E/S del correspondiente módulo de ampliación.

Tabla D-8 Bytes de marcas SMB8 a SMB21

Byte de marcas	Descripción	
Formato	Byte de número par: ID del módulo	Byte de número impar: Registro de errores del módulo
	<div><div>MSB</div><div>7</div><div>0</div><div>LSB</div><div><div>M</div><div>t</div><div>t</div><div>A</div><div>i</div><div>i</div><div>Q</div><div>Q</div></div></div>	<div><div>MSB</div><div>7</div><div>0</div><div>LSB</div><div>módulo</div><div><div>C</div><div>0</div><div>0</div><div>0</div><div>R</div><div>P</div><div>r</div><div>r</div></div></div>
	M: Módulo presente 0 = presente 1 = no presente	C: Error de configuración
	tt: 00 Módulo de ampliación 01 Reservado 10 Reservado 11 Reservado	R: Área excedida P: Error alimentac. usuario rr: Reservados
A Tipo de E/S 0 = digital 1 = analógico		
ii 00 Sin entradas QQ 00 Sin salidas 01 2 AI u 8 DI 01 2 AQ u 8 DQ 10 4 AI ó 16 DI 10 4 AQ ó 16 DQ 11 8 AI ó 32 DI 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	

Tabla D-8 Bytes de marcas SMB8 a SMB21, continued

Byte de marcas	Descripción
SMB14 SMB15	Identificador del módulo 3 Registro de errores del módulo 3
SMB16 SMB17	Identificador del módulo 4 Registro de errores del módulo 4
SMB18 SMB19	Identificador del módulo 5 Registro de errores del módulo 5
SMB20 SMB21	Identificador del módulo 6 Registro de errores del módulo 6

**SMW22 a SMW26: Tiempos de ciclo**

Como muestra la tabla D-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 D-9 Palabras de marcas SMW22 a SMW26

Palabra de marcas	Descripción
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 D-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 D-10 Bytes de marcas SMB28 y SMB29

Byte de marcas	Descripción
SMB28	Este byte almacena el valor leído del potenciómetro analógico 0. Dicho valor se actualiza una vez por ciclo en STOP/RUN.
SMB29	Este byte almacena el valor leído del potenciómetro analógico 1. Dicho 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 interfaces 0 y 1, respectivamente. SMB30 y SMB130 son marcas de lectura y escritura. Como muestra la tabla D-11, dichos bytes configuran la comunicación Freeport en los respectivos interfaces y permiten seleccionar si se debe asistir el modo Freeport o el protocolo de sistema.

Tabla D-11 Bytes de marcas SMB30

Interface 0	Interface 1	Descripción
Formato de SMB30	Formato de SMB130	<div style="display: flex; justify-content: space-between; align-items: center;"> <div style="text-align: center;"> <small>MSB</small> 7 </div> <div style="text-align: center;"> <small>LSB</small> 0 </div> </div> <div style="display: flex; align-items: center; margin-top: 5px;"> <div style="border: 1px solid black; padding: 2px; margin: 0 2px;">p</div> <div style="border: 1px solid black; padding: 2px; margin: 0 2px;">p</div> <div style="border: 1px solid black; padding: 2px; margin: 0 2px;">d</div> <div style="border: 1px solid black; padding: 2px; margin: 0 2px;">b</div> <div style="border: 1px solid black; padding: 2px; margin: 0 2px;">b</div> <div style="border: 1px solid black; padding: 2px; margin: 0 2px;">b</div> <div style="border: 1px solid black; padding: 2px; margin: 0 2px;">m</div> <div style="border: 1px solid black; padding: 2px; margin: 0 2px;">m</div> </div> <div style="margin-left: 10px;">Byte de control del modo Freeport</div>
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 (para la CPU 212: = 19.200 bits/s) 001 = 19.200 bits/s 010 = 9.600 bits/s 011 = 4.800 bits/s 100 = 2.400 bits/s 101 = 1.200 bits/s 110 = 600 bits/s 111 = 300 bits/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)

**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) bajo el control del 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 dicho 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 D-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 D-12 Byte de marcas SMB31 y palabra de marcas SMW32

Byte de marcas	Descripción
Formato	<div><div>SMB31: Comando de software</div><div><div>MSB</div><div>7</div><div>0</div><div>0</div><div>0</div><div>0</div><div>0</div><div>s</div><div>s</div><div>LSB</div><div>0</div></div></div> <div><div>SMW32: Dirección en la memoria V</div><div><div>MSB</div><div>15</div><div>Dirección en la memoria V</div><div>LSB</div><div>0</div></div></div>
SM31.0 y SM31.1	ss: Tamaño del valor a guardar 00 = bit 01 = byte 10 = palabra 11 = palabra doble
SM30.7	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 memorización.
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: Intervalos de interrupciones temporizadas**

Como muestra la tabla D-13, SMB34 y SMB35 especifican los intervalos de las interrupciones temporizadas 0 y 1, respectivamente. Los valores de dichos intervalos se pueden indicar (en incrementos de 1 ms) de 5 ms a 255 ms. La CPU capta dicho valor una vez que la correspondiente interrupción temporizada se asocia a una rutina de interrupción. Para modificar el intervalo es preciso reasociar la interrupción temporizada a la misma rutina de interrupción, o bien a una diferente. El evento de interrupción temporizada se finaliza desasociando el evento.

Tabla D-13 Bytes de marcas SMB34 y SMB35

Byte de marcas	Descripción
SMB34	Este byte indica el intervalo (en incrementos de 1 ms, de 5 ms a 255 ms) de la interrupción temporizada 0
SMB35	Este byte indica el intervalo (en incrementos de 1 ms, de 5 ms a 255 ms) de la interrupción temporizada 1

**SMB36 a SMB65: Bytes de programación de los contadores rápidos**

Como muestra la tabla D-14, los bytes de marcas SMB36 a SMB65 se utilizan para supervisar y controlar el funcionamiento de los contadores rápidos (HSC).

Tabla D-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 a SM37.2	Reservados
SM37.3	Bit de control del sentido de conteo de HSC0: 1 = contar adelante
SM37.4	Actualizar sentido de conteo de HSC0: 1 = sentido de actualización
SM37.5	Actualizar valor predeterminado de HSC0: 1 = escribir nuevo valor predeterminado en HSC0
SM37.6	Actualizar 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 = contar adelante
SM47.4	Actualizar sentido de conteo de HSC1: 1 = sentido de actualización
SM47.5	Actualizar valor predeterminado de HSC1: 1 = escribir nuevo valor predeterminado en HSC1
SM47.6	Actualizar 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.

Tabla D-14 Bytes de marcas SMB36 a SMB65, continuación

Byte de marcas	Descripción
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 = contar adelante
SM57.4	Actualizar sentido de conteo de HSC2: 1 = sentido de actualización
SM57.5	Actualizar valor predeterminado de HSC2: 1 = escribir nuevo valor predeterminado en HSC2
SM57.6	Actualizar 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 el byte menos significativo
SMB62 SMB63 SMB64 SMB65	Nuevo valor predeterminado de HSC2 SMB62 es el byte más significativo y SMB65 el byte menos significativo

**SMB66 a SMB85: Funciones PTO/PWM**

Como muestra la tabla D-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. Para obtener una descripción detallada de dichas operaciones, consulte el capítulo 10.

Tabla D-15 Bytes de marcas SMB66 a SMB85

Byte de marcas	Descripción
SM66.0 a SM66.5	Reservados
SM66.6	Desbordamiento pipeline PTO0: 0 = sin desbordamiento, 1 = desbordamiento
SM66.7	Bit de PTO0 en vacío: 0 = PTO en ejecución, 1 = PTO en vacío
SM67.0	Actualizar 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 $\mu$ s/ciclo; 1 = 1 ms/ciclo

Tabla D-15 Bytes de marcas SMB66 a SMB85, continuación

Byte de marcas	Descripción
SM67.4 y SM67.5	Reservados
SM67.6	Elegir función de PTO0/PWM0 0 = PTO, 1 = PWM
SM67.7	Bit de habilitación de PTO0/PWM0: 1 = habilitar
SMB68 SMB69	Valor de tiempo de ciclo de PTO0/PWM0 SMB68 es el byte más significativo y SMB69 el byte menos significativo.
SMB70 SMB71	Valor de ancho de impulsos de PWM0 SMB70 es el byte más significativo y SMB71 el byte menos significativo.
SMB72 SMB73 SMB74 SMB75	Valor de conteo de impulsos de PTO0 SMB72 es el byte más significativo y SMB75 el byte menos significativo.
SM76.0 a SM76.5	Reservados
SM76.6	Desbordamiento pipeline de PTO1: 0 = sin desbordamiento, 1 = desbordamiento
SM76.7	Bit de PTO1 en vacío: 0 = PTO en ejecución, 1 = PTO en vacío
SM77.0	Actualizar el valor del tiempo de ciclo de 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 y SM77.5	Reservados
SM77.6	Elegir función de PTO1/PWM1: 0 = PTO, 1 = PWM
SM77.7	Bit de habilitación de PTO1/PWM1: 1 = habilitar
SMB78 SMB79	Valor de tiempo de ciclo de PTO1/PWM1 SMB78 es el byte más significativo y SMB79 el byte menos significativo.
SMB80 SMB81	Valor de ancho de impulsos de PWM1 SMB80 es el byte más significativo y SMB81 el byte menos significativo.
SMB82 SMB83 SMB84 SMB85	Valor de conteo de impulsos de PTO1 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 D-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.

Interface 0	Interface 1	Descripción
SMB86	SMB186	<div><div><div>MSB 7</div><div><div>n</div><div>r</div><div>e</div><div>0</div><div>0</div><div>t</div><div>c</div><div>p</div></div><div>LSB 0</div></div><div>Byte de estado de recepción de mensajes</div></div> <p>n: 1 = El usuario ha inhibido la recepción de mensajes</p> <p>r: 1 = Se finalizó la recepción de mensajes: error en parámetros de entrada o falta condición inicial o final</p> <p>e: 1 = Carácter final recibido</p> <p>t: 1 = Se finalizó la recepción de mensajes: ha transcurrido la temporización</p> <p>c: 1 = Se finalizó la recepción de mensajes: se ha excedido el número máximo de caracteres</p> <p>p: 1 = Se finalizó la recepción de mensajes debido a un error de paridad</p>
SMB87	SMB187	<div><div><div>MSB 7</div><div><div>n</div><div>x</div><div>y</div><div>z</div><div>m</div><div>t</div><div>0</div><div>0</div></div><div>LSB 0</div></div><div>Byte de control de recepción de mensajes</div></div> <p>n: 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.</p> <p>x: 0 = Ignorar SMB88 o SMB188 1 = Utilizar el valor de SMB88 o de SMB188 para detectar el comienzo del mensaje</p> <p>y: 0 = Ignorar SMB89 o SMB189 1 = Utilizar el valor de SMB89 o de SMB189 para detectar el fin del mensaje</p> <p>z: 0 = Ignorar SMW90 o SMB190 1 = Utilizar el valor de SMW90 para detectar una condición de inactividad</p> <p>m: 0 = Utilizar el temporizador como temporizador entre caracteres 1 = Utilizar el temporizador como temporizador de mensajes</p> <p>t: 0 = Ignorar SMW92 o SMW192 1 = Finalizar la recepción si se excede el período de tiempo indicado en SMW92 o SMW192</p> <p>Estos bits definen los criterios para identificar el mensaje (incluyendo los criterios para el comienzo y el fin del mensaje). Para determinar el comienzo de un mensaje, los criterios habilitados a tal efecto se combinan mediante Y, debiendo presentarse en forma de secuencia (línea de inactividad seguida de un carácter inicial). Para determinar el fin de un mensaje, los criterios habilitados a tal efecto se combinan mediante O.</p> <p>Ecuaciones de los criterios para el comienzo y el fin de un mensaje: Comienzo del mensaje = z * x Fin del mensaje = y + t + número máximo de caracteres alcanzados</p> <p>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).</p>
SMB88	SMB188	Carácter de comienzo del mensaje
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 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.

**SMB110 a SMB115: Estado del protocolo estándar DP**

Como muestra la tabla D-17, los bytes de marcas SMB110 a SMB115 se utilizan para supervisar el estado del protocolo estándar DP.

**Nota**

Estas direcciones sólo pueden contener el estado. No efectúe operaciones de escritura en ellas. Las direcciones muestran valores activados por el maestro DP durante el proceso de configuración.

Tabla D-17 Bytes de marcas SMB110 a SMB115

Byte de marcas	Descripción								
SMB110	<div><div><div>MSB 7</div><div>LSB 0</div></div><table><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>s</td><td>s</td></tr></table></div> <div><div>Interface 1: Byte de estado del protocolo estándar DP</div><div><div>ss: Byte de estado del protocolo estándar DP</div><div>00 = Comunicación DP no iniciada desde el arranque</div><div>01 = Error de configuración o de parametrización</div><div>10 = Intercambio de datos activado</div><div>11 = Intercambio de datos desactivado</div></div><div>SM111 a SM115 se actualizan cada vez que la CPU acepta datos de configuración y/o de parametrización. Dichas direcciones se actualizan aun si se detecta un error de configuración y/o de parametrización. Las direcciones se borran en cada arranque de la CPU.</div></div>	0	0	0	0	0	0	s	s
0	0	0	0	0	0	s	s		
SMB111	Este byte indica la dirección del maestro del esclavo (0 a 126)								
SMB112 SMB113	Estos bytes indican la dirección en la memoria V del búfer de salida (desplazamiento de VB0). SMB112 es el byte más significativo y SMB113 es el byte menos significativo.								
SMB114	Este byte indica el número de bytes de los datos de salida								
SMB115	Este byte indica el número de bytes de los datos de entrada								

# Utilizar STEP 7-Micro/WIN con STEP 7 y STEP 7-Micro/DOS

# E

STEP 7-Micro/WIN 32 se puede incorporar en STEP 7, siendo posible ejecutar el software como todas las demás aplicaciones de STEP 7 (p.ej. el editor de símbolos o de programas). Para obtener informaciones más detalladas sobre el software de programación STEP 7, consulte la Ayuda online o el *Manual del usuario SIMATIC STEP 7*.

Asimismo, es posible importar archivos de programas que se hayan creado con STEP 7-Micro/DOS. Dichos archivos se pueden editar y cargar con STEP 7-Micro/WIN. Para obtener informaciones más detalladas sobre STEP 7-Micro/DOS, consulte la Ayuda online o el *Manual del usuario SIMATIC STEP 7-Micro/DOS*.

## Índice del capítulo

Apartado	Descripción	Página
E.1	Utilizar STEP 7-Micro/WIN con STEP 7	E-2
E.2	Importar archivos de STEP 7-Micro/DOS	E-4

## E.1 Utilizar STEP 7-Micro/WIN con STEP 7

STEP 7-Micro/WIN se puede integrar en el software STEP 7 para acceder al programa del sistema de automatización S7-200:

- Offline: Un equipo SIMATIC 200 se puede insertar en un proyecto de STEP 7.
- Online: A la CPU S7-200 se puede acceder mediante la lista de estaciones accesibles.

Cuando el software de programación STEP 7-Micro/WIN está integrado en STEP 7, su apariencia puede ser diferente a la de STEP 7-Micro/WIN como aplicación autónoma:

- Browsers: Si STEP 7-Micro/WIN está integrado en el software STEP 7, se utilizarán los browsers de STEP 7 para navegar a los objetos S7-200 que formen parte de la jerarquía de STEP 7. Sólo se podrá navegar a objetos S7-200 que conformen dicha jerarquía. Entonces no será posible abrir ningún objeto (proyecto, programa, bloque de datos o tabla de estado) almacenado en la jerarquía de STEP 7-Micro/WIN.
- Idioma y nemotécnica: Si STEP 7-Micro/WIN está integrado en el software STEP 7, se utilizarán los ajustes de idioma y nemotécnica ajustados para STEP 7.

### Crear una CPU S7-200 en un proyecto de STEP 7

Para crear una CPU S7-200 en el software de programación STEP 7, es preciso insertar un equipo SIMATIC 200 en un proyecto de STEP 7. STEP 7 creará entonces el equipo SIMATIC 200. A diferencia de los equipos S7-300 y S7-400, no hay otros objetos (p.ej. CPUs o redes) asociados al equipo S7-200. Un solo equipo S7-200 representa un proyecto entero de STEP 7-Micro/WIN que incluye el programa, el bloque de datos, así como las tablas de símbolos y de estado.

El software de programación STEP 7 se puede utilizar para copiar, mover, borrar o cambiar el nombre de un proyecto S7-200.

---

#### Nota

Una CPU S7-200 ("equipo SIMATIC 200") se puede insertar únicamente en la raíz del proyecto STEP 7, mas no debajo de ningún otro objeto. No existe interacción alguna entre el equipo SIMATIC 200 y los demás objetos de STEP 7.

---

Para crear un equipo S7-200:

1. Elija el comando de menú **Archivo ► Nuevo** para crear un nuevo proyecto en el Administrador SIMATIC.
2. Elija el comando de menú **Insertar ► Equipo ► Equipo SIMATIC 200** para crear un objeto S7-200.
3. Para editar el equipo S7-200, haga doble clic en el objeto S7-200 con objeto de abrir el equipo. STEP 7 inicia el software de programación STEP 7-Micro/WIN.

---

#### Nota

Sólo se puede tener abierta una versión del software de programación STEP 7-Micro/WIN a la vez. Si ya está abierto otro proyecto S7-200, es preciso cerrar el primer proyecto antes de abrir el segundo.

---

### Utilizar STEP 7 para editar una CPU S7-200 online

El Administrador SIMATIC incorpora una lista de las estaciones o equipos S7 que se encuentren online en la red ("Estaciones accesibles"). Dicha lista incluye todas las estaciones S7-200 conectadas a la red. Al seleccionarse la estación S7-200 de la lista de estaciones accesibles, STEP 7 inicia el software de programación STEP 7-Micro/WIN. STEP 7-Micro/WIN abre un proyecto vacío (sin título) y carga en la PG o el PC el programa de usuario, así como el bloque de datos y la configuración de la CPU S7-200.

#### Nota

Pueden existir redes diferentes a las que sea posible acceder sólo a través de STEP 7 o sólo mediante STEP 7-Micro/WIN. Si STEP 7-Micro/WIN está integrado en el software STEP 7, la lista de estaciones online en la red muestra sólo los equipos accesibles a través de STEP 7.

### Abrir un proyecto de STEP 7 desde STEP 7-Micro/WIN

Es posible acceder al programa de usuario de un equipo S7-200 almacenado en proyectos de STEP 7, aun cuando STEP 7-Micro/WIN no se ejecute integrado en el software STEP 7. Para editar el programa de usuario:

1. En el software de programación STEP 7-Micro/WIN, elija el comando de menú **Proyecto ► Nuevo** para crear un nuevo proyecto.
2. Elija los comandos de menú **Proyecto ► Importar ► Proyecto STEP 7** como muestra la figura E-1.
3. En el browser de proyectos de STEP 7, elija el equipo S7-200 del proyecto STEP 7 y haga clic en el botón "Abrir".

El programa de usuario y los demás componentes (bloque de datos, tabla de estado y tabla de símbolos) se abrirán en el proyecto de STEP 7-Micro/WIN (v. fig. E-1).

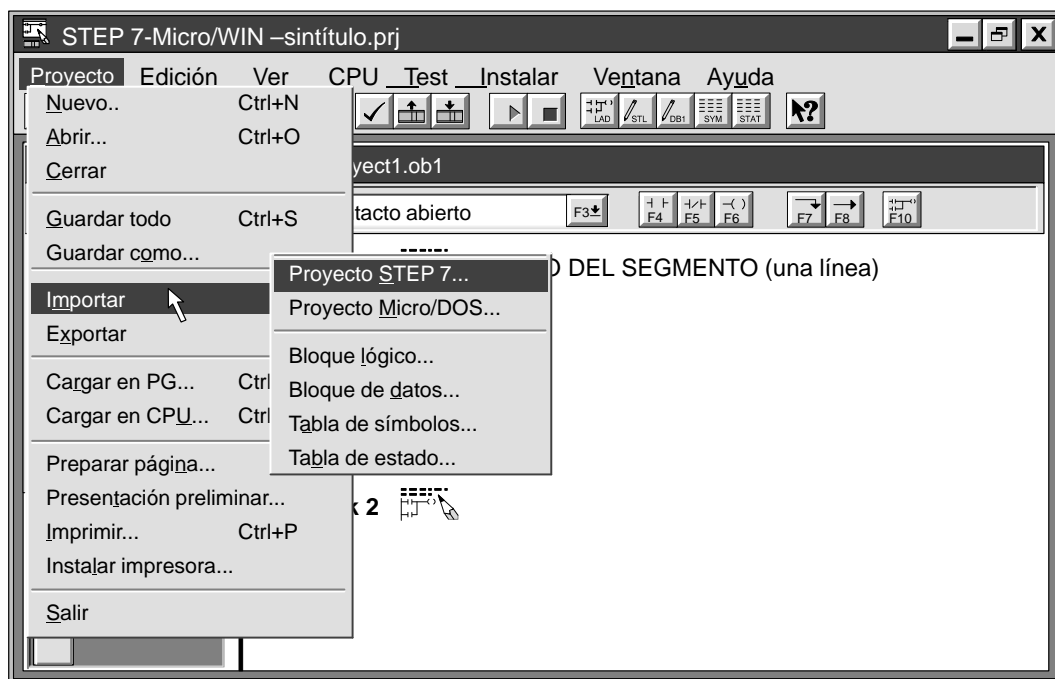


Figura E-1 Abrir un proyecto de STEP 7 desde STEP 7-Micro/WIN

## E.2 Importar archivos de STEP 7-Micro/DOS

Los programas creados con el software de programación STEP 7-Micro/DOS se pueden convertir en proyectos de STEP 7-Micro/WIN.

### Convertir programas de STEP 7-Micro/DOS

Para convertir un programa creado con STEP 7-Micro/DOS en un proyecto de STEP 7-Micro/WIN:

1. Elija el comando de menú **Proyecto ► Nuevo** para crear un proyecto sin título.
2. Elija los comandos de menú **Proyecto ► Importar ► Proyecto Micro/DOS...** como muestra la figura E-2.

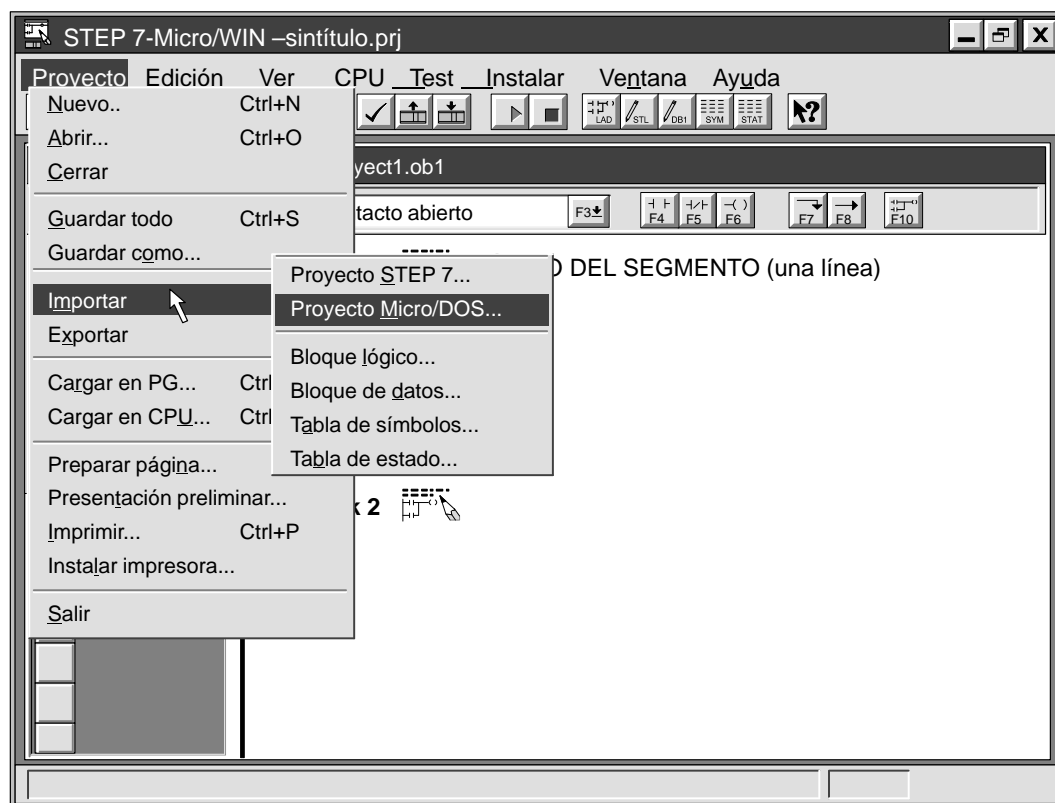


Figura E-2 Importar un archivo de STEP 7-Micro/DOS

3. Responda al mensaje (que indica que el programa de Micro/DOS sobrescribirá todo el programa) haciendo clic en el botón "Sí" para continuar. (El nuevo proyecto contiene un programa vacío). Si hace clic en el botón "No" se cancelará la operación.
4. En el cuadro de diálogo "Importar proyecto Micro/DOS" (que muestra la figura E-3), elija el directorio que contiene el programa de STEP 7-Micro/DOS que desea importar.

5. Haga doble clic en el archivo de STEP 7-Micro/DOS (o introduzca el nombre del mismo) como muestra la figura E-3.
6. Haga clic en el botón "Abrir". El programa importado y los archivos asociados se abrirán como proyecto sin título.

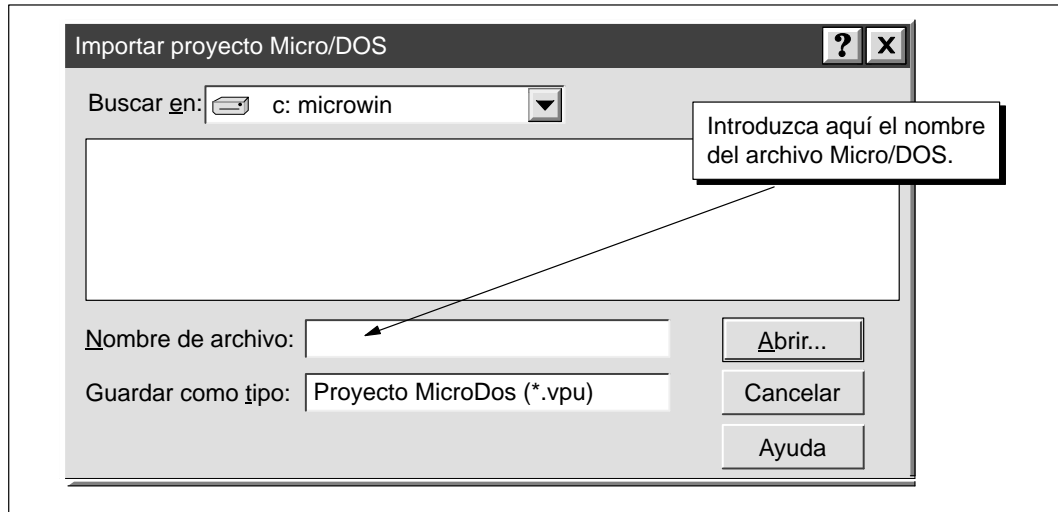


Figura E-3 Selección del programa de STEP 7-Micro/DOS

### Reglas y restricciones de conversión

Al importar un archivo de programa .vpu de STEP 7-Micro/DOS, una copia de los siguientes archivos de STEP 7-Micro/DOS se convertirá al formato de STEP 7-Micro/WIN una vez guardados los archivos:

- Archivos de programa
- Memoria V y datos
- Sinónimos y sus correspondientes comentarios
- Tabla de estado (su nombre debe coincidir con el del proyecto)

Cuando un programa creado con STEP 7-Micro/DOS se convierte en un proyecto de STEP 7-Micro/WIN sucede lo siguiente:

- Se conservan las constantes que se hayan definido en la memoria V.
- Los sinónimos de STEP 7-Micro/DOS se convierten en símbolos de STEP 7-Micro/WIN. No obstante, se truncan si exceden los 23 caracteres permitidos. Los comentarios de sinónimos, que pueden comprender hasta 144 caracteres, se truncan para que no excedan los 79 caracteres permitidos para los comentarios de símbolos en STEP 7-Micro/WIN.
- Los comentarios de segmento de Micro/DOS (que pueden comprender hasta 16 líneas de 60 caracteres) se conservan en los editores AWL y KOP.
- Si el nombre de una tabla de estado creada con STEP 7-Micro/DOS coincide con el del programa de STEP 7-Micro/DOS, se convierte en una tabla de estado de STEP 7-Micro/WIN. Por ejemplo, si el programa se llama TEST.VPU, comprendiendo las tablas de estado TEST.CH2 y TEST2.CH2, será posible convertir la tabla de estado TEST, mas no la tabla de estado TEST2.
- La dirección de red, la contraseña, el nivel de protección, la tabla de salidas y las áreas remanentes se ajustan conforme a los archivos de STEP 7-Micro/DOS. Dichos parámetros se pueden ver eligiendo el comando de menú **CPU ► Configurar....**

### Guardar el programa convertido

Para agregar el programa convertido al mismo directorio donde se encuentran los demás proyectos actuales de STEP 7-Micro/WIN:

1. Elija el comando de menú **Proyecto ► Guardar como...** y seleccione en la lista de directorios su directorio actual de STEP 7-Micro/WIN.
2. En el cuadro "Nombre de archivo", introduzca el nombre que desee asignar a los archivos de programa importados, utilizando la extensión .PRJ.
3. Haga clic en el botón "Aceptar".

---

#### Nota

Una vez guardado o cambiado el programa que se haya importado a STEP 7-Micro/WIN, no se podrá volver a convertir al formato STEP 7-Micro/DOS. No obstante, los archivos originales de Micro/DOS no se modifican. Por lo tanto, puede seguir utilizando los archivos originales con STEP 7-Micro/DOS.

---

# Tiempos de ejecución básicos 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. 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 S7-200.

Tabla F-1 Tiempo de ejecución de operaciones sin circulación de corriente

Operación sin circulación de corriente	CPU 212	CPU 214/215/216
Todas las operaciones AWL	10 µs	6 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 al direccionarse directamente los operandos o las constantes. Si en el programa se usan direcciones indirectas, se incrementa el tiempo de ejecución como muestra la tabla F-2.

Tabla F-2 Tiempo adicional para el direccionamiento indirecto

Operación de direccionamiento indirecto	CPU 212	CPU 214/215/216
Todas las operaciones, excepto R, RI, S y SI	76 µs	47 µs
R, RI, S y SI	185,3 µs	120,2 µs

## Impacto de las entradas y salidas analógicas sobre el tiempo de ejecución

El acceso a las entradas y salidas analógicas también afecta al tiempo de ejecución de una operación. La tabla F-3 muestra un factor que se debe añadir al tiempo de ejecución básico de cada operación que acceda a un valor analógico.

Tabla F-3 Impacto de las entradas y salidas (E/S) analógicas sobre el tiempo de ejecución de las operaciones

	Módulo de ampliación	CPU 212	CPU 214/215/216
Entradas analógicas	EM231, EM235	171 µs	139 µs
Salidas analógicas	EM232, EM235	99 µs	66 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  $\mu$ s)

Operación	Descripción	CPU 212 (en $\mu$ s)	CPU 214 (en $\mu$ s)	CPU 215 (en $\mu$ s)	CPU 216 (en $\mu$ s)
=	Tiempo de ejecución básico: I, Q M SM, T, C, V, S	1.2 4.8 6.0	0,8 3,2 4,0	0,8 3,2 4,0	0,8 3,2 4,0
+D	Tiempo de ejecución básico	143	95	95	95
−D	Tiempo de ejecución básico	144	96	96	96
+I	Tiempo de ejecución básico	110	73	73	73
−I	Tiempo de ejecución básico	111	74	74	74
=I	Tiempo de ejecución básico	63	42	42	42
+R	Tiempo de ejecución básico Tiempo de ejecución máximo	– –	220 350	220 350	220 350
−R	Tiempo de ejecución básico Tiempo de ejecución máximo	– –	225 355	225 355	225 355
*R	Tiempo de ejecución básico Tiempo de ejecución máximo	– –	255 320	255 320	255 320
/R	Tiempo de ejecución básico Tiempo de ejecución máximo	– –	810 870	810 870	810 870
A	Tiempo de ejecución básico: I, Q M SM, T, C, V, S	1.2 3.0 4.8	0,8 2,0 3,2	0,8 2,0 3,2	0,8 2,0 3,2
AB < =	Tiempo de ejecución si la comparación es verdadera Tiempo de ejecución si la comparación es falsa	65 68	43 45	43 45	43 45
AB =	Tiempo de ejecución si la comparación es verdadera Tiempo de ejecución si la comparación es falsa	65 68	43 45	43 45	43 45
AB > =	Tiempo de ejecución si la comparación es verdadera Tiempo de ejecución si la comparación es falsa	65 68	43 45	43 45	43 45
AD < =	Tiempo de ejecución si la comparación es verdadera Tiempo de ejecución si la comparación es falsa	137 140	91 93	91 93	91 93
AD =	Tiempo de ejecución si la comparación es verdadera Tiempo de ejecución si la comparación es falsa	137 140	91 93	91 93	91 93
AD > =	Tiempo de ejecución si la comparación es verdadera Tiempo de ejecución si la comparación es falsa	137 140	91 93	91 93	91 93
AI	Tiempo de ejecución básico	54	36	36	36
ALD	Tiempo de ejecución básico	1,2	0,8	0,8	0,8
AN	Tiempo de ejecución básico: I, Q M SM, T, C, V, S	1.2 3.0 4.8	0,8 2,0 3,2	0,8 2,0 3,2	0,8 2,0 3,2
ANDB	Tiempo de ejecución básico	–	–	49	49
ANDD	Tiempo de ejecución básico	137	91	91	91

Tabla F-4 Tiempos de ejecución de las operaciones AWL (en  $\mu$ s), continuación

Operación	Descripción	CPU 212 (en $\mu$ s)	CPU 214 (en $\mu$ s)	CPU 215 (en $\mu$ s)	CPU 216 (en $\mu$ s)
ANDW	Tiempo de ejecución básico	110	73	73	73
ANI	Tiempo de ejecución básico	54	36	36	36
AR=	Tiempo de ejecución básico	–	98	98	98
AR<=	Tiempo de ejecución básico	–	98	98	98
AR>=	Tiempo de ejecución básico	–	98	98	98
ATCH	Tiempo de ejecución básico	48	32	32	32
ATH	Total = tiempo básico + (longitud) * (ML) Tiempo de ejecución básico Multiplicador de longitud (ML)	729 62	486 41	486 41	486 41
ATT	Tiempo de ejecución básico	–	268	268	268
AW < =	Tiempo de ejecución si la comparación es verdadera Tiempo de ejecución si la comparación es falsa	110 113	73 75	73 75	73 75
AW=	Tiempo de ejecución si la comparación es verdadera Tiempo de ejecución si la comparación es falsa	110 113	73 75	73 75	73 75
AW > =	Tiempo de ejecución si la comparación es verdadera Tiempo de ejecución si la comparación es falsa	110 113	73 75	73 75	73 75
BCDI	Tiempo de ejecución básico	249	166	166	166
BMB	Total = tiempo básico + (longitud) * (ML) Tiempo de ejecución básico Multiplicador de longitud (ML)	633 32	422 21	422 21	422 21
BMD	Total = tiempo básico + (longitud) * (ML) Tiempo de ejecución básico Multiplicador de longitud (ML)	– –	– –	446 43	446 43
BMW	Total = tiempo básico + (longitud) * (ML) Tiempo de ejecución básico Multiplicador de longitud (ML)	636 51	424 34	424 34	424 34
CALL	Tiempo de ejecución básico	35	23	23	23
CRET	Tiempo de ejecución básico	26	17	17	17
CRETI	Tiempo de ejecución básico	75	50	50	50
CTU	Tiempo de ejecución básico	78	52	52	52
CTUD	Tiempo de ejecución básico	105	70	70	70
DECB	Tiempo de ejecución básico	–	–	37	37
DECD	Tiempo de ejecución básico	98	65	65	65
DECO	Tiempo de ejecución básico	84	56	56	56
DECW	Tiempo de ejecución básico	83	55	55	55
DISI	Tiempo de ejecución básico	36	24	24	24
DIV	Tiempo de ejecución básico	410	273	273	273
DTCH	Tiempo de ejecución básico	39	26	26	26

Tabla F-4 Tiempos de ejecución de las operaciones AWL (en  $\mu$ s), continuación

Operación	Descripción	CPU 212 (en $\mu$ s)	CPU 214 (en $\mu$ s)	CPU 215 (en $\mu$ s)	CPU 216 (en $\mu$ s)
DTR	Tiempo de ejecución básico Tiempo de ejecución máximo	–	108 135	108 135	108 135
ED	Tiempo de ejecución básico	32	21	21	21
ENCO	Tiempo de ejecución mínimo Tiempo de ejecución máximo	75 93	50 62	50 62	50 62
END	Tiempo de ejecución básico	1.8	1,2	1,2	1,2
ENI	Tiempo de ejecución básico	36	24	24	24
EU	Tiempo de ejecución básico	32	21	21	21
FIFO	Total = tiempo básico + (ML) * (longitud) Tiempo de ejecución básico Multiplicador de longitud (ML)	–	234 29	234 29	234 29
FILL	Total = tiempo básico + (ML) * (longitud) Tiempo de ejecución básico Multiplicador de longitud (ML)	578 18	385 12	385 12	385 12
FND <	Total = tiempo básico + (ML) * (longitud) Tiempo de ejecución básico Multiplicador de longitud (ML)	–	424 28	424 28	424 28
FND <>	Total = tiempo básico + (ML) * (longitud) Tiempo de ejecución básico Multiplicador de longitud (ML)	–	423 29	423 29	423 29
FND =	Total = tiempo básico + (ML) * (longitud) Tiempo de ejecución básico Multiplicador de longitud (ML)	–	431 25	431 25	431 25
FND >	Total = tiempo básico + (ML) * (longitud) Tiempo de ejecución básico Multiplicador de longitud (ML)	–	428 28	428 28	428 28
FOR	Total = tiempo básico + (ML) * (número de repeticiones) Tiempo de ejecución básico Multiplicador de lazos (ML)	–	135 129	135 129	135 129
HDEF	Tiempo de ejecución básico	80	53	53	53
HSC	Tiempo de ejecución básico	101.0	67	67	67
HTA	Total = tiempo básico + (ML) * (longitud) Tiempo de ejecución básico Multiplicador de longitud (ML)	714 35	476 23	476 23	476 23
IBCD	Tiempo de ejecución básico	186	124	124	124
INCB	Tiempo de ejecución básico	–	–	34	34
INCD	Tiempo de ejecución básico	96	64	64	64
INCW	Tiempo de ejecución básico	81	54	54	54
INT	Tiempo de ejecución típico con 1 interrupción	180	120	120	120
INVB	Tiempo de ejecución básico	–	–	40	40
INVD	Tiempo de ejecución básico	99	66	66	66
INVW	Tiempo de ejecución básico	84	56	56	56

Tabla F-4 Tiempos de ejecución de las operaciones AWL (en  $\mu$ s), continuación

Operación	Descripción	CPU 212 (en $\mu$ s)	CPU 214 (en $\mu$ s)	CPU 215 (en $\mu$ s)	CPU 216 (en $\mu$ s)
JMP	Tiempo de ejecución básico	1,2	0,8	0,8	0,8
LBL	Tiempo de ejecución básico	0	0	0	0
LD	Tiempo de ejecución básico: I, Q M SM, T, C, V, S	1.2 3.0 4.8	0,8 2,0 3,2	0,8 2,0 3,2	0,8 2,0 3,2
LDB <=	Tiempo de ejecución si la comparación es verdadera Tiempo de ejecución si la comparación es falsa	63 66	42 44	42 44	42 44
LDB =	Tiempo de ejecución si la comparación es verdadera Tiempo de ejecución si la comparación es falsa	63 66	42 44	42 44	42 44
LDB >=	Tiempo de ejecución si la comparación es verdadera Tiempo de ejecución si la comparación es falsa	63 66	42 44	42 44	42 44
LDD <=	Tiempo de ejecución si la comparación es verdadera Tiempo de ejecución si la comparación es falsa	135 138	90 92	90 92	90 92
LDD =	Tiempo de ejecución si la comparación es verdadera Tiempo de ejecución si la comparación es falsa	135 138	90 92	90 92	90 92
LDD > =	Tiempo de ejecución si la comparación es verdadera Tiempo de ejecución si la comparación es falsa	135 138	90 92	90 92	90 92
LDI	Tiempo de ejecución básico	50	33	33	33
LDN	Tiempo de ejecución básico: I, Q M SM, T, C, V, S	1.8 3.6 5.4	1,2 2,4 3,6	1,2 2,4 3,6	1,2 2,4 3,6
LDNI	Tiempo de ejecución básico	50	33	33	33
LDR=	Tiempo de ejecución básico	–	98	98	98
LDR<=	Tiempo de ejecución básico	–	98	98	98
LDR>=	Tiempo de ejecución básico	–	98	98	98
LDW <=	Tiempo de ejecución si la comparación es verdadera Tiempo de ejecución si la comparación es falsa	108 111	72 74	72 74	72 74
LDW =	Tiempo de ejecución si la comparación es verdadera Tiempo de ejecución si la comparación es falsa	108 111	72 74	72 74	72 74
LDW >=	Tiempo de ejecución si la comparación es verdadera Tiempo de ejecución si la comparación es falsa	108 111	72 74	72 74	72 74
LIFO	Tiempo de ejecución básico	–	261	261	261
LPP	Tiempo de ejecución básico	0.6	0,4	0,4	0,4
LPS	Tiempo de ejecución básico	1,2	0,8	0,8	0,8
LRD	Tiempo de ejecución básico	0.6	0,4	0,4	0,4
LSCR	Tiempo de ejecución básico	18	12	12	12
MEND	Tiempo de ejecución básico	1,2	0,8	0,8	0,8
MOVB	Tiempo de ejecución básico	45	30	30	30
MOVD	Tiempo de ejecución básico	81	54	54	54
MOVR	Tiempo de ejecución básico	81	54	54	54

Tabla F-4 Tiempos de ejecución de las operaciones AWL (en  $\mu$ s), continuación

Operación	Descripción	CPU 212 (en $\mu$ s)	CPU 214 (en $\mu$ s)	CPU 215 (en $\mu$ s)	CPU 216 (en $\mu$ s)
MOVW	Tiempo de ejecución básico	66	44	44	44
MUL	Tiempo de ejecución básico	210	140	140	140
NEXT	Tiempo de ejecución básico	–	0	0	0
NETR	Tiempo de ejecución básico	–	478	478	478
NETW	Total = tiempo básico + (ML) * (longitud) Tiempo de ejecución básico Multiplicador de longitud (ML)	–	460 16,8	460 16,8	460 16,8
NOP	Tiempo de ejecución básico	0	0	0	0
NOT	Tiempo de ejecución básico	1,2	0,8	0,8	0,8
O	Tiempo de ejecución básico: I, Q M SM, T, C, V, S	1.2 3.0 4.8	0,8 2,0 3,2	0,8 2,0 3,2	0,8 2,0 3,2
OB < =	Tiempo de ejecución si la comparación es verdadera Tiempo de ejecución si la comparación es falsa	65 68	43 45	43 45	43 45
OB =	Tiempo de ejecución si la comparación es verdadera Tiempo de ejecución si la comparación es falsa	65 68	43 45	43 45	43 45
OB > =	Tiempo de ejecución si la comparación es verdadera Tiempo de ejecución si la comparación es falsa	65 68	43 45	43 45	43 45
OD < =	Tiempo de ejecución si la comparación es verdadera Tiempo de ejecución si la comparación es falsa	138 140	92 93	92 93	92 93
OD =	Tiempo de ejecución si la comparación es verdadera Tiempo de ejecución si la comparación es falsa	138 140	92 93	92 93	92 93
OD > =	Tiempo de ejecución si la comparación es verdadera Tiempo de ejecución si la comparación es falsa	138 140	92 93	92 93	92 93
OI	Tiempo de ejecución básico	54	36	36	36
OLD	Tiempo de ejecución básico	1,2	0,8	0,8	0,8
ON	Tiempo de ejecución básico: I, Q M SM, T, C, V, S	1.2 3.0 4.8	0,8 2,0 3,2	0,8 2,0 3,2	0,8 2,0 3,2
ONI	Tiempo de ejecución básico	54	36	36	36
OR=	Tiempo de ejecución básico	–	98	98	98
OR<=	Tiempo de ejecución básico	–	98	98	98
OR >=	Tiempo de ejecución básico	–	98	98	98
ORB	Tiempo de ejecución básico	–	–	49	49
ORD	Tiempo de ejecución básico	137	91	91	91
ORW	Tiempo de ejecución básico	110	73	73	73
OW < =	Tiempo de ejecución si la comparación es verdadera Tiempo de ejecución si la comparación es falsa	108 111	72 74	72 74	72 74
OW =	Tiempo de ejecución si la comparación es verdadera Tiempo de ejecución si la comparación es falsa	108 111	72 74	72 74	72 74

Tabla F-4 Tiempos de ejecución de las operaciones AWL (en  $\mu$ s), continuación

Operación	Descripción	CPU 212 (en $\mu$ s)	CPU 214 (en $\mu$ s)	CPU 215 (en $\mu$ s)	CPU 216 (en $\mu$ s)
OW > =	Tiempo de ejecución si la comparación es verdadera Tiempo de ejecución si la comparación es falsa	108 111	72 74	72 74	72 74
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 $K_c$ , $T_d$ , $T_i$ , o $T_s$ se han modificado desde la anterior ejecución de esta operación o si se ha cambiado a modo automático.	– –	– –	2000 2600	2000 2600
PLS	Tiempo de ejecución básico	–	153	153	153
R	Total = tiempo de operando + (ML) * (longitud) Tiempo de ejecución del contador Tiempo de ejecución del temporizador Tiempo de ejecución de otros operandos Multiplicador de longitud (ML) del contador Multiplicador de longitud (ML) del temporizador Multiplicador de longitud (ML) de otros operandos Si la longitud se almacena en una variable, en vez de ser una constante, incrementar el tiempo de ejecución básico sumando:	33.9 32.9 39.9 28.8 49.7 5.6 109.8	23 21 27 19,2 33,1 3,7 73.2	23 22 27 19,2 33,1 3,7 73.2	23 22 27 19,2 33,1 3,7 73.2
RCV	Tiempo de ejecución básico	–	–	126	126
RET	Tiempo de ejecución básico	27	18	18	18
RETI	Tiempo de ejecución básico	75	50	50	50
RI	Total = tiempo básico + (ML) * (longitud) Tiempo de ejecución básico Multiplicador de longitud (ML) Si la longitud se almacena en una variable, en vez de ser una constante, incrementar el tiempo de ejecución básico sumando:	31.5 60 110	21 40 73	21 40 73	21 40 73
RLB	Total = tiempo básico + (ML) * (longitud) Tiempo de ejecución básico Multiplicador de longitud (ML)	– –	– –	62 1,2	62 1,2
RLD	Total = tiempo básico + (ML) * (longitud) Tiempo de ejecución básico Multiplicador de longitud (ML)	129 10.7	86 7,1	86 7,1	86 7,1
RLW	Total = tiempo básico + (ML) * (longitud) Tiempo de ejecución básico Multiplicador de longitud (ML)	116 6.9	77 4,6	77 4,6	77 4,6
RRB	Total = tiempo básico + (ML) * (longitud) Tiempo de ejecución básico Multiplicador de longitud (ML)	– –	– –	62 1,2	62 1,2
RRD	Total = tiempo básico + (ML) * (longitud) Tiempo de ejecución básico Multiplicador de longitud (ML)	135 10.4	90 6,9	90 6,9	90 6,9

Tabla F-4 Tiempos de ejecución de las operaciones AWL (en  $\mu$ s), continuación

Operación	Descripción	CPU 212 (en $\mu$ s)	CPU 214 (en $\mu$ s)	CPU 215 (en $\mu$ s)	CPU 216 (en $\mu$ s)
RRW	Total = tiempo básico + (ML) * (longitud) Tiempo de ejecución básico Multiplicador de longitud (ML)	117 6.6	78 4,4	78 4,4	78 4,4
S	Total = tiempo básico + (ML) * (longitud) Tiempo de ejecución básico Multiplicador de longitud (ML)  Si la longitud se almacena en una variable, en vez de ser una constante, incrementar el tiempo de ejecución básico sumando:	38 5.6  110	25 3.7  74	25 3.7  74	25 3.7  74
SBR	Tiempo de ejecución básico	0	0	0	0
SCRE	Tiempo de ejecución básico	0	0	0	0
SCRT	Tiempo de ejecución básico	31	21	21	21
SEG	Tiempo de ejecución básico	47	31	31	31
SHRB	Total = tiempo básico + (ML) * (longitud) Tiempo de ejecución básico Multiplicador de longitud (ML)	449 2.3	299 1,5	299 1,5	299 1,5
SI	Total = tiempo básico + (ML) * (longitud) Tiempo de ejecución básico Multiplicador de longitud (ML)  Si la longitud se almacena en una variable, en vez de ser una constante, incrementar el tiempo de ejecución básico sumando:	32 58  110	21 38  73	21 38  73	21 38  73
SLB	Total = tiempo básico + (ML) * (longitud) Tiempo de ejecución básico Multiplicador de longitud (ML)	– –	– –	64 1,6	64 1,6
SLD	Total = tiempo básico + (ML) * (longitud) Tiempo de ejecución básico Multiplicador de longitud (ML)	131 8.9	87 5,9	87 5,9	87 5,9
SLW	Total = tiempo básico + (ML) * (longitud) Tiempo de ejecución básico Multiplicador de longitud (ML)	119 5.1	79 3,4	79 3,4	79 3,4
SQRT	Tiempo de ejecución básico Tiempo de ejecución máximo	– 2110	1830 2110	1830 2110	1830 2110
SRB	Total = tiempo básico + (ML) * (longitud) Tiempo de ejecución básico Multiplicador de longitud (ML)	– –	– –	64 1,6	64 1,6
SRD	Total = tiempo básico + (ML) * (longitud) Tiempo de ejecución básico Multiplicador de longitud (ML)	137 8.6	91 5,7	91 5,7	91 5,7
SRW	Total = tiempo básico + (ML) * (longitud) Tiempo de ejecución básico Multiplicador de longitud (ML)	120 5.0	80 3,3	80 3,3	80 3,3
STOP	Tiempo de ejecución básico	13	9	9	9
SWAP	Tiempo de ejecución básico	65	43	43	43

Tabla F-4 Tiempos de ejecución de las operaciones AWL (en  $\mu$ s), continuación

Operación	Descripción	CPU 212 (en $\mu$ s)	CPU 214 (en $\mu$ s)	CPU 215 (en $\mu$ s)	CPU 216 (en $\mu$ s)
TODR	Tiempo de ejecución básico	–	282	282	282
TODW	Tiempo de ejecución básico	–	489	489	489
TON	Tiempo de ejecución básico	48	32	32	32
TONR	Tiempo de ejecución básico	74	49	49	49
TRUNC	Tiempo de ejecución básico Tiempo de ejecución máximo	– –	258 420	258 420	258 420
WDR	Tiempo de ejecución básico	21	14	14	14
XMT	Tiempo de ejecución básico	272	181	181	181
XORB	Tiempo de ejecución básico	–	–	49	49
XORD	Tiempo de ejecución básico	137	91	91	91
XORW	Tiempo de ejecución básico	110	73	73	73





## Números de referencia

CPU	Nº de referencia
CPU 212 con alimentación DC, entradas DC, salidas DC	6ES7 212-1AA01-0XB0
CPU 212 con alimentación AC, entradas DC, salidas de relé	6ES7 212-1BA01-0XB0
CPU 212 con alimentación AC, entradas AC, salidas AC	6ES7 212-1CA01-0XB0
CPU 212 con alimentación AC, entradas DC tipo fuente, salidas de relé	6ES7 212-1BA10-0XB0
CPU 212 con alimentación AC, entradas AC 24 V, salidas AC	6ES7 212-1DA01-0XB0
CPU 212 con alimentación AC 24 V, entradas DC tipo fuente, salidas de relé	6ES7 212-1FA01-0XB0
CPU 212 con alimentación AC, entradas AC, salidas de relé	6ES7 212-1GA01-0XB0
CPU 214 con alimentación DC, entradas DC, salidas DC	6ES7 214-1AC01-0XB0
CPU 214 con alimentación AC, entradas DC, salidas de relé	6ES7 214-1BC01-0XB0
CPU 214 con alimentación AC, entradas AC, salidas AC	6ES7 214-1CC01-0XB0
CPU 214 con alimentación AC, entradas DC tipo fuente, salidas de relé	6ES7 214-1BC10-0XB0
CPU 214 con alimentación AC, entradas AC 24 V, salidas AC	6ES7 214-1DC01-0XB0
CPU 214 con alimentación AC, entradas AC, salidas de relé	6ES7 214-1GC01-0XB0
CPU 215 con alimentación DC, entradas DC, salidas DC	6ES7 215-2AD00-0XB0
CPU 215 con alimentación DC, entradas DC, salidas DC	6ES7 215-2BD00-0XB0
CPU 216 con alimentación DC, entradas DC, salidas DC	6ES7 216-2AD00-0XB0
CPU 216 con alimentación AC, entradas DC, salidas de relé	6ES7 216-2BD00-0XB0

Módulos de ampliación	Nº de referencia
Módulo de ampliación EM 221, 8 entradas digitales DC 24 V	6ES7 221-1BF00-0XA0
Módulo de ampliación EM 221, 8 entradas digitales AC 120 V	6ES7 221-1EF00-0XA0
Módulo de ampliación EM 221, 8 entradas digitales tipo fuente DC 24 V	6ES7 221-1BF10-0XA0
Módulo de ampliación EM 221, 8 entradas digitales AC 24 V	6ES7 221-1JF00-0XA0
Módulo de ampliación EM 222, 8 salidas DC 24 V	6ES7 222-1BF00-0XA0
Módulo de ampliación EM 222, 8 salidas de relé	6ES7 222-1HF00-0XA0
Módulo de ampliación EM 222, 8 salidas AC 120/230 V	6ES7 222-1EF00-0XA0
Módulo de ampliación EM 223, 4 entradas digitales DC 24 V / 4 salidas digitales DC 24 V	6ES7 223-1BF00-0XA0
Módulo de ampliación EM 223, 4 entradas digitales DC 24 V / 4 salidas digitales de relé	6ES7 223-1HF00-0XA0
Módulo de ampliación EM 223, 4 entradas digitales AC 120 V / 4 salidas digitales AC 120/230 V	6ES7 223-1EF00-0XA0
Módulo de ampliación EM 223, 8 entradas digitales DC 24 V / 8 salidas digitales de relé	6ES7 223-1PH00-0XA0
Módulo de ampliación EM 223, 8 entradas digitales DC 24 V / 8 salidas digitales DC 24 V	6ES7 223-1BH00-0XA0

Módulos de ampliación	Nº de referencia
Módulo de ampliación EM 223, 16 entradas digitales DC 24 V / 16 salidas digitales de relé	6ES7 223-1PL00-0XA0
Módulo de ampliación EM 223, 16 entradas digitales DC 24 V / 16 salidas digitales DC 24 V	6ES7 223-1BL00-0XA0
Módulo de ampliación EM 231, 3 entradas analógicas AI de 12 bits	6ES7 231-0HC00-0XA0
Módulo de ampliación EM 232, 2 salidas analógicas AQ de 12 bits	6ES7 232-0HB00-0XA0
Módulo de ampliación EM 235, 3 entradas analógicas AI y 1 salida analógica AQ x de 12 bits	6ES7 235-0KD00-0XA0
CP 242-2 AS-Interface Master Module para S7-200	6GK7 242-2AX00-0XA0

Cables, conectores de red y repetidores	Nº de referencia
Cable para módulos de ampliación	6ES7 290-6BC50-0XA0
Cable MPI	6ES7 901-0BF00-0AA0
Cable PC/PPI	6ES7 901-3BF00-0XA0
Cable para redes PROFIBUS	6XV1 830-0AH10
Conector de bus con conector de interface de programación, salida vertical del cable	6ES7 972-0BB10-0XA0
Conector de bus (sin conector de interface de programación), salida vertical del cable	6ES7 972-0BA10-0XA0
Conector de bus RS 485 con salida axial del cable	6GK1 500-0EA00
Conector de bus RS 485 con salida de cable 30°	6ES7 972-0BA30-0XA0
Repetidor RS 485 IP 20	6ES7 972-0AA00-0XA0

Tarjetas de comunicación	Nº de referencia
Tarjeta MPI: Short AT ISA	6ES7 793-2AA01-0AA0
CP 5411: Short AT ISA	6GK1 541-1AA00
CP 5511: PCMCIA, tipo II, hardware "plug & play"	6GK1 551-1AA00
CP 5611: Short PCI, hardware "plug & play"	6GK1 561-1AA00

Visualizadores de textos y paneles de operador	Nº de referencia
Visualizador de textos TD 200	6ES7 272-0AA00-0YA0
Panel de operador OP3	6AV3 503-1DB10
Panel de operador OP7	6AV3 607-IJC20-0AX0
Panel de operador OP17	6AV3 617-IJC20-0AX0

General	Nº de referencia
Cartucho de memoria 8K x 8	6ES7 291-8GC00-0XA0
Cartucho de memoria 16k x 8	6ES7 291-8GD00-0XA0
Cartucho de pila	6ES7 291-8BA00-0XA0
Frenos para perfiles soporte	6ES5 728-8MA11
Bloque de 12 bornes de conexión para cableado de campo (CPU 212/215/216) paquete de 10	6ES7 290-2AA00-0XA0
Bloque de 14 bornes de conexión para cableado de campo (CPU 215/216 y módulos de ampliación) paquete de 10	6ES7 290-2CA00-0XA0
Bloque de 18 bornes de conexión para cableado de campo (CPU 214) paquete de 10	6ES7 290-2BA00-0XA0
Simulador de entradas DC para la CPU 212	6ES7 274-1XF00-0XA0
Simulador de entradas DC para la CPU 214	6ES7 274-1XH00-0XA0
Simulador de entradas DC para CPU 215/216	6ES7 274-1XK00-0XA0

Software de programación	Nº de referencia
STEP 7–Micro/WIN 16 (V2.1), licencia única	6ES7 810-2AA01-0YX0
STEP 7–Micro/WIN 16 (V2.1), licencia de copia	6ES7 810-2AA01-0YX1
STEP 7–Micro/WIN 16 (V2.1), actualización	6ES7 810-2AA01-0YX3
STEP 7–Micro/WIN 32 (V2.1), licencia única	6ES7 810-2AA11-0YX0
STEP 7–Micro/WIN 32 (V2.1), licencia de copia	6ES7 810-2AA11-0YX1
STEP 7–Micro/WIN 32 (V2.1), actualización	6ES7 810-2AA11-0YX3
STEP 7–Micro/DOS, licencia única	6ES7 810-2DA00-0YX0

Manuales	Nº de referencia
Sistemas de periferia descentralizada ET 200, Manual del sistema	6ES5 998-3ES22
Manual de la unidad de programación PG 702	6ES7 702-0AA00-8BA0
Manual del usuario del visualizador de textos TD 200	6ES7 272 0AA00-8BA0
Manual del CP242-2 AS-Interface Master Module	6GK7 242-2AX00-8BA0
Manual del usuario de STEP 7–Micro/DOS	6ES7 810-2DA10-8BA0



## Eliminar errores

Tabla H-1 Eliminar errores

Problema	Causas posibles	Solución
Las salidas han dejado de funcionar.	El dispositivo controlado ha causado una sobretensión que ha deteriorado la salida.	Al conectar la CPU a una carga inductiva (p.ej. un motor o un relé) es preciso utilizar un circuito de supresión adecuado (v. apt. 2.4).
El diodo "SF" (System Fault) de la CPU se enciende.	<p>La lista siguiente describe las causas más frecuentes:</p> <ul style="list-style-type: none"> <li>Error en el programa de usuario. <ul style="list-style-type: none"> <li>0003 Error de tiempo de vigilancia (watchdog).</li> <li>0011 Direccionamiento indirecto.</li> <li>0012 Comparación no válida.</li> </ul> </li> <li>Interferencia eléctrica. <ul style="list-style-type: none"> <li>0001 a 0009</li> </ul> </li> <li>Componente deteriorado. <ul style="list-style-type: none"> <li>0001 a 0010</li> </ul> </li> </ul>	<p>Lea el código del error fatal y consulte el apartado C.1:</p> <ul style="list-style-type: none"> <li>En caso de un error de programación, consulte la descripción de las operaciones FOR, NEXT, JMP, LBL y CMP.</li> <li>En caso de una interferencia eléctrica: <ul style="list-style-type: none"> <li>Consulte las reglas de cableado que se indican 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.</li> <li>Conecte a tierra el terminal M de la alimentación de sensores de DC 24 V.</li> </ul> </li> </ul>
Los valores de las entradas analógicas difieren de un muestreo a otro, aunque la señal de entrada es constante.	<p>Ello puede tener diversas causas:</p> <ul style="list-style-type: none"> <li>Interferencia eléctrica de la alimentación de corriente.</li> <li>Interferencia eléctrica en la señal de entrada.</li> <li>Puesta a tierra incorrecta.</li> <li>El valor retornado no se formatea conforme a lo esperado.</li> <li>Se trata de un módulo rápido que no permite filtrar a 50/60 Hz.</li> </ul>	<ul style="list-style-type: none"> <li>El valor retornado por el módulo no se ha filtrado. Al programa de usuario se le puede agregar una rutina de filtraje simple. Consulte el capítulo 5 para obtener informaciones más detalladas acerca del Asistente para filtrar las entradas analógicas.</li> <li>Compare la repetibilidad real del valor del módulo con el valor indicado en el Anexo A. Los módulos S7-200 retornan un valor no filtrado y justificado a la izquierda. Ello significa que cada variación de 1 conteo incrementará el valor en 8 en el módulo S7-200.</li> <li>Para determinar la fuente de la interferencia, intente cortocircuitar una entrada analógica no utilizada. Si el valor leído de la entrada en cortocircuito es diferente al de la entrada de sensor, la interferencia provendrá de los cables eléctricos. En caso contrario, la interferencia provendrá del sensor o del cableado del mismo. <ul style="list-style-type: none"> <li>Para obtener informaciones más detalladas acerca de las interferencias en el cableado de los sensores, consulte las reglas de instalación del EM231 (apt. A.33) o del EM235 (apt. A.35).</li> <li>En caso de interferencias en la fuente de alimentación, consulte las reglas de cableado en el apartado 2.3 o intente conectar a tierra los terminales M del módulo analógico y de la alimentación de sensores de la CPU.</li> </ul> </li> </ul>

Tabla H-1 Eliminar errores, continuación

Problema	Causas posibles	Solución
Fuente de alimentación deteriorada.	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	<ul style="list-style-type: none"> <li>• Puesta a tierra incorrecta.</li> <li>• Conducción del cableado en el armario eléctrico.</li> </ul>	Consulte las reglas de cableado que se indican 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.
Valores intermitentes de los módulos de ampliación	Vibración excesiva.	Los límites de vibración sinusoidal se indican en el apartado A.1.
	Montaje incorrecto en el perfil soporte.	Si el sistema se encuentra montado en un perfil soporte, consulte el apartado 2.2.
	Las uniones de plástico no se retiraron por completo al quitarse la tapa del interface de ampliación de bus.	Consulte el apartado 2.2 para obtener informaciones detalladas sobre cómo instalar los módulos de ampliación.
	Conector de bus defectuoso.	Sustituya el conector de bus.
Red de comunicación deteriorada al conectar un dispositivo externo. (Están deteriorados el puerto del PC, el interface de la CPU o el cable PC/PPI).	El interface RS-485 de la CPU S7-200 y el cable PC/PPI no tienen separación galvánica (a menos que se indique de otra forma en la hoja de datos técnicos).  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 deteriorar los circuitos.	<ul style="list-style-type: none"> <li>• Consulte el apartado 2.3 donde se indican las reglas de cableado y el capítulo 9 para obtener más información acerca de la comunicación en redes.</li> <li>• Sustituya el cable PC/PPI por un adaptador de RS485 a RS232 con separación galvánica (no suministrado por Siemens).</li> <li>• Utilice un repetidor de RS485 a RS485 con separación galvánica al conectar máquinas que no tengan una referencia eléctrica común.</li> </ul>
Problemas de comunicación en STEP 7–Micro/WIN		Para obtener informaciones más detalladas acerca de la comunicación en redes, consulte el capítulo 9.
Tratamiento de errores		Para obtener informaciones más detalladas acerca de los códigos de error, consulte el AnexoC.

# Índice alfabético

## A

### Acceso

- a áreas de memoria, modificar un puntero, 7-10
- áreas de memoria
  - & y \*, 7-9
  - direccionamiento indirecto, 7-9–7-11
- áreas de operandos, 10-3
- direccionamiento directo, 7-2
- Acceso a bits, 7-2
  - CPU 212/214/215/216, 10-3
- Acceso a bytes, 7-2
  - CPU 212/214/215/216, 10-3
  - utilizando punteros, 7-10
- Acceso a palabras, 7-2
  - CPU 212/214/215/216, 10-3
  - utilizando punteros, 7-10
- Acceso a palabras dobles, CPU 212/214/215/216, 10-3
- Activar contador rápido, definir modo para
  - contador rápido, 10-21
- Activar contador rápido (HSC), 10-21
- Actualización síncrona, función PWM, 10-41
- Actualizar, temporizadores, 10-14
- Acumuladores, direccionamiento, 7-6
- Adaptador, módem nulo, 3-19–3-20, 9-12
- Adaptador de módem nulo, 9-12
- Adaptador para módem nulo, 3-19–3-20
- Ajustar reloj de tiempo real, 10-49
- Ajustes en el cuadro de diálogo, "Interface PG/PC", 3-10
- Algoritmo para regulación PID, 10-55–10-59
- Algoritmo PID, 10-55–10-59
- Almacenar el programa en la memoria no volátil, 7-16
- ALT, combinaciones de teclas con, 5-9
- Archivo de datos maestros de los dispositivos (GSD), 9-23–9-25
  - acceder a, 9-23
  - utilización para maestros no SIMATIC, 9-24
- Archivo GSD
  - acceder a, 9-23
  - utilización para maestros no SIMATIC, 9-24

- Área de marcas, direccionamiento, 7-3
- Área de memoria de variables, direccionamiento, 7-3
- Áreas de funciones, crear, 6-2
- Áreas de memoria, 6-4
  - acceder a datos, 6-4
  - CPU 212/214/215/216, 10-2
- Áreas de memoria
  - acceder a los datos, 7-2
  - bits, 7-2
  - bytes, 7-2
  - CPU, 7-2
- Áreas de memoria, áreas de operandos, 10-3
- Áreas de operandos, CPU 212/214/215/216, 10-3
- Áreas remanentes, definir, 7-15
- Armario eléctrico
  - desmontaje del, 2-7
  - dimensiones
    - CPU 212, 2-3
    - CPU 214, 2-3
    - CPU 215, 2-4
    - CPU 216, 2-4
  - módulos de ampliación, 2-4
  - montaje, 2-5
    - cable de ampliación, 2-5–2-7
- Asignación de pines
  - interface de comunicación, 9-6
  - PC/PPI, A-82
- Asignar (bobina), 10-10
- Asignar directamente, 10-10
- Asistente de operaciones, S7-200
  - filtrar entradas analógicas, 5-14–5-16
  - seleccionar/utilizar, 5-12–5-14
- Asistente para filtrar entradas analógicas, 5-14–5-16
- Asistente, TD 200, 5-3
  - caracteres internacionales y especiales, 5-9
- Asociar interrupción, 10-116

AWL, 6-5  
 cambiar a KOP, 3-31  
 editor, 3-29  
 elementos básicos, 6-6  
 introducir el programa, 5-21  
 programa  
   imprimir, 5-23  
   introducir en STEP 7–Micro/WIN, 3-29  
 tiempos de ejecución de las operaciones,  
   F-1–F-11  
 visualizar en KOP, 3-29  
 visualizar un programa de STEP 7–Micro/WIN,  
   3-31

Ayuda

Ayuda online, STEP 7–Micro/WIN, 3-1

## B

Bias (suma integral), algoritmo PID, 10-57  
 Bits, 7-2  
 Bits de control, contadores rápidos, 10-28  
 Bits de estado (SMB0), D-1  
 Bits, marcas especiales, D-1–D-13  
 Bloque de bornes, bornes opcional, 2-10  
 Bloque de datos  
   crear en STEP 7-Micro/WIN, 3-32  
   ejemplos, 3-32  
   identificadores válidos de tamaños, 3-33  
   tipo de datos, 3-33  
 Bloque de parámetros (TD 200), 5-2  
   almacenar/visualizar, 5-11  
   configurar, 5-3  
   dirección, 5-7  
   ejemplo, 5-11  
 Bornes, bloque de bornes opcional, 2-10  
 Borrar primer registro de la tabla, 10-75  
 Borrar temporizador de vigilancia, 10-85–10-87  
   consideraciones, 10-85  
 Borrar último registro de la tabla, 10-74  
 Búfer de entrada, CPU 215, 9-18, 9-21  
 Búfer de salida, CPU 215, 9-18, 9-21  
 Buscar valor en tabla, 10-76  
 Buscar/reemplazar, 5-19  
 Búsqueda, operaciones de, 10-73–10-77  
 Byte de estado de los contadores rápidos, 10-30  
 Byte, formato de direcciones, 7-2  
 Bytes, y margen de enteros, 7-2  
 Bytes de programación de los contadores rápidos,  
   D-8

## C

Cable de ampliación  
 montaje, 2-5–2-7

Cable de E/S de ampliación  
   datos técnicos, A-81  
   instalación, A-81  
 Cable MPI, 3-8  
 Cable PC/PPI, 9-9  
   ajustar los interruptores DIP para seleccionar la  
     velocidad de transferencia, 9-10  
   ajustar los parámetros, 3-12  
   ajustes de los interruptores DIP, 3-7  
   conexión, 3-7  
   datos técnicos, A-82  
   definición de pines para el interface RS-232,  
     9-10  
   dimensiones, A-83  
   utilización con un módem, 3-19–3-20, 9-12  
   utilización en modo Freeport, 9-10–9-11

Cableado

  circuitos de supresión, 2-13–2-14  
   desmontar módulos, 2-7  
   entradas de los contadores rápidos, 10-26  
   reglas, 2-8–2-13  
     instalación con corriente alterna, 2-10  
     instalación con corriente continua, 2-11

Cableado (diagrama)

  CPU 212 24VAC/DC/relé, A-11  
   CPU 212 AC/AC/AC, A-13, A-17  
   CPU 212 AC/DC tipo fuente/relé, A-15  
   CPU 212 AC/DC/relé, A-9  
   CPU 212 DC/DC/DC, A-7  
   CPU 214 AC/AC/AC, A-25, A-29  
   CPU 214 AC/DC tipo fuente/relé, A-27  
   CPU 214 AC/DC/relé, A-23  
   CPU 214 DC/DC/DC, A-21  
   CPU 215 AC/DC/relé, A-35  
   CPU 215 DC/DC/DC, A-33  
   CPU 216 AC/DC/relé, A-39  
   CPU 216 DC/DC/DC, A-37  
   EM221, 8 entradas digitales AC 120 V, A-41  
   EM221, 8 entradas digitales AC 24 V, A-43  
   EM221, 8 entradas digitales DC 24 V, A-40  
   EM221, 8 entradas digitales tipo fuente  
     DC 24 V, A-42  
   EM222, 8 salidas de relé, A-45  
   EM222, 8 salidas digitales AC 120/230 V, A-47  
   EM222, 8 salidas digitales DC 24 V, A-44  
   EM223, 16 entradas digitales DC 24 V /  
     16 salidas digitales de relé, A-59  
   EM223, 4 entradas digitales AC 120 V /  
     4 salidas digitales AC 120/230 V, A-55  
   EM223, 4 entradas digitales DC 24 V / 4 salidas  
     digitales DC 24 V, A-49  
   EM223, 4 entradas digitales DC 24 V / 4 salidas  
     digitales de relé, A-54  
   EM223, 8 entradas digitales DC 24 V / 8 salidas  
     digitales de relé, A-57

- EM231, 3 entradas analógicas AI de 12 bits, A-60
- EM235 3 entradas analógicas AI / 1 salida analógica AQ de 12 bits, A-70
- Cableado de campo
  - diámetro, 2-8
  - instalación, 2-8
- Cableados, reglas, instalación en Norteamérica, 2-12
- Cables
  - cable de E/S de ampliación, datos técnicos, A-81
  - desmontar módulos, 2-7
  - instalar el cable de ampliación, 2-5-2-7
  - MPI, 3-8
  - número de referencia, G-2
  - para redes PROFIBUS, 9-8
  - PC/PPI, 9-9-9-11
    - ajustar los parámetros, 3-12
    - asignación de pines, A-82
    - datos técnicos, A-82
    - velocidad de transferencia, A-82
- Calcular requisitos de alimentación, 2-15
- Calibración
  - EM231, A-61
  - EM235, A-70, A-72
- Calibración de entradas, EM235, A-72
- Calibrar entradas, EM231, A-62
- Cambiar un puntero, 7-10
- Caracteres, Asistente TD 200, 5-9
- Caracteres internacionales, Asistente TD 200, 5-9
- Cargar, programa, 7-11
- Cargar en la CPU
  - mensajes de error, 4-15
  - modo de operación necesario, 6-13
  - programa de ejemplo, 4-15
  - programas, 3-30
  - requisitos para, 4-15
- Cartucho de memoria
  - códigos de error, C-2
  - copiar en, 7-17
  - datos técnicos, A-78
  - dimensiones, A-78
  - instalar, 7-17
  - número de referencia, G-3
  - restablecer el programa, 7-18
  - retirar, 7-17
  - utilización, 7-17
- Cartucho de pila, 7-11
  - datos técnicos, A-80
  - dimensiones, A-80
  - número de referencia, G-3
- Casilla de verificación
  - no se conectará como único maestro, 3-17
  - red multimaestro, 3-13
- Cerrar, redes, 9-7
- Certificado CE, A-3
- Ciclo
  - bits de estado, D-1
  - interrumpir, 6-11
  - tareas, 6-10
  - y la función de forzado, 6-18
  - y tabla de estado/forzado, 6-17
- Circuitos de seguridad cableados, diseñar, 6-3
- Circuitos de supresión, reglas
  - salidas AC, 2-14
  - transistores en DC, 2-13, 2-14
- Circulación de corriente, impacto en los tiempos de ejecución de las operaciones, F-1
- Codificar, 10-110
- Coherencia, datos, 9-20
- Coherencia de búfers, 9-20
- Coherencia de bytes, 9-20
- Coherencia de datos, CPU 215, 9-20
- Coherencia de palabras, 9-20
- Combinación O con bytes, 10-102
- Combinación O con palabras, 10-103
- Combinación O con palabras dobles, 10-104
- Combinación O-exclusiva con bytes, 10-102
- Combinación O-exclusiva con palabras, 10-103
- Combinación O-exclusiva con palabras dobles, 10-104
- Combinación Y con bytes, 10-102
- Combinación Y con palabras, 10-103
- Combinación Y con palabras dobles, 10-104
- Combinar primer y segundo valor mediante O, 10-99-10-101
- Combinar primer y segundo valor mediante Y, 10-99-10-101
- Comenzar rutina de interrupción, 10-114
- Comenzar subrutina, 10-88
- Comparación, CPUs S7-200, 1-3
- Comparar byte, 10-7
- Comparar entero palabra, 10-7
- Comparar entero palabra doble, 10-8
- Comparar real, 10-8
- Compatibilidad electromagnética, S7-200, A-5
- Compilación, errores
  - reacción del sistema, 6-20
  - violación de reglas, C-4
- Compilar, programa STEP 7-Micro/WIN, 3-29
- Comprobación de datos, 7-8
- Comprobar, programa, 6-16-6-18

## Comunicación

- ajustar parámetros, 3-9
- componentes de redes, 9-6
- comprobar configuración, 3-9
- con esclavos, 3-19
- con unidades periféricas, 9-15
- conectar el PC, 3-7
- configuración, 9-2
- configuración de una CPU 215 en calidad de esclavo DP, 9-17-9-19
- configurar, 3-7-3-24
- configurar durante la instalación, 3-12
- configurar en el Panel de control de Windows, 3-11
- DP, utilizando una CPU 215 en calidad de esclava, 3-19
- eliminar errores, 3-17
- en redes de periferia descentralizada (DP), 9-15-9-26
- estándar DP (periferia descentralizada), utilizando la CPU 215 en calidad de esclavo, 9-15
- hardware
  - instalación en Windows NT, 3-6
  - instalar/desinstalar, 3-4-3-6
- maestros/esclavos, 9-9
- módem, 3-19-3-24
- modo Freeport, 10-124, D-6
- MPI, 3-8, 9-3
- operaciones de, 10-124-10-136
- PPI, 3-7, 9-3
- procesar peticiones, 6-11
- programa de ejemplo para una CPU 215 en calidad de esclava DP, 9-26
- protocolo PROFIBUS-DP, 9-4
- protocolos, 9-2
- protocolos asistidos, 9-2
- seleccionar la parametrización, 3-12-3-13
- utilizando la tarjeta MPI, 9-13-9-14
- utilizando una tarjeta CP, 3-8, 9-13-9-14
- utilizando una tarjeta MPI, 3-8
- utilizar el cable PC/PPI, 9-9-9-11
- velocidades de transferencia, 9-2
- Comunicación DP, utilizando la CPU 215 en calidad de esclava, 3-19
- Comunicación DP (periferia descentralizada), programa de ejemplo, 9-26
- Comunicación en redes de periferia descentralizada, utilizar la CPU 215 en calidad de esclavo, 9-15
- Comunicación en redes de periferia descentralizada (DP)
- Comunicación MPI, 3-8
  - direcciones estándar, 3-17
  - eliminar errores, 3-17
  - tarjetas CP, 9-13

- Comunicación punto a punto, 1-3
- Comunidad Europea (CE), certificado, A-3
- Conceptos de programación, 6-4
- Condensador de alto rendimiento, 7-11
- Condiciones ambientales, A-4
- Conectar la alimentación, respaldar datos, 7-13-7-17
- Conector de bus, 2-5-2-7
  - desmontar módulos de ampliación, 2-7
- Conectores
  - interface de ampliación de bus, 2-5-2-7
  - retirar la tapa, 2-7
  - número de referencia, G-2
- Conectores de bus, redes, 9-7
- Configuración
  - archivo de datos maestros de los dispositivos PROFIBUS (GSD), 9-23-9-25
  - áreas remanentes, 7-15
  - bloque de parámetros, 5-3
  - calcular el consumo de corriente, B-1
  - crear planos, 6-3
  - de un PC con un CP y una unidad de programación, 9-14
  - de un PC con una tarjeta MPI y una unidad de programación, 9-14
  - de una CPU 215 en calidad de esclavo DP, 9-17-9-19
  - EM231, A-61
  - EM235, A-71
  - estado de las salidas, 8-6
  - hardware de comunicación, 3-4
  - maestro DP, 9-19
  - mensajes (TD 200), 5-3, 5-6-5-10
  - preferencias de programación, 3-25
- Configuraciones de E/S asistidas por la CPU 215, 9-19
- Configurar
  - comunicación, 3-7-3-24
  - la comunicación durante la instalación, 3-12
  - la comunicación en el Panel de control de Windows, 3-11
  - parámetros de comunicación, 3-9
- Congelar salidas, 8-6
- Consideraciones
  - al utilizar la operación Borrar temporizador de vigilancia, 10-85
  - entornos con vibraciones fuertes, 2-6
  - montaje del hardware, 2-2-2-4
  - montaje vertical, 2-6
  - utilización de frenos, 2-6
- Constantes, 7-8
- Consumo de corriente, tabla, B-1
- Contactos de comparación, ejemplo, 10-9
- Contador rápido, 8-7
  - ejemplos, 10-36

- Contadores
  - CPU 212/214/215/216, 10-2
  - direccionamiento, 7-5
  - tipos, 7-5
  - variables, 7-5
- Contadores rápidos, 10-21–10-40
  - ajustar los valores actuales y predeterminados, 10-29
  - byte de control, 10-28
  - byte de estado, 10-30
  - bytes de programación (SMB36 – SMB65), D-8
  - cableado de las entradas, 10-26
  - cambio de sentido, 10-35
  - cargar nuevo valor actual/predeterminado, 10-35
  - cronogramas, 10-22–10-25
  - direccionamiento, 7-7
  - ejemplos, 10-22–10-25
  - elegir el nivel de actividad, 10-28
  - inhibir, 10-35
  - interrupciones, 10-30
  - modos de inicialización, 10-31–10-34
  - modos de operación, 10-27
  - utilización, 10-22
- Contar adelante, 10-19
- Contar adelante/atrás, 10-19
- Contraseña
  - borrar, 6-15
  - configurar, 6-14
  - CPU, 6-14
  - habilitar la protección con contraseña (TD 200), 5-4
  - nivel de protección, 6-14
  - olvidada, 6-15
  - restringir el acceso, 6-14
- Control de escritura, D-6
- Control de interrupción de caracteres, 10-129
- Control del modo, lazos PID, 10-61
- Control directo de las entradas y salidas, 6-12
- Convertir
  - archivos de STEP 7-Micro/DOS, E-4
  - enteros de 16 bits a números reales, 10-59
  - entradas del lazo, 10-59
  - guardar un programa convertido, E-6
  - números reales a valores normalizados, 10-59
  - reglas y restricciones, E-5
- Convertir de ASCII a hexadecimal, 10-112
- Convertir de BCD a entero, 10-108
- Convertir de entero a BCD, 10-108
- Convertir de entero de palabra doble a real, 10-108
- Convertir de hexadecimal a ASCII, 10-112
- Copiar segundo valor, 10-99–10-101
- CP (procesador de comunicaciones), 9-13
  - conexión, 3-8
  - configuración con un PC, 9-14
- CP 5411, 9-13
  - ajustar los parámetros de la tarjeta MPI (MPI), 3-16–3-17
  - ajustar los parámetros de la tarjeta MPI (PPI), 3-14
  - número de referencia, G-2
- CP 5511, 9-13
  - ajustar los parámetros de la tarjeta MPI (MPI), 3-16–3-17
  - ajustar los parámetros de la tarjeta MPI (PPI), 3-14
  - número de referencia, G-2
- CP 5611, 9-13
  - ajustar los parámetros de la tarjeta MPI (MPI), 3-16–3-17
  - ajustar los parámetros de la tarjeta MPI (PPI), 3-14
  - número de referencia, G-2
- CPU
  - ajustar el modo de operación, 6-13
  - áreas de memoria, 7-2
  - áreas de operandos, 10-3
  - borrar memoria, 6-15
  - cargar un programa de STEP 7-Micro/WIN, 3-30
  - ciclo, 6-10
  - conectar un módem, 3-19–3-24
  - contraseña, 6-14
  - datos técnicos generales, A-4
  - desmontaje, 2-7
  - dimensiones
    - CPU 212, 2-3
    - CPU 214, 2-3
    - CPU 215, 2-4
    - CPU 216, 2-4
    - módulos de ampliación, 2-4
    - tornillos para el montaje, 2-3–2-5
  - eliminación de errores, 6-19
  - errores fatales, C-2
  - espacio necesario para el montaje, 2-2
  - funcionamiento básico, 6-4
  - identificador (SMB6), D-4
  - montaje
    - armario eléctrico, 2-5
    - cable de ampliación, 2-5–2-7
    - orientación correcta del módulo, 2-5–2-8
    - perfil soporte, 2-6
  - números de referencia, G-1
  - pila lógica, 6-6
  - procedimiento, desmontaje, 2-7
  - protocolos de comunicación, 9-2
  - requisitos de alimentación, 2-15
  - tornillos para el montaje, 2-3–2-5

## CPU 212

- áreas de operandos, 10-3
- comunicación, 9-2
- copias de seguridad, 1-3
- datos técnicos, A-6–A-15
  - simulador de entradas, A-84
- E/S, 1-3
- ejemplo de numeración de E/S, 8-3
- eventos de interrupción, 10-117
- filtros de entrada, 1-3
- funciones, 10-2
- hardware asistido para la comunicación en redes, 3-4
- interrupciones, número máximo, 10-120
- interrupciones asistidas, 1-3, 10-118
- memoria, 1-3
  - márgenes, 10-2
- módulo, 1-5
- módulos de ampliación, 1-3
- número de referencia, G-1
- operaciones, tiempos de ejecución, F-1–F-10
- operaciones asistidas, 1-3
  - Asignar, 10-10
  - Asignar directamente, 10-10
  - Asociar/Desasociar interrupción, 10-116
  - Borrar temporizador de vigilancia, 10-85
  - Cambio de señal, 10-5
  - Cargar directamente/Cargar valor negado directamente, 10-4
  - Cargar relé de control secuencial, 10-92
  - Cargar/Cargar valor negado, 10-4
  - Codificar, 10-110
  - Combinación O con palabras, 10-103
  - Combinación O con palabras dobles, 10-104
  - Combinación O-exclusiva con palabras, 10-103
  - Combinación O-exclusiva con palabras dobles, 10-104
  - Combinación Y con palabras, 10-103
  - Combinación Y con palabras dobles, 10-104
  - Combinar primer y segundo valor mediante O, 10-99
  - Combinar primer y segundo valor mediante Y, 10-99
  - Comenzar rutina de interrupción, 10-114
  - Comenzar subrutina, 10-88
  - Comparar byte, 10-7
  - Comparar entero palabra, 10-7
  - Comparar entero palabra doble, 10-8
  - Contactos directos, 10-4
  - Contactos estándar, 10-4
  - Contar adelante, 10-19
  - Contar adelante/atrás, 10-19
  - Convertir de ASCII a hexadecimal, 10-112
  - Convertir de BCD a entero, 10-108
  - Convertir de entero a BCD, 10-108
  - Convertir de hexadecimal a ASCII, 10-112
  - Copiar segundo valor, 10-99
  - Decodificar, 10-110
  - Decrementar palabra, 10-66
  - Decrementar palabra doble, 10-67
  - Definir modo para contador rápido, 10-21
  - Desplazar palabra a la derecha/Desplazar palabra a la izquierda, 10-80
  - Desplazar palabra doble a la derecha/Desplazar palabra doble a la izquierda, 10-81
  - Detectar flanco positivo y negativo, 10-5
  - Dividir enteros de 16 bits, 10-52
  - Duplicar primer valor, 10-99
  - END/MEND, 10-84
  - Fin condicional/fin absoluto, 10-84
  - Fin del relé de control secuencial, 10-92
  - Habilitar/Inhibir todos los eventos de interrupción, 10-116
  - Incrementar palabra, 10-66
  - Incrementar palabra doble, 10-67
  - Inicializar memoria, 10-72
  - Invertir bytes de una palabra, 10-70
  - Invertir palabra, 10-106
  - Invertir palabra doble, 10-106
  - Llamar subrutina, 10-88
  - Multiplicar enteros de 16 bits, 10-52
  - NOT, 10-5
  - O directa/O-NO directa, 10-4
  - O/O-NO, 10-4
  - Operación nula, 10-11
  - Poner a 1 directamente/Poner a 0 directamente, 10-11
  - Poner a 1/Poner a 0, 10-10
  - Registro de desplazamiento, 10-78
  - Restar enteros de 16 bits, 10-50
  - Restar enteros de 32 bits, 10-50
  - Retorno condicional/absoluto de subrutina, 10-88
  - Retorno condicional/absoluto desde rutina de interrupción, 10-114
  - Rotar palabra a la derecha/Rotar palabra a la izquierda, 10-82
  - Rotar palabra doble a la derecha/Rotar palabra doble a la izquierda, 10-82
  - Sacar primer valor, 10-99
  - Saltar a meta/Definir meta, 10-87
  - Segmento, 10-110
  - STOP, 10-84
  - Sumar enteros de 16 bits, 10-50
  - Sumar enteros de 32 bits, 10-50
  - Temporizador de retardo a la conexión, 10-13
  - Temporizador de retardo a la conexión memorizado, 10-13
  - Transferir byte, 10-68
  - Transferir bytes en bloque, 10-69

- Transferir palabra, 10-68
- Transferir palabra doble, 10-68
- Transferir palabras en bloque, 10-69
- Transición del relé de control secuencial, 10-92
- Transmitir mensaje, 10-124
- Y directa/Y-NO directa, 10-4
- Y/Y-NO, 10-4
- protocolos asistidos, 1-3
- puertos de comunicación, 1-3
- resumen, 1-3
- velocidades de transferencia asistidas, 9-2
- CPU 214
  - áreas de operandos, 10-3
  - comunicación, 9-2
  - copias de seguridad, 1-3
  - datos técnicos, A-20–A-29
    - simulador de entradas, A-85
  - E/S, 1-3
  - ejemplo de numeración de E/S, 8-3
  - eventos de interrupción, 10-117
  - filtros de entrada, 1-3
  - funciones, 10-2
  - hardware asistido para la comunicación en redes, 3-4
  - interrupciones, número máximo, 10-120
  - interrupciones asistidas, 1-3, 10-118
  - memoria, 1-3
    - áreas, 10-2
  - módulo, 1-5
  - módulos de ampliación, 1-3
  - número de referencia, G-1
  - operaciones, tiempos de ejecución, F-1–F-10
  - operaciones asistidas, 1-3
    - Ajustar reloj de tiempo real, 10-49
    - Asignar, 10-10
    - Asignar directamente, 10-10
    - Asociar/Desasociar interrupción, 10-116
    - Borrar primer registro de la tabla, 10-75
    - Borrar temporizador de vigilancia, 10-85
    - Borrar último registro de la tabla, 10-74
    - Buscar valor en tabla, 10-76
    - Cambio de señal, 10-5
    - Cargar directamente/Cargar valor negado directamente, 10-4
    - Cargar relé de control secuencial, 10-92
    - Cargar/Cargar valor negado, 10-4
    - Codificar, 10-110
    - Combinación O con palabras, 10-103
    - Combinación O con palabras dobles, 10-104
    - Combinación O-exclusiva con palabras, 10-103
    - Combinación O-exclusiva con palabras dobles, 10-104
    - Combinación Y con palabras, 10-103
    - Combinación Y con palabras dobles, 10-104
    - Combinar primer y segundo valor mediante O, 10-99
    - Combinar primer y segundo valor mediante Y, 10-99
    - Comenzar rutina de interrupción, 10-114
    - Comenzar subrutina, 10-88
    - Comparar byte, 10-7
    - Comparar entero palabra, 10-7
    - Comparar entero palabra doble, 10-8
    - Comparar real, 10-8
    - Contactos directos, 10-4
    - Contactos estándar, 10-4
    - Contar adelante, 10-19
    - Contar adelante/atrás, 10-19
    - Convertir de ASCII a hexadecimal, 10-112
    - Convertir de BCD a entero, 10-108
    - Convertir de entero a BCD, 10-108
    - Convertir de entero de palabra doble a real, 10-108
    - Convertir de hexadecimal a ASCII, 10-112
    - Copiar segundo valor, 10-99
    - Decodificar, 10-110
    - Decrementar palabra, 10-66
    - Decrementar palabra doble, 10-67
    - Definir modo para contador rápido, 10-21
    - Desplazar palabra a la derecha/Desplazar palabra a la izquierda, 10-80
    - Desplazar palabra doble a la derecha/Desplazar palabra doble a la izquierda, 10-81
    - Detectar flanco positivo y negativo, 10-5
    - Dividir enteros de 16 bits, 10-52
    - Dividir reales, 10-53
    - Duplicar primer valor, 10-99
    - END/MEND, 10-84
    - Fin condicional/fin absoluto, 10-84
    - Fin del relé de control secuencial, 10-92
    - FOR/NEXT, 10-90
    - Habilitar/Inhibir todos los eventos de interrupción, 10-116
    - Incrementar palabra, 10-66
    - Incrementar palabra doble, 10-67
    - Inicializar memoria, 10-72
    - Invertir bytes de una palabra, 10-70
    - Invertir palabra, 10-106
    - Invertir palabra doble, 10-106
    - Leer de la red/Escribir en la red, 10-133
    - Leer reloj de tiempo real, 10-49
    - Llamar subrutina, 10-88
    - Multiplicar enteros de 16 bits, 10-52
    - Multiplicar reales, 10-53
    - NEXT, 10-90
    - NOT, 10-5
    - O directa/O-NO directa, 10-4
    - O/O-NO, 10-4
    - Operación nula, 10-11

- Poner a 1 directamente/Poner a 0 directamente, 10-11
- Poner a 1/Poner a 0, 10-10
- Raíz cuadrada, 10-53
- Registrar valor en tabla, 10-73
- Registro de desplazamiento, 10-78
- Restar enteros de 16 bits, 10-50
- Restar enteros de 32 bits, 10-50
- Restar reales, 10-51
- Retorno condicional/absoluto de subrutina, 10-88
- Retorno condicional/absoluto desde rutina de interrupción, 10-114
- Rotar palabra a la derecha/Rotar palabra a la izquierda, 10-82
- Rotar palabra doble a la derecha/Rotar palabra doble a la izquierda, 10-82
- Sacar primer valor, 10-99
- Salida de impulsos, 10-37
- Saltar a meta/Definir meta, 10-87
- Segmento, 10-110
- STOP, 10-84
- Sumar enteros de 16 bits, 10-50
- Sumar enteros de 32 bits, 10-50
- Sumar reales, 10-51
- Temporizador de retardo a la conexión, 10-13
- Temporizador de retardo a la conexión memorizado, 10-13
- Transferir byte, 10-68
- Transferir bytes en bloque, 10-69
- Transferir palabra, 10-68
- Transferir palabra doble, 10-68
- Transferir palabras en bloque, 10-69
- Transferir real, 10-68
- Transición del relé de control secuencial, 10-92
- Transmitir mensaje, 10-124
- Truncar, 10-108
- Y directa/Y-NO directa, 10-4
- Y/Y-NO, 10-4
- protocolos asistidos, 1-3
- puertos de comunicación, 1-3
- resumen, 1-3
- velocidades de transferencia asistidas, 9-2
- CPU 215
  - áreas de operandos, 10-3
  - búfer de entrada, 9-18, 9-21
  - búfer de salida, 9-18, 9-21
  - coherencia de datos, 9-20
  - como esclavo DP, 3-19
  - comunicación, 9-2
  - configuraciones de E/S asistidas, 9-19
  - configurar en calidad de esclavo DP, 9-17–9-19
  - copias de seguridad, 1-3
  - datos técnicos, A-32–A-35
  - simulador de entradas, A-86
  - E/S, 1-3
  - ejemplo de numeración de E/S, 8-3
  - en calidad de esclava, 3-19
  - en calidad de esclavo DP, 9-15
  - eventos de interrupción, 10-117
  - filtros de entrada, 1-3
  - funciones, 10-2
  - hardware asistido para la comunicación en redes, 3-4
  - información de estado en calidad de esclavo DP, 9-21
  - intercambio de datos con el maestro DP, 9-21
  - interface DP, 3-19
  - interrupciones, número máximo, 10-120
  - interrupciones asistidas, 1-3, 10-118
  - LEDs de estado para la comunicación DP, 9-22
  - memoria, 1-3
    - áreas, 10-2
  - módulo, 1-5
  - módulos de ampliación, 1-3
  - número de referencia, G-1
  - operaciones, tiempos de ejecución, F-1–F-10
  - operaciones asistidas, 1-3
    - Ajustar reloj de tiempo real, 10-49
    - Asignar, 10-10
    - Asignar directamente, 10-10
    - Asociar/Desasociar interrupción, 10-116
    - Borrar primer registro de la tabla, 10-75
    - Borrar temporizador de vigilancia, 10-85
    - Borrar último registro de la tabla, 10-74
    - Buscar valor en tabla, 10-76
    - Cambio de señal, 10-5
    - Cargar directamente/Cargar valor negado directamente, 10-4
    - Cargar relé de control secuencial, 10-92
    - Cargar/Cargar valor negado, 10-4
    - Codificar, 10-110
    - Combinación O con bytes, 10-102
    - Combinación O con palabras, 10-103
    - Combinación O con palabras dobles, 10-104
    - Combinación O-exclusiva con bytes, 10-102
    - Combinación O-exclusiva con palabras, 10-103
    - Combinación O-exclusiva con palabras dobles, 10-104
    - Combinación Y con bytes, 10-102
    - Combinación Y con palabras, 10-103
    - Combinación Y con palabras dobles, 10-104
    - Combinar primer y segundo valor mediante O, 10-99
    - Combinar primer y segundo valor mediante Y, 10-99
    - Comenzar rutina de interrupción, 10-114
    - Comenzar subrutina, 10-88
    - Comparar byte, 10-7

- Comparar entero palabra, 10-7
- Comparar entero palabra doble, 10-8
- Comparar real, 10-8
- Contactos directos, 10-4
- Contactos estándar, 10-4
- Contar adelante, 10-19
- Contar adelante/atrás, 10-19
- Convertir de ASCII a hexadecimal, 10-112
- Convertir de BCD a entero, 10-108
- Convertir de entero a BCD, 10-108
- Convertir de entero de palabra doble a real, 10-108
- Convertir de hexadecimal a ASCII, 10-112
- Copiar segundo valor, 10-99
- Decodificar, 10-110
- Decrementar byte, 10-66
- Decrementar palabra, 10-66
- Decrementar palabra doble, 10-67
- Definir modo para contador rápido, 10-21
- Desplazar byte a la derecha/Desplazar byte a la izquierda, 10-80
- Desplazar palabra a la derecha/Desplazar palabra a la izquierda, 10-80
- Desplazar palabra doble a la derecha/Desplazar palabra doble a la izquierda, 10-81
- Detectar flanco positivo y negativo, 10-5
- Dividir enteros de 16 bits, 10-52
- Dividir reales, 10-53
- Duplicar primer valor, 10-99
- END/MEND, 10-84
- Fin condicional/fin absoluto, 10-84
- Fin del relé de control secuencial, 10-92
- FOR/NEXT, 10-90
- Habilitar/Inhibir todos los eventos de interrupción, 10-116
- Incrementar byte, 10-66
- Incrementar palabra, 10-66
- Incrementar palabra doble, 10-67
- Inicializar memoria, 10-72
- Invertir byte, 10-106
- Invertir bytes de una palabra, 10-70
- Invertir palabra, 10-106
- Invertir palabra doble, 10-106
- Leer de la red/Escribir en la red, 10-133
- Leer reloj de tiempo real, 10-49
- Llamar subrutina, 10-88
- Multiplicar enteros de 16 bits, 10-52
- Multiplicar reales, 10-53
- NEXT, 10-90
- NOT, 10-5
- O directa/O-NO directa, 10-4
- O/O-NO, 10-4
- Operación nula, 10-11
- Poner a 1 directamente/Poner a 0 directamente, 10-11
- Poner a 1/Poner a 0, 10-10
- Raíz cuadrada, 10-53
- Recibir mensaje, 10-124
- Registrar valor en tabla, 10-73
- Registro de desplazamiento, 10-78
- Regulación PID, 10-55
- Restar enteros de 16 bits, 10-50
- Restar enteros de 32 bits, 10-50
- Restar reales, 10-51
- Retorno condicional/absoluto de subrutina, 10-88
- Retorno condicional/absoluto desde rutina de interrupción, 10-114
- Rotar byte a la derecha/Rotar byte a la izquierda, 10-81
- Rotar palabra a la derecha/Rotar palabra a la izquierda, 10-82
- Rotar palabra doble a la derecha/Rotar palabra doble a la izquierda, 10-82
- Sacar primer valor, 10-99
- Salida de impulsos, 10-37
- Saltar a meta/Definir meta, 10-87
- Segmento, 10-110
- STOP, 10-84
- Sumar enteros de 16 bits, 10-50
- Sumar enteros de 32 bits, 10-50
- Sumar reales, 10-51
- Temporizador de retardo a la conexión, 10-13
- Temporizador de retardo a la conexión memorizado, 10-13
- Transferir byte, 10-68
- Transferir bytes en bloque, 10-69
- Transferir palabra, 10-68
- Transferir palabra doble, 10-68
- Transferir palabras dobles en bloque, 10-69
- Transferir palabras en bloque, 10-69
- Transferir real, 10-68
- Transición del relé de control secuencial, 10-92
- Transmitir mensaje, 10-124
- Truncar, 10-108
- Y directa/Y-NO directa, 10-4
- Y/Y-NO, 10-4
- programa de ejemplo para un esclavo DP, 9-26
- protocolos asistidos, 1-3
- puertos de comunicación, 1-3
- reglas de configuración, 9-19
- resumen, 1-3
- tamaño del búfer de datos, 9-19
- velocidades de transferencia asistidas, 9-2
- CPU 216
  - áreas de operandos, 10-3
  - comunicación, 9-2
  - copias de seguridad, 1-3
  - datos técnicos, A-36–A-39
  - simulador de entradas, A-86

- E/S, 1-3
- ejemplo de numeración de E/S, 8-4
- eventos de interrupción, 10-117
- filtros de entrada, 1-3
- funciones, 10-2
- hardware asistido para la comunicación en redes, 3-4
- interrupciones, número máximo, 10-120
- interrupciones asistidas, 1-3, 10-118
- memoria, 1-3
  - áreas, 10-2
- módulo, 1-5
- módulos de ampliación, 1-3
- número de referencia, G-1
- operaciones, tiempos de ejecución, F-1–F-10
- operaciones asistidas, 1-3
  - Ajustar reloj de tiempo real, 10-49
  - Asignar, 10-10
  - Asignar directamente, 10-10
  - Asociar/Desasociar interrupción, 10-116
  - Borrar primer registro de la tabla, 10-75
  - Borrar temporizador de vigilancia, 10-85
  - Borrar último registro de la tabla, 10-74
  - Buscar valor en tabla, 10-76
  - Cambio de señal, 10-5
  - Cargar directamente/Cargar valor negado directamente, 10-4
  - Cargar relé de control secuencial, 10-92
  - Cargar/Cargar valor negado, 10-4
  - Codificar, 10-110
  - Combinación O con bytes, 10-102
  - Combinación O con palabras, 10-103
  - Combinación O con palabras dobles, 10-104
  - Combinación O-exclusiva con bytes, 10-102
  - Combinación O-exclusiva con palabras, 10-103
  - Combinación O-exclusiva con palabras dobles, 10-104
  - Combinación Y con bytes, 10-102
  - Combinación Y con palabras, 10-103
  - Combinación Y con palabras dobles, 10-104
  - Combinar primer y segundo valor mediante O, 10-99
  - Combinar primer y segundo valor mediante Y, 10-99
  - Comenzar rutina de interrupción, 10-114
  - Comenzar subrutina, 10-88
  - Comparar byte, 10-7
  - Comparar entero palabra, 10-7
  - Comparar entero palabra doble, 10-8
  - Comparar real, 10-8
  - Contactos directos, 10-4
  - Contactos estándar, 10-4
  - Contar adelante, 10-19
  - Contar adelante/atrás, 10-19
  - Convertir de ASCII a hexadecimal, 10-112
  - Convertir de BCD a entero, 10-108
  - Convertir de entero a BCD, 10-108
  - Convertir de entero de palabra doble a real, 10-108
  - Convertir de hexadecimal a ASCII, 10-112
  - Copiar segundo valor, 10-99
  - Decodificar, 10-110
  - Decrementar byte, 10-66
  - Decrementar palabra, 10-66
  - Decrementar palabra doble, 10-67
  - Definir modo para contador rápido, 10-21
  - Desplazar byte a la derecha/Desplazar byte a la izquierda, 10-80
  - Desplazar palabra a la derecha/Desplazar palabra a la izquierda, 10-80
  - Desplazar palabra doble a la derecha/Desplazar palabra doble a la izquierda, 10-81
  - Detectar flanco positivo y negativo, 10-5
  - Dividir enteros de 16 bits, 10-52
  - Dividir reales, 10-53
  - Duplicar primer valor, 10-99
  - END/MEND, 10-84
  - Fin condicional/fin absoluto, 10-84
  - Fin del relé de control secuencial, 10-92
  - FOR/NEXT, 10-90
  - Habilitar/Inhibir todos los eventos de interrupción, 10-116
  - Incrementar byte, 10-66
  - Incrementar palabra, 10-66
  - Incrementar palabra doble, 10-67
  - Inicializar memoria, 10-72
  - Invertir byte, 10-106
  - Invertir bytes de una palabra, 10-70
  - Invertir palabra, 10-106
  - Invertir palabra doble, 10-106
  - Leer de la red/Escribir en la red, 10-133
  - Leer reloj de tiempo real, 10-49
  - Llamar subrutina, 10-88
  - Multiplicar enteros de 16 bits, 10-52
  - Multiplicar reales, 10-53
  - NEXT, 10-90
  - NOT, 10-5
  - O directa/O-NO directa, 10-4
  - O/O-NO, 10-4
  - Operación nula, 10-11
  - Poner a 1 directamente/Poner a 0 directamente, 10-11
  - Poner a 1/Poner a 0, 10-10
  - Raíz cuadrada, 10-53
  - Recibir mensaje, 10-124
  - Registrar valor en tabla, 10-73
  - Registro de desplazamiento, 10-78
  - Regulación PID, 10-55
  - Restar enteros de 16 bits, 10-50
  - Restar enteros de 32 bits, 10-50
  - Restar reales, 10-51

Retorno condicional/absoluto de subrutina, 10-88  
 Retorno condicional/absoluto desde rutina de interrupción, 10-114  
 Rotar byte a la derecha/Rotar byte a la izquierda, 10-81  
 Rotar palabra a la derecha/Rotar palabra a la izquierda, 10-82  
 Rotar palabra doble a la derecha/Rotar palabra doble a la izquierda, 10-82  
 Sacar primer valor, 10-99  
 Salida de impulsos, 10-37  
 Saltar a meta/Definir meta, 10-87  
 Segmento, 10-110  
 STOP, 10-84  
 Sumar enteros de 16 bits, 10-50  
 Sumar enteros de 32 bits, 10-50  
 Sumar reales, 10-51  
 Temporizador de retardo a la conexión, 10-13  
 Temporizador de retardo a la conexión memorizado, 10-13  
 Transferir byte, 10-68  
 Transferir bytes en bloque, 10-69  
 Transferir palabra, 10-68  
 Transferir palabra doble, 10-68  
 Transferir palabras dobles en bloque, 10-69  
 Transferir palabras en bloque, 10-69  
 Transferir real, 10-68  
 Transición del relé de control secuencial, 10-92  
 Transmitir mensaje, 10-124  
 Truncar, 10-108  
 Y directa/Y-NO directa, 10-4  
 Y/Y-NO, 10-4  
 protocolos asistidos, 1-3  
 puertos de comunicación, 1-3  
 resumen, 1-3  
 velocidades de transferencia asistidas, 9-2  
 Crear, proyectos de STEP 7-Micro/WIN, 3-26  
 Crear un programa, ejemplo: ajustar una interrupción temporizada, 6-9  
 Cuadro de diálogo  
   "Interface PG/PC", 3-10  
   "Recursos" para Windows NT, 3-6  
 Cuadro de diálogo "Instalar/desinstalar tarjetas", 3-3  
 Cuadro Registro de desplazamiento (SHRB), 10-78  
 Cuadro Salida de impulsos (PLS), 8-7, 10-37

## D

### Datos técnicos

cable de E/S de ampliación, A-81  
 cable PC/PPI, A-82  
 cartucho de memoria, A-78  
 cartucho de pila, A-80  
 CPU 212, A-6–A-15  
 CPU 214, A-20–A-29  
 CPU 215, A-32–A-35  
 CPU 216, A-36–A-39  
 EM221, A-40–A-43  
 EM222, A-44–A-46  
 EM223, A-48–A-54  
 EM231, A-60–A-64  
 EM235, A-69–A-75  
 gama S7-200, A-4  
 simulador de entradas  
   CPU 212, A-84  
   CPU 214, A-85  
   CPU 215/216, A-86  
 Decodificar, 10-110  
 Decrementar, ejemplo, 10-67  
 Decrementar byte, 10-66  
 Decrementar palabra, 10-66  
 Decrementar palabra doble, 10-67  
 Definir mensajes (TD 200), 5-8  
 Definir meta, 10-87  
 Definir modo para contador rápido, 10-21  
   modo de conteo, 10-28  
 Definir modo para contador rápido (HDEF), 10-21  
 Desasociar interrupción, 10-116  
 Desbordamiento de la cola de espera (SMB4), D-3  
 Desmontaje  
   CPU, 2-7  
   dimensiones  
     CPU 212, 2-3  
     CPU 214, 2-3  
     CPU 215, 2-4  
     CPU 216, 2-4  
   módulos de ampliación, 2-4  
   tornillos para el montaje, 2-3–2-5  
   espacio necesario para el montaje, 2-2  
   módulos de ampliación, 2-7  
   orientación correcta del módulo, 2-7  
   procedimiento, 2-7  
   tapa del interface de ampliación de bus, 2-5–2-7  
   tornillos, 2-3–2-5  
 Desplazamiento, operaciones de, 10-68–10-77

Desplazar byte a la derecha, 10-80  
 Desplazar byte a la izquierda, 10-80  
 Desplazar palabra a la derecha, 10-80  
 Desplazar palabra a la izquierda, 10-80  
 Desplazar palabra doble a la derecha, 10-81  
 Desplazar palabra doble a la izquierda, 10-81  
 Detectar flanco negativo, 10-5  
 Detectar flanco positivo, 10-5  
 Determinación del tipo de datos, 7-8  
 Dimensiones  
   cable PC/PPI, A-83  
   cartucho de memoria, A-78  
   cartucho de pila, A-80  
   CPU 212, 2-3  
   CPU 214, 2-3  
   CPU 215, 2-4  
   CPU 216, 2-4  
   módulos de ampliación, 2-4  
   tornillos para el montaje, 2-3–2-5  
 Diodos supresores, 2-13  
 Dirección de estación más alta, 9-31  
 Direccionamiento  
   acumuladores, 7-6  
   área de marcas, 7-3  
   áreas de memoria, 7-2  
   byte.bit, 7-2  
   contadores, 7-5  
   contadores rápidos, 7-7  
   de dispositivos en una red, 9-2  
   E/S de los módulos de ampliación, 8-2  
   E/S integradas, 8-2  
   entradas analógicas, 7-6  
   imagen del proceso de las entradas, 7-3  
   imagen del proceso de las salidas, 7-3  
   indirecto (punteros), 7-9–7-11  
     modificar un puntero, 7-10  
   lista de elementos utilizados, 5-18  
   marcas especiales, 7-4  
   márgenes, observar, 5-18  
   memoria de variables, 7-3  
   relés de control secuencial, 7-3  
   salidas analógicas, 7-6  
   temporizadores, 7-4  
   utilizar símbolos, 3-36  
 Direccionamiento directo, 7-2  
 Direccionamiento indirecto, 7-9–7-11  
   impacto en los tiempos de ejecución de las  
     operaciones, F-1  
   modificar un puntero, 7-10  
 Direccionamiento simbólico, 3-36  
 Direcciones  
   absolutas, 6-4  
   comunicación MPI, 3-17  
   observar, 5-17, 5-18  
   simbólicas, 6-4  
   tabla de estado/de forzado, 3-35

Direcciones de E/S, de un maestro PROFIBUS-DP,  
 9-18  
 Diseñar una solución de automatización con un Mi-  
 cro-PLC, 6-2  
 Diseño de sistemas, Micro-PLC, 6-2  
 Dispositivos, utilizar maestros no SIMATIC, 9-24  
 Dividir enteros de 16 bits, 10-52  
 Dividir reales, 10-53  
 Duplicar primer valor, 10-99–10-101

## E

E/S analógicas, impacto en los tiempos de ejecu-  
 ción de las operaciones, F-1  
 E/S integradas, direccionamiento, 8-2  
 Editor de bloques de datos, 3-32  
 EEPROM, 7-11, 7-13  
   códigos de error, C-2  
   copiar memoria de variables en, 7-16  
   guardar de la memoria V, D-6  
 Ejemplos  
   agregar visualizadores de textos TD 200 a la  
     red, 9-14  
   ajustar una interrupción temporizada, 6-9  
   archivo GSD, 9-24  
   aritmética, 10-54  
   bloque de datos, 3-32  
   bloque de parámetros, 5-11  
   borrar primer registro de la tabla, 10-75  
   borrar último registro de la tabla, 10-74  
   buscar valor en tabla, 10-77  
   calcular requisitos de alimentación, 2-15  
   contador rápido, 10-36  
     funcionamiento con puesta a 0 y arranque,  
       10-23  
     funcionamiento con puesta a 0 y sin arran-  
       que, 10-22  
   funcionamiento de HSC0 en modo 0 y HSC1  
     o HSC2 en modos 0, 1, ó 2, 10-23  
   funcionamiento de HSC1 o HSC2 en modo  
     3, 4, ó 5, 10-24  
   funcionamiento de HSC1 o HSC2 en modo  
     9, 10 u 11 (frecuencia cuádruple), 10-25  
   funcionamiento de HSC1 o HSC2 en modo  
     9, 10 u 11 (frecuencia simple), 10-25  
   funcionamiento de HSC1 o HSC2, en modo  
     6, 7 u 8, 10-24  
   conversión de números reales, 10-109  
   convertir de ASCII a hexadecimal, 10-113  
   convertir y truncar, 10-109  
   de operación lógica de pila, 10-101–10-103  
   de una operación de contaje, 10-20

- Decodificar/Codificar, 10-111
- decrementar, 10-67
- desplazamiento y rotación, 10-83–10-85
- FOR/NEXT, 10-91–10-93
- incrementar, 10-67
- inicialización de HSC1, 10-21
- Inicializar memoria, 10-72–10-74
- Invertir, 10-107–10-109
- Leer de la red/Escribir en la red, 10-134–10-136
- Llamar subrutina, 10-89–10-91
- Modulación del ancho de impulsos, 10-47
- numeración de E/S, 8-2, 8-3
- operación Registro de desplazamiento, 10-79–10-81
- operaciones con contactos, 10-6
- operaciones con rutinas de interrupción, 10-122
- operaciones con salidas, 10-12
- operaciones de comparación, 10-9
- operaciones de transmisión, 10-130
- potenciómetros analógicos, 8-8
- programa de ejemplo, 4-2
- programa para la comunicación DP, 9-26
- red con token passing, 9-28
- Registrar valor en tabla, 10-73
- regulación PID, 10-63–10-65
- relé de control secuencial, 10-93–10-98
  - convergir cadenas secuenciales, 10-96–10-99
  - dividir cadenas secuenciales, 10-94
  - transiciones condicionales, 10-98
- Saltar a meta, 10-87–10-89
- Segmento, 10-111
- STOP, END y Borrar temporizador de vigilancia, 10-86–10-88
- tabla de estado/de forzado, 3-34
- tabla de símbolos, 3-36
- tarjeta MPI con maestros/esclavos, 3-9
- temporizador de retardo a la conexión, 10-17
- temporizador de retardo a la conexión memorizado, 10-18
- transferencia de bloques, 10-71–10-73
- transferir e invertir, 10-70–10-72
- Tren de impulsos, 10-45
- truncar, 10-109
- Y, O y O-exclusiva, 10-105–10-107
- Eliminación
  - errores de compilación, C-4
  - errores de programación del tiempo de ejecución, C-3
  - errores fatales, C-2
- Eliminación de errores, 6-19
  - contraseña olvidada, 6-15
  - fatales, 6-19
  - instalación de STEP 7-Micro/WIN, 3-2
  - lazo PID, 10-62
  - leer de la red/escribir en la red, 10-133
  - no fatales, 6-20
  - rearrancar la CPU después de un error fatal, 6-19
  - responder a errores, 6-19
- Eliminar errores, comunicación MPI, 3-17
- EM221, datos técnicos, A-40–A-43
- EM222, datos técnicos, A-44–A-46
- EM223, datos técnicos, A-48–A-54
- EM231
  - calibración, A-61
  - configuración, área de entradas analógicas, A-61
  - datos técnicos, A-60–A-64
  - esquema de conexiones de las entradas, A-63
  - formato de la palabra de datos, A-62
  - interruptores DIP, A-61
    - ubicación, A-61
  - reglas de instalación, A-64
- EM235
  - Calibración, A-70
  - configuración, margen de entradas analógicas, A-71
  - datos técnicos, A-69–A-75
  - esquema de conexiones de las entradas, A-73
  - esquema de conexiones de las salidas, A-74
  - formato de la palabra de datos, A-72, A-74
  - interruptores DIP
    - ajustar, A-71
    - ubicación, A-70
  - reglas de instalación, A-75
- END, 10-84
- Enlaces, MPI, lógicos, 9-3, 9-4
- Enlaces lógicos, MPI, 9-3, 9-4
- Enteros de 16 bits, convertir a números reales, 10-59
- Entornos con vibraciones fuertes, utilización de frenos, 2-6
- Entradas, funcionamiento básico, 6-4
- Entradas analógicas
  - acceder a, 6-10
  - direccionamiento, 7-6
  - leer el valor con una rutina de interrupción, 10-123
- Entradas digitales, leer, 6-10

Entradas y salidas rápidas, 8-7  
Equipos necesarios  
  S7-200, 1-2  
  STEP 7-Micro/WIN, 3-1  
Errores  
  de programación del tiempo de ejecución, C-3  
  fatales, C-2  
  lazo PID, 10-62  
  Leer de la red/Escribir en la red, 10-133  
  no fatales, C-3, C-4  
  SMB1, errores de ejecución, D-2  
  violación de reglas de compilación, C-4  
Errores de programación del tiempo de ejecución, C-3  
Errores durante el tiempo de ejecución, reacción del sistema, 6-20  
Errores fatales, C-2  
  y el funcionamiento de la CPU, 6-19  
Errores no fatales  
  eliminación de, 6-20  
  reacción del sistema, 6-20  
  y el funcionamiento de la CPU, 6-20  
Escalar la salida del lazo, 10-60  
Esclavos  
  comunicación, 3-19, 9-9  
  CPU 215 como esclava DP, 3-19  
  CPU 215 en calidad de esclavo DP, 9-15  
Escribir en la red, 10-133  
  ejemplo, 10-134–10-136  
  errores, 10-133  
Espacio necesario para el montaje, 2-2  
Esquema de conexiones de las entradas, EM231, A-63, A-73  
Esquema de conexiones de las salidas, EM235, A-74  
Estación esclava, CPU 215, 3-19  
Estación más alta, dirección. *See* Highest station address  
Estaciones de operador, definir, 6-3  
Estado de las entradas y salidas (SMB5), D-3  
ET 200, manual, G-3

## F

Factor de actualización GAP, 9-31

Fecha, ajustar, 10-49  
Filtrar entradas analógicas, 5-14–5-16  
Filtro de entrada, supresión de ruidos, 8-5  
FOR, 10-90  
Formatear, valores de datos en textos, 5-10  
Formato de la palabra de datos, EM231, A-62  
Formato de la palabra de datos de entrada, EM235, A-72  
Formato de la palabra de datos de salida, EM235, A-74  
Forzado, habilitar en el TD 200, 5-4  
Forzar valores, 6-17  
Forzar variables, tabla de estado/de forzado, 3-35  
Frecuencia de actualización (TD 200), seleccionar, 5-5  
Freeport, definición del modo, 10-118  
Funciones PTO/PWM  
  ancho y conteo de impulsos, 10-39  
  bit de estado, 10-39  
  bits de control, 10-39  
  byte de control, 10-38  
  efectos en las salidas, 10-43  
  imagen del proceso, 10-44  
  inicialización, 10-40  
  pipeline PTO, 10-38  
  registro de control, 10-40  
  registros de control, SMB66–SMB85, D-9  
  tabla de referencia hexadecimal, 10-40  
  tiempo de ciclo, 10-39  
Funciones PTO/PWM (SMB66–SMB85), D-9

## G

Ganancia del lazo, convertir, 10-59  
GSD  
Guardar  
  el programa en la memoria no volátil, 7-16  
  proyectos de STEP 7-Micro/WIN, 3-26  
  valor en EEPROM, D-6

## H

Habilitar todos los eventos de interrupción, 10-116  
Homologaciones, A-3

Hora, ajustar, 10-49

## I

### Identificación de terminales de conexión

- CPU 212 24VAC/DC/relé, A-11
- CPU 212 AC/AC/AC, A-13, A-17
- CPU 212 AC/DC tipo fuente/relé, A-15
- CPU 212 AC/DC/relé, A-9
- CPU 212 DC/DC/DC, A-7
- CPU 214 AC/AC/AC, A-25, A-29
- CPU 214 AC/DC tipo fuente/relé, A-27
- CPU 214 AC/DC/relé, A-23
- CPU 214 DC/DC/DC, A-21
- CPU 215 AC/DC/relé, A-35
- CPU 215 DC/DC/DC, A-33
- CPU 216 AC/DC/relé, A-39
- CPU 216 DC/DC/DC, A-37
- EM221, 8 entradas digitales AC 120 V, A-41
- EM221, 8 entradas digitales AC 24 V, A-43
- EM221, 8 entradas digitales DC 24 V, A-40
- EM221, 8 entradas digitales tipo fuente DC 24 V, A-42
- EM222, 8 salidas de relé, A-45
- EM222, 8 salidas digitales AC 120/230 V, A-47
- EM222, 8 salidas digitales DC 24 V, A-44
- EM223, 16 entradas digitales DC 24 V / 16 salidas digitales de relé, A-59
- EM223, 4 entradas digitales AC 120 V / 4 salidas digitales AC 120/230 V, A-55
- EM223, 4 entradas digitales DC 24 V / 4 salidas digitales DC 24 V, A-49
- EM223, 4 entradas digitales DC 24 V / 4 salidas digitales de relé, A-54
- EM223, 8 entradas digitales DC 24 V / 8 salidas digitales de relé, A-57
- EM231, 3 entradas analógicas AI de 12 bits, A-60
- EM235 3 entradas analógicas AI / 1 salida analógica AQ de 12 bits, A-70
- Idioma, visualizador de textos, 5-4
- Imagen del proceso de las entradas, 6-12
  - direccionamiento, 7-3
  - funcionamiento, 6-10
- Imagen del proceso de las salidas, 6-11, 6-12
  - direccionamiento, 7-3
  - función PTO/PWM, 10-44
- Importar, archivos de STEP 7-Micro/DOS, E-4
- Imprimir, programa AWL o KOP, 5-23
- Incrementar byte, 10-66
- Incrementar palabra, 10-66
- Incrementar palabra doble, 10-67
- Incrementar un puntero, 7-10
- Indicadores (LEDs) de estado, CPU 215 en calidad de esclavo DP, 9-22

Información de estado, CPU 215 en calidad de esclavo DP, 9-21

Información de estado DP, CPU 215 en calidad de esclavo DP, 9-21

Inhibir todos los eventos de interrupción, 10-116

### Inicialización

- contadores rápidos, 10-31–10-34
- función PWM, 10-41
- modo Freeport, 10-126
- PTO/PWM, 10-40
- Tren de impulsos rápidos (PTO), 10-42

Inicializar memoria, 10-68–10-77

### Instalación

- cable de E/S de ampliación, A-81
- cartucho de memoria, 7-17
- dimensiones
  - CPU 212, 2-3
  - CPU 214, 2-3
  - CPU 215, 2-4
  - CPU 216, 2-4
  - módulos de ampliación, 2-4
  - perfil soporte, 2-3
  - tornillos para el montaje, 2-3–2-5
- EM231, A-64
- EM235, A-75
- en entornos con vibraciones fuertes, utilizando frenos, 2-6
- en sentido vertical, utilizando frenos, 2-6
- hardware de comunicación, 3-4–3-6
  - instrucciones especiales para usuarios de Windows NT, 3-6
- Instalación del simulador de entradas DC para las CPUs 215/216, A-86
  - procedimiento
    - armario eléctrico, 2-5
    - módulo de ampliación, 2-5–2-7
    - orientación correcta del módulo, 2-5–2-8
    - perfil soporte, 2-6
- simulador de entradas DC para la CPU 212, A-84
- simulador de entradas DC para la CPU 214, A-85
- STEP 7-Micro/WIN
  - Windows 3.1, 3-2
  - Windows 95, 3-2
  - Windows NT, 3-2
  - tornillos para el montaje, 2-3–2-5
- Instalación con corriente alterna, reglas, 2-10
- Instalación con corriente continua, reglas, 2-11
- Instalación del simulador de entradas DC, instalación, A-86
- Instalación del simulador de entradas DC para las CPUs 215/216, instalación, A-86
- Instalación en Norteamérica, reglas, 2-12

Intercambio de datos, entre el maestro DP y la CPU  
215, 9-21

Interface de ampliación de bus, retirar la tapa,  
2-5-2-7

Interface de comunicación  
asignación de pines, 9-6  
interrupciones, 10-118

Interface DP, CPU 215, 3-19

Interrupción temporizada, ejemplo, 6-9, 10-123

Interrupciones  
asistencia del sistema, 10-114  
ciclo de la CPU, 6-11  
colas de espera, 10-120  
configurar, 10-116  
contadores rápidos, 10-30  
CPU 212/214/215/216, 10-2  
datos compartidos con el programa principal ,  
10-115  
E/S, 10-118  
flanco positivo/negativo, 10-118  
habilitar e inhibir, 10-116  
marcas para el desbordamiento de colas de es-  
pera, 10-120  
prioridades, 10-120  
rutinas, 10-114  
temporizadas, 10-119, D-7  
    configurar para leer entradas analógicas,  
    10-123  
tipos y números de eventos  
    CPU 212/214/215/216, 10-117  
    prioridades, 10-121  
uso restringido, 10-114

Interrupciones temporizadas, SMB34, SMB35, D-7

Interruptores DIP  
configuración del EM 231, A-61  
configuración del EM235, A-71  
EM235, configuración, A-70

Interruptores DIP, ajustes, Cable PC/PPI, 3-7

Invertir byte, 10-106

Invertir bytes de una palabra, 10-70

Invertir palabra, 10-106

Invertir palabra doble, 10-106

## J

Juego alternativo de caracteres, TD 200, 5-4

Juego de parámetros del módulo  
cable PC/PPI (PPI), 3-12-3-13  
seleccionar, 3-12-3-13  
tarjeta MPI (MPI), 3-16-3-17  
tarjeta MPI (PPI), 3-14

## K

### KOP

cambiar a AWL, 3-31  
editor, 3-27  
elementos básicos, 6-5  
estado del programa, 6-17  
imprimir un programa, 5-23  
introducir el programa, 5-21  
programa, introducir en STEP 7-Micro/WIN,  
3-27  
programa de ejemplo, 4-5, 4-10  
visualizar un programa de STEP 7-Micro/WIN,  
3-31

## L

Lazo de regulación (PID), 10-55-10-65

### Lazos PID

acción positiva/negativa, 10-60  
ajustar la suma integral (bias), 10-61  
convertir salidas, 10-60  
elegir el tipo de regulación, 10-58  
errores, 10-62  
márgenes, variables, 10-60  
modos, 10-61  
programa de ejemplo, 10-63-10-65  
tabla del lazo, 10-62

LEDs de estado, CPU 215 en calidad de esclavo  
DP, 9-22

Leer de la red, 10-133

ejemplo, 10-134-10-136  
errores, 10-133

Leer reloj de tiempo real, 10-49

Lenguaje de programación, concepto, 6-5

Lista de elementos utilizados, 5-18

Lista de instrucciones, programa de ejemplo, 4-4

Llamar subrutina, 10-88

## M

### Maestro DP

aplicaciones de configuración, 9-19  
intercambio de datos con la CPU 215, 9-21

Maestro PROFIBUS-DP, área de direcciones de  
E/S, 9-18

## Maestros

- protocolo PROFIBUS-DP, 9-4
- archivo GSD, 9-24
- comunicación, 9-9
- módem, 3-19
- protocolo MPI, 9-3, 9-13
- protocolo PPI, 9-3
- utilizar dispositivos no SIMATIC, 9-24

## Manuales, número de referencia, G-3

## Marcas de habilitación de mensajes (TD 200), 5-7

## Marcas especiales, D-1–D-13

- direccionamiento, 7-4
- SMB0: bits de estado, D-1
- SMB1: bits de estado, D-2
- SMB110–SMB115: estado del protocolo estándar DP, D-12
- SMB186–SMB194: control de recepción de mensajes, D-10
- SMB2: búfer de recepción de caracteres en modo Freeport, D-2
- SMB28 y SMB29: potenciómetros analógicos, D-5
- SMB3: error de paridad en modo Freeport, D-2
- SMB30, SMB130 (control del modo Freeport), 10-126
- SMB30, SMB130: registros de control del modo Freeport, D-6
- SMB31: control de escritura en la memoria no volátil (EEPROM), D-6
- SMB34/SMB35: intervalos de interrupciones temporizadas, D-7
- SMB36–SMB65: bytes de programación de los contadores rápidos, D-8
- SMB4: desbordamiento de la cola de espera, D-3
- SMB5: estado de las entradas y salidas, D-3
- SMB6: identificador de la CPU, D-4
- SMB66–SMB85: funciones PTO/PWM, D-9
- SMB7: reservado, D-4
- SMB8–SMB21: registro de errores e identificadores de los módulos de ampliación, D-4
- SMB86–SMB94: control de recepción de mensajes, D-10
- SMW222–SMW26: tiempos de ciclo, D-5
- SMW32: Control de escritura en la memoria no volátil (EEPROM), D-6

## Márgenes válidos para las CPUs, 10-2

## Memoria

- borrar, 6-15
- lista de elementos utilizados, 5-18

## Memoria de byte, 7-2

## Memoria V, copiar en la EEPROM, 7-16

## Mensajes

- definir, 5-8
- direcciones, 5-7
- formatear valores de datos intercalados, 5-10
- intercalar valores, 5-8
- marcas de habilitación, TD 200, 5-7
- red con token passing (paso de testigo), 9-29
- tamaño/cantidad, 5-6

## Módem

- adaptador de módem nulo, 9-12
- comunicación en redes, 3-19–3-24
- conectar un PC/PG a una CPU, 3-19–3-20
- requisitos para el cable, 3-19
- utilización con el cable PC/PPI, 9-12

## Modificar un puntero (direccionamiento indirecto), 7-10

## Modo Freeport

- control de interrupción de caracteres, 10-129
- funcionamiento, 10-124
- habilitar, 10-125
- inicializar, 10-126
- protocolo definido por el usuario, 9-5
- SMB2, búfer de recepción de caracteres, D-2
- SMB3, error de paridad, D-2
- SMB30, SMB130 (control del modo Freeport), 10-126
- SMB30, SMB130: registros de control del modo Freeport, D-6
- utilizar el cable PC/PPI, 9-10–9-11
- y modos de operación, 10-124

## Modos

## Modos de operación

- bits de estado, D-1
- cambiar, 6-13
- cambiar la CPU a RUN en el programa de ejemplo, 4-15
- comunicación Freeport, 10-124
- contadores rápidos, 10-27
- y la función de forzado, 6-17

Modulación del ancho de impulsos (PWM), 8-7,  
10-37  
  cambiar el ancho de impulsos, 10-38, 10-41  
  ejemplo, 10-47  
  inicializar, 10-41

Módulo de ampliación analógico, direcciona-  
miento, 8-2

Módulo de ampliación digital, direccionamiento,  
8-2

Módulos de ampliación, 1-4  
  cable de ampliación, instalar, 2-5–2-7  
  desmontaje, 2-7  
  dimensiones  
    8, 16 y 32 E/S, 2-4  
    CPU 212, 2-3  
    CPU 214, 2-3  
    CPU 215, 2-4  
    CPU 216, 2-4  
    tornillos para el montaje, 2-3–2-5  
  direccionamiento de E/S, 8-2  
  espacio necesario para el montaje, 2-2  
  identificadores y registro de errores (SMB8 a  
    SMB21), D-4  
  montaje  
    armario eléctrico, 2-5  
    cable de ampliación, 2-5–2-7  
    orientación correcta del módulo, 2-5–2-8  
    perfil soporte, 2-6  
    retirar la tapa del interface de ampliación de  
      bus, 2-5–2-7  
  números de referencia, G-1  
  requisitos de alimentación, 2-15  
  tornillos para el montaje, 2-3–2-5

Montaje  
  configuración, 2-2  
  dimensiones  
    CPU 212, 2-3  
    CPU 214, 2-3  
    CPU 215, 2-4  
    CPU 216, 2-4  
    módulos de ampliación, 2-4  
    perfil soporte, 2-3  
    tornillos para el montaje, 2-3–2-5  
  en entornos con vibraciones fuertes, utilización  
    de frenos, 2-6  
  en sentido vertical, utilizar frenos, 2-6  
  espacio necesario para el montaje, 2-2  
  procedimiento  
    armario eléctrico, 2-5  
    módulo de ampliación, 2-5–2-7  
    orientación correcta del módulo, 2-5–2-8  
    perfil soporte, 2-6  
    tornillos para el montaje, 2-3–2-5

MPI (interface multipunto), protocolo, 9-3  
  velocidad de transferencia, 9-13

MPI, comunicación, 9-3

MPI, enlaces lógicos, 9-3, 9-4

Muestreo de entradas analógicas, 5-14–5-16

Multimaestro, redes, 9-13

Multiplicar enteros de 16 bits, 10-52

Multiplicar reales, 10-53

## N

NEXT, 10-90

Nombres simbólicos, elaborar, 6-3

Normas, nacionales e internacionales, A-3

NOT, 10-5

Números  
  representación de, 7-2  
  utilizar valores de constantes, 7-8

Números de referencia, G-1

## O

OB1 (programa de usuario), 3-27

Observar  
  direcciones, 5-17  
  direcciones/márgenes, 5-18  
  estado del programa, 6-17  
  programa, 6-16–6-18  
  programa de ejemplo, 4-16

Operación nula, 10-11

Operaciones  
  Activar contador rápido (HSC), 10-21  
  ajustar reloj de tiempo real, 10-49  
  aritméticas, 10-50–10-65  
  Asignar (bobina), 10-10  
  Asignar directamente, 10-10  
  Asociar interrupción, 10-116  
  Borrar primer registro de la tabla, 10-75  
  Borrar temporizador de vigilancia, 10-85–10-87  
  Borrar último registro de la tabla, 10-74  
  Buscar valor en tabla, 10-76  
  buscar/reemplazar, 5-19  
  Codificar, 10-110  
  Combinación O con bytes, 10-102  
  Combinación O con palabras, 10-103  
  Combinación O con palabras dobles, 10-104  
  Combinación O-exclusiva con bytes, 10-102  
  Combinación O-exclusiva con palabras, 10-103  
  Combinación O-exclusiva con palabras dobles,  
    10-104  
  Combinación Y con bytes, 10-102  
  Combinación Y con palabras, 10-103  
  Combinación Y con palabras dobles, 10-104  
  Combinar primer y segundo valor mediante O,  
    10-99–10-101  
  Combinar primer y segundo valor mediante Y,  
    10-99–10-101  
  Comenzar rutina de interrupción, 10-114

- Comenzar subrutina, 10-88
- Comparar byte, 10-7
- Comparar entero palabra, 10-7
- Comparar entero palabra doble, 10-8
- Comparar real, 10-8
- con contadores, 10-13–10-49
- con contadores rápidos, 8-7
- con temporizadores, 10-13–10-49
- contactos, 10-4–10-6
- contactos directos, 10-4
- contactos estándar, 10-4
- contador rápido, 10-21
- contadores rápidos, 10-13, 10-21–10-49
- Contar adelante, 10-19
- Contar adelante/atrás, 10-19
- control del programa, 10-84–10-98
- Convertir de ASCII a hexadecimal, 10-112
- Convertir de BCD a entero, 10-108
- Convertir de entero a BCD, 10-108
- Convertir de entero de palabra doble a real, 10-108
- Convertir de hexadecimal a ASCII, 10-112
- Copiar segundo valor, 10-99–10-101
- Cuadro Registro de desplazamiento (SHRB), 10-78
- Cuadro Salida de impulsos (PLS), 8-7, 10-37
- de conversión, 10-108–10-113
- Decodificar, 10-110
- Decrementar, 10-50–10-65
- Decrementar byte, 10-66
- Decrementar palabra, 10-66
- Decrementar palabra doble, 10-67
- Definir modo para contador rápido, 10-21
- Definir modo para contador rápido (HDEF), 10-21
- del reloj de tiempo real, 10-13
- Desasociar interrupción, 10-116
- Desplazar byte a la derecha, 10-80
- Desplazar byte a la izquierda, 10-80
- Desplazar palabra a la derecha, 10-80
- Desplazar palabra a la izquierda, 10-80
- Desplazar palabra doble a la derecha, 10-81
- Desplazar palabra doble a la izquierda, 10-81
- Detectar flanco negativo, 10-5
- Detectar flanco positivo, 10-5
- Dividir enteros de 16 bits, 10-52
- Dividir reales, 10-53
- Duplicar primer valor, 10-99–10-101
- END, 10-84
- Escribir en la red, 10-133
- FOR, 10-90
- Habilitar todos los eventos de interrupción, 10-116
- Incrementar, 10-50–10-65
- Incrementar byte, 10-66
- Incrementar palabra, 10-66
- Incrementar palabra doble, 10-67
- incrementar un puntero, 7-10
- Inhibir todos los eventos de interrupción, 10-116
- Inicializar memoria, 10-68–10-77
- interrupción, 10-114–10-136
- Invertir byte, 10-106
- Invertir bytes de una palabra, 10-70
- Invertir palabra, 10-106
- Invertir palabra doble, 10-106
- Lazo de regulación (PID), 10-55
- Leer de la red, 10-133
- leer reloj de tiempo real, 10-49
- Llamar subrutina, 10-88
- lógicas, 10-102–10-107
- modificar un puntero, 7-10
- Multiplicar enteros de 16 bits, 10-52
- Multiplicar reales, 10-53
- NEXT, 10-90
- NOT, 10-5
- Operación nula, 10-11
- PID, 10-55–10-65
- pila lógica, 10-99–10-101
- Poner a 0, 10-10
- Poner a 0 directamente, 10-11
- Poner a 1, 10-10
- Poner a 1 directamente, 10-11
- Raíz cuadrada, 10-53
- rápidas de salida, 8-7
- rápidas de salida de impulsos, 10-37–10-49
- Recibir mensaje, 10-124
- Registrar valor en tabla, 10-73
- Registro de desplazamiento, 10-78
- Registro de desplazamiento (SHRB), 10-78
- relé de control secuencial, 10-92
- reloj de tiempo real, 10-49
- Restar enteros de 16 bits, 10-50
- Restar enteros de 32 bits, 10-50
- Restar reales, 10-51
- Retorno de subrutina, 10-88
- Retorno desde rutina de interrupción, 10-114
- Rotar byte a la derecha, 10-81
- Rotar byte a la izquierda, 10-81
- Rotar palabra a la derecha, 10-82
- Rotar palabra a la izquierda, 10-82
- Rotar palabra doble a la derecha, 10-82
- Rotar palabra doble a la izquierda, 10-82
- Sacar primer valor, 10-99–10-101
- Salida de impulsos, 10-37
- salida de impulsos, 10-37
- Salida de impulsos (PLS), 8-7, 10-37
- salidas, 10-10–10-12
- Saltar a meta, 10-87
- Segmento, 10-110
- STOP, 10-84
- Sumar enteros de 16 bits, 10-50
- Sumar enteros de 32 bits, 10-50

- Sumar reales, 10-51
- Temporizador de retardo a la conexión, 10-13
- Temporizador de retardo a la conexión memorizado, 10-13
- tiempos de ejecución, F-1–F-9
- Transferir byte, 10-68
- Transferir bytes en bloque, 10-69
- Transferir palabra, 10-68
- Transferir palabra doble, 10-68
- Transferir palabras dobles en bloque, 10-69
- Transferir palabras en bloque, 10-69
- Transferir real, 10-68
- Transmitir mensaje, 10-124
- Truncar, 10-108
- Operaciones aritméticas, 10-50–10-65
  - Dividir enteros de 16 bits, 10-52
  - Dividir reales, 10-53
  - ejemplo, 10-54
  - Multiplicar enteros de 16 bits, 10-52
  - Multiplicar reales, 10-53
  - Raíz cuadrada, 10-53
  - Restar enteros de 16 bits, 10-50
  - Restar enteros de 32 bits, 10-50
  - Restar reales, 10-51
  - Sumar enteros de 16 bits, 10-50
  - Sumar enteros de 32 bits, 10-50
  - Sumar reales, 10-51
- Operaciones con contactos, 10-4–10-6
  - contactos directos, 10-4
  - contactos estándar, 10-4
  - Detectar flanco negativo, 10-5
  - Detectar flanco positivo, 10-5
  - ejemplo, 10-6
  - NOT, 10-5
- Operaciones con contactos directos, 10-4
- Operaciones con contactos estándar, 10-4
- Operaciones con contadores rápidos, 10-13, 10-21–10-49
- Operaciones con salidas, 10-10–10-12
  - Asignar (bobina), 10-10
  - Asignar directamente, 10-10
  - ejemplo, 10-12
  - Operación nula, 10-11
  - Poner a 0, 10-10
  - Poner a 0 directamente, 10-11
  - Poner a 1, 10-10
  - Poner a 1 directamente, 10-11
- Operaciones con salidas de impulsos
- Operaciones de búsqueda, 10-73–10-77
  - Borrar primer registro de la tabla, 10-75
  - Borrar último registro de la tabla, 10-74
  - Buscar valor en tabla, 10-76
  - Registrar valor en tabla, 10-73
- Operaciones de comparación
  - Comparar byte, 10-7
  - Comparar entero palabra, 10-7
  - Comparar entero palabra doble, 10-8
  - Comparar real, 10-8
  - ejemplo, 10-9
- Operaciones de comunicación, 10-124–10-136
  - Escribir en la red, 10-133
  - Leer de la red, 10-133
  - Recibir mensaje, 10-124
  - Transmitir mensaje, 10-124
- Operaciones de conteo, 10-13–10-49
  - Contar adelante, 10-19
  - Contar adelante/atrás, 10-19
  - ejemplo, 10-20
  - funcionamiento, 10-19
- Operaciones de control del programa, 10-84–10-98
  - Borrar temporizador de vigilancia, 10-85–10-87
  - ejemplo, 10-86–10-88
  - Comenzar subrutina, 10-88
  - END, 10-84
  - ejemplo, 10-86–10-88
  - FOR, 10-90
  - FOR/NEXT, ejemplo, 10-91–10-93
  - Llamar subrutina, 10-88
  - ejemplo, 10-89–10-91
  - NEXT, 10-90
  - relé de control secuencial, 10-92
  - Retorno de subrutina, 10-88
  - Saltar a meta, 10-87
  - ejemplo, 10-87–10-89
  - STOP, 10-84
  - ejemplo, 10-86–10-88
- Operaciones de conversión, 10-108–10-113
  - Codificar, 10-110
  - Convertir de ASCII a hexadecimal, 10-112
  - Convertir de BCD a entero, 10-108
  - Convertir de entero a BCD, 10-108
  - Convertir de entero de palabra doble a real, 10-108
  - Convertir de hexadecimal a ASCII, 10-112
  - Decodificar, 10-110
  - Segmento, 10-110
  - Truncar, 10-108
- Operaciones de desplazamiento, 10-68–10-77
  - Desplazar byte a la derecha, 10-80
  - Desplazar byte a la izquierda, 10-80
  - Desplazar palabra a la derecha, 10-80
  - Desplazar palabra a la izquierda, 10-80
  - Desplazar palabra doble a la derecha, 10-81
  - Desplazar palabra doble a la izquierda, 10-81
  - ejemplo de desplazamiento y rotación, 10-83–10-85
  - ejemplo de registro de desplazamiento, 10-79–10-81
  - Registro de desplazamiento, 10-78
- Operaciones de inicialización, ejemplo, 10-72–10-74

- Operaciones de interrupción, 10-114–10-136
  - Asociar interrupción, 10-116
  - Comenzar rutina de interrupción, 10-114
  - Desasociar interrupción, 10-116
  - descripción, 10-116
  - ejemplo, 10-122
  - Habilitar todos los eventos de interrupción, 10-116
  - Inhibir todos los eventos de interrupción, 10-116
  - Retorno desde rutina de interrupción, 10-114
- Operaciones de rotación, 10-68–10-77
  - ejemplo de desplazamiento y rotación, 10-83–10-85
  - Rotar byte a la derecha, 10-81
  - Rotar byte a la izquierda, 10-81
  - Rotar palabra a la derecha, 10-82
  - Rotar palabra a la izquierda, 10-82
  - Rotar palabra doble a la derecha, 10-82
  - Rotar palabra doble a la izquierda, 10-82
- Operaciones de segmentación (operaciones SCR), 10-93
- Operaciones de tabla, 10-73–10-77
  - Borrar primer registro de la tabla, 10-75
  - Borrar último registro de la tabla, 10-74
  - Buscar valor en tabla, 10-76
  - Registrar valor en tabla, 10-73
- Operaciones de temporización, 10-13–10-49
  - ejemplo de un temporizador de retardo a la conexión, 10-17
  - ejemplo de un temporizador de retardo a la conexión memorizado, 10-18
  - Temporizador de retardo a la conexión, 10-13
  - Temporizador de retardo a la conexión memorizado, 10-13
- Operaciones de transferencia, 10-68–10-77
  - ejemplo de transferir e invertir, 10-70–10-72
  - ejemplo de una operación de transferencia de bloques, 10-71–10-73
  - Invertir bytes de una palabra, 10-70
  - Transferir byte, 10-68
  - Transferir bytes en bloque, 10-69
  - Transferir palabra, 10-68
  - Transferir palabra doble, 10-68
  - Transferir palabras dobles en bloque, 10-69
  - Transferir palabras en bloque, 10-69
  - Transferir real, 10-68
- Operaciones del relé de control secuencial, 10-92
  - ejemplos, 10-93–10-97
- Operaciones del reloj de tiempo real, 10-13, 10-49
  - ajustar reloj de tiempo real, 10-49
  - leer reloj de tiempo real, 10-49
- Operaciones incrementar, ejemplo, 10-67
- Operaciones lógicas, 10-102–10-107
  - Combinación O con bytes, 10-102
  - Combinación O con palabras, 10-103
  - Combinación O con palabras dobles, 10-104
  - Combinación O-exclusiva con bytes, 10-102
  - Combinación O-exclusiva con palabras, 10-103
  - Combinación O-exclusiva con palabras dobles, 10-104
  - Combinación Y con bytes, 10-102
  - Combinación Y con palabras, 10-103
  - Combinación Y con palabras dobles, 10-104
  - ejemplo
    - Invertir, 10-107–10-109
    - Y, O y O-exclusiva, 10-105–10-107
  - Invertir byte, 10-106
  - Invertir palabra, 10-106
  - Invertir palabra doble, 10-106
- Operaciones lógicas con contactos, ejemplo, 10-6
- Operaciones lógicas de pilas, 10-99–10-101
  - Combinar primer y segundo valor mediante O, 10-99–10-101
  - Combinar primer y segundo valor mediante Y, 10-99–10-101
  - Copiar segundo valor, 10-99–10-101
  - Duplicar primer valor, 10-99–10-101
  - ejemplo, 10-101–10-103
  - funcionamiento, 10-100
  - Sacar primer valor, 10-99–10-101
- Operaciones para contador rápido, 10-21
  - activar contador rápido, 10-21
- Operaciones para decrementar, 10-50–10-65
  - Decrementar byte, 10-66
  - Decrementar palabra, 10-66
  - Decrementar palabra doble, 10-67
  - Restar enteros de 16 bits, 10-50
  - Restar enteros de 32 bits, 10-50
  - Restar reales, 10-51
- Operaciones para incrementar, 10-50–10-65
  - Incrementar byte, 10-66
  - Incrementar palabra, 10-66
  - Incrementar palabra doble, 10-67
  - Sumar enteros de 16 bits, 10-50
  - Sumar enteros de 32 bits, 10-50
  - Sumar reales, 10-51
- Operaciones PID, ejemplo, 10-63–10-65
- Operaciones rápidas, Salida de impulsos, 10-37
- Operaciones rápidas de salida, cambiar el ancho de impulsos, 8-7
- Operaciones rápidas de salida de impulsos, 10-37–10-49
- Orientación correcta del módulo, 2-5–2-8

## P

- Palabra de datos de entrada, EM235, A-72
- Palabra de datos de salida, EM235, A-74
- Palabras, y margen de enteros, 7-2
- Palabras dobles, y margen de enteros, 7-2
- Parametrización
  - Cable PC/PPI (PPI), 3-12–3-13
  - seleccionar, 3-12–3-13
  - tarjeta MPI (MPI), 3-16–3-17
  - tarjeta MPI (PPI), 3-14
- Parámetros, buscar/reemplazar, 5-19
- PC/PPI, cable, 9-9–9-11
- Perfil soporte
  - desmontaje, 2-7
  - dimensiones, 2-3
  - entornos con vibraciones fuertes, 2-6
  - espacio necesario para el montaje, 2-2–2-4
  - montaje, 2-6
  - montaje vertical, 2-6
  - número de referencia, G-3
  - utilización de frenos, 2-6
- Periferia descentralizada (DP), comunicación en
  - redes, 9-15–9-26
- PID, 10-55–10-65
  - modos, 10-61
- Pila lógica
  - funcionamiento, 6-6
  - relés de control secuencial (SCRs), 10-92
- Polarizar, redes, 9-7
- Poner a 0, 10-10
- Poner a 0 directamente, 10-11
- Poner a 1, 10-10
- Poner a 1 directamente, 10-11
- Potenciómetros
  - EM231, A-61
  - SMB28 y SMB29, 8-8
- Potenciómetros analógicos, 8-8
  - SMB28, SMB29, D-5
- Potenciómetros analógicos SMB28 y SMB29, 8-8
- Potenciómetros analógicos: SMB28 y SMB29, D-5
- PPI (interface punto a punto)
  - comunicación, 3-7
  - conectar cables, 9-9
  - conexión a redes, 9-9
  - protocolo, 9-3
- PPI, comunicación, 9-3
- Preferencias, ajustar, 3-25
- Procesador de comunicaciones (CP), número de
  - referencia, G-2
- PROFIBUS
  - archivo de datos maestros de los dispositivos
    - (GSD), 9-23–9-25
  - asignación de pines, 9-6
  - cables para redes, 9-8
  - coherencia de datos, 9-20
  - repetidores, 9-8
  - PROFIBUS-DP, 9-15, 9-17
    - protocolo, 9-4
  - PROFIBUS-DP, comunicación, 9-4
  - Programa
    - almacenamiento, 7-11–7-14, 7-17
    - cargar, 7-11
    - cargar en la CPU con STEP 7-Micro/WIN, 3-30
    - compilar con STEP 7-Micro/WIN, 3-29
    - comprobar, 6-16–6-18
    - de ejemplo, 4-2–4-19
    - de STEP 7-Micro/WIN, visualizar, 3-31
    - ejecutar, 6-11
    - elementos básicos, 6-8
    - entradas analógicas, 6-10
    - entradas/salidas (E/S), 6-4
    - estructura, 6-8
    - guardar datos en la memoria no volátil, 7-16
    - importar de STEP 7-Micro/DOS, E-4
    - imprimir, 5-23
    - introducir, 5-21
    - introducir comentarios, 5-21
    - observar, 6-16–6-18
    - observar el estado, 6-17
    - reglas y restricciones de conversión, E-5
    - restablecer con el cartucho de memoria, 7-18
    - utilizar subrutinas, 10-88
    - utilizar una tabla de estado/forzado, 6-16
  - Programa de ejemplo
    - cambiar el modo de operación de la CPU, 4-15
    - cargar en la CPU, 4-15
    - compilar, 4-13
    - crear un proyecto, 4-6
    - crear una tabla de estado, 4-14
    - crear una tabla de símbolos, 4-8
    - guardar, 4-13
    - introducir en KOP, 4-10–4-14
    - KOP, 4-5
    - Lista de instrucciones, 4-4
    - observar, 4-16
    - requisitos de sistema, 4-2
    - tareas, 4-3
  - Programa de usuario (OB1), 3-27
  - Programar, preferencias para STEP 7-Micro/WIN,
    - 3-25
  - Programas, crear en STEP 7-Micro/WIN,
    - 3-27–3-31
  - Protocolo definido por el usuario, modo Freeport,
    - 9-5
  - Protocolo estándar DP, vigilar el estado, D-12
  - Protocolos
  - Proyecto
    - cargar en la CPU, 3-30
    - componentes, 3-30
    - crear, 4-6
    - crear en STEP 7-Micro/WIN, 3-26
    - guardar en STEP 7-Micro/WIN, 3-26

programa de ejemplo, 4-6  
 PTO/PWM, 10-38–10-44  
 PTO/PWM, tabla de referencia hexadecimal, 10-40  
 Puesta a tierra para circuitos, reglas de cableado, 2-9  
 Punteros, 7-9–7-11  
   modificar un puntero, 7-10

## R

Raíz cuadrada, 10-53  
 Rearrancar la CPU, después de un error fatal, 6-19  
 Recibir mensaje, 10-124, 10-127  
   SMB86-SMB94, SMB186–SMB194, D-10  
 Red  
   seleccionar la parametrización, 3-12  
   tiempo de rotación del testigo (token), 9-29–9-32  
 Red con token passing, ejemplo, 9-28  
 Red multimaestro  
   tarjeta MPI, 9-13  
   tarjetas CP, 9-13  
 Redes  
   cerrar, 9-7  
   componentes, 9-6  
   conectar cables, 9-9  
   conectores, 9-7  
   configurar la comunicación, 3-7–3-24  
   datos técnicos de los cables, 9-8  
   dirección de estación más alta, 9-31  
   direcciones, 9-2  
   enviar mensajes, 9-29  
   esclavos, 9-2  
   factor de actualización GAP, 9-31  
   instalar hardware de comunicación, 3-4–3-6  
   interface de comunicación, 9-6  
   maestros, 9-2  
   multimaestro, 9-13  
   optimizar el rendimiento, 9-31  
   polarizar, 9-7  
   rendimiento, 9-28  
   repetidores, 9-8  
   restricciones, 9-28  
   segmentos, 9-2  
   utilizar maestros no SIMATIC, 9-24  
 Redes de resistencia/condensador, aplicaciones de relé, 2-14  
 Redes PC/PPI, 9-9  
 Reemplazar, 5-19  
 Referencias cruzadas, 5-17  
 Registrar valor en tabla, 10-73  
 Registro de desplazamiento, 10-78  
 Registro de desplazamiento (SHRB), 10-78

Registro de errores e identificadores de los módulos de ampliación (SMB8-SMB21), D-4  
 Reglas  
   cableado, 2-8  
     aislamiento, 2-9  
   circuitos de supresión, 2-13  
     salidas AC, 2-14  
     transistores en DC, 2-14  
   diseñar una solución de automatización, 6-2–6-4  
   entornos con vibraciones fuertes, 2-6  
   instalación con corriente alterna, 2-10  
   instalación con corriente continua, 2-11  
   instalación del EM235, A-75  
   instalación en Norteamérica, 2-12  
   introducir direcciones simbólicas, 3-36  
   modificar un puntero para el direccionamiento indirecto, 7-10  
   montaje vertical, 2-6  
   puesta a tierra para circuitos, 2-9  
   utilización de frenos, 2-6  
 Reglas de cableado para fases unipolares, 2-10  
 Reglas para el cableado para tres fases, 2-12  
 Reglas para instalaciones aisladas con corriente continua, 2-11  
 Regulación del lazo, convertir entradas, 10-59  
 Regulación PID  
   acción positiva/negativa, 10-60  
   ajustar suma integral (bias), 10-61  
   bits de historial, 10-61  
   convertir entradas, 10-59  
   convertir salidas, 10-60  
   CPU 212/214/215/216, 10-2  
   ejemplo, 10-63–10-65  
   elegir el tipo, 10-58  
   errores, 10-62  
   márgenes/variables, 10-60  
   modos, 10-61  
   programa de ejemplo, 10-63–10-65  
   tabla del lazo, 10-62  
 Relé corriente continua, 2-14  
 Relés, redes de resistencia/condensador, 2-14  
 Relés de control secuencial  
   CPU 212/214/215/216, 10-2  
   direccionamiento, 7-3  
 Reloj  
   bits de estado, D-1  
   habilitar (TD 200), 5-4  
 Reloj de tiempo real, 10-49  
 Reloj de tiempo real (TOD), habilitar el menú (TD 200), 5-4  
 Repetidor, número de referencia, G-2

Repetidores, red PROFIBUS, 9-8  
 Requisitos de alimentación  
     calcular, 2-15  
     CPU, 2-15  
     módulos de ampliación, 2-15  
 Respalda datos, 7-11–7-16  
     áreas remanentes, 7-15  
     cartucho de pila (opcional), 7-11  
     condensador de alto rendimiento, 7-11  
     conectar la alimentación, 7-13–7-17  
     EEPROM, 7-11, 7-13  
     en la EEPROM, 7-16  
 Respalda la memoria, 7-11–7-16  
 Restar enteros de 32 bits, 10-50  
 Restar reales, 10-51  
 Restringir el acceso a la CPU  
 Resumen de las CPUs S7-200  
     áreas de operandos, 10-3  
     funciones, 1-3  
 Resumen de las CPUs S7-200 CPU, áreas de memoria, 10-2  
 Retirar, cartucho de memoria, 7-17  
 Retorno de subrutina, 10-88  
 Retorno desde rutina de interrupción, 10-114  
 Rotación, operaciones de, 10-68–10-77  
 Rotar byte a la derecha, 10-81  
 Rotar byte a la izquierda, 10-81  
 Rotar palabra a la derecha, 10-82  
 Rotar palabra a la izquierda, 10-82  
 Rotar palabra doble a la derecha, 10-82  
 Rotar palabra doble a la izquierda, 10-82  
 RUN, 6-13  
 Rutinas de interrupción, reglas, 6-8

## S

S7-200  
     Asistente de operaciones, 5-12–5-16  
         filtrar entradas analógicas, 5-14–5-16  
     compatibilidad electromagnética, A-5  
     componentes, 1-4  
     Componentes de sistema, 1-2  
     condiciones ambientales, A-4  
     CPU, desmontaje, 2-7  
     datos técnicos, A-4  
     desmontaje, 2-7  
     dimensiones  
         CPU 212, 2-3  
         CPU 214, 2-3  
         CPU 215, 2-4  
         CPU 216, 2-4  
         módulos de ampliación, 2-4  
         tornillos para el montaje, 2-3–2-5  
     espacio necesario para el montaje, 2-2  
     módulos de ampliación, 1-4  
         desmontaje, 2-7

montaje  
     armario eléctrico, 2-5  
     cable de ampliación, 2-5–2-7  
     orientación correcta del módulo, 2-5–2-8  
     perfil soporte, 2-6  
     resumen de las CPUs, 1-3  
     tornillos para el montaje, 2-3–2-5  
 Sacar primer valor, 10-99–10-101  
 Salida de impulsos, 8-7, 10-37  
     cambiar el ancho de impulsos, 10-38  
     función PTO/PWM, 10-38–10-44  
     funciones PTO/PWM, SMB66–SMB85: marcas especiales, D-9  
 Salida de impulsos (PLS), 8-7, 10-37  
 Salidas  
     congelar, 8-6  
     de impulsos rápidos, 8-7  
     funcionamiento básico, 6-4  
 Salidas AC, 2-14  
 Salidas analógicas  
     acceder a, 6-11  
     direccionamiento, 7-6  
 Salidas de impulsos, 8-7  
     funcionamiento, 10-37  
 Salidas digitales, escribir, 6-11  
 Saltar a meta, 10-87  
 Segmento (operaciones de conversión), 10-110  
 Segmentos  
     buscar/reemplazar, 5-19  
     redes, 9-2  
 Selector, de modos de operación, 6-13  
 Símbolos, buscar/reemplazar, 5-19  
 Simulador  
 Simulador de entradas  
     CPU 212, A-84  
     CPU 214, A-85  
     CPU 215/216, A-86  
     número de referencia, G-3  
 Simulador de entradas DC, instalación, A-84, A-85  
 Simulador de entradas DC para la CPU 212,  
     instalación, A-84  
 Simulador de entradas DC para la CPU 214,  
     instalación, A-85  
 SM0.2 (marca Datos remanentes perdidos), 7-14  
 SMB0: bits de estado, D-1  
 SMB1: bits de estado, D-2  
 SMB110–SMB115: estado del protocolo estándar  
     DP, D-12  
 SMB186–SMB194: control de recepción de  
     mensajes, D-10  
 SMB2 (recepción de caracteres en modo Freeport),  
     control de interrupción de caracteres, 10-129  
 SMB2: búfer de recepción de caracteres en modo  
     Freeport, D-2  
 SMB3 (error de paridad en modo Freeport), control  
     de interrupción de caracteres, 10-129  
 SMB3: error de paridad en modo Freeport, D-2

SMB30, SMB130 (control del modo Freeport) , 10-126  
 SMB30, SMB130: registros de control del modo Freeport, D-6  
 SMB34/SMB35: intervalos de interrupciones temporizadas, D-7  
 SMB36–SMB65: bytes de programación de los contadores rápidos, D-8  
 SMB5: estado de las entradas y salidas, D-3  
 SMB6: identificador de la CPU, D-4  
 SMB7 (reservado), D-4  
 SMB86–SMB94: control de recepción de mensajes, D-10  
 SMW22–SMW26: tiempos de ciclo, D-5  
 Software de programación, números de referencia, G-3  
 STEP 7-Micro/DOS, importar archivos, E-4  
 STEP 7-Micro/WIN  
   Ayuda online, 3-1  
   cargar un programa en la CPU, 3-30  
   comunicación con módems, 3-19–3-24  
   configurar la comunicación en, 3-10  
   crear programas, 3-27–3-31  
   crear un proyecto, 3-26  
   eliminación de errores durante la instalación, 3-2  
   equipos necesarios, 3-1  
   guardar un proyecto, 3-26  
   hardware para la comunicación en redes, 3-4  
   instalación, 3-2  
   instalar hardware de comunicación, 3-4–3-6  
   número de referencia, G-3  
   número de referencia (actualización), G-3  
   número de referencia (licencia de copia), G-3  
   preferencias de programación, 3-25  
   tabla de estado/de forzado, 3-34  
   visualizar un programa, 3-31  
 STEP 7-Micro/DOS, convertir archivos, E-4  
 STEP 7-Micro/WIN  
   compilar un programa, 3-29  
   convertir archivos de STEP 7-Micro/DOS, E-4  
   crear un bloque de datos, 3-32  
   Editor de bloque de datos, 3-32  
 STOP, 6-13, 10-84  
 Subrutina, ejemplo, 6-9  
 Subrutinas, reglas, 6-8  
 Suma integral (bias), ajustar, regulación PID, 10-61  
 Sumar enteros de 16 bits, 10-50  
 Sumar enteros de 32 bits, 10-50  
 Sumar reales, 10-51  
 Supresión de ruidos, filtro de entrada, 8-5

## T

Tabla, operaciones de, 10-73–10-77  
 Tabla de elementos utilizados, imprimir, 5-23

Tabla de estado  
   para el programa de ejemplo, 4-14  
   programa de ejemplo, 4-14  
 Tabla de estado/de forzado  
   editar direcciones, 3-35  
   forzar variables, 3-35  
   leer y escribir variables, 3-34  
   observar/modificar valores, 4-17  
   STEP 7-Micro/WIN, 3-34  
 Tabla de estado/forzado  
   ciclo de la CPU, 6-17  
   modificar el programa, 6-16  
 Tabla de referencia hexadecimal PTO/PWM, 10-40  
 Tabla de referencias cruzadas, imprimir, 5-23  
 Tabla de salidas, configurar el estado de las salidas, 8-6  
 Tabla de símbolos  
   crear, 4-8  
   funciones de edición, 3-37  
   ordenar por nombres/ordenar por direcciones, 3-37  
   programa de ejemplo, 4-8  
   STEP 7-Micro/WIN, 3-36  
 Tabla del lazo, 10-62  
 Tabla del lazo PID, 10-62  
 Tamaño, módulos de ampliación, 2-4  
 Tamaño de los módulos  
   CPU 212, 2-3  
   CPU 214, 2-3  
   CPU 215, 2-4  
   CPU 216, 2-4  
   tornillos para el montaje, 2-3–2-5  
 Tamaño físico  
   CPU 212, 2-3  
   CPU 214, 2-3  
   CPU 215, 2-4  
   CPU 216, 2-4  
   módulos de ampliación, 2-4  
   tornillos para el montaje, 2-3–2-5  
 Tarjeta MPI, 3-8, 9-13  
   ajustar los parámetros de la tarjeta MPI (MPI), 3-16–3-17  
   ajustar los parámetros de la tarjeta MPI (PPI), 3-14  
   conexión, 3-8  
   configuración con un PC, 9-14  
   parámetros MPI, 3-16  
   parámetros PPI, 3-14  
 Tarjeta MPI (interface multipunto), número de referencia, G-2

TD 200, 5-2–5-9  
 Asistente de configuración, 5-3  
 bloque de parámetros, 5-2  
 configurar el bloque de parámetros, 5-3  
 crear mensajes, 5-8  
 frecuencia de actualización, 5-5  
 función de forzado, 5-4  
 idioma de los menús, 5-4  
 juego alternativo de caracteres, 5-4  
 mensajes, 5-6–5-10  
 protección con contraseña, 5-4  
 teclas de función, 5-5  
 Teclas de función (TD 200), habilitar, 5-5  
 Temporizador de retardo a la conexión, 10-13  
 Temporizador de retardo a la conexión  
 memorizado, 10-13  
 Temporizadores  
 actualizar, 10-14–10-18  
 CPU 212/214/215/216, 10-2  
 direccionamiento, 7-4  
 funcionamiento, 10-13  
 número, 10-13  
 resolución, 10-13  
 Temporizadores T32/T96, interrupciones, 10-119  
 TERM, 6-13  
 Término diferencial, algoritmo PID, 10-58  
 Término integral, algoritmo PID, 10-57  
 Término proporcional, algoritmo PID, 10-57  
 Test de aislamiento de alto potencial, A-5  
 Tiempo de ciclo, Tren de salida de impulsos (PTO),  
 10-42  
 Tiempo de ciclo, SMW22 a SMW26), D-5  
 Tiempo de rotación del testigo (token), 9-29–9-32  
 Tiempo de rotación del testigo, número de estacio-  
 nes, 9-30  
 Tiempos de ejecución, operaciones AWL, F-1–F-11  
 Tiempos de ejecución de las operaciones  
 impacto de la circulación de corriente, F-1  
 impacto de las E/S analógicas, F-1  
 impacto del direccionamiento indirecto, F-1  
 Tornillos (para el montaje), 2-3–2-5  
 Transferencia, operaciones de, 10-68–10-77  
 Transferir byte, 10-68  
 Transferir bytes en bloque, 10-69  
 Transferir palabra, 10-68  
 Transferir palabra doble, 10-68  
 Transferir palabras dobles en bloque, 10-69  
 Transferir palabras en bloque, 10-69  
 Transferir real, 10-68  
 Transistores en DC, proteger, 2-13  
 Transmitir mensaje, 10-124, 10-127  
 ejemplo, 10-130  
 Tren de impulsos (PTO), 8-7  
 ejemplo, 10-45  
 Tren de impulsos rápidos (PTO), 10-37  
 inicializar, 10-42

Tren de salida de impulsos (PTO)  
 cambiar el tiempo de ciclo, 10-42  
 cambiar el tiempo de ciclo y el valor de conteje  
 de impulsos, 10-43  
 cambiar el valor de conteje, 10-42  
 Truncar, 10-108

## U

Ubicación de los potenciómetros, EM235, A-70  
 Unidades periféricas, comunicación con, 9-15  
 Utilizar punteros, 7-9  
 modificar un puntero, 7-10  
 Utilizar subrutinas, 10-88

## V

Valores  
 bloque de datos, 3-33  
 intercalar en mensajes de texto, 5-8  
 Valores actuales de los temporizadores, actualizar,  
 10-16  
 Valores de datos intercalados (en mensajes de  
 texto), formatear, 5-10  
 Valores de datos intercalados (mensajes de texto),  
 5-8  
 Valores en coma flotante  
 regulación PID, 10-59  
 representación de, 7-2  
 Variable del proceso, convertir, 10-59  
 Variables, forzar, 3-35, 6-17  
 Velocidad de transferencia, cable PC/PPI, A-82  
 Velocidades de transferencia  
 ajustes de los interruptores DIP del cable PC/  
 PPI, 3-7  
 CPUs, 9-2  
 interfaces de comunicación, 9-2  
 posición de los interruptores DIP en el cable  
 PC/PPI, 9-10  
 Vibraciones fuertes, utilizar frenos, 2-6  
 Visualizador de textos, TD 200, 5-2  
 Visualizar, programas, 3-31

## W

Windows 3.1  
 eliminar errores de comunicación MPI, 3-17  
 instalación de STEP 7-Micro/WIN, 3-2  
 Windows 95, instalación de STEP 7-Micro/WIN,  
 3-2  
 Windows NT  
 eliminar errores de comunicación MPI, 3-18  
 instalación de STEP 7-Micro/WIN, 3-2  
 instalar hardware, 3-6

Siemens AG  
A&D AS E46

Östliche Rheinbrückenstr. 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:

- |  |   |
|--|---|
| <input type="checkbox"/> Industria del automóvil   | <input type="checkbox"/> Industria farmacéutica |
| <input type="checkbox"/> Industria química         | <input type="checkbox"/> Industria del plástico |
| <input type="checkbox"/> Industria eléctrica       | <input type="checkbox"/> Industria papelera     |
| <input type="checkbox"/> Industria alimentaria     | <input type="checkbox"/> Industria textil       |
| <input type="checkbox"/> Control e instrumentación | <input type="checkbox"/> Transportes            |
| <input type="checkbox"/> Industria mecánica        | <input type="checkbox"/> Otros _ _ _ _ _        |
| <input type="checkbox"/> 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 ?           | <input type="checkbox"/> |
| 2. | ¿ Resulta fácil localizar las informaciones requeridas ?           | <input type="checkbox"/> |
| 3. | ¿ Es comprensible el texto ?                                       | <input type="checkbox"/> |
| 4. | ¿ Corresponde el nivel de los detalles técnicos a sus exigencias ? | <input type="checkbox"/> |
| 5. | ¿ Qué opina de la calidad de las ilustraciones y tablas ?          | <input type="checkbox"/> |

En las líneas siguientes puede exponer los problemas concretos que se le hayan planteado al manejar el manual:

-----

-----

-----

-----

-----

-----

-----

-----

-----

-----