

SIEMENS

SIMATIC

S7

Controlador programable S7-1200




Manual de sistema

Prólogo	
Sinopsis del producto	1
Software de programación STEP 7	2
Montaje	3
Principios básicos del PLC	4
Configuración de dispositivos	5
Principios básicos de programación	6
Instrucciones básicas	7
Instrucciones avanzadas	8
Instrucciones tecnológicas	9
Comunicación	10
Servidor web	11
Procesador de comunicaciones	12
Comunicación TeleService (correo electrónico SMTP)	13
Herramientas online y diagnóstico	14
Datos técnicos	A
Calcular la corriente necesaria	B
Referencias	C

Notas jurídicas

Filosofía en la señalización de advertencias y peligros

Este manual contiene las informaciones necesarias para la seguridad personal así como para la prevención de daños materiales. Las informaciones para su seguridad personal están resaltadas con un triángulo de advertencia; las informaciones para evitar únicamente daños materiales no llevan dicho triángulo. De acuerdo al grado de peligro las consignas se representan, de mayor a menor peligro, como sigue.

 PELIGRO
Significa que, si no se adoptan las medidas preventivas adecuadas se producirá la muerte, o bien lesiones corporales graves.
 ADVERTENCIA
Significa que, si no se adoptan las medidas preventivas adecuadas puede producirse la muerte o bien lesiones corporales graves.
 PRECAUCIÓN
con triángulo de advertencia significa que si no se adoptan las medidas preventivas adecuadas, pueden producirse lesiones corporales.
PRECAUCIÓN
sin triángulo de advertencia significa que si no se adoptan las medidas preventivas adecuadas, pueden producirse daños materiales.
ATENCIÓN
significa que puede producirse un resultado o estado no deseado si no se respeta la consigna de seguridad correspondiente.


Si se dan varios niveles de peligro se usa siempre la consigna de seguridad más estricta en cada caso. Si en una consigna de seguridad con triángulo de advertencia se alarma de posibles daños personales, la misma consigna puede contener también una advertencia sobre posibles daños materiales.

Personal cualificado

El producto/sistema tratado en esta documentación sólo deberá ser manejado o manipulado por **personal cualificado** para la tarea encomendada y observando lo indicado en la documentación correspondiente a la misma, particularmente las consignas de seguridad y advertencias en ella incluidas. Debido a su formación y experiencia, el personal cualificado está en condiciones de reconocer riesgos resultantes del manejo o manipulación de dichos productos/sistemas y de evitar posibles peligros.

Uso previsto o de los productos de Siemens

Considere lo siguiente:

 ADVERTENCIA
Los productos de Siemens sólo deberán usarse para los casos de aplicación previstos en el catálogo y la documentación técnica asociada. De usarse productos y componentes de terceros, éstos deberán haber sido recomendados u homologados por Siemens. El funcionamiento correcto y seguro de los productos exige que su transporte, almacenamiento, instalación, montaje, manejo y mantenimiento hayan sido realizados de forma correcta. Es preciso respetar las condiciones ambientales permitidas. También deberán seguirse las indicaciones y advertencias que figuran en la documentación asociada.

Marcas registradas

Todos los nombres marcados con ® son marcas registradas de 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 sus titulares.

Exención de responsabilidad

Hemos comprobado la concordancia del contenido de esta publicación con el hardware y el software descritos. Sin embargo, como es imposible excluir desviaciones, no podemos hacernos responsable de la plena concordancia. El contenido de esta publicación se revisa periódicamente; si es necesario, las posibles las correcciones se incluyen en la siguiente edición.

Prólogo

Objeto del manual

La gama S7-1200 abarca distintos controladores lógicos programables (PLC) que pueden utilizarse para numerosas tareas. Gracias a su diseño compacto, bajo costo y amplio juego de instrucciones, los PLCs S7-1200 son idóneos para controlar una gran variedad de aplicaciones. Los modelos S7-1200 y el software de programación basado en Windows ofrecen la flexibilidad necesaria para solucionar las tareas de automatización.

Este manual contiene información sobre cómo montar y programar los PLCs S7-1200 y está dirigido a ingenieros, programadores, técnicos de instalación y electricistas que dispongan de conocimientos básicos sobre los controladores lógicos programables.

Nociones básicas

Para comprender este manual se requieren conocimientos básicos en el campo de la automatización y de los controladores lógicos programables.

Alcance del manual

Este manual describe los productos siguientes:

- STEP 7 V11 Basic y Professional
- Versión de firmware V3.0 de la CPU S7-1200

En los datos técnicos (Página 737) encontrará una lista completa de los productos S7-1200 descritos en el manual.

Homologaciones, marcado CE, C-Tick y otras normas

Para más información, consulte los datos técnicos (Página 737).

Service & Support

Además de la documentación, ofrecemos nuestros conocimientos técnicos en Internet, en la página web de atención al cliente (<http://www.siemens.com/automation/>).

Contacte con el representante de Siemens más próximo si tiene consultas de carácter técnico, así como para obtener información sobre los cursillos de formación o para pedir productos S7. Puesto que los representantes de Siemens han sido debidamente aleccionados y tienen conocimientos detallados sobre las operaciones, los procesos y la industria, así como sobre los distintos productos de Siemens empleados, pueden solucionar cualquier problema de forma rápida y eficiente.

Documentación e información

S7-1200 y STEP 7 ofrecen una documentación variada y otros recursos para encontrar la información técnica requerida.

- El manual de sistema S7-1200 ofrece información específica sobre el funcionamiento, la programación y las especificaciones de toda la familia de productos S7-1200. Además del manual de sistema, S7-1200 Easy Book ofrece una visión más amplia de las prestaciones de la familia S7-1200.

Tanto el manual de sistema como el Easy Book están disponibles en formato electrónico (PDF) e impreso. Los manuales electrónicos se pueden descargar de la página web de atención al cliente y también se incluyen en el disco que se suministra con cada CPU S7-1200.

- El sistema de información online de STEP 7 ofrece un acceso inmediato a la información conceptual y a instrucciones específicas que describen el funcionamiento y las funciones del paquete de programación, así como el funcionamiento básico de las CPUs SIMATIC.
- My Documentation Manager accede a las versiones en formato electrónico (PDF) del conjunto de documentos de SIMATIC, incluidos el manual de sistema, el Easy Book y el sistema de información de STEP 7. Con My Documentation Manager, puede arrastrar temas de varios documentos para crear su propio manual personalizado.

El portal de acceso al servicio de atención al cliente

(<http://support.automation.siemens.com>) ofrece un enlace a My Documentation Manager en mySupport.

- La página web de atención al cliente también ofrece podcasts, FAQs y otros documentos útiles para S7-1200 y STEP 7. Los podcasts emplean presentaciones breves de vídeos de formación que se centran en situaciones o funciones específicas para mostrar las interacciones, la comodidad y la eficacia que ofrece STEP 7. Visite las páginas web siguientes para acceder a la recopilación de podcasts:
 - Página web de STEP 7 Basic (<http://www.automation.siemens.com/mcms/simatic-controller-software/en/step7/step7-basic/Pages/Default.aspx>)
 - Página web de STEP 7 Professional (<http://www.automation.siemens.com/mcms/simatic-controller-software/en/step7/step7-professional/Pages/Default.aspx>)
- También puede seguir discusiones sobre productos y participar en ellas a través del foro técnico de Service & Support (<https://www.automation.siemens.com/WW/forum/guests/Conferences.aspx?Language=en&siteid=csius&treeLang=en&groupid=4000002&extranet=standard&viewreg=WW&nodeid0=34612486>). Los foros permiten interactuar con diferentes expertos sobre productos.
 - Foro de S7-1200 (<https://www.automation.siemens.com/WW/forum/guests/Conference.aspx?SortField=LastPostDate&SortOrder=Descending&ForumID=258&Language=en&onlyInternet=False>)
 - Foro de STEP 7 Basic (<https://www.automation.siemens.com/WW/forum/guests/Conference.aspx?SortField=LastPostDate&SortOrder=Descending&ForumID=265&Language=en&onlyInternet=False>)

Índice

	Prólogo	3
1	Sinopsis del producto	19
1.1	Introducción al PLC S7-1200	19
1.2	Capacidad de expansión de la CPU	22
1.3	Módulos S7-1200	25
1.4	Nuevas funciones	26
1.5	Paneles HMI Basic	27
2	Software de programación STEP 7	29
2.1	Requisitos del sistema	30
2.2	Diferentes vistas que facilitan el trabajo	31
2.3	Herramientas fáciles de utilizar	32
2.3.1	Introducir instrucciones en el programa de usuario	32
2.3.2	Acceder a instrucciones desde la barra de herramientas "Favoritos"	32
2.3.3	Crear una ecuación compleja con una instrucción sencilla	33
2.3.4	Agregar entradas o salidas a una instrucción KOP o FUP	35
2.3.5	Instrucciones ampliables	35
2.3.6	Seleccionar la versión de una instrucción	36
2.3.7	Modificar la apariencia y configuración de STEP 7	36
2.3.8	Arrastrar y soltar elementos entre los distintos editores	37
2.3.9	Cambiar el estado operativo de la CPU	37
2.3.10	Modificar el tipo de llamada de un DB	38
2.3.11	Desconectar temporalmente dispositivos de una red	39
2.3.12	Desconexión virtual de dispositivos desde la configuración	40
3	Montaje	41
3.1	Directrices para montar dispositivos S7-1200	41
3.2	Corriente necesaria	42
3.3	Procedimientos de montaje y desmontaje	44
3.3.1	Dimensiones de montaje de los dispositivos S7-1200	44
3.3.2	Montaje y desmontaje de la CPU	47
3.3.3	Montaje y desmontaje de SB, CB o BB	49
3.3.4	Instalación y desmontaje de un SM	51
3.3.5	Montaje y desmontaje de un CM o CP	53
3.3.6	Extraer e insertar el conector del bloque de terminales del S7-1200	54
3.3.7	Instalación y desmontaje de un cable de ampliación	55
3.3.8	TS (TeleService) Adapter	57
3.3.8.1	Conectar el adaptador de TeleService	57
3.3.8.2	Instalar la tarjeta SIM	58
3.3.8.3	Montaje del adaptador TS	60
3.3.8.4	Montaje del adaptador TS en una pared	60

3.4	Directrices de cableado.....	61
4	Principios básicos del PLC	67
4.1	Ejecución del programa de usuario	67
4.1.1	Estados operativos de la CPU	70
4.1.2	Procesamiento del ciclo en estado operativo RUN	73
4.1.3	Bloques de organización (OBs)	74
4.1.4	Prioridades y colas de espera para la ejecución de eventos.....	76
4.1.5	Vigilancia del tiempo de ciclo.....	81
4.1.6	Memoria de la CPU.....	83
4.1.6.1	Marcas de sistema y de ciclo.....	85
4.1.7	Búfer de diagnóstico	87
4.1.8	Reloj en tiempo real	88
4.1.9	Configurar las salidas en una transición de RUN a STOP	88
4.2	Almacenamiento de datos, áreas de memoria, E/S y direccionamiento	89
4.2.1	Acceder a los datos del S7-1200	89
4.3	Procesamiento de valores analógicos	94
4.4	Tipos de datos.....	95
4.4.1	Tipos de datos Bool, Byte, Word y DWord	96
4.4.2	Tipos de datos de entero	97
4.4.3	Tipos de datos de real en coma flotante.....	98
4.4.4	Tipos de datos de fecha y hora.....	98
4.4.5	Tipos de datos Carácter y Cadena	100
4.4.6	Tipo de datos ARRAY	101
4.4.7	Tipo de datos Estructura de datos	103
4.4.8	Tipo de datos PLC	103
4.4.9	Tipos de datos de puntero	103
4.4.9.1	Tipo de datos de puntero "Pointer"	104
4.4.9.2	Tipo de datos de puntero "Any"	105
4.4.9.3	Tipo de datos de puntero "Variant"	106
4.4.10	Acceder a un "segmento" de un tipo de datos de variable	107
4.4.11	Acceso a una variable con una superposición de AT.....	108
4.5	Utilizar una Memory Card	111
4.5.1	Insertar una Memory Card en la CPU.....	112
4.5.2	Configurar los parámetros de arranque de la CPU antes de copiar el proyecto en la Memory Card	113
4.5.3	Utilizar la Memory Card como tarjeta de "Transferencia"	114
4.5.4	Utilizar la Memory Card como tarjeta de "Programa"	116
4.5.5	Actualización de firmware	119
4.6	Recuperación si se olvida la contraseña	122
5	Configuración de dispositivos	123
5.1	Insertar una CPU	124
5.2	Detectar la configuración de una CPU sin especificar.....	125
5.3	Agregar módulos a la configuración	126
5.4	Configurar el funcionamiento de la CPU.....	127
5.5	Configurar los parámetros de los módulos	130

5.6	Configurar la CPU para la comunicación.....	131
5.6.1	Crear una conexión de red	131
5.6.2	Configurar la vía de conexión local/de interlocutor.....	132
5.6.3	Parámetros de la conexión PROFINET	134
5.6.4	Asignar direcciones IP (Internet Protocol)	137
5.6.4.1	Asignar direcciones IP a los dispositivos de programación y red.....	137
5.6.4.2	Comprobar la dirección IP de la programadora.....	139
5.6.4.3	Asignar una dirección IP a una CPU online.....	139
5.6.4.4	Configurar una dirección IP para una CPU en el proyecto.....	141
5.6.5	Comprobar la red PROFINET.....	144
5.6.6	Localizar la dirección Ethernet (MAC) en la CPU.....	145
5.6.7	Configurar la sincronización del Network Time Protocol (NTP)	147
5.6.8	Tiempo de arranque de dispositivo, asignación de nombre y de dirección en PROFINET	148
6	Principios básicos de programación	149
6.1	Directrices para diseñar un sistema PLC	149
6.2	Estructurar el programa de usuario	150
6.3	Utilizar bloques para estructurar el programa.....	152
6.3.1	Bloque de organización (OB).....	153
6.3.2	Función (FC).....	154
6.3.3	Bloque de función (FB)	155
6.3.4	Bloque de datos (DB).....	156
6.4	Principios básicos de la coherencia de datos.....	157
6.5	Lenguaje de programación	158
6.5.1	Esquema de contactos (KOP)	159
6.5.2	Diagrama de funciones (FUP)	160
6.5.3	SCL.....	160
6.5.4	EN y ENO para KOP, FUP y SCL.....	167
6.6	Protección	168
6.6.1	Protección de acceso a la CPU	168
6.6.2	Protección de know-how.....	170
6.6.3	Protección anticopia.....	171
6.7	Cargar los elementos del programa en la CPU.....	172
6.8	Cargar desde la CPU.....	172
6.8.1	Copia de elementos del proyecto	172
6.8.2	Utilización de la función de comparación	174
6.9	Depurar y comprobar el programa.....	174
6.9.1	Vigilar y modificar datos de la CPU	174
6.9.2	Tablas de observación y tablas de forzado	175
6.9.3	Referencia cruzada para mostrar la utilización.....	175
6.9.4	Estructura de llamadas para ver la jerarquía de llamadas	176
7	Instrucciones básicas	179
7.1	Instrucciones lógicas con bits	179
7.1.1	Contactos y bobinas en instrucciones lógicas con bits	179
7.1.2	Instrucciones "Activar salida" y "Desactivar salida".....	182
7.1.3	Instrucciones "Consultar flanco de señal ascendente de un operando" y "Consultar flanco de señal descendente de un operando"	185

7.2	Temporizadores	187
7.3	Contadores.....	196
7.4	Comparación.....	202
7.4.1	Comparación.....	202
7.4.2	Instrucciones "Valor dentro del rango" y "Valor fuera del rango"	203
7.4.3	Instrucciones "Comprobar validez" y "Comprobar invalidez"	204
7.5	Funciones matemáticas	205
7.5.1	Instrucción Calcular.....	205
7.5.2	Instrucciones "Sumar", "Restar", "Multiplicar" y "Dividir"	206
7.5.3	Instrucción Modulo	207
7.5.4	Instrucción de negación	208
7.5.5	Instrucciones "Incrementar" y "Decrementar".....	209
7.5.6	Instrucción "Calcular valor absoluto"	209
7.5.7	Instrucciones de mínimo y máximo.....	210
7.5.8	Instrucción Limit	211
7.5.9	Instrucciones matemáticas en coma flotante.....	212
7.6	Desplazamiento	214
7.6.1	Instrucciones "Copiar valor" y "Copiar área".....	214
7.6.2	Instrucciones FieldRead y FieldWrite	216
7.6.3	Instrucciones de relleno	218
7.6.4	Instrucción "Cambiar disposición".....	220
7.7	Convertir.....	221
7.7.1	Instrucción CONV	221
7.7.2	Instrucciones de conversión de SCL	222
7.7.3	Instrucciones "Redondear número" y "Truncar a entero doble"	225
7.7.4	Instrucciones "Crear el siguiente número entero superior a partir del número en coma flotante" y "Crear el siguiente número entero inferior a partir del número en coma flotante"	226
7.7.5	Instrucciones "Escalar" y "Normalizar".....	227
7.8	Control del programa	230
7.8.1	Vista general de las instrucciones de control del programa de SCL	230
7.8.2	Instrucción IF-THEN.....	231
7.8.3	Instrucción CASE	232
7.8.4	Instrucción FOR	233
7.8.5	Instrucción WHILE-DO.....	234
7.8.6	Instrucción REPEAT-UNTIL.....	235
7.8.7	Instrucción CONTINUE	236
7.8.8	Instrucción EXIT	236
7.8.9	Instrucción GOTO	237
7.8.10	Instrucción RETURN.....	238
7.8.11	Instrucciones de salto y etiquetas.....	238
7.8.12	Instrucción JMP_LIST	239
7.8.13	Instrucción SWITCH.....	240
7.8.14	Instrucción de control de ejecución "Retroceder"	242
7.8.15	Instrucción "Reiniciar tiempo de vigilancia del ciclo"	243
7.8.16	Instrucción "Parar ciclo del PLC"	244
7.8.17	Instrucciones GET_ERROR.....	244
7.9	Operaciones lógicas con palabras.....	248
7.9.1	Instrucciones Y, O y O-exclusiva	248

7.9.2	Instrucción "Complemento a uno".....	249
7.9.3	Instrucciones "Codificar" y "Decodificar".....	249
7.9.4	Instrucciones Seleccionar, Multiplexar y Demultiplexar	251
7.10	Instrucciones de desplazamiento y rotación.....	253
7.10.1	Instrucciones de desplazamiento.....	253
7.10.2	Instrucciones de rotación	254
8	Instrucciones avanzadas	257
8.1	Fecha y hora	257
8.1.1	Instrucciones de fecha y hora	257
8.1.2	Ajustar y leer el reloj del sistema	259
8.1.3	Instrucción Contador de horas de funcionamiento	261
8.1.4	Instrucción SET_TIMEZONE	263
8.2	Cadena y carácter.....	264
8.2.1	Sinopsis del tipo de datos String.....	264
8.2.2	Instrucción S_MOVE.....	265
8.2.3	Instrucciones de conversión de cadenas.....	265
8.2.3.1	Conversión de cadenas en valores y de valores en cadenas	265
8.2.3.2	Conversiones de cadena a carácter y de carácter a cadena	274
8.2.3.3	Conversiones de ASCII a Hex y de Hex a ASCII	276
8.2.4	Instrucciones con cadenas	278
8.2.4.1	LEN	278
8.2.4.2	CONCAT	279
8.2.4.3	LEFT, RIGHT y MID.....	280
8.2.4.4	DELETE	281
8.2.4.5	INSERT	282
8.2.4.6	REPLACE	283
8.2.4.7	FIND.....	284
8.3	E/S distribuidas (PROFINET, PROFIBUS o AS-i).....	285
8.3.1	Instrucciones de E/S descentralizadas.....	285
8.3.2	RDREC y WRREC	286
8.3.3	RALRM.....	289
8.3.4	Parámetro STATUS para RDREC, WRREC y RALRM.....	292
8.3.5	DPRD_DAT y DPWR_DAT	296
8.3.6	DPNRM_DG.....	298
8.4	Alarmas	301
8.4.1	Instrucciones ATTACH y DETACH.....	301
8.4.2	Alarmas cíclicas	304
8.4.2.1	SET_CINT (Ajustar parámetros de alarma cíclica).....	304
8.4.2.2	QRY_CINT (Consultar parámetros de alarma cíclica).....	306
8.4.3	Alarmas de retardo	307
8.4.4	Alarmas de evento asíncrono	309
8.5	Diagnóstico (PROFINET o PROFIBUS).....	311
8.5.1	Instrucciones de diagnóstico.....	311
8.5.2	Eventos de diagnóstico de E/S descentralizadas.....	311
8.5.3	Instrucción LED.....	312
8.5.4	Instrucción DeviceStates	313
8.5.5	Instrucción ModuleStates.....	314
8.5.6	Instrucción GET_DIAG.....	316

8.6	Impulso.....	323
8.6.1	Instrucción CTRL_PWM.....	323
8.6.2	Manejo de las salidas de impulso	325
8.6.3	Configurar un canal de impulsos para PWM	327
8.7	Registrar datos.....	328
8.7.1	Estructura del registro.....	328
8.7.2	Instrucciones de programa que controlan registros.....	330
8.7.2.1	DataLogCreate.....	330
8.7.2.2	DataLogOpen.....	333
8.7.2.3	DataLogClose	335
8.7.2.4	DataLogWrite	336
8.7.2.5	DataLogNewFile.....	337
8.7.3	Trabajar con registros de datos	339
8.7.4	Límites del tamaño de los archivos de registro.....	341
8.7.5	Programa de ejemplo de registros de datos.....	344
8.8	Control de bloques de datos	348
8.8.1	READ_DBL, WRIT_DBL (Leer de o escribir en un bloque de datos de la memoria de carga).....	348
8.9	Códigos de error comunes para las instrucciones "Avanzadas"	351
9	Instrucciones tecnológicas.....	353
9.1	Contador rápido	353
9.1.1	Funcionamiento del contador rápido.....	355
9.1.2	Configuración del HSC.....	361
9.2	Control PID.....	362
9.2.1	Insertar la instrucción PID y un objeto tecnológico.....	364
9.2.2	Instrucción PID_Compact.....	367
9.2.3	Parámetros de la instrucción ErrorBit de PID_Compact.....	372
9.2.4	Instrucción PID_3STEP.....	373
9.2.5	Parámetros de la instrucción ErrorBit de PID_3STEP	379
9.2.6	Configurar el regulador PID	381
9.2.7	Puesta en servicio del regulador PID.....	383
9.3	Motion control.....	384
9.3.1	Configurar el eje.....	387
9.3.2	Configurar TO_CommandTable_PTO.....	390
9.3.3	Instrucciones de Motion Control	393
9.3.3.1	Instrucción MC_Power	393
9.3.3.2	Instrucción MC_Reset.....	396
9.3.3.3	Instrucción MC_Home.....	397
9.3.3.4	Instrucción MC_Halt	399
9.3.3.5	Instrucción MC_MoveAbsolute	401
9.3.3.6	Instrucción MC_MoveRelative.....	403
9.3.3.7	Instrucción MC_MoveVelocity	405
9.3.3.8	Instrucción MC_MoveJog.....	407
9.3.3.9	Instrucción MC_CommandTable.....	410
9.3.3.10	MC_ChangeDynamic	412
9.3.4	Operación de Motion Control para S7-1200	414
9.3.4.1	Salidas de la CPU utilizadas para control de movimiento	414
9.3.4.2	Finales de carrera por hardware y software para Motion Control.....	416
9.3.4.3	Referenciación	419

9.3.4.4	Limitación de tirones	424
9.3.5	Puesta en servicio.....	425
9.3.6	Vigilar comandos activos	428
9.3.6.1	Vigilar instrucciones MC con un parámetro de salida "Done"	428
9.3.6.2	Vigilar la instrucción MC_Velocity	432
9.3.6.3	Vigilar la instrucción MC_MoveJog	436
10	Comunicación.....	441
10.1	Número de conexiones soportadas de comunicación asíncrona	442
10.2	PROFINET	443
10.2.1	Conexión local/interlocutor.....	443
10.2.2	Open User Communication.....	445
10.2.2.1	ID de conexión para las instrucciones PROFINET	445
10.2.2.2	Protocolos	448
10.2.2.3	Modo ad hoc	449
10.2.2.4	TCP y ISO on TCP.....	450
10.2.2.5	UDP.....	465
10.2.2.6	T_CONFIG	470
10.2.2.7	Parámetros comunes para instrucciones	476
10.2.3	Comunicación con una programadora.....	478
10.2.3.1	Establecer la conexión de hardware.....	479
10.2.3.2	Configurar los dispositivos	479
10.2.3.3	Asignar direcciones IP (Internet Protocol)	480
10.2.3.4	Comprobar la red PROFINET	480
10.2.4	Comunicación entre dispositivos HMI y el PLC	480
10.2.4.1	Configurar las conexiones de red lógicas entre dos dispositivos	482
10.2.5	Comunicación entre PLCs	482
10.2.5.1	Configurar las conexiones de red lógicas entre dos dispositivos	483
10.2.5.2	Configurar la vía de conexión local/interlocutor entre dos dispositivos	483
10.2.5.3	Configurar los parámetros de transmisión y recepción	484
10.2.6	Configurar una CPU y un dispositivo PROFINET IO	486
10.2.7	Diagnóstico	490
10.2.8	Instrucciones de E/S descentralizadas.....	490
10.2.9	Instrucciones de diagnóstico.....	491
10.2.10	Eventos de diagnóstico de E/S descentralizadas.....	491
10.3	PROFIBUS.....	491
10.3.1	Módulos de comunicación para PROFIBUS.....	492
10.3.1.1	Conexión a PROFIBUS	492
10.3.1.2	Servicios de comunicación de los CMs PROFIBUS.....	493
10.3.1.3	Otras propiedades de los CMs PROFIBUS	494
10.3.1.4	Ejemplos de configuración para PROFIBUS	495
10.3.2	Configurar un maestro DP y un dispositivo esclavo	496
10.3.2.1	Agregar el módulo CM 1243-5 (maestro DP) y un esclavo DP	496
10.3.2.2	Configurar las conexiones de red lógicas entre dos dispositivos PROFIBUS	497
10.3.2.3	Asignar direcciones PROFIBUS al módulo CM 1243-5 y al esclavo DP	497
10.3.3	Instrucciones de E/S descentralizadas.....	499
10.3.4	Instrucciones de diagnóstico.....	499
10.3.5	Eventos de diagnóstico de E/S descentralizadas.....	499
10.4	AS-i	499
10.4.1	Configuración de un maestro AS-i y un dispositivo esclavo.....	500
10.4.1.1	Agregar el módulo maestro AS-i CM 1243-2 y esclavo AS-i.....	500

10.4.1.2	Configuración de las conexiones de red lógicas entre dos dispositivos AS-i.....	501
10.4.1.3	Configurar las propiedades del maestro AS-i CM1243-2	501
10.4.1.4	Asignar una dirección AS-i a un esclavo AS-i.....	502
10.4.2	Intercambio de datos entre el programa de usuario y los esclavos AS-i.....	503
10.4.2.1	Configuración básica de STEP 7	503
10.4.2.2	Configurar esclavos con STEP 7	505
10.4.3	Instrucciones de E/S descentralizadas	507
10.4.4	Trabajar con herramientas online AS-i	507
10.5	Comunicación S7	509
10.5.1	Instrucciones GET y PUT.....	509
10.5.2	Crear una conexión S7	513
10.5.3	Configurar la vía de conexión local/interlocutor entre dos dispositivos	513
10.5.4	Asignación de parámetros de conexión GET/PUT	514
10.5.4.1	Parámetros de la conexión	514
10.5.4.2	Configurar una conexión S7 de CPU a CPU	517
11	Servidor web.....	523
11.1	Habilitar el servidor web.....	524
11.2	Páginas web estándar.....	525
11.2.1	Acceder a las páginas web estándar desde el PC	525
11.2.2	Representación de las páginas web estándar	527
11.2.3	Introducción.....	530
11.2.4	Arranque.....	531
11.2.5	Identificación	532
11.2.6	Búfer de diagnóstico	532
11.2.7	Información del módulo.....	533
11.2.8	Comunicación	535
11.2.9	Estado de las variables	537
11.2.10	Registros	539
11.2.11	Update Firmware.....	541
11.3	Páginas web definidas por el usuario	543
11.3.1	Creación de páginas HTML	544
11.3.2	Comandos AWP soportados por el servidor web del S7-1200.....	545
11.3.2.1	Leer variables.....	546
11.3.2.2	Escribir variables.....	547
11.3.2.3	Leer variables especiales.....	549
11.3.2.4	Escribir variables especiales.....	551
11.3.2.5	Utilizar un alias para una referencia de variable.....	552
11.3.2.6	Definir tipos de enumeración	553
11.3.2.7	Referenciar variables de CPU con un tipo de enumeración.....	554
11.3.2.8	Crear fragmentos	555
11.3.2.9	Importar fragmentos.....	556
11.3.2.10	Combinar definiciones	557
11.3.2.11	Procesamiento de nombres de variable que contienen caracteres especiales	557
11.3.3	Configurar el uso de las páginas web definidas por el usuario	560
11.3.4	Programar la instrucción WWW para páginas web definidas por el usuario.....	561
11.3.5	Cargar los bloques de programa en la CPU	563
11.3.6	Acceder a las páginas web definidas por el usuario desde el PC.....	563
11.3.7	Limitaciones específicas de las páginas web definidas por el usuario.....	564
11.3.8	Ejemplo de una página web definida por el usuario	565
11.3.8.1	Página web para vigilar y controlar una turbina de viento	565

11.3.8.2	Leer y visualizar datos del controlador	567
11.3.8.3	Utilizar un tipo de enumeración	568
11.3.8.4	Escribir una entrada del usuario en el controlador	569
11.3.8.5	Escribir una variable especial	570
11.3.8.6	Referencia: Lista HTML de la página web para la vigilancia remota de la turbina de viento....	570
11.3.8.7	Configuración en STEP 7 de la página web de ejemplo	574
11.3.9	Configurar páginas web definidas por el usuario en varios idiomas	576
11.3.9.1	Crear la estructura de carpetas	576
11.3.9.2	Programar el cambio de idioma	577
11.3.9.3	Configurar STEP 7 para utilizar una estructura de página multilingüe.....	579
11.3.10	Control avanzado de páginas web definidas por el usuario	580
11.4	Limitaciones	583
11.4.1	Funciones restringidas al deshabilitar JavaScript.....	584
11.4.2	Funciones restringidas cuando se no se permiten cookies	585
11.4.3	Importar el certificado de seguridad Siemens	586
11.4.4	Importar registros de datos en formato CSV a versiones de Microsoft Excel que no son estadounidenses ni británicas.....	587
12	Procesador de comunicaciones.....	589
12.1	Utilizar las interfaces de comunicación serie.....	589
12.2	Polarizar y terminar un conector de red RS485.....	590
12.3	Comunicación punto a punto (PtP)	591
12.3.1	Instrucciones de comunicación punto a punto.....	592
12.3.1.1	Parámetros comunes de las instrucciones de comunicación punto a punto.....	592
12.3.1.2	Instrucción PORT_CFG	594
12.3.1.3	Instrucción SEND_CFG	596
12.3.1.4	Instrucción RCV_CFG.....	598
12.3.1.5	Instrucción SEND_PTP.....	602
12.3.1.6	Instrucción RCV_PTP	606
12.3.1.7	Instrucción RCV_RST	607
12.3.1.8	Instrucción SGN_GET.....	608
12.3.1.9	Instrucción SGN_SET	609
12.3.2	Configurar los puertos de comunicación	611
12.3.2.1	Gestionar el control de flujo	613
12.3.3	Configurar los parámetros de transmisión y recepción	614
12.3.3.1	Configurar los parámetros de transmisión.....	615
12.3.3.2	Configurar los parámetros de recepción.....	616
12.3.4	Programar la comunicación PtP	624
12.3.4.1	Arquitectura de sondeo.....	625
12.3.5	Ejemplo: Comunicación punto a punto.....	626
12.3.5.1	Configurar los módulos de comunicación.....	627
12.3.5.2	Configurar RS422 y RS485	629
12.3.5.3	Programar el programa de STEP 7	632
12.3.5.4	Configurar el emulador de terminal.....	633
12.3.5.5	Ejecutar el programa de ejemplo	634
12.4	Comunicación de interfaz serie universal (USS)	634
12.4.1	Requisitos para utilizar el protocolo USS	636
12.4.2	Instrucción USS_DRV.....	639
12.4.3	Instrucción USS_PORT	642
12.4.4	Instrucción USS_RPM	643

12.4.5	Instrucción USS_WPM.....	644
12.4.6	Códigos de estado USS.....	646
12.4.7	Información general sobre la configuración del accionamiento.....	648
12.5	Comunicación Modbus.....	651
12.5.1	Vista general de las comunicaciones Modbus RTU y TCP	651
12.5.2	Modbus TCP	654
12.5.2.1	MB_CLIENT (Modbus TCP).....	654
12.5.2.2	MB_SERVER (Modbus TCP).....	661
12.5.2.3	Ejemplo de MB_SERVER: Conexiones TCP múltiples	666
12.5.2.4	Ejemplo 1 de MB_CLIENT: Peticiones múltiples con conexión TCP común.....	667
12.5.2.5	Ejemplo 2 de MB_CLIENT: Peticiones múltiples con varias conexiones TCP	668
12.5.2.6	Ejemplo 3 de MB_CLIENT: Petición de escritura de imagen de salida	669
12.5.2.7	Ejemplo 4 de MB_CLIENT: Coordinación de varias peticiones.....	670
12.5.3	Modbus RTU	671
12.5.3.1	MB_COMM_LOAD	672
12.5.3.2	MB_MASTER	675
12.5.3.3	MB_SLAVE	681
12.5.3.4	Ejemplo de programa de maestro Modbus RTU	687
12.5.3.5	Ejemplo de programa de esclavo Modbus RTU	689
12.6	Telecontrol y TeleService con el CP 1242-7.....	690
12.6.1	Conexión a una red GSM.....	690
12.6.2	Aplicaciones del CP 1242-7	691
12.6.3	Otras propiedades del CP.....	693
12.6.4	Accesorios.....	694
12.6.5	Ejemplos de configuración para Telecontrol.....	696
13	Comunicación TeleService (correo electrónico SMTP).....	701
13.1	Instrucción de transferencia de correos electrónicos TM_Mail.....	701
14	Herramientas online y diagnóstico.....	709
14.1	LEDs de estado.....	709
14.2	Establecer una conexión online con una CPU.....	711
14.3	Asignar un nombre a un dispositivo PROFINET IO online	712
14.4	Ajustar la dirección IP y la hora	714
14.5	Restablecimiento de los ajustes de fábrica.....	714
14.6	Panel de control de la CPU online	715
14.7	Vigilar el tiempo de ciclo y la carga de la memoria.....	716
14.8	Visualizar los eventos de diagnóstico de la CPU	717
14.9	Comparar CPUs online y offline.....	718
14.10	Vigilar y modificar valores en la CPU.....	719
14.10.1	Conexión online para observar los valores en la CPU	720
14.10.2	Visualización del estado en el editor de programas	720
14.10.3	Obtener los valores online de un DB para restablecer los valores iniciales.....	721
14.10.4	Utilizar una tabla de observación para observar y modificar valores en la CPU.....	722
14.10.4.1	Utilizar un disparador para observar o forzar variables PLC.....	724
14.10.4.2	Desbloquear las salidas en modo STOP	725
14.10.5	Forzar valores permanentemente en la CPU	725

14.10.5.1	Utilizar la tabla de forzado permanente	725
14.10.5.2	Funcionamiento de la función de forzado permanente	726
14.11	Carga en estado operativo RUN.....	728
14.11.1	Requisitos para la "Carga en estado operativo RUN"	729
14.11.2	Modificar el programa en el estado operativo RUN.....	730
14.11.3	Descargar bloques seleccionados	731
14.11.4	Descargar un único bloque seleccionado con un error de compilación en otro bloque	732
14.11.5	Reacción del sistema si falla la descarga	733
14.11.6	Descarga del programa en estado operativo RUN.....	734
A	Datos técnicos	737
A.1	Datos técnicos generales.....	737
A.2	CPU 1211C	744
A.2.1	Especificaciones generales y propiedades.....	744
A.2.2	Temporizadores, contadores y bloques lógicos soportados por la CPU 1211C	745
A.2.3	Entradas y salidas digitales	747
A.2.4	Entradas analógicas	749
A.2.4.1	Respuesta a un escalón de las entradas analógicas integradas en la CPU	750
A.2.4.2	Tiempo de muestreo para los puertos analógicos integrados en la CPU	750
A.2.5	Diagramas de cableado de la CPU 1211C	751
A.3	CPU 1212C	754
A.3.1	Especificaciones generales y propiedades.....	754
A.3.2	Temporizadores, contadores y bloques lógicos soportados por la CPU 1212C	755
A.3.3	Entradas y salidas digitales	758
A.3.4	Entradas analógicas	759
A.3.4.1	Respuesta a un escalón de las entradas analógicas integradas en la CPU	760
A.3.4.2	Tiempo de muestreo para los puertos analógicos integrados en la CPU	760
A.3.5	Diagramas de cableado de la CPU 1212C	761
A.4	CPU 1214C	764
A.4.1	Especificaciones generales y propiedades.....	764
A.4.2	Temporizadores, contadores y bloques lógicos soportados por la CPU 1214C	765
A.4.3	Entradas y salidas digitales	767
A.4.4	Entradas analógicas	769
A.4.4.1	Respuesta a un escalón de las entradas analógicas integradas en la CPU	770
A.4.4.2	Tiempo de muestreo para los puertos analógicos integrados en la CPU	770
A.4.5	Diagramas de cableado de la CPU 1214C	771
A.5	CPU 1215C	774
A.5.1	Especificaciones generales y propiedades.....	774
A.5.2	Temporizadores, contadores y bloques lógicos soportados por la CPU 1215C	776
A.5.3	Entradas y salidas digitales	778
A.5.4	Entradas y salidas analógicas	780
A.5.4.1	Datos técnicos de la entrada analógica.....	780
A.5.4.2	Respuesta a escalón de las entradas analógicas integradas en la CPU	780
A.5.4.3	Tiempo de muestreo para los puertos analógicos integrados en la CPU	781
A.5.4.4	Especificaciones de salidas analógicas.....	781
A.5.5	Diagramas de cableado de la CPU 1215C	782
A.6	Módulos de señales digitales (SMs)	786
A.6.1	Datos técnicos del módulo de entradas digitales SM 1221	786
A.6.2	Datos técnicos del módulo de salidas digitales SM 1222 de 8 salidas	788

A.6.3	Datos técnicos del módulo de salidas digitales SM 1222 de 16 salidas	790
A.6.4	Datos técnicos del módulo de entradas/salidas digitales V DC SM 1223	795
A.6.5	Datos técnicos del módulo de entradas/salidas digitales AC SM 1223.....	800
A.7	Módulos de señales analógicos (SMs)	803
A.7.1	Datos técnicos del módulo de entradas analógicas SM 1231	803
A.7.2	Datos técnicos del módulo de salidas analógicas SM 1232.....	807
A.7.3	Datos técnicos del módulo de entradas/salidas analógicas SM 1234.....	810
A.7.4	Respuesta a un escalón de las entradas analógicas	813
A.7.5	Tiempo de muestreo y tiempos de actualización para entradas analógicas	813
A.7.6	Rangos de medida de entradas analógicas de tensión.....	814
A.7.7	Rangos de medida de las entradas analógicas de intensidad	814
A.7.8	Rangos de medición de salidas (AQ) de tensión e intensidad (SB y SM).....	815
A.8	Módulos de señales RTD y de termopar (SMs).....	816
A.8.1	SM 1231 de termopar	816
A.8.1.1	Funcionamiento básico de un termopar.....	819
A.8.1.2	Tablas de selección de termopares para el SM 1231	820
A.8.2	SM 1231 RTD	822
A.8.2.1	Tablas de selección para el SM 1231 RTD	825
A.9	Signal Boards digitales (SBs).....	828
A.9.1	Datos técnicos de la SB 1221 de entradas digitales 200 kHz	828
A.9.2	Datos técnicos de la SB 1222 de salidas digitales 200 kHz.....	831
A.9.3	Datos técnicos de la SB 1223 de entradas/salidas digitales 200 kHz.....	834
A.9.4	Datos técnicos de la SB 1223 2 entradas x 24 V DC / 2 salidas x 24 V DC	837
A.10	Signal Boards analógicas (SBs)	839
A.10.1	Datos técnicos de la SB 1231 de 1 entrada analógica	839
A.10.2	Datos técnicos de la SB 1232 de 1 salida analógica.....	842
A.10.3	Rangos de medida para entradas y salidas analógicas	844
A.10.3.1	Respuesta a un escalón de las entradas analógicas	844
A.10.3.2	Tiempo de muestreo y tiempos de actualización para entradas analógicas	844
A.10.3.3	Rangos de medida de entradas analógicas de tensión.....	845
A.10.3.4	Rangos de medida de las entradas analógicas de intensidad	845
A.10.3.5	Rangos de medición de salidas (AQ) de tensión e intensidad (SB y SM).....	846
A.10.4	SBs de termopar	847
A.10.4.1	Datos técnicos de la SB 1231 de termopar de 1 entrada analógica	847
A.10.4.2	Funcionamiento básico de un termopar.....	848
A.10.5	SBs RTD	851
A.10.5.1	Datos técnicos de la SB 1231 de 1 entrada analógica RTD.....	851
A.10.5.2	Tablas de selección para la SB 1231 RTD.....	854
A.11	BB 1297 Battery Board	856
A.12	Interfaces de comunicación	858
A.12.1	PROFIBUS	858
A.12.1.1	CM 1242-5.....	858
A.12.1.2	CM 1243-5.....	859
A.12.2	GPRS	861
A.12.2.1	CP 1242-7	861
A.12.3	CM 1243-2 AS-i Master.....	864
A.12.3.1	Datos técnicos del maestro AS-i CM 1243-2.....	864
A.12.3.2	Conexiones eléctricas del maestro AS-i CM 1243-2	865
A.12.4	RS232, RS422 y RS485	866

A.12.4.1	Datos técnicos de CB 1241 RS485	866
A.12.4.2	Datos técnicos del CM 1241 RS232	869
A.12.4.3	Datos técnicos del CM 1241 RS422/485	870
A.13	TeleService (TS Adapter y TS Adapter modular)	871
A.14	SIMATIC Memory Cards	872
A.15	Simuladores de entradas	872
A.16	Cable para módulos de ampliación	873
A.17	Productos adicionales	874
A.17.1	Módulo de alimentación PM 1207	874
A.17.2	Módulo de conmutación compacto CSM 1277	874
B	Calcular la corriente necesaria	875
C	Referencias	879
C.1	Módulos CPU	879
C.2	Módulos de señales (SM), Signal Boards (SB) y Battery Boards (BB)	879
C.3	Comunicación	881
C.4	Otros módulos	882
C.5	Memory Cards	882
C.6	Dispositivos HMI Basic	882
C.7	Repuestos y hardware adicional	883
C.8	Software de programación	883
C.9	Documentación	884
	Índice alfabético	885

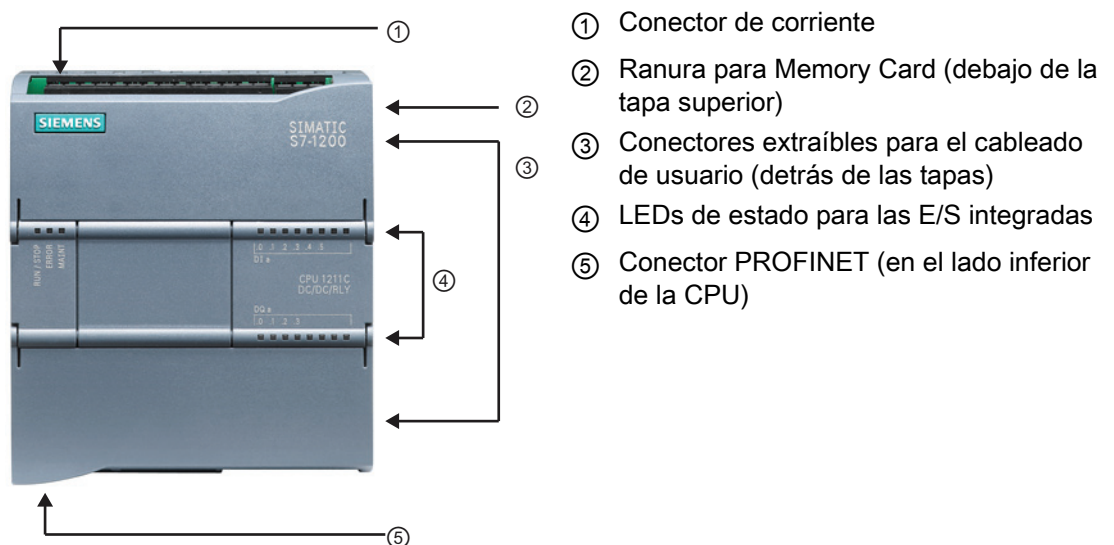
Sinopsis del producto

1.1 Introducción al PLC S7-1200

El controlador S7-1200 ofrece la flexibilidad y potencia necesarias para controlar una gran variedad de dispositivos para las distintas necesidades de automatización. Gracias a su diseño compacto, configuración flexible y amplio juego de instrucciones, el S7-1200 es idóneo para controlar una gran variedad de aplicaciones.

La CPU incorpora un microprocesador, una fuente de alimentación integrada, circuitos de entrada y salida, PROFINET integrado, E/S de control de movimiento de alta velocidad y entradas analógicas incorporadas, todo ello en una carcasa compacta, conformando así un potente controlador. Una vez cargado el programa en la CPU, ésta contiene la lógica necesaria para vigilar y controlar los dispositivos de la aplicación. La CPU vigila las entradas y cambia el estado de las salidas según la lógica del programa de usuario, que puede incluir lógica booleana, instrucciones de contaje y temporización, funciones matemáticas complejas, así como comunicación con otros dispositivos inteligentes.

La CPU incorpora un puerto PROFINET para la comunicación en una red PROFINET. Hay disponibles módulos adicionales para la comunicación en redes PROFIBUS, GPRS, RS485 o RS232.



1.1 Introducción al PLC S7-1200

Numerosas funciones de seguridad protegen el acceso tanto a la CPU como al programa de control:

- Todas las CPU ofrecen protección por contraseña (Página 168) que permite configurar el acceso a sus funciones.
- Es posible utilizar la "protección de know-how" (Página 170) para ocultar el código de un bloque específico.
- Además, la protección anticopia (Página 171) permite vincular el programa a una Memory Card o CPU específica.

Tabla 1- 1 Comparación de los modelos de CPU

Función		CPU 1211C	CPU 1212C	CPU 1214C	CPU 1215C
Dimensiones físicas (mm)		90 x 100 x 75	90 x 100 x 75	110 x 100 x 75	130 x 100 x 75
Memoria de usuario	Trabajo	30 kB	50 KB	75 kB	100 kB
	Carga	1 MB	1 MB	4 MB	4 MB
	Remanente	10 kB	10 kB	10 kB	10 kB
E/S integradas locales	Digital	6 entradas/4 salidas	8 entradas/6 salidas	14 entradas/10 salidas	14 entradas/10 salidas
	Analógico	2 entradas	2 entradas	2 entradas	2 entradas/2 salidas
Tamaño de la memoria imagen de proceso	Entradas (I)	1024 bytes	1024 bytes	1024 bytes	1024 bytes
	Salidas (Q)	1024 bytes	1024 bytes	1024 bytes	1024 bytes
Área de marcas (M)		4096 bytes	4096 bytes	8192 bytes	8192 bytes
Ampliación con módulo de señales (SM)		Ninguna	2	8	8
Signal board (SB), Battery Board (BB) o Communication Board (CB)		1	1	1	1
Módulo de comunicación (CM) (ampliación en el lado izquierdo)		3	3	3	3
Contadores rápidos	Total	3 E/S incorporadas, 5 con SB	4 E/S incorporadas, 6 con SB	6	6
	Fase simple	3 a 100 kHz SB: 2 a 30 kHz	3 a 100 kHz 1 a 30 kHz SB: 2 a 30 kHz	3 a 100 kHz 3 a 30 kHz	3 a 100 kHz 3 a 30 kHz
	Fase en cuadratura	3 a 80 kHz SB: 2 a 20 kHz	3 a 80 kHz 1 a 20 kHz SB: 2 a 20 kHz	3 a 80 kHz 3 a 20 kHz	3 a 80 kHz 3 a 20 kHz
Generadores de impulsos ¹		4	4	4	4
Memory Card		SIMATIC Memory Card (opcional)			
Tiempo de respaldo del reloj de tiempo real		20 días típ./12 días mín. a 40 °C (condensador de alto rendimiento sin mantenimiento)			
PROFINET		1 puerto de comunicación Ethernet			2 puertos de comunicación Ethernet

Función	CPU 1211C	CPU 1212C	CPU 1214C	CPU 1215C
Velocidad de ejecución de funciones matemáticas con números reales	2,3 µs/instrucción			
Velocidad de ejecución booleana	0,08 µs/instrucción			

¹ Para modelos de CPU con salidas de relé, se debe instalar una Signal Board (SB) digital para emplear las salidas de impulsos.

Cada CPU proporciona conexiones HMI dedicadas que admiten un máximo de 3 dispositivos HMI. El número total de HMI depende del tipo de paneles HMI indicados en la configuración. Así, por ejemplo, pueden conectarse a la CPU hasta tres SIMATIC Basic Panels o bien hasta dos SIMATIC Comfort Panel con un Basic Panel.

Los diferentes modelos de CPU ofrecen una gran variedad de funciones y prestaciones que permiten crear soluciones efectivas destinadas a numerosas aplicaciones. Para más información sobre una CPU en particular, consulte los datos técnicos (Página 737).

Tabla 1-2 Bloques, temporizadores y contadores que soporta el S7-1200

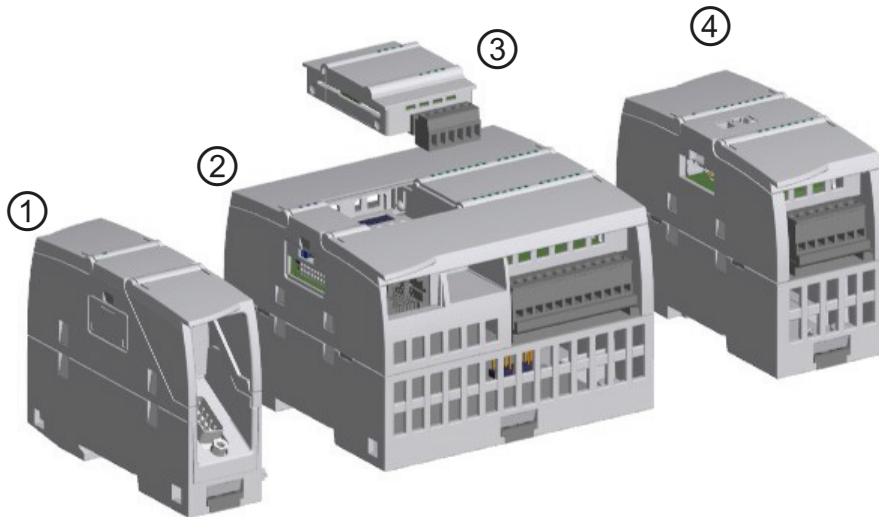
Elemento	Descripción	
Bloques	Tipo	OB, FB, FC, DB
	Tamaño	30 KB (CPU 1211C) 50 KB (CPU 1212C) 64 KB (CPU 1214C y CPU 1215C)
	Cantidad	Un total de hasta 1024 bloques (OB + FB + FC + DB)
	Rango de direcciones para FB, FC y DB	de 1 a 65535 (p. ej. del FB 1 al FB 65535)
	Profundidad de anidamiento	16 del OB de arranque o de ciclo; 4 del OB de alarma de retardo, alarma horaria, alarma cíclica, alarma de proceso, alarma de error de tiempo o alarma de diagnóstico
	Observar	Se puede observar a la vez el estado de 2 bloques lógicos
OB	Ciclo del programa	Múltiple: OB 1, de OB 200 a OB 65535
	Arranque	Múltiple: OB 100, de OB 200 a OB 65535
	Alarmas de retardo y alarmas cíclicas	4 ¹ (1 por evento): de OB 200 a OB 65535
	Alarmas de proceso (flancos y HSC)	50 (1 por evento): de OB 200 a OB 65535
	Alarmas de error de tiempo	1: OB 80
	Alarmas de error de diagnóstico	1: OB 82
Temporizadores	Tipo	CEI
	Cantidad	Sólo limitada por el tamaño de la memoria
	Almacenamiento	Estructura en DB, 16 bytes por temporizador
Contadores	Tipo	CEI
	Cantidad	Sólo limitada por el tamaño de la memoria

Elemento	Descripción
Almacenamiento	Estructura en DB, tamaño dependiente del tipo de contaje <ul style="list-style-type: none">• SInt, USInt: 3 bytes• Int, UInt: 6 bytes• DInt, UDInt: 12 bytes

¹ Las alarmas de retardo y cíclicas usan los mismos recursos en la CPU. Sólo se puede contar con un total máximo de 4 de estas alarmas (suma de alarmas de retardo y cíclicas). No puede haber 4 alarmas de retardo y 4 alarmas cíclicas.

1.2 Capacidad de expansión de la CPU

La familia S7-1200 ofrece diversos módulos y placas de conexión para ampliar las capacidades de la CPU con E/S adicionales y otros protocolos de comunicación. Para más información sobre un módulo en particular, consulte los datos técnicos (Página 737).



- ① Módulo de comunicación (CM), procesador de comunicaciones (CP) o TS Adapter
- ② CPU
- ③ Signal Board (SB), communication board (CB) o battery board (BB)
- ④ Módulo de señales (SM)

Tabla 1- 3 Módulos de señales digitales y Signal Boards

Tipo	Sólo entradas	Sólo salidas	Combinación de entradas y salidas
③ SB digital	<ul style="list-style-type: none"> Entrada: 4 x 24 V DC, 200 kHz Entrada: 4 x 5 V DC, 200 kHz 	<ul style="list-style-type: none"> Salida: 4 x 24 V DC, 200 kHz Salida: 4 x 5 V DC, 200 kHz 	<ul style="list-style-type: none"> Entrada: 2 x 24 V DC / Salida: 2 x 24 V DC Entrada: 2 x 24 V DC / Salida: 2 x 24 V DC, 200 kHz Entrada: 2 x 5 V DC / Salida: 2 x 5 V DC, 200 kHz
④ SM digital	<ul style="list-style-type: none"> Entrada: 8 x 24 V DC 	<ul style="list-style-type: none"> Salida: 8 x 24 V DC 8 salidas de relé 8 salidas de relé (conmutador) 	<ul style="list-style-type: none"> Entrada: 8 x 24 V DC / Salida: 8 x 24 V DC Entrada: 8 x 24 V DC / 8 salidas de relé 8 x 120/230VAC In / 8 x salidas de relé
	<ul style="list-style-type: none"> Entrada: 16 x 24 V DC 	<ul style="list-style-type: none"> Salida: 16 x 24 V DC 16 salidas de relé 	<ul style="list-style-type: none"> Entrada: 16 x 24 V DC / Salida: 16 x 24 V DC Entrada: 16 x 24 V DC / 16 salidas de relé

Tabla 1- 4 Módulos de señales analógicas y Signal Boards

Tipo	Sólo entradas	Sólo salidas	Combinación de entradas y salidas
③ SB analógica	<ul style="list-style-type: none"> 1 entrada analógica de 12 bits 1 RTD de 16 bits 1 termopar de 16 bits 	<ul style="list-style-type: none"> 1 salida analógica 	-
④ SM analógico	<ul style="list-style-type: none"> 4 entradas analógicas 4 entradas analógicas de 16 bits 8 entradas analógicas Termopar: <ul style="list-style-type: none"> 4 TC de 16 bits 8 TC de 16 bits RTD: <ul style="list-style-type: none"> 4 RTD de 16 bits 8 RTD de 16 bits 	<ul style="list-style-type: none"> 2 salidas analógicas 4 salidas analógicas 	<ul style="list-style-type: none"> 4 entradas analógicas/2 salidas analógicas

Tabla 1- 5 Interfaces de comunicación

Módulo	Tipo	Descripción
① Módulo de comunicación (CM)	RS232	Dúplex
	RS422/485	Dúplex (RS422) Semidúplex (RS485)
	Maestro PROFIBUS	DPV1
	Esclavo PROFIBUS	DPV1
	Maestro AS-i (CM 1243-2)	ASInterface
① Procesador de comunicaciones (CP)	Conectividad de módems	GPRS
① Placa de comunicación (CB)	RS485	Semidúplex
① TeleService	TS Adapter IE Basic ¹	Conexión con la CPU
	TS Adapter GSM	GSM/GPRS
	TS Adapter Modem	Módem
	TS Adapter ISDN	ISDN
	TS Adapter RS232	RS232

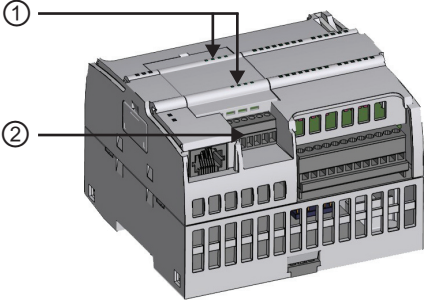
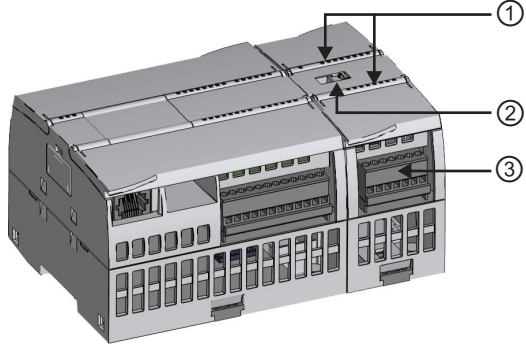
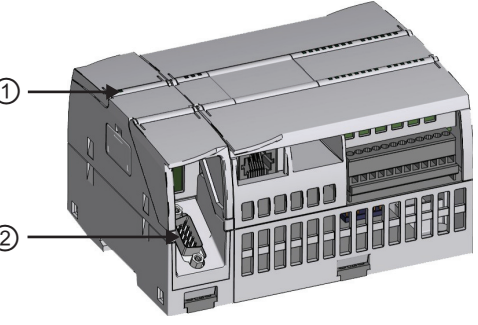
¹ El TS Adapter permite conectar varias interfaces de comunicación al puerto PROFINET de la CPU. El TS Adapter se instala en el lado izquierdo de la CPU y el TS Adapter modular (un máximo de 3) se conecta al TS Adapter.

Tabla 1- 6 Otras placas

Módulo	Descripción
③ Battery board (placa de batería)	Se enchufa en la interfaz de la placa de ampliación en la parte frontal de la CPU. Ofrece respaldo a largo plazo del reloj en tiempo real.

1.3 Módulos S7-1200

Tabla 1- 7 Módulos de ampliación S7-1200

Tipo de módulo	Descripción		
<p>La CPU soporta una placa de ampliación tipo plug-in:</p> <ul style="list-style-type: none"> • Una Signal Board (SB) proporciona E/S adicionales a la CPU. La SB se conecta en la parte frontal de la CPU. • Una placa de comunicación (CB) permite agregar un puerto de comunicación adicional a la CPU. • Una placa de batería (BB) ofrece respaldo a largo plazo del reloj en tiempo real. 		①	LEDs de estado en la SB
<p>Los módulos de señales (SM) agregan funciones a la CPU. Los SM se conectan en el lado derecho de la CPU.</p> <ul style="list-style-type: none"> • E/S digitales • E/S analógicas • RTD y termopar 		①	LEDs de estado
<p>Los módulos de comunicación (CM) y los procesadores de comunicaciones (CP) agregan opciones de comunicación a la CPU, p. ej. para la conectividad de PROFIBUS o RS232 / RS485 (para PtP, Modbus o USS) o el maestro AS-i. Un CP ofrece funcionalidades para otros tipos de comunicación, como conectar la CPU a través de una red GPRS.</p> <ul style="list-style-type: none"> • La CPU soporta hasta 3 CMs o CPs • Cada CM o CP se conecta en el lado izquierdo de la CPU (o en el lado izquierdo de otro CM o CP) 		①	LEDs de estado
		②	Conector de comunicación

1.4 Nuevas funciones

La presente versión ofrece las siguientes nuevas funciones:

- Página estándar de servidor web para actualizar el firmware de la CPU (Página 541)
- Posibilidad de utilizar tres módulos maestros PROFIBUS DP CM 1243-5 o tres módulos maestros AS-i CM 1243-2

Nota

Para utilizar tres módulos AS-i como maestros, debe actualizar el firmware de los módulos AS-i.

Nuevos módulos para S7-1200

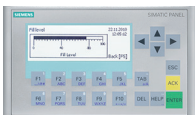
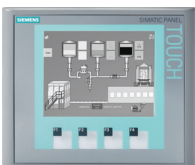
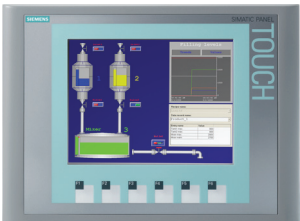
Diversos módulos nuevos amplían la potencia de la CPU S7-1200 y ofrecen la flexibilidad necesaria para cubrir las necesidades de automatización.

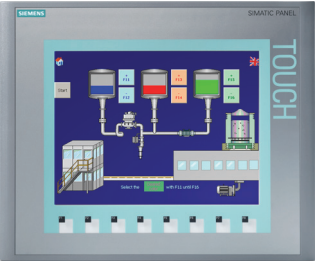
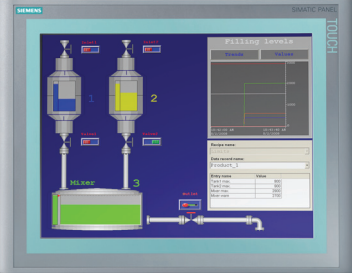
- CPU nuevas y mejoradas:
 - Las nuevas CPU 1215C DC/DC/DC, CPU 1215C DC/DC/relé y CPU 1215C AC/DC/relé ofrecen una memoria de trabajo de 100 KB, dual Ethernet y salidas analógicas.
 - Las nuevas y mejoradas CPU 1211C, CPU 1212C y CPU 1214C tienen tiempos de procesamiento menores, la posibilidad de 4 PTO (la CPU 1211C requiere una Signal Board), mayor memoria remanente (10 KB) y mayor tiempo de mantenimiento de hora (20 días).
- Nuevo módulo de señales E/S: El SM 1231 AI 4 x 16 bits ofrece mayor frecuencia de muestreo y más bits.
- La nueva placa de batería (BB 1297) ofrece respaldo a largo plazo del reloj en tiempo real. La BB 1297 se puede conectar en la ranura para Signal Board de la CPU S7-1200 (firmware 3.0 y posteriores).

Para usar los nuevos módulos debe utilizarse STEP 7 V11 SP2 Actualización 3 o posterior (Basic o Professional) y se debe descargar el hardware support package (HSP) para los nuevos módulos con el comando de menú de STEP 7 **Opciones > Support Packages**. Para agregar módulos al catálogo de hardware para STEP 7 (TIA Portal) siga las instrucciones de la página del Service & Support de Siemens (<http://support.automation.siemens.com>).

1.5 Paneles HMI Basic

Puesto que la visualización se está convirtiendo cada vez más en un componente estándar de la mayoría de las máquinas, los SIMATIC HMI Basic Panels ofrecen dispositivos con pantalla táctil para tareas básicas de control y supervisión. Todos los paneles ofrecen el grado de protección IP65 y certificación CE, UL, cULus y NEMA 4x.

Panel HMI Basic	Descripción	Datos técnicos
 <p>KP 300 Basic PN</p>	<p>Teclado de membrana de 3,6" con 10 teclas táctiles que se pueden configurar libremente</p> <ul style="list-style-type: none"> • Mono (STN, blanco y negro) • 87 mm x 31 mm (3,6") • Color de retroiluminación programado (blanco, verde, amarillo o rojo) • Resolución: 240 x 80 	<ul style="list-style-type: none"> • 250 variables • 50 sinópticos de proceso • 200 avisos • 25 curvas • 40 KB memoria de recetas • 5 recetas, 20 registros, 20 entradas
 <p>KTP 400 Basic PN</p>	<p>Pantalla táctil de 4 pulgadas con 4 teclas táctiles</p> <ul style="list-style-type: none"> • Mono (STN, escala de grises) • 76,79 mm x 57,59 mm (3,8") Horizontal o vertical • Resolución: 320 x 240 	<ul style="list-style-type: none"> • 250 variables • 50 sinópticos de proceso • 200 avisos • 25 curvas • 40 KB memoria de recetas • 5 recetas, 20 registros, 20 entradas
 <p>KTP 600 Basic PN</p>	<p>Pantalla táctil de 6 pulgadas con 6 teclas táctiles</p> <ul style="list-style-type: none"> • Color (TFT, 256 colores) o monocromo (STN, escala de grises) • 115,2 mm x 86,4 mm (5,7") Horizontal o vertical • Resolución: 320 x 240 	<ul style="list-style-type: none"> • 500 variables • 50 sinópticos de proceso • 200 avisos • 25 curvas • 40 KB memoria de recetas • 5 recetas, 20 registros, 20 entradas

Panel HMI Basic	Descripción	Datos técnicos
	<p>Pantalla táctil de 10 pulgadas con 8 teclas táctiles</p> <ul style="list-style-type: none"> • Color (TFT, 256 colores) • 211,2 mm x 158,4 mm (10,4") • Resolución: 640 x 480 	<ul style="list-style-type: none"> • 500 variables • 50 sinópticos de proceso • 200 avisos • 25 curvas • 40 KB memoria de recetas • 5 recetas, 20 registros, 20 entradas
<p>KTP 1000 Basic PN</p>	<p>Pantalla táctil de 15 pulgadas</p> <ul style="list-style-type: none"> • Color (TFT, 256 colores) • 304,1 mm x 228,1 mm (15,1") • Resolución: 1024 x 768 	<ul style="list-style-type: none"> • 500 variables • 50 sinópticos de proceso • 200 avisos • 25 curvas • 40 KB memoria de recetas (memoria flash integrada) • 5 recetas, 20 registros, 20 entradas
	<p>TP 1500 Basic PN</p>	

Software de programación STEP 7

STEP 7 ofrece un entorno confortable que permite desarrollar, editar y observar la lógica del programa necesaria para controlar la aplicación, incluyendo herramientas para gestionar y configurar todos los dispositivos del proyecto, tales como controladores y dispositivos HMI. Para poder encontrar la información necesaria, STEP 7 ofrece un completo sistema de ayuda en pantalla.

STEP 7 proporciona lenguajes de programación estándar, que permiten desarrollar de forma cómoda y eficiente el programa de control.

- KOP (esquema de contactos) es un lenguaje de programación gráfico. Su representación se basa en esquemas (Página 159) de circuitos.
- FUP (diagrama de funciones) es un lenguaje de programación que se basa en los símbolos lógicos gráficos empleados en el álgebra (Página 160) booleana.
- SCL (structured control language) es un lenguaje de programación de alto nivel basado en texto.

Al crear un bloque lógico, se debe seleccionar el lenguaje de programación que empleará dicho bloque. El programa de usuario puede emplear bloques lógicos creados con cualquiera de los lenguajes de programación.

Nota

STEP 7 es el componente de software para programación y configuración del TIA Portal. El TIA Portal, además de STEP 7, también contiene WinCC para el diseño y la ejecución de la visualización de procesos en runtime, con ayuda en pantalla para WinCC y STEP 7.

2.1 Requisitos del sistema

Para instalar el software STEP 7 en un equipo con el sistema operativo Windows XP o Windows 7, es preciso iniciar la sesión con derechos de administrador.

Tabla 2- 1 Requisitos del sistema

Hardware/software	Requisitos
Tipo de procesador	Pentium M, 1,6 GHz o similar
RAM	1 GB
Espacio disponible en el disco duro	2 GB en la unidad de disco C:\
Sistemas operativos	<ul style="list-style-type: none">• Windows XP Professional SP3• Windows 2003 Server R2 StdE SP2• Windows 7 Home Premium (solo STEP 7 Basic, no compatible con STEP 7 Professional)• Windows 7 (Professional, Enterprise, Ultimate)• Windows 2008 Server StdE R2
Tarjeta gráfica	32 MB RAM Intensidad de color de 24 bits
Resolución de la pantalla	1024 x 768
Red	Ethernet de 20 Mbits/s o más rápido
Unidad óptica	DVD-ROM

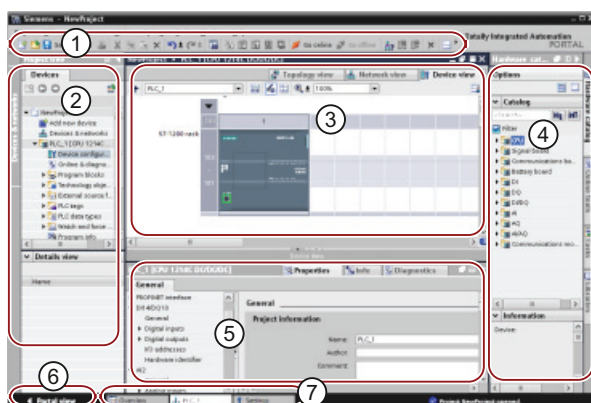
2.2 Diferentes vistas que facilitan el trabajo

STEP 7 proporciona un entorno de fácil manejo para programar la lógica del controlador, configurar la visualización de HMI y definir la comunicación por red. Para aumentar la productividad, STEP 7 ofrece dos vistas diferentes del proyecto, a saber: Distintos portales orientados a tareas y organizados según las funciones de las herramientas (vista del portal) o una vista orientada a los elementos del proyecto (vista del proyecto). El usuario puede seleccionar la vista que considere más apropiada para trabajar eficientemente. Con un solo clic es posible cambiar entre la vista del portal y la vista del proyecto.



Vista del portal

- ① Portales para las diferentes tareas
- ② Tareas del portal seleccionado
- ③ Panel de selección para la acción seleccionada
- ④ Cambia a la vista del proyecto



Vista del proyecto

- ① Menús y barra de herramientas
- ② Árbol del proyecto
- ③ Área de trabajo
- ④ Task Cards
- ⑤ Ventana de inspección
- ⑥ Cambia a la vista del portal
- ⑦ Barra del editor

Puesto que todos estos componentes se encuentran en un solo lugar, es posible acceder fácilmente a todas las áreas del proyecto. La ventana de inspección, por ejemplo, muestra las propiedades e información acerca del objeto seleccionado en el área de trabajo. Si se seleccionan varios objetos, la ventana de inspección muestra las propiedades que pueden configurarse. La ventana de inspección incluye fichas que permiten ver información de diagnóstico y otros mensajes.

La barra de editores agiliza el trabajo y mejora la eficiencia, ya que muestra todos los editores que están abiertos. Para cambiar entre los editores abiertos, basta con hacer clic sobre el editor en cuestión. También es posible visualizar dos editores simultáneamente, ya sea en mosaico vertical u horizontal. Esta función permite mover elementos entre los editores mediante Drag & Drop.

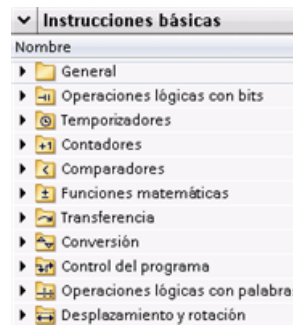
2.3 Herramientas fáciles de utilizar

2.3.1 Introducir instrucciones en el programa de usuario

STEP 7 dispone de Task Cards que contienen las instrucciones del programa. Las instrucciones se agrupan por funciones.

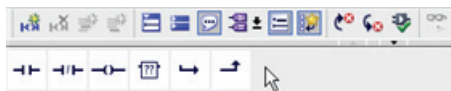


Para crear el programa, arrastre las instrucciones desde las Task Cards a los diferentes segmentos mediante Drag & Drop.

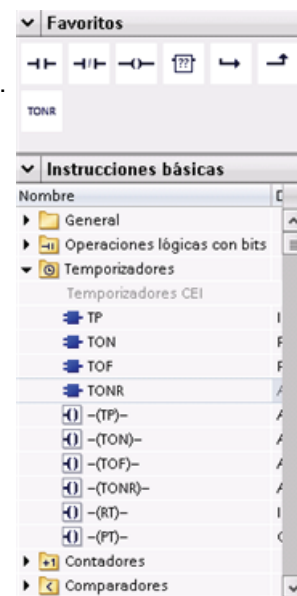
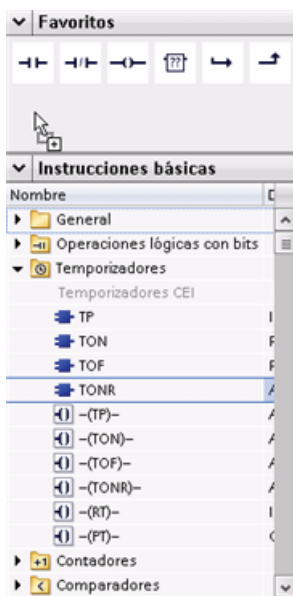


2.3.2 Acceder a instrucciones desde la barra de herramientas "Favoritos"

STEP 7 ofrece una barra de herramientas de "Favoritos" que permite acceder rápidamente a las instrucciones utilizadas con mayor frecuencia. Sólo tiene que hacer clic en el botón de la instrucción que desea insertar en el segmento.

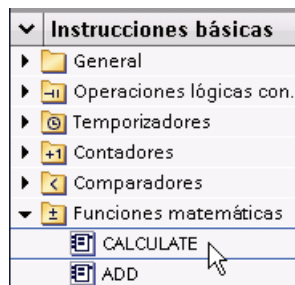


(Haga doble clic en el icono para ver los "Favoritos" en el árbol de instrucciones.)

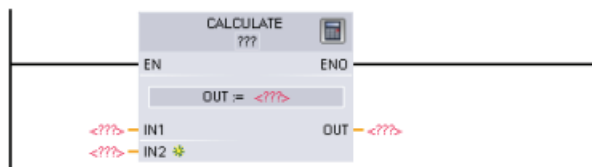


2.3.3 Crear una ecuación compleja con una instrucción sencilla

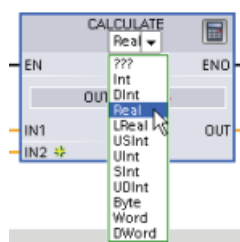
La instrucción Calculate permite crear una función matemática que se ejecuta con múltiples parámetros de entrada para obtener el resultado en función de la ecuación definida.



En el árbol de instrucciones Basic, amplíe la carpeta de funciones matemáticas. Haga doble clic en la instrucción Calculate para insertar la instrucción en el programa de usuario.



La instrucción Calculate no configurada ofrece dos parámetros de entrada y uno de salida.

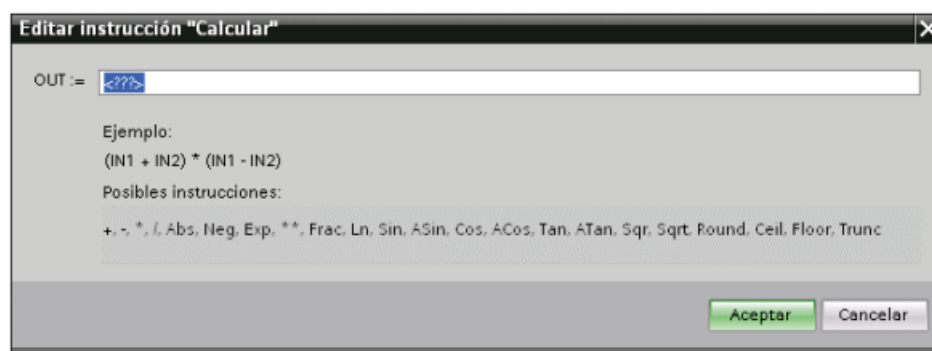


Haga clic en "???" y seleccione los tipos de datos de los parámetros de entrada y salida. (Los parámetros de entrada y salida deben tener un mismo tipo de datos).

En este ejemplo, seleccione el tipo de datos "Real".



Haga clic en el icono "Editar ecuación" para introducir la ecuación.



En este ejemplo, introduzca la ecuación siguiente para escalar un valor analógico bruto. (Las designaciones "In" y "Out" corresponden a los parámetros de la instrucción Calcular.)

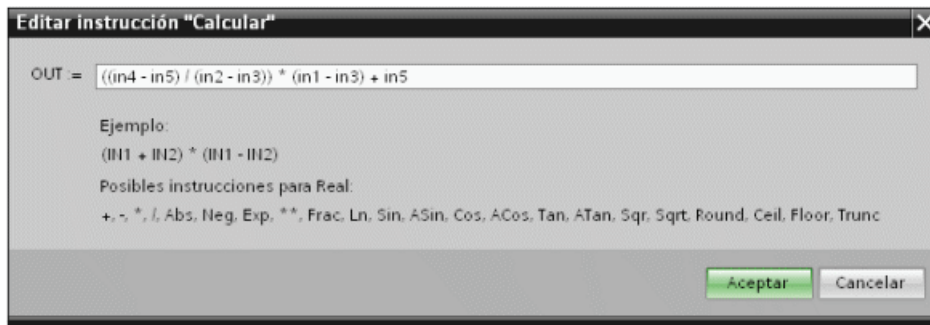
$$\text{Out value} = ((\text{Out high} - \text{Out low}) / (\text{In high} - \text{In low})) * (\text{In value} - \text{In low}) + \text{Out low}$$

$$\text{Out} = ((\text{in4} - \text{in5}) / (\text{in2} - \text{in3})) * (\text{in1} - \text{in3}) + \text{in5}$$

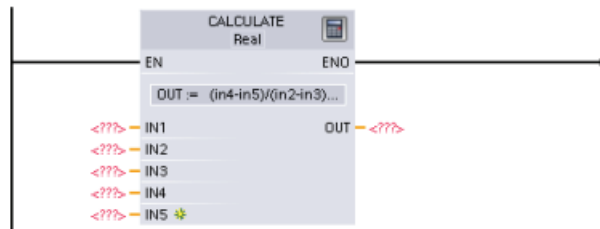
Donde:	Out value	(Out)	Valor de salida escalado
	In value	(in1)	Valor de entrada analógica
	In high	(in2)	Límite superior del valor de entrada escalado
	In low	(in3)	Límite inferior del valor de entrada escalado
	Out high	(in4)	Límite superior del valor de salida escalado
	Out low	(in5)	Límite inferior del valor de salida escalado

En el cuadro "Editar cálculo", introduzca la ecuación con los nombres de los parámetros:

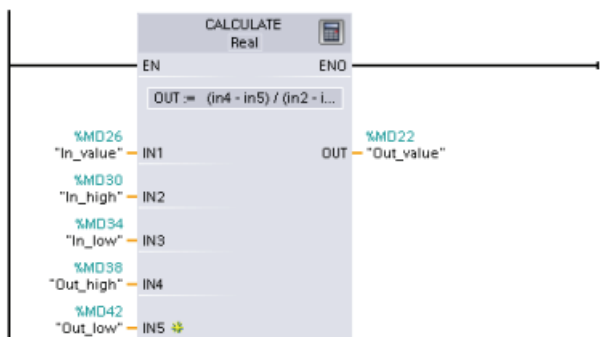
$$\text{OUT} = ((\text{in4} - \text{in5}) / (\text{in2} - \text{in3})) * (\text{in1} - \text{in3}) + \text{in5}$$



Al hacer clic en "Aceptar", la instrucción Calcular crea las entradas necesarias para la instrucción.



Introduzca los nombres de las variables de los valores que corresponden a los parámetros.



2.3.4 Agregar entradas o salidas a una instrucción KOP o FUP

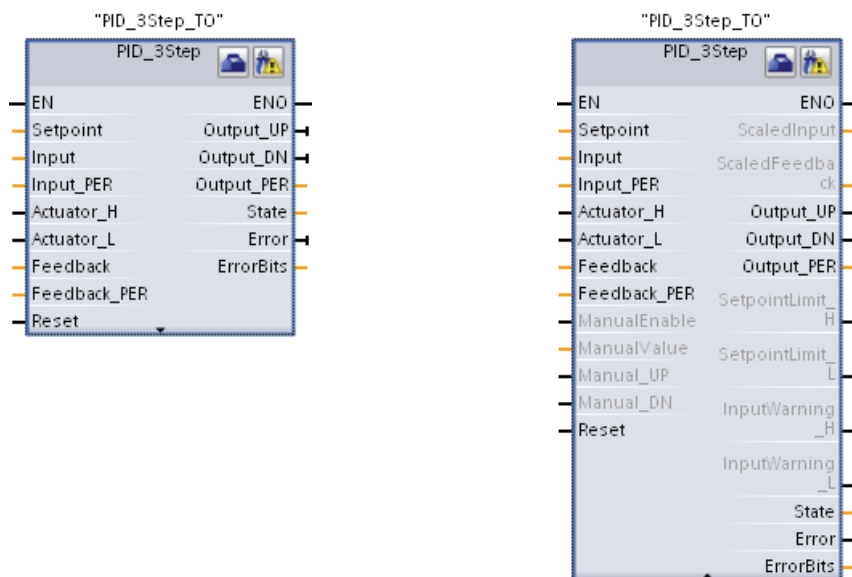


Algunas de las instrucciones permiten crear entradas o salidas adicionales.

- Para agregar una entrada o salida, haga clic en el icono "Crear" o haga clic con el botón derecho del ratón en el terminal de entrada del parámetro IN u OUT existente y seleccione el comando "Insertar entrada".
- Para quitar una entrada o salida, haga clic con el botón derecho del ratón en el conector de uno de los parámetros IN o OUT existentes (si hay más de las dos entradas originales) y seleccione el comando "Borrar".

2.3.5 Instrucciones ampliables

Algunas de las instrucciones más complejas pueden ampliarse, de modo que se visualicen únicamente las entradas y salidas clave. Para ver las entradas y salidas, haga clic en la flecha situada en la parte inferior de la instrucción.



2.3.6 Seleccionar la versión de una instrucción

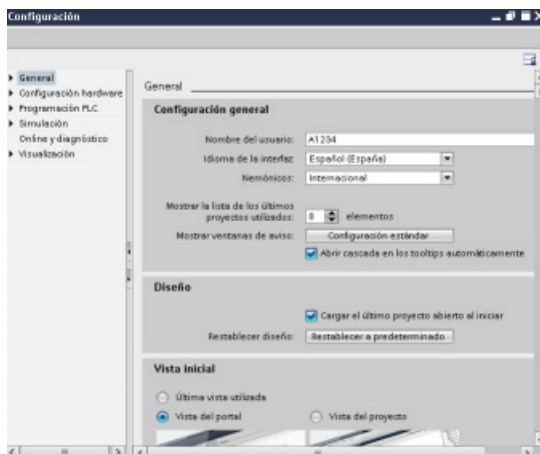
Los ciclos de desarrollo y lanzamiento de determinados conjuntos de instrucciones (como Modbus, PID y de movimiento) han dado lugar a múltiples versiones de dichas instrucciones. Para ayudar a garantizar la compatibilidad y la migración de proyectos más antiguos, STEP 7 permite seleccionar la versión de la instrucción que se va a insertar en el programa de usuario.



Haga clic en el icono de la Task Card del árbol de instrucciones para activar los encabezados y columnas del árbol de instrucciones.

Para cambiar la versión de la instrucción, seleccione la versión correspondiente en la lista desplegable.

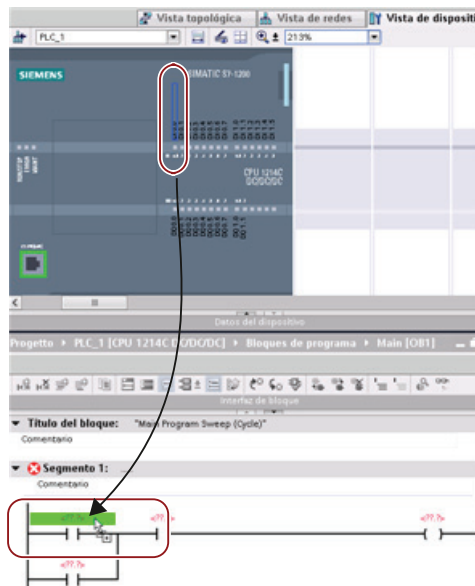
2.3.7 Modificar la apariencia y configuración de STEP 7



Es posible seleccionar una serie de ajustes, como p. ej. la apariencia de la interfaz, el idioma o la carpeta en la que guardar el trabajo.

Elija el comando "Configuración" del menú "Opciones" para cambiar dichos ajustes.

2.3.8 Arrastrar y soltar elementos entre los distintos editores

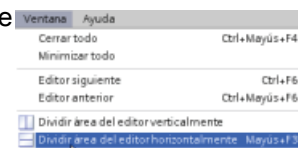


Para ayudar a realizar las tareas rápida y fácilmente, STEP 7 permite arrastrar y soltar elementos de un editor a otro. Así, es posible, por ejemplo, arrastrar una entrada de la CPU a la dirección de una instrucción del programa de usuario.

Es necesario ampliar el zoom como mínimo al 200% para poder seleccionar las entradas o salidas de la CPU.

Recuerde que los nombres de las variables no sólo se muestran en la tabla de variables PLC, sino también en la CPU.

Para ver dos editores al mismo tiempo, utilice los comandos de menú "Dividir el área del editor" o los botones correspondientes de la barra de herramientas.



Para cambiar entre los editores abiertos, haga clic en los botones de la barra de editores.



2.3.9 Cambiar el estado operativo de la CPU

Consulte

La CPU no dispone de interruptores físicos para cambiar entre los modos de operación (STOP o RUN).

Utilice los botones "Arrancar CPU" o "Parar CPU" de la barra de herramientas para cambiar el estado operativo de la CPU.



Al configurar la CPU en la configuración de dispositivos se define el comportamiento de arranque en las propiedades de la CPU.

El portal "Online y diagnóstico" ofrece también un panel de mando que permite cambiar el estado operativo de la CPU online. Para utilizar el panel de mando de la CPU es necesario que exista una conexión online con la CPU. La Task Card "Herramientas online" muestra un panel de mando en el que se indica el modo de operación de la CPU. El panel también permite cambiar el modo de operación de la CPU.

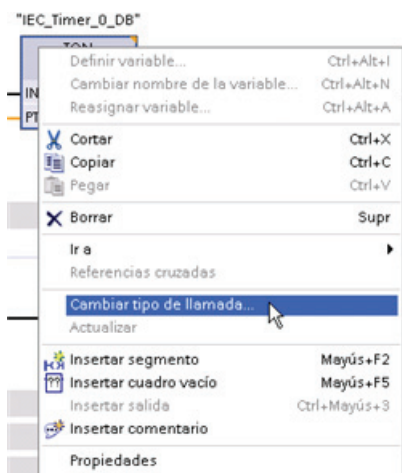


Utilice el botón del panel de mando para cambiar el modo de operación (STOP o RUN). El panel de mando también dispone de un botón MRES para inicializar la memoria.

El color del indicador RUN/STOP muestra el modo de operación actual de la CPU. El amarillo indica el estado operativo STOP y el verde RUN.

Consulte Estados operativos de la CPU en el Manual de sistema S7-1200 (Página 70) para configurar el estado operativo predeterminado al arrancar.

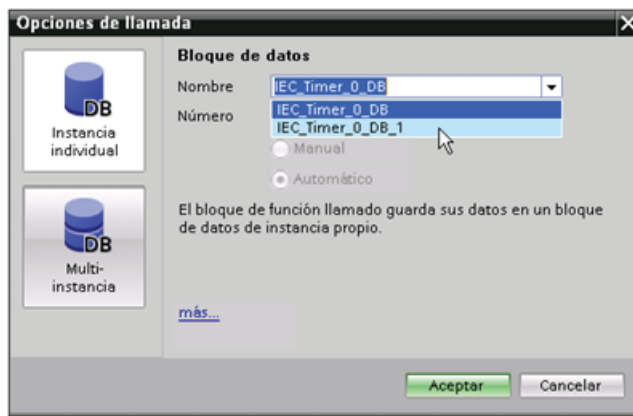
2.3.10 Modificar el tipo de llamada de un DB



STEP 7 permite crear o modificar fácilmente la asignación de un DB de una instrucción o un FB que está en un FB.

- Es posible conmutar la asignación entre diferentes DBs.
- Es posible conmutar la asignación entre un DB monoinstancia y un DB multiinstancia.
- Es posible crear un DB de instancia (si falta o no está disponible).

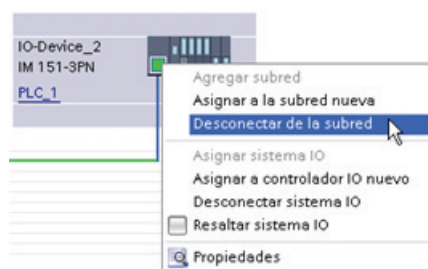
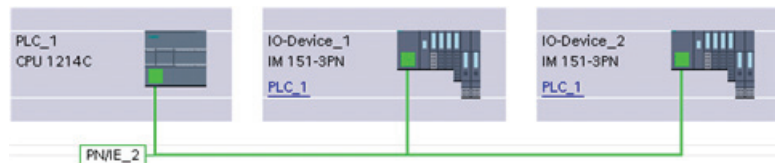
Al comando "Modificar tipo de llamada" se accede haciendo clic con el botón derecho del ratón en la instrucción o FB en el editor de programas o bien seleccionando el comando "Llamada de bloque" del menú "Opciones".



El cuadro de diálogo "Opciones de llamada" permite seleccionar un DB monoinstancia o multiinstancia. También existe la posibilidad de seleccionar DBs concretos de una lista desplegable de DBs disponibles.

2.3.11 Desconectar temporalmente dispositivos de una red

Existe la posibilidad de desconectar dispositivos de red concretos de la subred. Puesto que la configuración del dispositivo no se elimina del proyecto, resulta fácil restablecer la conexión con el dispositivo.



Haga clic con el botón derecho del ratón en el puerto de interfaz del dispositivo de red y seleccione el comando "Desconectar de la subred" del menú contextual.

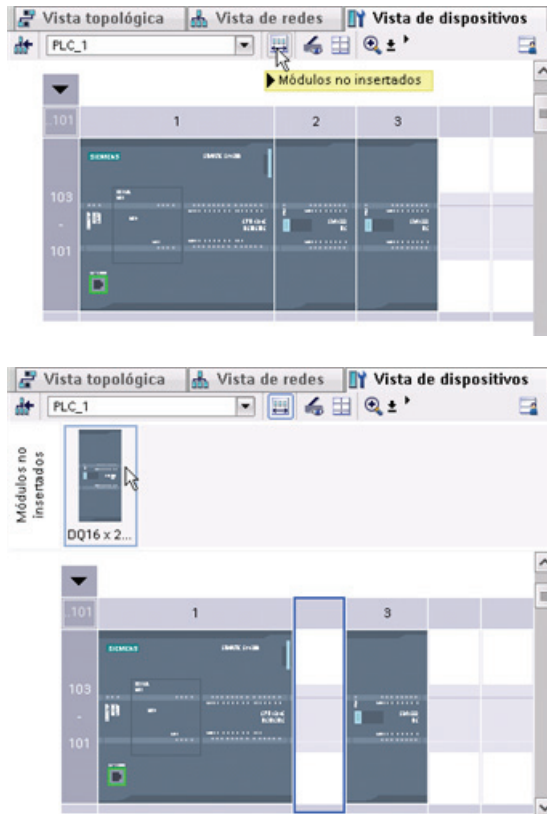
STEP 7 reconfigura las conexiones de red pero no elimina el dispositivo desconectado del proyecto. Mientras la conexión de red siga borrada, las direcciones de interfaz no cambiarán.



Cuando se descargan las nuevas conexiones de red, la CPU debe estar en modo STOP.

Para volver a conectar el dispositivo, sólo hay que crear una nueva conexión de red con el puerto del dispositivo.

2.3.12 Desconexión virtual de dispositivos desde la configuración




STEP 7 dispone de un área de almacenamiento para módulos "no enchufados". Puede arrastrar un módulo desde el rack para guardar la configuración del módulo en cuestión. Estos módulos desenchufados se guardan con el proyecto, permitiendo así volver a insertarlos en el futuro sin necesidad de configurar los parámetros.

Uno de los usos de esta función corresponde al mantenimiento temporal. Consideremos una situación en la que se está esperando un módulo de recambio y se planifica un uso temporal de un módulo diferente a modo de sustituto provisional. Es posible arrastrar el módulo configurado desde el rack a los "módulos no enchufados" y a continuación insertar el módulo provisional.

Montaje

3.1 Directrices para montar dispositivos S7-1200

Los equipos S7-1200 son fáciles de montar. El S7-1200 puede montarse en un panel o en un rail DIN, bien sea horizontal o verticalmente. El tamaño pequeño del S7-1200 permite ahorrar espacio.

 ADVERTENCIA
<p>Los PLCs S7-1200 SIMATIC son controladores abiertos. Por este motivo, el S7-1200 debe montarse en una carcasa, un armario eléctrico o una sala de control. Sólo el personal autorizado debe tener acceso a la carcasa, el armario eléctrico o la sala de control.</p> <p>Si no se cumplen los requisitos de montaje, pueden producirse la muerte, lesiones corporales graves y/o daños materiales.</p> <p>Vigile siempre los requisitos de montaje de los PLCs S7-1200.</p>

Alejar los dispositivos S71200 de fuentes de calor, alta tensión e interferencias


Como regla general para la disposición de los dispositivos del sistema, los aparatos que generan altas tensiones e interferencias deben mantenerse siempre alejados de los equipos de baja tensión y de tipo lógico, tales como el S71200.

Al configurar la disposición del S7-1200 en el panel, se deben tener en cuenta los aparatos que generan calor y disponer los equipos electrónicos en las zonas más frías del armario eléctrico. Si se reduce la exposición a entornos de alta temperatura, aumentará la vida útil de cualquier dispositivo electrónico.

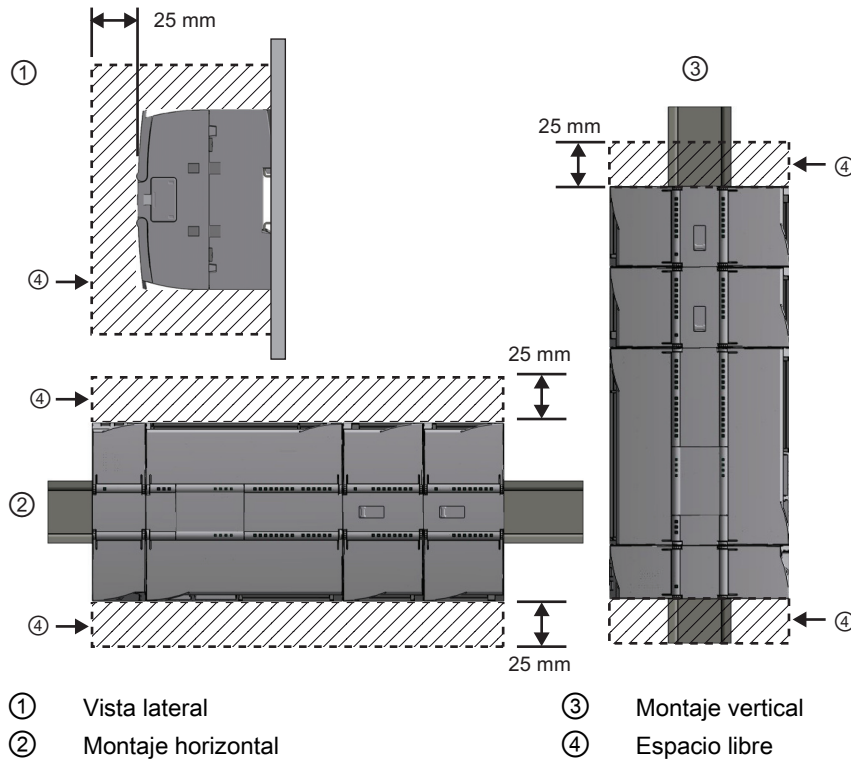
También se debe considerar la ruta del cableado de los dispositivos montados en el panel. Evite tender las líneas de señales de baja tensión y los cables de comunicación en un mismo canal junto con los cables AC y DC de alta energía y conmutación rápida.

Prever espacio suficiente para la refrigeración y el cableado

La refrigeración de los dispositivos S71200 se realiza por convección natural. Para la refrigeración correcta es preciso dejar un espacio mínimo de 25 mm por encima y por debajo de los dispositivos. Asimismo, se deben prever como mínimo 25 mm de profundidad entre el frente de los módulos y el interior de la carcasa.

 PRECAUCIÓN
<p>En el montaje vertical, la temperatura ambiente máxima admisible se reduce en 10 grados centígrados. Las S71200 montadas en vertical deben orientarse de la forma mostrada en la figura.</p>

Al planificar la disposición del sistema S71200, prevea espacio suficiente para el cableado y la conexión de los cables de comunicación.



3.2 Corriente necesaria

La CPU dispone de una fuente de alimentación interna que suministra energía eléctrica a la CPU, los módulos de señales, la Signal Board y los módulos de comunicación, así como otros consumidores de 24 V DC.

En los datos técnicos (Página 737) encontrará más información sobre la corriente de 5 V DC que suministra la CPU y la corriente de 5 V DC que requieren los módulos de señales, las Signal Boards y los módulos de comunicación. En "Calcular la corriente necesaria" (Página 875) encontrará más información sobre cómo determinar cuánta energía (o corriente) puede proveer la CPU para la configuración.

La CPU ofrece una alimentación de sensores de 24 V DC que suministra 24 V DC a las entradas y bobinas de relé de los módulos de señales, así como a otros consumidores. Si los requisitos de corriente de 24 V DC exceden la capacidad de alimentación de los sensores, es preciso añadir una fuente de alimentación externa de 24 V DC al sistema. En los datos técnicos (Página 737) se indica la corriente necesaria para la alimentación de sensores de 24 V DC de las distintas CPU S7-1200.

Nota

El CM 1243-5 (módulo maestro PROFIBUS) requiere alimentación de la alimentación de sensores de 24 V DC de la CPU.

Si se requiere una fuente de alimentación externa de 24 V DC, vigile que no se conecte en paralelo con la alimentación de sensores de la CPU. Para aumentar la protección contra interferencias, se recomienda conectar los cables neutros (M) de las distintas fuentes de alimentación.

 **ADVERTENCIA**

Si se conecta una fuente de alimentación externa de 24 V DC en paralelo a la alimentación de sensores de 24 V DC, puede surgir un conflicto entre ambas fuentes, ya que cada una intentará establecer su propio nivel de tensión de salida.

Este conflicto puede reducir la vida útil u ocasionar la avería inmediata de una o ambas fuentes de alimentación y, en consecuencia, el funcionamiento imprevisible del sistema PLC. El funcionamiento imprevisible puede producir la muerte, lesiones corporales graves y/o daños materiales.

La fuente de alimentación DC de sensores y cualquier fuente de alimentación externa deben alimentar diferentes puntos.

Algunos bornes de entrada de la alimentación de 24 V DC del sistema S7-1200 están interconectados, teniendo un circuito lógico común que conecta varios bornes M. Por ejemplo, los circuitos siguientes están interconectados si no tienen aislamiento galvánico según las hojas de datos técnicos: la fuente de alimentación de 24 V DC de la CPU, la entrada de alimentación de la bobina de relé de un SM, o bien la fuente de alimentación de una entrada analógica no aislada. Todos los bornes M sin aislamiento galvánico deben conectarse al mismo potencial de referencia externo.

 **ADVERTENCIA**

Si los bornes M sin aislamiento galvánico se conectan a diferentes potenciales de referencia, circularán corrientes indeseadas que podrían averiar o causar reacciones inesperadas en el PLC y los equipos conectados.

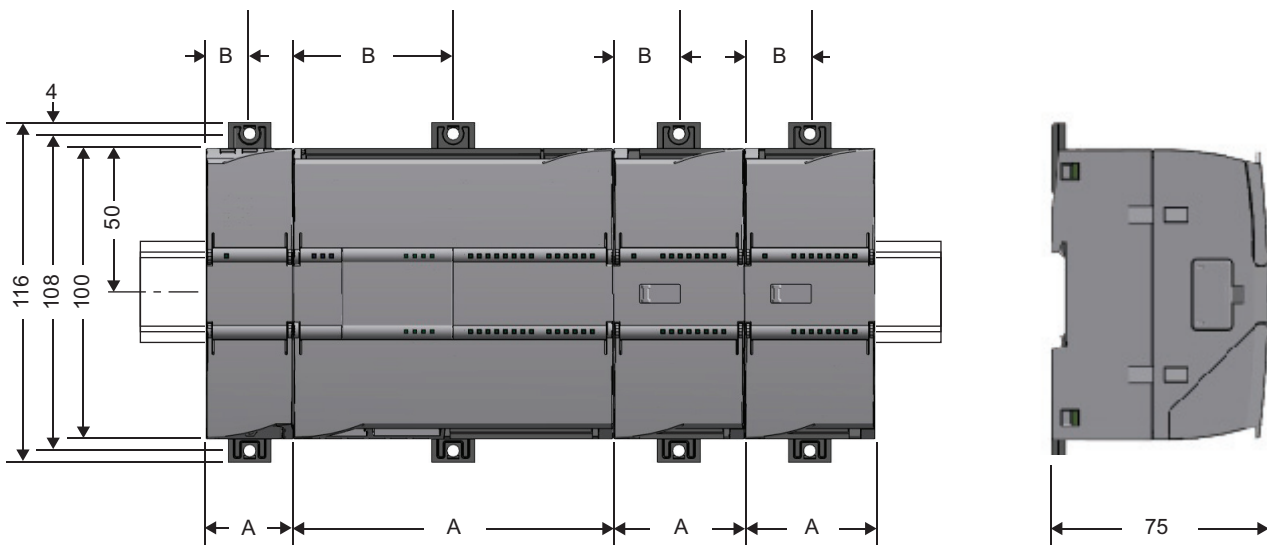
Si no se cumplen estas directrices, es posible que se produzcan averías o reacciones inesperadas que podrían causar la muerte, lesiones corporales graves y/o daños materiales.

Asegúrese que todos los bornes M sin aislamiento galvánico de un sistema S7-1200 están conectados al mismo potencial de referencia.

3.3 Procedimientos de montaje y desmontaje

3.3.1 Dimensiones de montaje de los dispositivos S7-1200

CPU 1211C, CPU 1212C, CPU 1214C



CPU 1215C

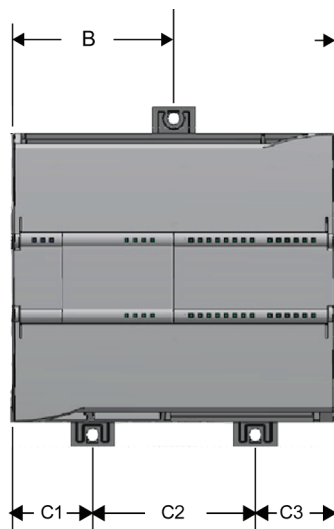


Tabla 3- 1 Dimensiones de montaje (mm)

Dispositivos S7-1200		Ancho A (mm)	Ancho B (mm)	Ancho C (mm)
CPU	CPU 1211C y CPU 1212C	90	45	--
	CPU 1214C	110	55	--
	CPU 1215C	130	65 (parte superior)	Parte inferior: C1: 32,5 C2: 65 C3: 32,5
Módulos de señales	Digitales 8 y 16 puntos Analógicas 2, 4 y 8 puntos Termopar 4 y 8 puntos RTD 4 puntos	45	22,5	--
	Digital DQ 8 x Relay (inversor)	70	22,5	--
	Analógico 16 puntos RTD 8 puntos	70	35	--
Interfaces de comunicación	CM 1241 RS232 y CM 1241 RS422/485 CM 1243-5 PROFIBUS maestro y CM 1242-5 PROFIBUS esclavo CM 1242-2 AS-i Master CP 1242-7 GPRS	30	15	--
	TS AdapterIE Basic	60 ¹	15	--

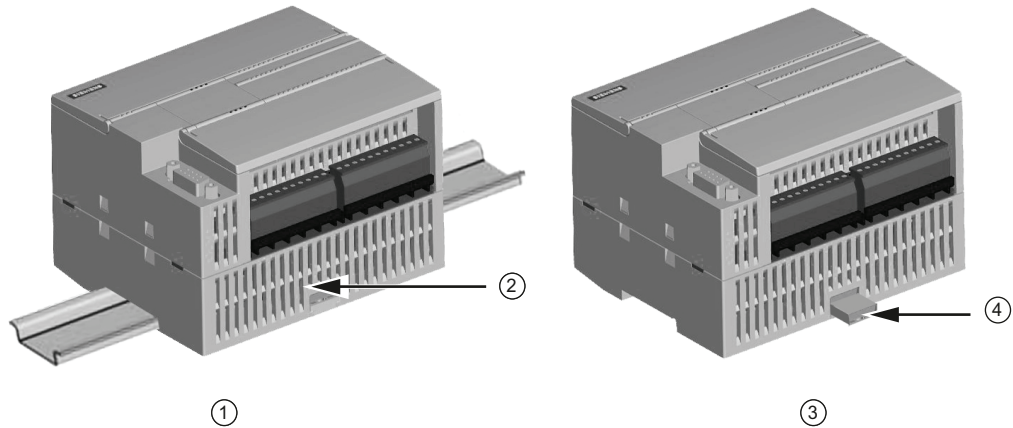
¹ Debido a que hay que instalar un TS Adapter modular con el TS Adapter, el ancho total ("ancho A") es de 60 mm.

Cada CPU, SM, CM y CP admite el montaje en un perfil DIN o en un panel. Utilice los clips del módulo previstos para el perfil DIN para fijar el dispositivo al perfil. Estos clips también pueden extenderse a otra posición para poder montar la unidad directamente en un panel. La dimensión interior del orificio para los clips de fijación en el dispositivo es 4,3 mm.

Es preciso prever una zona de disipación de 25 mm por encima y por debajo de la unidad para que el aire pueda circular libremente.

Montaje y desmontaje de dispositivos S7-1200

La CPU se puede montar fácilmente en un perfil estándar o en un panel. Los clips de fijación permiten fijar el dispositivo al perfil DIN. Estos clips también encajan en una posición extendida para proveer orificios de montaje que permiten montar el dispositivo directamente en un panel.



- ① Montaje en perfil DIN
- ② Clip de fijación al perfil enclavado
- ③ Montaje en panel
- ④ Clip de fijación en posición extendida para el montaje en panel

Antes de montar o desmontar cualquier dispositivo eléctrico, asegúrese que se ha desconectado la alimentación. Asegúrese también que está desconectada la alimentación eléctrica de todos los dispositivos conectados.

ADVERTENCIA

Si el S7-1200 o los dispositivos conectados se montan o desmontan estando conectada la alimentación, puede producirse un choque eléctrico o un funcionamiento inesperado de los dispositivos.

Si la alimentación del S7-1200 y de los dispositivos conectados no se desconecta por completo antes del montaje o desmontaje, podrían producirse la muerte, lesiones corporales graves y/o daños materiales debidos a choques eléctricos o al funcionamiento inesperado de los equipos.

Respete siempre las medidas de seguridad necesarias y asegúrese que la alimentación del S7-1200 está desconectada antes de montar o desmontar las CPU S7-1200 o los equipos conectados.

Al sustituir o montar un dispositivo S7-1200, vigile que se utilice siempre el módulo correcto o un dispositivo equivalente.

! ADVERTENCIA

El montaje incorrecto de un módulo S7-1200 puede ocasionar el funcionamiento impredecible del programa del S7-1200.

Si un dispositivo S7-1200 no se sustituye por el mismo modelo o si no se monta con la orientación correcta y en el orden previsto, podrían producirse la muerte, lesiones corporales graves y/o daños materiales debido al funcionamiento inesperado del equipo.

Sustituya un dispositivo S7-1200 por el mismo modelo y móntelo con la orientación y posición correctas.

! ADVERTENCIA

No desconecte ningún dispositivo en presencia de una atmósfera inflamable o combustible.

La desconexión de dispositivos en presencia de atmósferas inflamables o combustibles puede provocar un incendio o una explosión lo que puede producir la muerte, lesiones corporales graves y/o daños materiales.

PRECAUCIÓN

Las descargas electrostáticas pueden deteriorar el dispositivo o la ranura para tarjetas en la CPU.

Cuando utilice el dispositivo, deberá estar en contacto con una superficie conductiva puesta a tierra o llevar puesta una pulsera puesta a tierra.

3.3.2 Montaje y desmontaje de la CPU

La CPU se puede montar en un panel o en un perfil DIN.

Nota

Conecte los módulos de comunicación necesarios a la CPU y monte el conjunto en forma de unidad. Los módulos de señales se montan por separado una vez montada la CPU.

3.3 Procedimientos de montaje y desmontaje

Al montar las unidades en un perfil DIN o panel deben considerarse los siguientes puntos:

- Para el montaje en un raíl DIN, asegúrese de que el clip de fijación superior está en la posición enclavada (interior) y que el clip de fijación inferior está extendido, tanto en la CPU como en los CMs acoplados.
- Una vez montados los dispositivos en el perfil DIN, enclave los clips de sujeción para sujetar los dispositivos al raíl.
- Para el montaje en un panel, asegúrese de que los clips de fijación al raíl DIN están en posición extendida.

Para montar la CPU en un panel, proceda del siguiente modo:

1. Posicione y taladre los orificios de montaje (M4), según las dimensiones indicadas en la tabla Dimensiones de montaje (mm) (Página 44).
2. Asegúrese de que la CPU y todo el equipamiento S7-1200 están desconectados de la tensión eléctrica.
3. Extienda los clips de fijación del módulo. Asegúrese que los clips de fijación al perfil DIN en los lados superior e inferior de la CPU están en posición extendida.
4. Atornille el módulo al panel utilizando un tornillo M4 de cabeza alomada con una arandela elástica y otra plana. No utilice un tornillo de cabeza avellanada.

Nota

El tipo de tornillo viene determinado por el material en el que se monta. Aplique el par adecuado hasta que la arandela elástica quede plana. No aplique un par excesivo a los tornillos de montaje. No utilice un tornillo de cabeza avellanada.

Nota

Si el sistema está sometido a vibraciones fuertes o si se monta verticalmente, el montaje en panel ofrece mayor protección al S7-1200.

Tabla 3- 2 Instalar la CPU en un perfil DIN

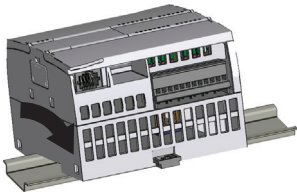
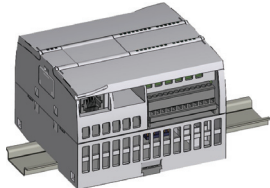
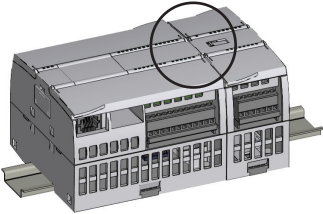
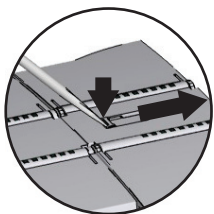
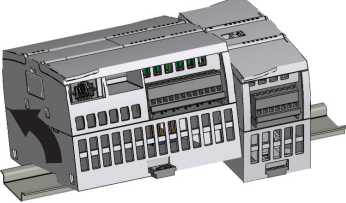
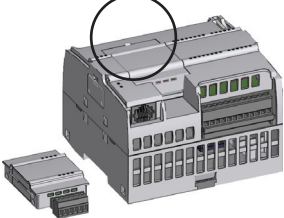
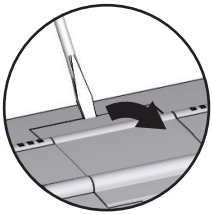
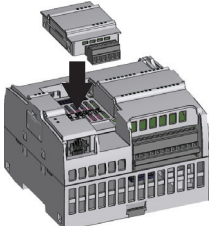
Tarea	Procedimiento
	<ol style="list-style-type: none"> 1. Monte el perfil DIN. Atornille el perfil al panel de montaje dejando un espacio de 75 mm entre tornillo y tornillo. 2. Asegúrese de que la CPU y todo el equipamiento S7-1200 están desconectados de la tensión eléctrica. 3. Enganche la CPU por el lado superior del perfil. 4. Extraiga el clip de fijación en el lado inferior de la CPU de manera que asome por encima del perfil.
	<ol style="list-style-type: none"> 5. Gire la CPU hacia abajo para posicionarla correctamente en el perfil. 6. Oprima los clips hasta que la CPU encaje en el perfil.

Tabla 3- 3 Retirar la CPU de un perfil DIN

Tarea	Procedimiento	
		<ol style="list-style-type: none"> 1. Asegúrese de que la CPU y todo el equipamiento S7-1200 están desconectados de la tensión eléctrica. 2. Desconecte los conectores de E/S, el cableado y los cables restantes de la CPU (Página 54). 3. Desmonte la CPU y los módulos de comunicación conectados en forma de conjunto. Todos los módulos de señales deben permanecer montados.
	<ol style="list-style-type: none"> 4. Si hay un SM conectado a la CPU, retire el conector de bus: <ul style="list-style-type: none"> – Coloque un destornillador junto a la lengüeta en el lado superior del módulo de señales. – Oprima hacia abajo para desenclavar el conector de la CPU. – Desplace la lengüeta por completo hacia la derecha. 5. Desmonte la CPU: <ul style="list-style-type: none"> – Extraiga el clip de fijación para desenclavar la CPU del perfil DIN. – Gire la CPU hacia arriba, extráigala del perfil y retírela del sistema. 	

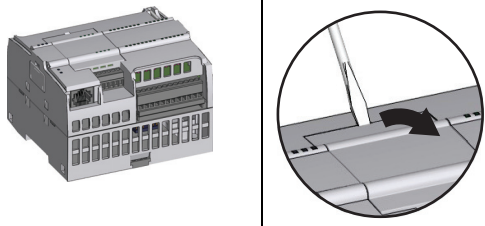
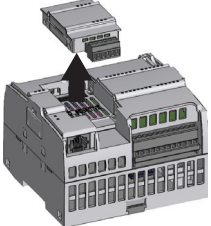
3.3.3 Montaje y desmontaje de SB, CB o BB

Tabla 3- 4 Montaje de SB, CB o BB 1297

Tarea	Procedimiento	
	<ol style="list-style-type: none"> 1. Asegúrese de que la CPU y todo el equipamiento S7-1200 están desconectados de la tensión eléctrica. 2. Retire las cubiertas de bloque de terminales superior e inferior de la CPU. 3. Inserte un destornillador en la ranura arriba de la CPU en el lado posterior de la tapa. 4. Haga palanca suavemente para levantar la tapa y retírela de la CPU. 	
		<ol style="list-style-type: none"> 5. Coloque el módulo recto en su posición de montaje en el lado superior de la CPU. 6. Presione firmemente el módulo hasta que encaje en su posición. 7. Coloque nuevamente las tapas de los bloques de terminales.

3.3 Procedimientos de montaje y desmontaje

Tabla 3- 5 Desmontaje de SB, CB o BB 1297

Tarea	Procedimiento
	<ol style="list-style-type: none"> 1. Asegúrese de que la CPU y todo el equipamiento S7-1200 están desconectados de la tensión eléctrica. 2. Retire las cubiertas de bloque de terminales superior e inferior de la CPU. 3. Inserte un destornillador en la ranura en el lado superior del módulo. 4. Haga palanca suavemente para desacoplar el módulo de la CPU.
	<ol style="list-style-type: none"> 5. Retire el módulo recto desde arriba de su posición de montaje en el lado superior de la CPU. 6. Vuelva a colocar la cubierta en la CPU. 7. Coloque nuevamente las tapas de los bloques de terminales.

Instalación o sustitución de la batería en la BB 1297

La BB 1297 requiere una batería de tipo CR1025. La batería no se suministra con la BB 1297 y debe adquirirla el usuario. Para instalar o sustituir la batería, proceda del siguiente modo:

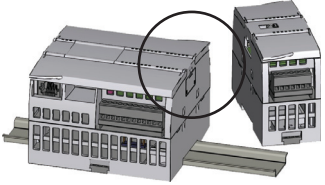
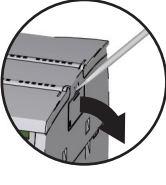
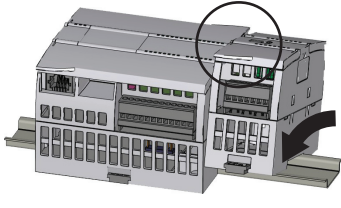

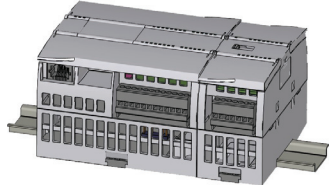
1. En la BB 1297, instale una batería nueva con el lado positivo de la batería hacia arriba y el lado negativo junto a la tarjeta de circuito impreso.
2. La BB 1297 está lista para su instalación en la CPU. Siga las instrucciones de instalación anteriores para instalar la BB 1297.

Para sustituir la batería de la BB 1297:

1. Extraiga la BB 1297 de la CPU según las instrucciones de extracción indicadas anteriormente.
2. Retire con cuidado la batería vieja con la ayuda de un destornillador pequeño. Extraiga la batería de debajo del clip.
3. Instale una nueva batería de repuesto CR1025 con el lado positivo de la batería hacia arriba y el lado negativo junto a la tarjeta de circuito impreso.
4. Vuelva a instalar la BB 1297 según las instrucciones de instalación indicadas anteriormente.

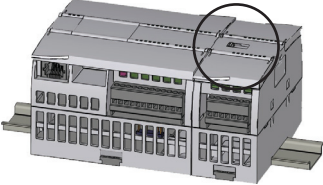
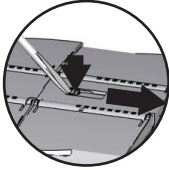
3.3.4 Instalación y desmontaje de un SM

Tabla 3- 6 Instalación de un SM

Tarea	Procedimiento	
		<p>El SM se monta una vez montada la CPU.</p> <ol style="list-style-type: none"> 1. Asegúrese de que la CPU y todo el equipamiento S7-1200 están desconectados de la tensión eléctrica. 2. Retire la tapa del conector en el lado derecho de la CPU. 3. Inserte un destornillador en la ranura arriba de la tapa. 4. Haga palanca suavemente en el lado superior de la tapa y retírela. Guarde la tapa para poder reutilizarla.
		<p>Conecte el SM a la CPU:</p> <ol style="list-style-type: none"> 1. Coloque el SM junto a la CPU. 2. Enganche el SM por el lado superior del perfil DIN. 3. Extraiga el clip de fijación inferior para colocar el SM sobre el perfil. 4. Gire el SM hacia abajo hasta su posición junto a la CPU y oprima el clip de fijación inferior para enclavar el SM en el perfil.
	<p>Al extender el conector de bus se crean las conexiones mecánicas y eléctricas para el SM.</p> <ol style="list-style-type: none"> 1. Coloque un destornillador junto a la lengüeta en el lado superior del SM. 2. Desplace la lengüeta por completo hacia la izquierda para extender el conector de bus hacia la CPU. <p>Siga el mismo procedimiento para montar un módulo de señales en otro módulo de señales.</p>	

3.3 Procedimientos de montaje y desmontaje

Tabla 3-7 Desmontaje de un SM

Tarea	Procedimiento
	<p>Cualquier SM se puede desmontar sin necesidad de desmontar la CPU u otros SMs.</p> <ol style="list-style-type: none"> 1. Asegúrese de que la CPU y todo el equipamiento S7-1200 están desconectados de la tensión eléctrica. 2. Desconecte los conectores de E/S y el cableado del SM (Página 54). 3. Retraiga el conector de bus. <ul style="list-style-type: none"> - Coloque un destornillador junto a la lengüeta en el lado superior del SM. - Oprima hacia abajo para desenclavar el conector de la CPU. - Desplace la lengüeta por completo hacia la derecha. <p>Si hay otro SM en el lado derecho, repita este procedimiento para ese SM.</p>
	<p>Desmonte el SM:</p> <ol style="list-style-type: none"> 1. Extraiga el clip de fijación inferior para desenclavar el SM del perfil DIN. 2. Gire el SM hacia arriba y extráigalo del perfil. Retire el SM del sistema. 3. En caso necesario, cubra el conector de bus de la CPU para impedir que se ensucie. <p>Siga el mismo procedimiento para desmontar un módulo de señales de otro módulo de señales.</p>

3.3.5 Montaje y desmontaje de un CM o CP

Conecte los módulos de comunicación necesarios a la CPU y monte el conjunto como una unidad, tal y como se muestra en Instalación y desmontaje de una CPU (Página 47).

Tabla 3- 8 Instalación de un CM o CP

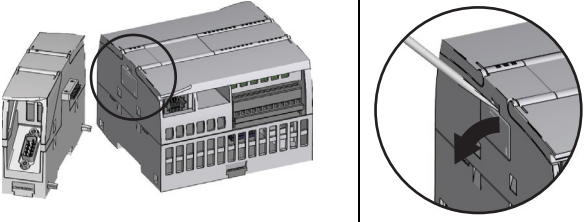
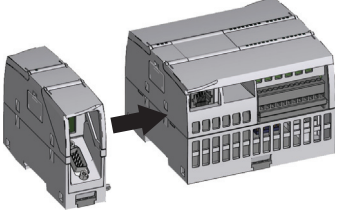
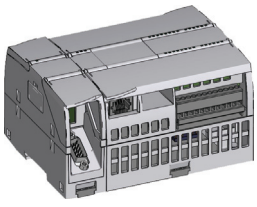
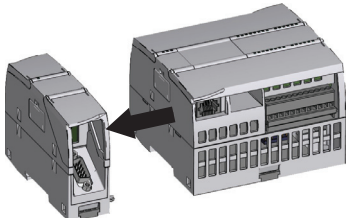
Tarea	Procedimiento
	<ol style="list-style-type: none"> 1. Asegúrese de que la CPU y todo el equipamiento S7-1200 están desconectados de la tensión eléctrica. 2. Acople el CM a la CPU antes de montar el conjunto en forma de unidad en el perfil DIN o panel. 3. Retire la tapa de bus en el lado izquierdo de la CPU: <ul style="list-style-type: none"> – Inserte un destornillador en la ranura arriba de la tapa de bus. – Haga palanca suavemente en el lado superior de la tapa.
	<ol style="list-style-type: none"> 4. Retire la tapa de bus. Guarde la tapa para poder reutilizarla. 5. Conecte el CM o CP a la CPU: <ul style="list-style-type: none"> – Alinee el conector de bus y las clavijas del CM con los orificios de la CPU. – Empuje firmemente una unidad contra la otra hasta que encajen las clavijas. 6. Instale la CPU y el CP en un perfil DIN o panel.

Tabla 3- 9 Desmontaje de un CM o CP

Tarea	Procedimiento
	<p>Desmonte la CPU y el CM en forma de unidad del raíl DIN o panel.</p> <ol style="list-style-type: none"> 1. Asegúrese de que la CPU y todo el equipamiento S7-1200 están desconectados de la tensión eléctrica. 2. Desconecte los conectores de E/S y retire el cableado y demás cables de la CPU y los CMs. 3. Para el montaje en un raíl DIN, extienda los clips de sujeción inferiores de la CPU y los CMs.
	<ol style="list-style-type: none"> 4. Desmonte la CPU y los CMs del raíl DIN o panel. 5. Sujete la CPU y los CMs firmemente con las manos y sepárelos.

PRECAUCIÓN

No utilice herramientas para separar los módulos, puesto que podrían deteriorarse.

3.3.6 Extraer y reinsertar el conector del bloque de terminales del S7-1200

La CPU, la SB y los módulos SM incorporan conectores extraíbles que facilitan la conexión del cableado.

Tabla 3- 10 Extracción del conector

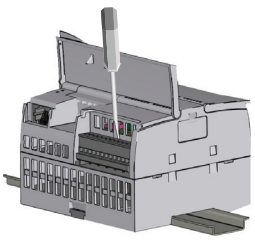
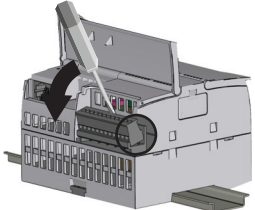
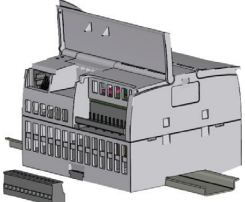
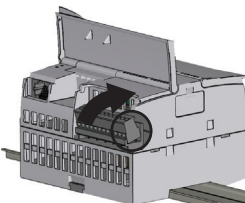
Tarea	Procedimiento
	<p>Prepare el sistema para la extracción del bloque de terminales desconectando la alimentación de la CPU y abriendo la tapa situada sobre el bloque de terminales.</p> <ol style="list-style-type: none">1. Asegúrese de que la CPU y todo el equipamiento S7-1200 están desconectados de la tensión eléctrica.2. Busque la ranura para insertar la punta del destornillador en el lado superior del conector.3. Inserte un destornillador en la ranura.4. Haga palanca suavemente en el lado superior del conector para extraerlo de la CPU. El conector se desenclava audiblemente.5. Sujete el conector con las manos y extráigalo de la CPU.
	

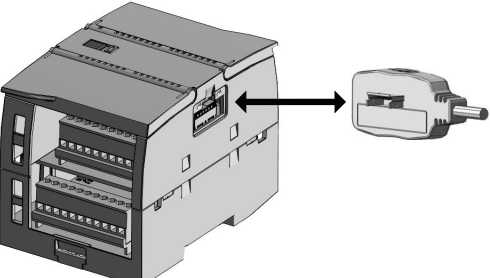
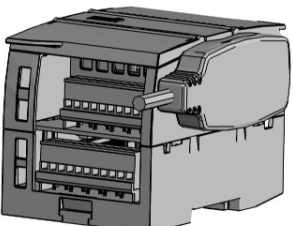
Tabla 3- 11 Instalación del conector

Tarea	Procedimiento
	<p>Prepare los componentes para el montaje del bloque de terminales desconectando la alimentación de la CPU y abriendo la tapa para el conector.</p> <ol style="list-style-type: none"> 1. Asegúrese de que la CPU y todo el equipamiento S7-1200 están desconectados de la tensión eléctrica. 2. Alinee el conector a los pines del dispositivo. 3. Alinee el lado de cableado del conector en el zócalo. 4. Con un movimiento giratorio, empuje firmemente el conector hacia abajo hasta que encaje.
	<p>Compruebe si el conector está bien alineado y encajado correctamente.</p>

3.3.7 Instalación y desmontaje de un cable de ampliación

El cable de ampliación S7-1200 proporciona una flexibilidad adicional a la hora de configurar la estructura del sistema S7-1200. Sólo se permite un cable de ampliación por sistema de CPU. El cable de ampliación se instala o bien entre la CPU y el primer SM, o bien entre dos SMs.

Tabla 3- 12 Instalación y desmontaje del conector macho del cable de ampliación

Tarea	Procedimiento
	<p>Para montar el conector macho:</p> <ol style="list-style-type: none"> 1. Asegúrese de que la CPU y todo el equipamiento S7-1200 están desconectados de la tensión eléctrica. 2. Presione el conector en el conector de bus del lado derecho del módulo de señales o la CPU. <p>Para retirar el conector macho:</p> <ol style="list-style-type: none"> 1. Asegúrese de que la CPU y todo el equipamiento S7-1200 están desconectados de la tensión eléctrica. 2. Tire del conector hembra para soltarlo del módulo de señales o la CPU.
	

3.3 Procedimientos de montaje y desmontaje

Tabla 3- 13 Montaje del conector hembra del cable de ampliación

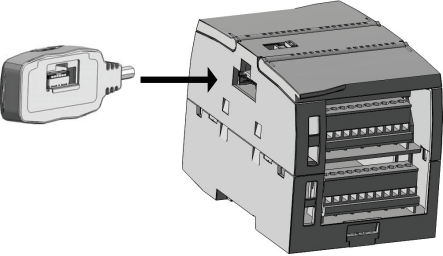
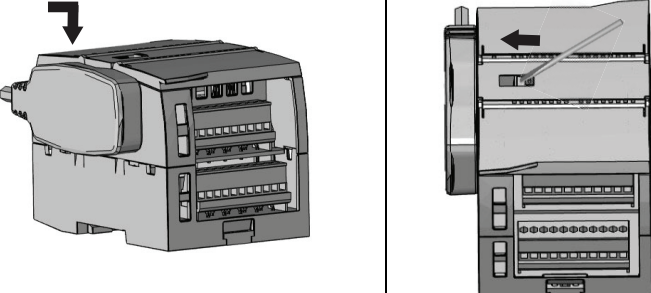
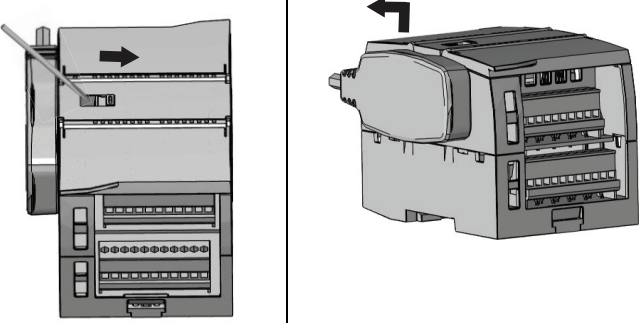
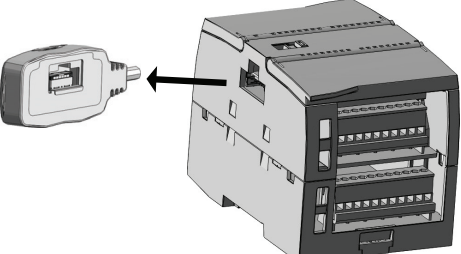
Tarea	Procedimiento
	<ol style="list-style-type: none"> 1. Asegúrese de que la CPU y todo el equipamiento S7-1200 están desconectados de la tensión eléctrica. 2. Coloque el conector hembra en el conector de bus del lado izquierdo del módulo de señales. 3. Deslice la extensión con gancho del conector hembra en el interior de la carcasa en el conector de bus y presione suavemente hacia abajo para encastrar el gancho.
	<ol style="list-style-type: none"> 4. Bloquee el conector en posición: <ul style="list-style-type: none"> - Coloque un destornillador junto a la lengüeta en el lado superior del módulo de señales. - Desplace la lengüeta por completo hacia la izquierda. <p>Para encastrar el conector, deslice la lengüeta al máximo hacia la izquierda. La lengüeta debe quedar bloqueada en posición.</p>

Tabla 3- 14 Desmontaje del conector hembra del cable de ampliación

Tarea	Procedimiento
	<ol style="list-style-type: none"> 1. Asegúrese de que la CPU y todo el equipamiento S7-1200 están desconectados de la tensión eléctrica. 2. Desbloquee el conector: <ul style="list-style-type: none"> - Coloque un destornillador junto a la lengüeta en el lado superior del módulo de señales. - Presione hacia abajo suavemente y deslice la lengüeta por completo hacia la derecha. 3. Levante el conector ligeramente para desenclavar la ampliación de gancho. 4. Retire el conector hembra.
	

3.3.8 TS (TeleService) Adapter

3.3.8.1 Conectar el adaptador de TeleService

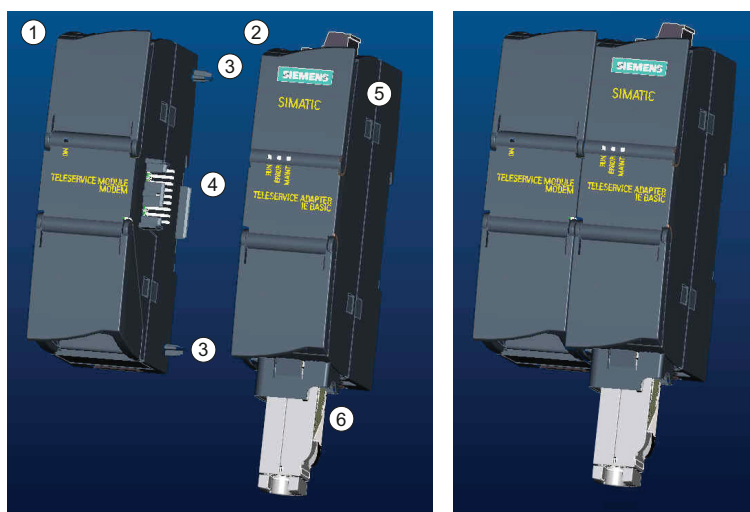
Antes de instalar el TS (TeleService) Adapter IE Basic hay que conectar el TS Adapter y un módulo TS.

Módulos TS disponibles:

- Módulo TS RS232
- Módulo TS Módem
- Módulo TS GSM
- Módulo TS RDSI

PRECAUCIÓN

Es posible que el módulo TS se dañe al tocar los contactos del conector base ④ del módulo TS. Observe las normas ESD para evitar dañar el módulo TS debido a una descarga electrostática. Antes de conectar un módulo TS y el TS Adapter, asegúrese de que ambos están inactivos.



- | | | | |
|---|------------|---|-----------------------------|
| ① | Módulo TS | ④ | Conector base del módulo TS |
| ② | TS Adapter | ⑤ | No se puede abrir |
| ③ | Elementos | ⑥ | Puerto Ethernet |

PRECAUCIÓN

Antes de conectar un módulo TS con la unidad básica del TS Adapter, asegúrese de que las clavijas ④ no estén dobladas. Al realizar la conexión, asegúrese de que el conector macho y los elementos de guiado están bien colocados.

Un módulo TS solo debe conectarse al TS Adapter. No conecte a la fuerza el TS Adapter a un dispositivo distinto, p. ej. a una CPU S7-1200. No modifique la construcción mecánica del conector ni elimine o destruya los elementos de guiado.

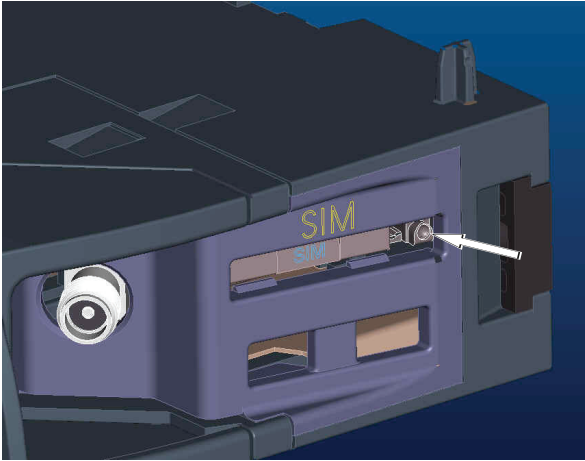
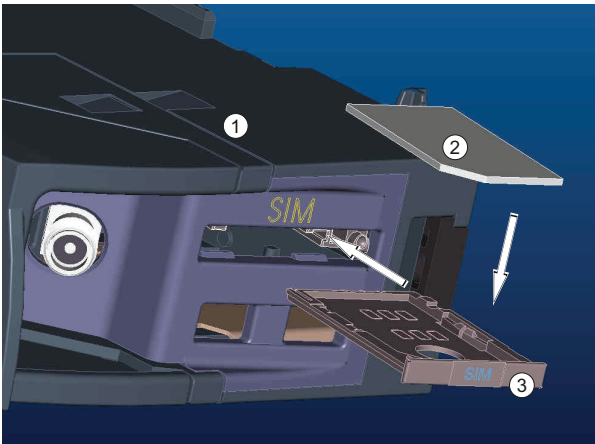
3.3.8.2 Instalar la tarjeta SIM

Localice el slot de la tarjeta SIM en la parte inferior del TS module GSM.

ATENCIÓN

La tarjeta SIM solo debe retirarse o insertarse si el TS module GSM no tiene tensión.

Tabla 3- 15 Instalar la tarjeta SIM

Procedimiento	Tarea							
<p>Utilice un objeto puntiagudo para pulsar el botón de expulsión de la bandeja de la tarjeta SIM (en la dirección de la flecha) y saque la bandeja.</p>								
<p>Coloque la tarjeta SIM en la bandeja en la posición mostrada y vuelva a colocar la bandeja de la tarjeta SIM en el slot.</p>		<table border="1"> <tr> <td data-bbox="1155 887 1219 927">①</td> <td data-bbox="1219 887 1479 927">TS Module GSM</td> </tr> <tr> <td data-bbox="1155 927 1219 967">②</td> <td data-bbox="1219 927 1479 967">Tarjeta SIM</td> </tr> <tr> <td data-bbox="1155 967 1219 1350">③</td> <td data-bbox="1219 967 1479 1350">Bandeja de tarjeta SIM</td> </tr> </table>	①	TS Module GSM	②	Tarjeta SIM	③	Bandeja de tarjeta SIM
①	TS Module GSM							
②	Tarjeta SIM							
③	Bandeja de tarjeta SIM							

Nota

Asegúrese de que la bandeja de la tarjeta SIM está bien orientada. De lo contrario, la tarjeta SIM no hará contacto con el módulo y es posible que el botón de expulsión no expulse la bandeja.

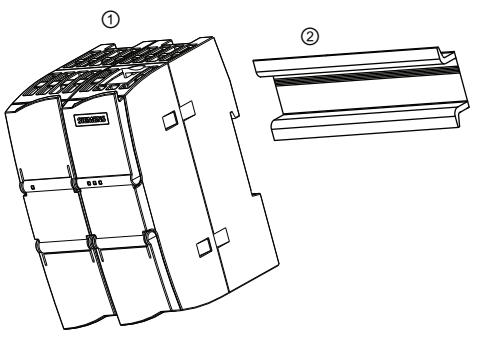
3.3.8.3 Montaje del adaptador TS


Requisitos: el TS Adapter y el módulo TS deben estar conectados y el perfil DIN debe estar instalado.

Nota

Si se instala la unidad TS en posición vertical o en un entorno con muchas vibraciones, es posible que el módulo TS se desconecte del TS Adapter. Utilice un ángulo final 8WA1 808 en el perfil DIN para que los módulos permanezcan conectados.

Tabla 3- 16 Montar y desmontar el TS Adapter

Tarea	Procedimiento
	<p>Montaje:</p> <ol style="list-style-type: none"> Enganche el TS Adapter con el módulo TS conectado ① en el perfil DIN ②. Gire la unidad hacia atrás hasta que encaje. Presione el clip de fijación en cada módulo para fijarlos al perfil DIN. <p>Desmontaje:</p> <ol style="list-style-type: none"> Retire el cable analógico y el cable Ethernet de la parte inferior del TS Adapter. Desconecte la alimentación del TS Adapter. Suelte los clips del perfil en ambos módulos utilizando un destornillador. Gire la unidad hacia arriba para quitarla del perfil DIN.

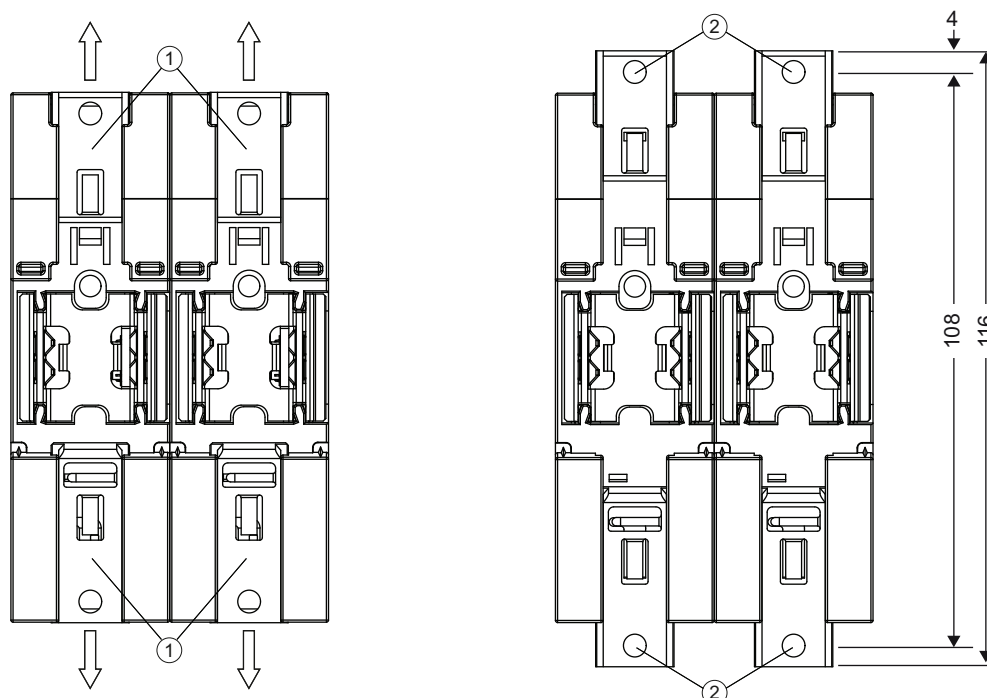
 ADVERTENCIA
<p>Antes de desconectar la alimentación de la unidad, desconecte la toma de tierra del TS Adapter quitando el cable analógico y el cable Ethernet.</p>

3.3.8.4 Montaje del adaptador TS en una pared

Requisitos: el TS Adapter y el módulo TS deben estar conectados.

- Desplace la lengüeta deslizante de fijación ① situada en la parte posterior del TS Adapter y el módulo TS en la dirección de la flecha hasta que encaje.
- Atornille el TS Adapter y el módulo TS en la posición señalada por ② en la pared de montaje deseada.

La figura siguiente muestra la parte posterior del TS Adapter, con las lengüetas deslizantes de fijación ① en ambas posiciones:



- ① Lengüeta deslizante de fijación
- ② Orificios para el montaje en la pared

3.4 Directrices de cableado

La puesta a tierra y el cableado correctos de todos los equipos eléctricos es importante para garantizar el funcionamiento óptimo del sistema y aumentar la protección contra interferencias de la aplicación y del S7-1200. Encontrará los diagramas de cableado del S7-1200 en los datos técnicos (Página 737).

Requisitos

Antes de poner a tierra o cablear cualquier dispositivo eléctrico, asegúrese que la alimentación está desconectada. Asegúrese también que está desconectada la alimentación eléctrica de todos los equipos conectados.

Vigile que se respeten todos los reglamentos eléctricos vinculantes al cablear el S7-1200 y los equipos conectados. El equipo se debe montar y operar conforme a todas las normas nacionales y locales vigentes. Contacte con las autoridades locales para determinar qué reglamentos y normas rigen en su caso específico.

 **ADVERTENCIA**

Si el S7-1200 o los equipos conectados se montan o cablean estando conectada la alimentación, puede producirse un choque eléctrico o un funcionamiento inesperado de los equipos. Si la alimentación del S7-1200 y de los equipos conectados no se desconecta por completo antes del montaje o desmontaje, pueden producirse la muerte, lesiones corporales graves y/o daños debidos a choques eléctricos o al funcionamiento inesperado de los equipos.

Respete siempre las medidas de seguridad necesarias y asegúrese que la alimentación eléctrica del S7-1200 está desconectada antes de montar o desmontar el S7-1200 o los equipos conectados.

Considere siempre los aspectos de seguridad al configurar la puesta a tierra y el cableado del sistema S7-1200. Los dispositivos de control electrónicos, tales como el S7-1200, pueden fallar y causar reacciones inesperadas de los equipos que se están controlando o vigilando. Por este motivo, se recomienda prever medidas de seguridad independientes del S7-1200 para evitar lesiones corporales y/o daños materiales.

 **ADVERTENCIA**

Los dispositivos de control pueden fallar y provocar condiciones no seguras, causando a su vez reacciones inesperadas de los equipos controlados. Las reacciones inesperadas podrían producir la muerte, lesiones corporales graves y/o daños materiales.

Prevea dispositivos de parada de emergencia, dispositivos de protección electromecánicos y otras medidas redundantes de seguridad que sean independientes del S7-1200.

Directrices de aislamiento galvánico

Los límites de la alimentación AC del S7-1200 y de las E/S a los circuitos AC se han diseñado y aprobado para proveer un aislamiento galvánico seguro entre las tensiones de línea AC y los circuitos de baja tensión. Estos límites incluyen un aislamiento doble o reforzado, o bien un aislamiento básico más uno adicional, según las distintas normas. Los componentes que cruzan estos límites, tales como optoacopladores, condensadores, transformadores y relés se han aprobado, ya que proveen un aislamiento galvánico seguro. Los límites de aislamiento que cumplen estos requisitos se identifican en las hojas de datos de los productos S7-1200, indicando que tienen un aislamiento de 1500 V AC o superior. Esta indicación se basa en una prueba de fábrica rutinaria de $(2U_e + 1000 \text{ V AC})$ o equivalente, según los métodos aprobados. Los límites de aislamiento galvánico seguro del S7-1200 se han comprobado hasta 4242 V DC.

La salida de la fuente de alimentación de sensores, los circuitos de comunicación y los circuitos lógicos internos de un S7-1200 con fuente de alimentación AC incluida tienen una fuente SELV (pequeña tensión de seguridad) conforme a EN 61131-2.

Para conservar el carácter seguro de los circuitos de baja tensión del S7-1200, las conexiones externas a puertos de comunicación, circuitos analógicos y todas las fuentes de alimentación nominales de 24 V y circuitos E/S deben ser alimentados por fuentes aprobadas que cumplan los requisitos de SELV, PELV, clase 2, tensión limitada o intensidad limitada, según distintas normas.

 **ADVERTENCIA**

La utilización de fuentes de alimentación no aisladas o con aislamiento simple para abastecer los circuitos de baja tensión desde un conductor AC pueden causar tensiones peligrosas en circuitos considerados no peligrosos (seguros al tacto), tales como los circuitos de comunicación y el cableado de sensores de baja tensión.

Las altas tensiones inesperadas podrían causar choques eléctricos que pueden producir la muerte, lesiones corporales graves y/o daños materiales.

Utilice sólo convertidores de alta a baja tensión aprobados como fuentes de circuitos de tensión limitada seguros al tacto.

Directrices de puesta a tierra del S7-1200

La mejor forma de poner a tierra la aplicación es garantizar que todos los conductores neutros y de masa del S7-1200 y de los equipos conectados se pongan a tierra en un mismo punto. Este punto debería conectarse directamente a la toma de tierra del sistema.

Todos los cables de puesta a tierra deberían tener la menor longitud posible y una sección grande, p. ej. 2 mm² (14 AWG).

Al definir físicamente las tierras es necesario considerar los requisitos de puesta a tierra de protección y el funcionamiento correcto de los dispositivos protectores.

Directrices de cableado del S7-1200

Al diseñar el cableado del S7-1200, prevea un interruptor unipolar para cortar simultáneamente la alimentación de la CPU S7-1200, de todos los circuitos de entrada y de todos los circuitos de salida. Prevea dispositivos de protección contra sobreintensidad (p. ej. fusibles o cortacircuitos) para limitar las corrientes de fallo en el cableado de alimentación. Para mayor protección es posible disponer un fusible u otro limitador de sobreintensidad en todos los circuitos de salida.

Utilice dispositivos de supresión de sobretensiones apropiados en el cableado sujeto a perturbaciones por descargas atmosféricas.

Evite colocar las líneas de señales de baja tensión y los cables de comunicación en una misma canalización junto con los cables AC y los cables DC de alta energía y conmutación rápida. El cableado deberá efectuarse por pares; con el cable de neutro o común combinado con el hilo caliente o de señal.

Utilice el cable más corto posible y vigile que tenga una sección suficiente para conducir la corriente necesaria. El conector de la CPU y el SM soporta cables con una sección de 2 mm² a 0,3 mm² (14 AWG a 22 AWG). El conector de la SB soporta cables con una sección de 1,3 mm² a 0,3 mm² (16 AWG a 22 AWG). Utilice cables apantallados para obtener una protección óptima contra interferencias. Por lo general, los mejores resultados se obtienen poniendo a tierra la pantalla del S7-1200.

Al cablear circuitos de entrada alimentados por una fuente externa, prevea dispositivos protectores contra sobrecorriente en estos circuitos. La protección externa no se requiere en los circuitos alimentados por la alimentación de sensores de 24 V DC del S7-1200, puesto que la alimentación de sensores ya está protegida contra sobrecorriente.

Todos los módulos S7-1200 incorporan conectores extraíbles para el cableado de usuario. Para evitar conexiones flojas, asegúrese que el conector está encajado correctamente y que el cable está insertado de forma segura en el conector. No apriete excesivamente los tornillos para impedir que se deteriore el conector. El par máximo de apriete de los tornillos del conector de la CPU y el SM es de 0,56 Nm (5 pulgadas-libra). El par máximo de apriete de los tornillos del conector de la SB es de 0,33 Nm (3 pulgadas-libra).

Para impedir flujos de corriente indeseados en la instalación, el S7-1200 provee límites de aislamiento galvánico en ciertos puntos. Tenga en cuenta estos límites de aislamiento al planificar el cableado del sistema. En los datos técnicos encontrará más información acerca de la ubicación de los puntos de aislamiento galvánico y la capacidad que ofrecen. Los aislamientos con valores nominales inferiores a 1500 V AC no deben tomarse para definir barreras de seguridad.

Directrices para las cargas de lámpara

Las cargas de lámpara pueden averiar los contactos de relé, debido a la elevada sobrecorriente momentánea de conexión. Esta sobrecorriente momentánea es nominalmente 10 a 15 veces superior a la corriente en régimen permanente de una lámpara de tungsteno. Se recomienda intercalar un relé sustituible o un limitador de sobretensión para las cargas de lámparas que deben conmutarse con frecuencia durante la vida útil de la aplicación.

Directrices relativas a las cargas inductivas

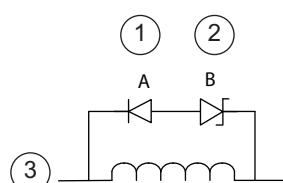
Se recomienda equipar las cargas inductivas con circuitos de supresión que limiten el incremento de tensión producido al desactivarse las salidas. Los circuitos de supresión protegen las salidas contra fallos prematuros debidos a altas tensiones al desconectar las cargas inductivas. Además, estos circuitos limitan las interferencias generadas al conmutar las cargas inductivas. La manera más efectiva de reducir las interferencias es disponer un circuito de supresión externo paralelo eléctricamente a la carga y ubicado físicamente cerca de la carga.

Las salidas DC del S7-1200 incluyen circuitos de supresión internos adecuados para las cargas inductivas en la mayoría de las aplicaciones. Puesto que los contactos de salida de relé del S7-1200 pueden utilizarse para conmutar cargas tanto DC como AC, no proporcionan protección interna.

Nota

La eficacia de un determinado circuito de supresión depende de la aplicación. Por tanto, debe verificarse para cada caso en particular. Asegúrese que los todos componentes utilizados en el circuito de supresión se adecúan para la aplicación en cuestión.

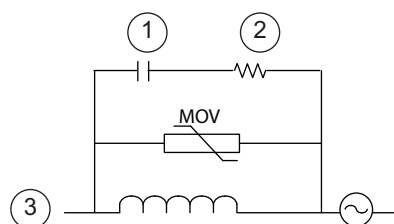
Circuito de supresión típico para salidas de relé o DC que conmutan cargas inductivas DC



- ① Diodo 1N4001 o equivalente
- ② Zener de 8,2 V (salidas DC)
Zener de 36 V (salidas de relé)
- ③ Salida

En la mayoría de las aplicaciones es suficiente prever adicionalmente un diodo (A) paralelo a una carga inductiva DC. No obstante, si la aplicación requiere tiempos de desconexión más rápidos, se recomienda utilizar un diodo Zener (B). Vigile que el diodo Zener tenga suficiente capacidad para la cantidad de corriente en el circuito de salida.

Circuito de supresión típico para salidas de relé que conmutan cargas inductivas AC



- ① 0,1 μ F
- ② 100 a 120 Ω
- ③ Punto de salida

Cuando se utiliza una salida de relé para conmutar cargas de 115 V/230 V AC, hay que instalar, paralelo a la carga AC, un circuito adecuado de varistores de óxido metálico (MOV) con resistores/condensadores. Vigile que la tensión de trabajo del varistor MOV sea como mínimo un 20% superior a la tensión de línea nominal.

Principios básicos del PLC

4.1 Ejecución del programa de usuario

La CPU soporta los siguientes tipos de bloques lógicos que permiten estructurar eficientemente el programa de usuario:

- Los bloques de organización (OBs) definen la estructura del programa. Algunos OBs tienen reacciones y eventos de arranque predefinidos. No obstante, también es posible crear OBs con eventos de arranque personalizados. Los rangos de números de OB válidos se indican en Prioridades de ejecución de eventos y cola de espera (Página 76).
- Las funciones (FCs) y los bloques de función (FBs) contienen el código de programa correspondiente a tareas específicas o combinaciones de parámetros. Cada FC o FB provee parámetros de entrada y salida para compartir datos con el bloque invocante. Un FB utiliza también un bloque de datos asociado (denominado DB instancia) para conservar el estado de valores durante la ejecución que pueden utilizar otros bloques del programa. Los números válidos para FC y FB van de 1 a 65535.
- Los bloques de datos (DBs) almacenan datos que pueden ser utilizados por los bloques del programa. Los números válidos para DB van de 1 a 65535.

La ejecución del programa de usuario comienza con uno o varios bloques de organización (OB) de arranque que se ejecutan una vez al cambiar a estado operativo RUN, seguidos de uno o varios OB de ciclo que se ejecutan cíclicamente. También es posible asociar un OB a un evento de alarma que puede ser un evento estándar o de error y que se ejecuta cada vez que ocurre el evento en cuestión.

Una función (FC) o un bloque de función (FB) es un bloque de código del programa que puede llamarse desde un OB, o bien desde otra FC u otro FB. Son posibles las profundidades de anidamiento siguientes:

- 16 desde OB de ciclo o de arranque
- 4 desde OB de alarma de retardo, de alarma cíclica, de alarma horaria, de alarma de proceso, de alarma de error de tiempo o de alarma de error de diagnóstico

Las FCs no están asociadas a ningún bloque de datos (DB) en particular, mientras que los FBs están vinculados directamente a un DB que utilizan para transferir parámetros, así como para almacenar valores intermedios y resultados.


El tamaño del programa de usuario, los datos y la configuración está limitado por la memoria de carga disponible y la memoria de trabajo de la CPU. No hay un límite determinado para el número de cada bloque OB, FC, FB y DB individual. No obstante, el número total de bloques se limita a 1024.

En cada ciclo se escribe en las salidas, se leen las entradas, se ejecutan las instrucciones del programa de usuario y se realiza el procesamiento en segundo plano. En inglés, el ciclo también se llama "scan cycle" o "scan".

Los módulos (SM, SB, BB, CB, CM o CP) solo se detectan e incorporan en el proceso de arranque.

4.1 Ejecución del programa de usuario

- No está permitido insertar o extraer un módulo del rack central con la alimentación conectada (en caliente). No inserte ni extraiga nunca un módulo del rack central cuando la CPU tenga tensión.

 ADVERTENCIA
La inserción o extracción de un módulo (SM, SB, BB, CD, CM o CP) del rack central cuando la CPU tiene tensión podría causar un comportamiento impredecible que podría provocar daños en el equipo y/o lesiones personales.
Asegúrese siempre de que se ha desconectado la alimentación de la CPU y del rack central antes de insertar o extraer un módulo del rack central.

- Una SIMATIC Memory Card sí puede insertarse o extraerse mientras la CPU tiene tensión. Sin embargo, la inserción o extracción de una Memory Card cuando la CPU está en RUN provoca el paso a STOP de la CPU.

PRECAUCIÓN
La inserción o extracción de una Memory Card mientras la CPU está en RUN provoca el paso a STOP de la CPU, lo que podría causar daños en el equipo o en el proceso que se está controlando.
Siempre que se inserta o extrae una Memory Card, la CPU pasa inmediatamente al estado operativo STOP. Antes de insertar o extraer una Memory Card, asegúrese siempre de que la CPU no está controlando activamente una máquina o un proceso. Prevea siempre un circuito de parada de emergencia para la aplicación o el proceso.

- Si se inserta o extrae un módulo en un rack de E/S distribuidas (PROFINET o PROFIBUS) cuando la CPU está en RUN, la CPU genera una entrada en el búfer de diagnóstico y permanece en RUN.

De forma predeterminada, todas las E/S digitales y analógicas locales se actualizan de forma sincrónica con el ciclo, utilizando un área de memoria interna denominada memoria imagen de proceso. La memoria imagen de proceso contiene una instantánea de las entradas y salidas físicas (es decir, las E/S físicas de la CPU, de la Signal Board y de los módulos de señales).

La CPU ejecuta las siguientes tareas:

- La CPU escribe las salidas desde la memoria imagen de proceso de las salidas en las salidas físicas.
- La CPU lee las entradas físicas inmediatamente antes de ejecutar el programa de usuario y almacena los valores de entrada en la memoria imagen de proceso de las entradas. Así se garantiza que estos valores sean coherentes durante la ejecución de las instrucciones programadas.
- La CPU ejecuta la lógica de las instrucciones programadas y actualiza los valores de salida en la memoria imagen de proceso de las salidas, en vez de escribirlos en las salidas físicas reales.

Este proceso ofrece una lógica coherente al ejecutar las instrucciones programadas durante un ciclo determinado y previene la fluctuación de las salidas físicas cuyo estado puede cambiar varias veces en la memoria imagen de proceso de las salidas.

Se puede especificar si las E/S digitales y analógicas deben actualizarse y guardarse automáticamente en la memoria imagen de proceso. Si un módulo se inserta en la vista de dispositivos, sus datos se encontrarán en la memoria imagen de proceso de la CPU (ajuste predeterminado). La CPU procesa el intercambio de datos entre el módulo y el área de la memoria imagen de proceso automáticamente durante la actualización de ésta. Para excluir E/S digitales o analógicas de la actualización automática de la memoria imagen de proceso, seleccione el dispositivo en cuestión en la "Configuración de dispositivos", abra la ficha "Propiedades", expanda en caso necesario para localizar las E/S deseadas y seleccione luego "Direcciones I/O/identificador HW". A continuación, cambie la entrada en "Memoria imagen de proceso:" de "IP cíclica" a "---". Para volver a incluir las E/S en la actualización automática de la memoria imagen de proceso, cambie de nuevo esta selección a "IP cíclica".

Es posible leer inmediatamente los valores de las entradas físicas y escribir inmediatamente los valores de las salidas físicas cuando se ejecuta una instrucción. Una lectura inmediata accede al estado actual de la entrada física y no actualiza la memoria imagen de proceso de las entradas, independientemente de si se ha configurado que la entrada se almacene en la memoria imagen de proceso. Una lectura inmediata en una salida física actualiza tanto la memoria imagen de proceso de las salidas (si se ha configurado que la salida se almacene en la memoria imagen de proceso) y la salida física. Añada el sufijo ":P" a la dirección E/S si desea que el programa acceda inmediatamente a los datos de E/S directamente desde la E/S física, en vez de utilizar la memoria imagen de proceso.

La CPU soporta E/S descentralizadas para redes PROFINET y PROFIBUS (Página 441).

4.1.1 Estados operativos de la CPU

La CPU tiene tres estados operativos, a saber: STOP, ARRANQUE y RUN. Los LEDs de estado en el frente de la CPU indican el estado operativo actual.

- En modo STOP, la CPU no ejecuta el programa. Se puede descargar un proyecto.
- En estado operativo ARRANQUE, los OBs de arranque (si existen) se ejecutan una vez. Los eventos de alarma no se procesan durante el modo de arranque.
- En modo RUN, los OBs cíclicos se ejecutan repetidamente. Los eventos de interrupción pueden ocurrir y procesarse en cualquier punto del modo RUN. Algunas partes de un proyecto se pueden descargar en modo RUN (Página 728).

La CPU soporta el arranque en caliente para pasar al estado operativo RUN. El arranque en caliente no incluye la inicialización de la memoria. Los datos de sistema no remanentes y los datos de usuario se inicializan en un arranque en caliente. Se conservan los datos de usuario remanentes.

El borrado total borra toda la memoria de trabajo, así como las áreas de memoria remanentes y no remanentes. Además, copia la memoria de carga en la memoria de trabajo. El borrado total no borra el búfer de diagnóstico ni tampoco los valores almacenados permanentemente de la dirección IP.

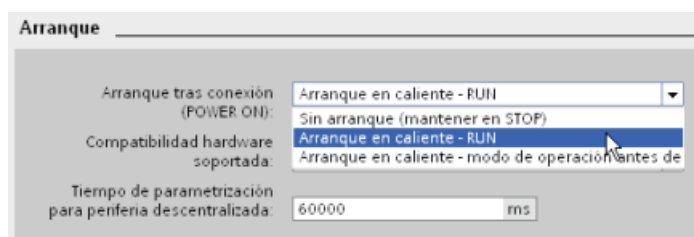
Nota

Si se descargan uno o más DB desde STEP 7 V11 a una CPU S7-1200 V2, los valores remanentes y no remanentes de dichos DB se ajustan a sus valores iniciales. En la siguiente transición a RUN, se realizará un re arranque en caliente, ajustando todos los datos no remanentes a sus valores iniciales y todos los datos remanentes a sus valores retenidos.

Si se descargan elementos de un proyecto (como una configuración de dispositivo, bloques lógicos o DBs) desde STEP 7 V10.5 a una CPU S7-1200 o desde STEP 7 V11 a una CPU S7-1200 V1 (o una CPU V2 configurada como V1), en la siguiente transición a RUN se inicializarán **todos** los DBs del proyecto a sus valores iniciales.

Se puede configurar el ajuste "arranque tras POWER ON" de la CPU. Este ajuste se encuentra en la "Configuración de dispositivos" de la CPU en "Arranque". Cuando se aplica tensión, la CPU ejecuta una secuencia de tests de diagnóstico de arranque e inicialización del sistema. Durante la inicialización del sistema, la CPU elimina toda el área de marcas no remanente e inicializa todos los contenidos de DB no remanentes a los valores iniciales de la memoria de carga. La CPU retiene el área de marcas remanente y los contenidos de DB remanentes y, a continuación, entra en el modo operativo correspondiente. Determinados errores impiden que la CPU pase al estado operativo RUN. La CPU admite las siguientes opciones de configuración:

- Sin rearranque (permanecer en modo STOP)
- Arranque en caliente - RUN
- Arranque en caliente - modo previo a POWER OFF



PRECAUCIÓN

La CPU puede pasar a STOP debido a errores reparables, como el fallo de un módulo de señales sustituible, o a errores temporales, como perturbaciones en el cable de alimentación o eventos de arranque imprevisibles.

Si la CPU se ha configurado para un "Arranque en caliente - modo de operación previo a la desconexión", no regresará al modo RUN al reparar o eliminar el error hasta que reciba un nuevo comando de STEP 7 para pasar a RUN. Hasta que no llegue el nuevo comando se mantendrá el modo STOP como modo previo a la desconexión.

Las CPUs destinadas a funcionar con independencia de una conexión STEP 7 deben configurarse como norma general para "Arranque en caliente - RUN" para que la CPU puede regresar al modo RUN con un ciclo de desconexión y reconexión tras eliminar las condiciones de error.

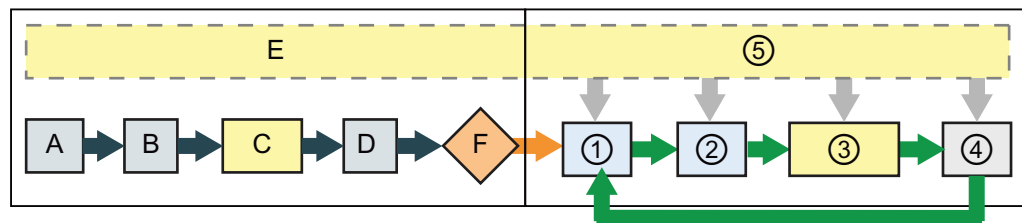
4.1 Ejecución del programa de usuario

El estado operativo actual se puede cambiar con los comandos "STOP" o "RUN" de las herramientas online del software de programación. También se puede insertar una instrucción STP en el programa para cambiar la CPU a STOP. Esto permite detener la ejecución del programa en función de la lógica.

- En estado operativo STOP, la CPU procesa las peticiones de comunicación (según sea necesario) y realiza el autodiagnóstico. La CPU no ejecuta el programa de usuario y la memoria imagen de proceso no se actualiza automáticamente.

El proyecto sólo se puede cargar en la CPU si está se encuentra en estado operativo STOP.

- En estado operativo ARRANQUE y RUN, la CPU ejecuta las tareas que muestra la figura siguiente.



ARRANQUE

- A Borra el área de memoria I (imagen)
- B Inicializa las salidas con el último valor o el valor sustitutivo
- C Ejecuta los OBs de arranque
- D Copia el estado de las entradas físicas en la memoria I
- E Almacena los eventos de alarma de la cola de espera que deben procesarse una vez que se haya pasado al estado operativo RUN
- F Habilita la escritura de la memoria Q en las salidas físicas

RUN

- ① Escribe la memoria Q en las salidas físicas
- ② Copia el estado de las entradas físicas en la memoria I
- ③ Ejecuta los OBs de ciclo
- ④ Realiza autodiagnóstico
- ⑤ Procesa alarmas y comunicaciones en cualquier parte del ciclo

Procesamiento del ARRANQUE

Cada vez que el modo operativo cambia de STOP a RUN, la CPU borra las entradas de la memoria imagen de proceso, inicializa las salidas de la memoria imagen de proceso y procesa los OBs de arranque. En los accesos de lectura a las entradas de la memoria imagen de proceso realizados por instrucciones de los OBs de arranque se lee cero, en vez del valor actual de la entrada física. Por tanto, para leer el estado actual de una entrada física durante el estado operativo ARRANQUE, es preciso realizar una lectura inmediata. Luego se ejecutan los OBs de arranque, así como los FBs y FCs asociados. Si existe más de un OB de arranque, cada uno de ellos se ejecuta en el orden correspondiente al número de OB, comenzando con el número de OB más bajo.

Todo OB de arranque incluye información de arranque que ayuda a determinar la validez de los datos remanentes y el reloj en tiempo real. Es posible programar instrucciones dentro de los OBs de arranque para examinar estos valores de arranque y realizar las acciones apropiadas. Los OBs de arranque soportan las siguientes ubicaciones de arranque:

Tabla 4- 1 Ubicaciones de arranque soportadas por el OB de arranque

Entrada	Tipo de datos	Descripción
LostRetentive	Bool	Este bit es verdadero (TRUE) si se han perdido las áreas de almacenamiento de datos remanentes
LostRTC	Bool	Este bit es verdadero (TRUE) si se ha perdido el reloj en tiempo real

La CPU también ejecuta las siguientes tareas durante el procesamiento del arranque.

- Las alarmas se ponen en cola de espera, pero no se procesan durante la fase de arranque
- El tiempo de ciclo no se vigila durante la fase de arranque
- La configuración de HSC (contadores rápidos), PWM (modulación del ancho de pulso) y módulos PtP (comunicación punto a punto) se puede modificar durante el arranque
- Los HSC, la PWM y los módulos de comunicación punto a punto sólo funcionan en estado operativo RUN

Una vez finalizada la ejecución de los OBs de arranque, la CPU pasa a estado operativo RUN y procesa las tareas de control en un ciclo continuo.

Consulte también

Instrucción "Parar ciclo del PLC" (Página 244)

Panel de control de la CPU online (Página 715)

4.1.2 Procesamiento del ciclo en estado operativo RUN

En cada ciclo, la CPU escribe en las salidas, lee las entradas, ejecuta el programa de usuario, actualiza los módulos de comunicación y reacciona a los eventos de alarma de usuario y peticiones de comunicación. Las peticiones de comunicación se procesan periódicamente durante el ciclo.

Estas acciones (excepto los eventos de alarma de usuario) se procesan con regularidad y en orden secuencial. Los eventos de alarma habilitados se procesan según su prioridad en el orden en que aparecen.

4.1 Ejecución del programa de usuario

El sistema garantiza que el ciclo se procese dentro de un periodo denominado tiempo de ciclo máximo. De lo contrario, se generará un evento de error de tiempo.

- Todo ciclo comienza con la consulta de los valores actuales de las salidas digitales y analógicas de la memoria imagen de proceso. Estos valores se escriben luego en las salidas físicas de la CPU, la SB y los módulos SM configurados para la actualización automática de E/S (configuración predeterminada). Cuando una instrucción accede a una salida física, se actualizan tanto la memoria imagen de proceso de las salidas como la salida física.
- El ciclo continúa con la lectura de los valores actuales de las entradas digitales y analógicas de la CPU, la SB y los SMs configurados para la actualización automática de E/S (configuración predeterminada). Estos valores se escriben luego en la memoria imagen de proceso. Cuando una instrucción accede a una entrada física, se modifica el valor de ésta, pero no se actualiza la memoria imagen de proceso de las entradas.
- Tras leer las entradas, el programa de usuario se ejecuta desde la primera hasta la última instrucción. Esto incluye todos los OBs de ciclo, así como sus FCs y FBs asociados. Los OBs de ciclo se ejecutan en el orden correspondiente al número de OB, comenzando con el número de OB más bajo.

Las comunicaciones se procesan periódicamente durante todo el ciclo, siendo posible que se interrumpa la ejecución del programa de usuario.

El autodiagnóstico incluye comprobaciones periódicas del sistema y de estado de los módulos de E/S.

Las alarmas pueden ocurrir en cualquier parte del ciclo y son controladas por eventos. Cuando ocurre un evento, la CPU interrumpe el ciclo y llama el OB configurado para procesar ese evento. Una vez que el OB haya finalizado el procesamiento del evento, la CPU reanuda la ejecución del programa de usuario en el punto de interrupción.

4.1.3 Bloques de organización (OBs)

Los OB controlan la ejecución del programa de usuario. Todo OB debe tener un número de OB unívoco. Los números inferiores a 200 están reservados para los números de OB predeterminados. La numeración de los demás OB debe comenzar a partir de 200.

Determinados eventos de la CPU disparan la ejecución de un bloque de organización. Un OB no puede llamar a otro. Tampoco es posible llamar un OB desde una FC o un FB. Sólo un evento de arranque, p. ej. una alarma de diagnóstico o un intervalo, puede iniciar la ejecución de un OB. La CPU procesa los OB según su clase de prioridad. Los OB de mayor prioridad se ejecutan antes que los de menor prioridad. La clase de prioridad más baja es 1 (para el ciclo del programa principal) y la más alta es 26 (para las alarmas de error de tiempo).

Los OB controlan los siguientes procesos:

- Los OB de ciclo se ejecutan cíclicamente cuando la CPU se encuentra en estado operativo RUN. El bloque principal del programa es un OB de ciclo. Éste contiene las instrucciones que controlan el programa y permite llamar otros bloques de usuario. Es posible utilizar varios OB de ciclo. Éstos se ejecutan en orden numérico. El OB 1 es el bloque predeterminado. Los demás OB de ciclo deben identificarse como OB 200 o superior.
- Los OB de arranque se ejecutan una vez cuando el estado operativo de la CPU cambia de STOP a RUN, al arrancar a estado operativo RUN y en una transición ordenada de STOP a RUN. Una vez finalizado, se comienza a ejecutar el OB de ciclo. Es posible utilizar varios OB de arranque. El OB 100 es el bloque predeterminado. El número de los demás OB debe ser 200 o superior.
- Los OB de alarma cíclica se ejecutan en intervalos periódicos. Los OB de alarma cíclica interrumpen la ejecución cíclica del programa en intervalos definidos, p. ej. cada 2 segundos. Es posible configurar como máximo un total de 4 eventos de retardo y cíclicos en cualquier momento. Por cada evento de retardo o cíclico configurado se permite un OB. El número del OB debe ser 200 o superior.
- Los OB de alarma de proceso se ejecutan cuando ocurre el evento de hardware correspondiente, incluyendo flancos ascendentes y descendentes en las entradas digitales integradas y eventos de contadores rápidos (HSC). Los OB de alarma de proceso interrumpen la ejecución cíclica del programa como reacción a una señal de un evento de hardware. Los eventos se definen en las propiedades de la configuración hardware. Por cada evento de hardware configurado se permite un OB. El número del OB debe ser 200 o superior.

4.1 Ejecución del programa de usuario

- Un OB de alarma de error de tiempo se ejecuta cuando se excede el tiempo de ciclo máximo o se produce un evento de error de tiempo. El OB para procesar la alarma de error de tiempo es el OB 80. Si se dispara, se ejecuta, interrumpiendo la ejecución cíclica normal del programa o cualquier otro OB de evento. A continuación se describen los eventos que disparan la alarma de error de tiempo y la reacción de la CPU a dichos eventos:
 - Rebase del tiempo de ciclo máximo: el tiempo de ciclo máximo se configura en las propiedades de la CPU. Si el OB 80 no existe, la reacción de la CPU al excederse el tiempo máximo es cambiar a STOP.
 - Errores de tiempo: Si el OB 80 no existe, la reacción de la CPU es permanecer en RUN. Los errores de tiempo se producen cuando un evento de hora del día falta o se repite, una cola se desborda, o bien cuando se inicia un OB de evento (evento de retardo, evento de hora del día o una alarma cíclica) antes de que la CPU finalice la ejecución del primero.

La aparición de cualquiera de esos eventos genera una entrada en el búfer de diagnóstico que describe el evento. La entrada del búfer de diagnóstico se genera independientemente de la existencia del OB 80.

- Los OB de alarma de diagnóstico se ejecutan cuando se detecta y notifica un error de diagnóstico. Los OB de alarma de diagnóstico interrumpen la ejecución cíclica del programa cuando el módulo apto para diagnóstico detecta un error (si se ha habilitado la alarma de diagnóstico para ese módulo). El OB 82 es el único número de OB soportado para el evento de error de diagnóstico. Es posible incluir una instrucción STP (poner CPU a STOP) en el OB 82 para que la CPU pase al estado operativo STOP en cuanto reciba este tipo de error. Si no hay ningún OB de diagnóstico en el programa, la CPU ignora el error (permanece en RUN).

4.1.4 Prioridades y colas de espera para la ejecución de eventos

El procesamiento de la CPU es controlado por eventos. Un evento dispara la ejecución de un OB de alarma. Se puede definir el OB de alarma para un evento al crear el bloque, al configurar dispositivos o con una instrucción ATTACH o DETACH. Algunos eventos ocurren con regularidad, tales como los eventos de ciclo o cíclicos. Otros eventos ocurren una sola vez, tales como el evento de arranque y los eventos de alarma de retardo. Algunos eventos ocurren cuando se produce un cambio disparado por hardware, p. ej. un flanco en una entrada o un evento de contador rápido. Asimismo, hay eventos p. ej. de error de diagnóstico o de error de tiempo que ocurren solamente cuando se produce un error. Las prioridades de eventos y las colas de espera sirven para determinar el orden de procesamiento de los OB de alarma.

El evento de ciclo ocurre una vez por ciclo del programa. Durante el ciclo del programa, la CPU escribe en las salidas, lee las entradas y ejecuta los OB de ciclo. El evento de ciclo es necesario y siempre está habilitado. Es posible no tener OB de ciclo, o bien tener varios OB de ciclo seleccionados para el evento de ciclo. Una vez disparado el evento de ciclo, se ejecuta el OB de ciclo con el número más bajo (normalmente el OB 1). Los demás OB de ciclo se ejecutan secuencialmente (en orden numérico) dentro del ciclo de programa.

Los eventos de alarma cíclica permiten configurar la ejecución de un OB de alarma en un tiempo de ciclo configurado. El tiempo de ciclo inicial se configura al crear el OB y seleccionar que sea de alarma cíclica. Un evento cíclico interrumpe el ciclo del programa y ejecuta el OB de alarma cíclica (el evento cíclico pertenece a una clase de mayor prioridad que el evento de ciclo del programa).

Un solo OB de alarma cíclica puede asignarse a un evento cíclico.

A cada evento cíclico se le puede asignar un desfase. De este modo, la ejecución de alarmas cíclicas con el mismo tiempo de ciclo puede ejecutarse con un offset entre ellas equivalente a la cantidad de desfase. El desfase predeterminado es 0. Para modificar el desfase inicial, o para cambiar el tiempo de ciclo inicial de un evento cíclico, haga clic con el botón derecho del ratón en el OB de alarma cíclica del árbol de proyectos, haga clic en "Propiedades" y, a continuación, haga clic en "Alarma cíclica" e introduzca los valores iniciales nuevos. También se puede consultar y modificar el tiempo de ciclo y el desfase desde el programa con las instrucciones de consulta de alarma cíclica (QRY_CINT) y ajuste de alarma cíclica (SET_CINT). Los valores de tiempo de ciclo y desfase definidos en la instrucción SET_CINT no se conservan tras desconectar y conectar la alimentación, o tras pasar a estado operativo STOP; los valores de tiempo de ciclo y desfase vuelven a los valores iniciales tras desconectar y conectar la alimentación, o tras pasar a estado operativo STOP. La CPU soporta un total de cuatro eventos de alarmas cíclicas y de retardo.

El evento de arranque ocurre una vez al producirse un cambio de STOP a RUN y lanza la ejecución de los OB de arranque. Es posible seleccionar varios OB para el evento de arranque. Los OB de arranque se ejecutan en orden numérico.

Los eventos de alarma de retardo permiten configurar la ejecución de un OB de alarma transcurrido un tiempo de retardo definido. El tiempo de retardo se especifica con la instrucción SRT_DINT. Los eventos de alarma de retardo interrumpen el ciclo del programa, con el fin de ejecutar el OB de alarma de retardo. Un solo OB de alarma de retardo puede asignarse a un evento de retardo. La CPU soporta cuatro eventos de retardo.

Los eventos de alarma de proceso son disparados por un cambio en el hardware, p. ej. un flanco ascendente o descendente en una entrada, o bien un evento de contador rápido (HSC). Sólo un OB de alarma puede estar seleccionado para cada evento de alarma de proceso. Los eventos de alarma de proceso se habilitan en la "Configuración de dispositivos". Los OB se definen para el evento en la "Configuración de dispositivos" o con una instrucción ATTACH en el programa de usuario. La CPU soporta varios eventos de alarma de proceso. Los eventos exactos dependen del modelo de CPU y del número de entradas.

Los eventos de error de tiempo y diagnóstico son disparados cuando la CPU detecta un error. Estos eventos pertenecen a una clase de mayor prioridad que los demás eventos de alarma y pueden interrumpir la ejecución de los eventos de alarma de retardo, alarma cíclica y alarma de proceso. Es posible definir un OB de alarma para cada uno de los eventos de error de tiempo y diagnóstico.

Prioridades y colas de espera para la ejecución de eventos

El número de eventos pendientes (en cola de espera) de una sola fuente se limita utilizando una cola diferente para cada tipo de evento. Al alcanzar el límite de eventos pendientes de un determinado tipo, se pierde el evento siguiente. Para más información sobre el desbordamiento de colas de espera, consulte el apartado "Eventos de error de tiempo".

4.1 Ejecución del programa de usuario

Todo evento de la CPU tiene asignada una prioridad. No es posible modificar la prioridad de un OB. Generalmente, los eventos se procesan según su prioridad (primero los de mayor prioridad). Los eventos de igual prioridad se procesan según su orden de aparición.

Tabla 4-2 Eventos de OB

Evento	Número de OB	Cantidad permitida	Evento de arranque	Prioridad de OB
Ciclo del programa	OB 1, de OB 200 a OB 65535	1 evento de ciclo de programa Se admiten varios OB	<ul style="list-style-type: none"> OB de arranque finaliza Último OB de ciclo de programa finaliza 	1
Arranque	OB 100, de OB 200 a OB 65535	1 evento de arranque ^{1, 2} Se admiten varios OB	Transición de STOP a RUN	1
Tiempo	De OB 200 a OB 65535	Hasta 4 eventos de tiempo ³ 1 OB por evento	Programación de evento de OB de retardo	3
			Programación de evento de OB de ciclo	7
Proceso	De OB 200 a OB 65535	Hasta 50 eventos de proceso ⁴ 1 OB por evento	Flancos: <ul style="list-style-type: none"> Eventos de flanco ascendente: 16 máx. Eventos de flanco descendente: 16 máx. 	5
			Para HSC: <ul style="list-style-type: none"> CV=PV: 6 máx. Cambio de sentido: 6 máx. Inicialización externa: 6 máx. 	6
Error de diagnóstico	OB 82	1 evento (sólo si se ha cargado el OB 82)	El módulo transmite un error	9
Error de tiempo	OB 80	1 evento (sólo si se ha cargado el OB 80) ⁵	<ul style="list-style-type: none"> Tiempo de ciclo máximo excedido Una segunda alarma de tiempo (cíclica o de retardo) se ha iniciado antes de que la CPU haya terminado de ejecutar la primera alarma 	26

- Los eventos de arranque y de ciclo de programa no ocurren nunca simultáneamente, ya que el evento de arranque debe haber finalizado antes de poder iniciar el evento de ciclo de programa (controlado por el sistema operativo).
- Sólo el evento de error de diagnóstico (OB 82) puede interrumpir el evento de arranque. Los demás eventos se ponen en cola para procesarse una vez que haya finalizado el evento de arranque.
- La CPU proporciona un total de 4 eventos de tiempo repartidos entre los OB de retardo y los OB cíclicos. El número de OB de retardo y cíclicos del programa de usuario no puede ser mayor de 4.
- Se puede contar con más de 50 eventos de proceso si se utilizan las instrucciones DETACH y ATTACH.
- Es posible configurar la CPU de modo que permanezca en RUN si se excede el tiempo de ciclo máximo o bien utilizar la instrucción RE_TRIGR para resetear el tiempo de ciclo. Sin embargo, la CPU pasa a STOP la segunda vez que se excede el tiempo de ciclo máximo en un mismo ciclo.

Tras iniciarse la ejecución de un OB con una prioridad de 2 a 25, su procesamiento no se puede interrumpir al aparecer otro evento, excepto el OB 80 (evento de error de tiempo, que tiene una prioridad de 26). Todos los demás eventos se ponen en cola de espera para su procesamiento posterior, permitiendo que finalice el OB actual.

Latencia de alarmas

La latencia de los eventos de alarma (es decir, el tiempo que transcurre desde que la CPU notifica que ha ocurrido un evento hasta que comienza la ejecución de la primera instrucción en el OB que procesa este evento) es de aproximadamente 175 µs, siempre que un OB de ciclo de programa sea el único subprograma activo que procese el evento de alarma en el momento de su aparición.

Eventos de error de tiempo

La aparición de cualquiera de las condiciones de error de tiempo diferentes causa un evento de error de tiempo. Se soportan los siguientes errores de tiempo:

- Rebase del tiempo de ciclo máximo
- No se puede iniciar el OB solicitado
- Desbordamiento de la cola de espera

El error de rebase del tiempo de ciclo máximo ocurre si el ciclo de programa no finaliza dentro del tiempo de ciclo máximo especificado. Encontrará más información acerca de la condición de tiempo de ciclo máximo y sobre cómo configurar el tiempo de ciclo máximo e inicializar la vigilancia del tiempo de ciclo en el apartado "Vigilancia del tiempo de ciclo (Página 81)" del manual de sistema S7-1200.

El OB solicitado no se puede iniciar si una alarma cíclica, una alarma de retardo o una alarma horaria solicita un OB, pero éste ya se está ejecutando.

La cola de espera se desborda si las alarmas ocurren más rápidamente de lo que pueden procesarse. El número de eventos pendientes (en cola de espera) se limita utilizando una cola diferente para cada tipo de evento. Si ocurre un evento estando llena la cola de espera correspondiente, se genera un evento de error de tiempo.

Todos los eventos de error de tiempo disparan la ejecución del OB 80 (si existe). Si el programa de usuario no incluye un OB 80, la configuración de dispositivo de la CPU determina la reacción de la CPU al error de tiempo:

- La configuración predeterminada para errores de tiempo, como el inicio de una segunda alarma cíclica antes de que la CPU haya terminado de ejecutar la primera, es que la CPU permanezca en RUN.
- La configuración predeterminada para el rebase del tiempo máximo prevé que la CPU cambie a STOP.

Es posible utilizar la instrucción RE_TRIGR para resetear el tiempo de ciclo máximo. Sin embargo, si el tiempo de ciclo máximo se rebasa dos veces en un mismo ciclo del programa sin que se inicialice el temporizador de vigilancia del ciclo, la CPU pasará a STOP, independientemente de si existe el OB 80. Consulte el apartado "Vigilancia del tiempo de ciclo" (Página 81) del manual de sistema S7-1200.

4.1 Ejecución del programa de usuario

El OB 80 incluye información de arranque que permite determinar qué evento y OB ha generado el error de tiempo. Es posible programar instrucciones dentro del OB 80 para examinar estos valores de arranque y realizar las acciones apropiadas.

Tabla 4- 3 Información de arranque para OB 80

Entrada	Tipo de datos	Descripción
fault_id	BYTE	16#01 - rebase del tiempo de ciclo máximo 16#02 - no se puede iniciar el OB solicitado 16#07 y 16#09 - desbordamiento de la cola de espera
csg_OBnr	OB_ANY	Número de OB que se estaba ejecutando cuando ocurrió el error
csg_prio	UINT	Prioridad del OB que ha causado el error

Cuando se crea un proyecto nuevo, no existe ningún OB 80 de error de tiempo. Si desea agregar un OB 80 de error de tiempo al proyecto, haga doble clic en "Agregar nuevo bloque" en "Bloques de programa" en el árbol del proyecto, seleccione luego "Bloque de organización" y después "OB de error de tiempo".

Eventos de error de diagnóstico

Los dispositivos analógicos (locales), PROFINET y PROFIBUS pueden detectar y notificar errores de diagnóstico. La aparición o eliminación de cualquiera de las diferentes condiciones de error de diagnóstico ocasiona un evento de error de diagnóstico. Se soportan los siguientes errores de diagnóstico:

- Falta alimentación externa
- Límite alto excedido
- Límite bajo excedido
- Rotura de hilo
- Cortocircuito

Los eventos de error de diagnóstico disparan la ejecución del OB 82 (si existe). Si el OB 82 no existe, la CPU ignora el error. Cuando se crea un proyecto nuevo, no existe ningún OB 82 de alarma de diagnóstico. Si desea agregar un OB 82 de alarma de diagnóstico al proyecto, haga doble clic en "Agregar nuevo bloque" en "Bloques de programa" en el árbol del proyecto, seleccione luego "Bloque de organización" y después "OB de alarma de diagnóstico".

Nota

Errores de diagnóstico para dispositivos analógicos locales multicanal (E/S, RTD y termopar)

El OB 82 de alarma de diagnóstico no puede notificar más de un error de diagnóstico de canal al mismo tiempo.

Si dos canales de un dispositivo multicanal tienen un error, el segundo error sólo dispara el OB 82 en las condiciones siguientes: el primer error de canal se borra, la ejecución del OB 82 disparado por el primer error ha finalizado y el segundo error persiste.

El OB 82 incluye información de arranque que ayuda a determinar si el evento se debe a la aparición o desaparición de un error, así como el dispositivo y canal que han notificado el error. Es posible programar instrucciones dentro del OB 82 para examinar estos valores de arranque y realizar las acciones apropiadas.

Tabla 4- 4 Información de arranque del OB 82

Entrada	Tipo de datos	Descripción
IOstate	WORD	Estado de E/S del dispositivo: <ul style="list-style-type: none"> • Bit 0 = 1 si la configuración es correcta, y = 0 si la configuración ya no es correcta. • El bit 4 es 1 si existe un error (p. ej. una rotura de hilo). (Bit 4 = 0 si no hay ningún error.) • Bit 5 = 1 si la configuración no es correcta, y = 0 si la configuración vuelve a ser correcta. • Bit 6 = 1 si se ha producido un error de acceso a E/S. Véase laddr para conocer el identificador de hardware de E/S con error de acceso. (Bit 6 = 0 si no hay ningún error.)
laddr	HW_ANY	ID de hardware del dispositivo o unidad funcional que ha notificado el error ¹
channel	UINT	Número de canal
multierror	BOOL	TRUE (verdadero) si ha ocurrido más de un error

¹ La entrada en KOP contiene el identificador de hardware del dispositivo o unidad funcional que ha devuelto el error. El identificador de hardware se asigna automáticamente cuando se insertan componentes en la vista de dispositivos o redes, y aparece en la ficha "Constantes" de "Variables PLC". También se asigna automáticamente un nombre al identificador de hardware. Estas entradas de la ficha "Constantes" de "Variables PLC" no se pueden modificar.

4.1.5 Vigilancia del tiempo de ciclo

El tiempo de ciclo es el tiempo que requiere el sistema operativo de la CPU para ejecutar la fase cíclica del estado operativo RUN. La CPU ofrece dos métodos para vigilar el tiempo de ciclo:

- Tiempo de ciclo máximo
- Tiempo de ciclo mínimo fijo

La vigilancia del tiempo de ciclo comienza una vez finalizado el evento de arranque. Esta función se configura en la "Configuración de dispositivos" de la CPU en "Tiempo de ciclo".

La CPU vigila siempre el ciclo y reacciona si se rebasa el tiempo de ciclo máximo. Si se rebasa el tiempo de ciclo máximo configurado, se generará un error que se procesa de dos maneras posibles:

- Si el programa de usuario no incluye ningún OB 80, la CPU genera un error y pasa a STOP. (Es posible modificar la configuración de la CPU para que ignore este error de tiempo y permanezca en RUN. La configuración predeterminada prevé que la CPU pase a STOP.)
- Si el programa de usuario incluye un OB 80, la CPU lo ejecuta.

4.1 Ejecución del programa de usuario

La instrucción RE_TRIGR (Volver a lanzar la vigilancia del tiempo de ciclo) permite resetear el temporizador que mide el tiempo de ciclo. No obstante, esta instrucción funciona únicamente si se ejecuta en un OB de ciclo. La instrucción RE_TRIGR se ignorará si se ejecuta en el OB 80. Si el tiempo de ciclo máximo se rebasa dos veces en un mismo ciclo del programa, sin que la instrucción RE_TRIGR se ejecute entre los dos rebases, la CPU cambiará inmediatamente al estado operativo STOP. Si la instrucción RE_TRIGR se ejecuta repetidas veces, ello puede dar origen a un bucle infinito o un ciclo muy prolongado.

Generalmente, el ciclo se ejecuta tan rápido como sea posible y el ciclo siguiente comienza cuando finaliza el ciclo actual. En función del programa de usuario y las tareas de comunicación, el tiempo de ciclo puede fluctuar de ciclo en ciclo. Para eliminar esta fluctuación, la CPU soporta un tiempo de ciclo mínimo fijo opcional (o "ciclo fijo"). Si está habilitada esta función opcional y se ha definido un tiempo de ciclo mínimo fijo en ms, la CPU mantendrá el tiempo de ciclo mínimo con una tolerancia de ± 1 ms para la finalización de cada ciclo.

Si la CPU finaliza el ciclo normal antes del tiempo de ciclo mínimo especificado, aprovechará el tiempo restante para realizar tareas de diagnóstico en runtime y/o procesar peticiones de comunicación. De esta manera, la CPU utiliza siempre un tiempo fijo para finalizar un ciclo.

Si la CPU no finaliza el ciclo normal dentro del tiempo de ciclo mínimo especificado, lo finalizará normalmente (incluyendo el procesamiento de las peticiones de comunicación), sin que el rebase del tiempo de ciclo mínimo cause una reacción del sistema. La tabla siguiente muestra los rangos y valores predeterminados para las funciones de vigilancia del tiempo de ciclo.

Tabla 4- 5 Rango para el tiempo de ciclo

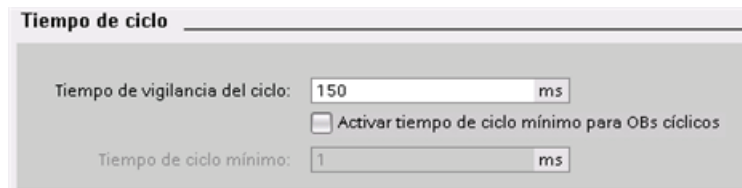
Tiempo de ciclo	Rango (ms)	Valor predeterminado
Tiempo de ciclo máximo ¹	1 a 6000	150 ms
Tiempo de ciclo mínimo fijo ²	1 hasta tiempo de ciclo máximo	Inhibido

- ¹ El tiempo de ciclo máximo siempre está habilitado. Configure un tiempo de ciclo comprendido entre 1 y 6000 ms. El valor predeterminado es 150 ms.
- ² El tiempo de ciclo mínimo fijo es opcional y está inhibido de forma predeterminada. En caso necesario, configure un tiempo de ciclo comprendido entre 1 ms y el tiempo de ciclo máximo.

Configurar el tiempo de ciclo y la carga de comunicación

Las propiedades de la CPU en la "Configuración de dispositivos" permiten configurar los siguientes parámetros:

- **Tiempo de ciclo:** Es posible introducir un tiempo de ciclo máximo. También se puede definir un tiempo de ciclo mínimo fijo.



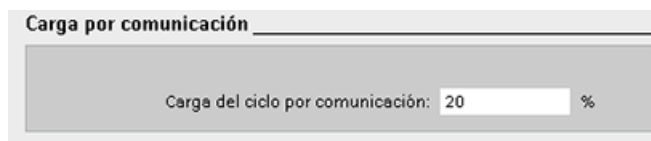
Tiempo de ciclo

Tiempo de vigilancia del ciclo: 150 ms

Activar tiempo de ciclo mínimo para OBs cíclicos

Tiempo de ciclo mínimo: 1 ms

- **Carga de comunicación:** Es posible configurar un porcentaje del tiempo que debe dedicarse a las tareas de comunicación.



Carga por comunicación

Carga del ciclo por comunicación: 20 %

Encontrará más información sobre el ciclo en el apartado "Vigilancia del tiempo de ciclo". (Página 81)

4.1.6 Memoria de la CPU

Gestión de la memoria

La CPU provee las áreas de memoria siguientes para almacenar el programa de usuario, los datos y la configuración:

- La memoria de carga permite almacenar de forma no volátil el programa de usuario, los datos y la configuración. Cuando un proyecto se carga en la CPU, se almacena primero en el área de memoria de carga. Esta área se encuentra bien sea en una Memory Card (si está disponible) o en la CPU. Esta área de memoria no volátil se conserva incluso tras una pérdida de potencia. La Memory Card ofrece mayor espacio de almacenamiento que el integrado en la CPU.
- La memoria de trabajo ofrece almacenamiento volátil para algunos elementos del proyecto mientras se ejecuta el programa de usuario. La CPU copia algunos elementos del proyecto desde la memoria de carga en la memoria de trabajo. Esta área volátil se pierde si se desconecta la alimentación. La CPU la restablece al retornar la alimentación.

- La memoria remanente permite almacenar de forma no volátil un número limitado de valores de la memoria de trabajo. El área de memoria remanente se utiliza para almacenar los valores de algunas posiciones de memoria durante una pérdida de potencia. Cuando se produce una caída o un corte de la alimentación, la CPU restaura esos valores remanentes al restablecer la alimentación.

Para ver el uso de memoria del proyecto actual, haga clic con el botón derecho del ratón en la CPU (o uno de sus bloques) y elija el comando "Carga de la memoria" del menú contextual. Para ver el uso de memoria de la CPU actual, haga doble clic en "Online y diagnóstico", expanda "Diagnóstico" y seleccione "Memoria".

Memoria remanente

Para impedir la pérdida de datos tras un corte de alimentación, es posible definir que ciertos datos sean remanentes. Los siguientes datos pueden configurarse para que sean remanentes:

- Área de marcas (M): El ancho preciso de la memoria para el área de marcas puede definirse en la tabla de variables PLC o el plano de ocupación. El área de marcas remanente comienza siempre en MB0, abarcando consecutivamente un determinado número de bytes. Para definir este valor, haga clic en el botón "Remanencia" de la barra de herramientas de la tabla de variables PLC o del plano de ocupación. Introduzca el número de bytes M que deben ser remanentes a partir de MB0.
- Variables de un bloque de función (FB): Si un FB se ha creado estando seleccionado "Optimizado", la interfaz del bloque de este FB incluirá la columna "Remanencia". En esta columna es posible seleccionar "Remanente", "No remanente" o "Ajustar en IDB" individualmente para cada una de las variables. Un DB de instancia que haya sido creado al insertar este FB en el editor de programas muestra asimismo la columna "Remanencia". El estado remanente de una variable solo se puede modificar desde el editor de la interfaz del DB de instancia si se ha seleccionado "Ajustar en IDB" (activado en el bloque de datos de instancia) en la selección "Remanencia" de la variable en el FB optimizado.

Si se ha creado un FB estando seleccionado "Estándar - compatible con S7-300/400", el editor de la interfaz de este FB no incluirá la columna "Remanencia". Un DB de instancia que haya sido creado al insertar este FB en el editor de programas muestra y permite editar la columna "Remanencia". En este caso, si se activa la opción "Remanente" para alguna de las variables, se seleccionarán **todas** las variables. Por analogía, si se desactiva la opción "Remanente" para alguna de las variables, se deseleccionarán **todas** las variables. Si un FB se ha configurado con el atributo "Estándar - compatible con S7-300/400", el estado remanente se puede cambiar desde el editor del DB de instancia, pero todas las variables se ajustan conjuntamente al mismo estado remanente.

Tras haber creado el FB no es posible modificar la opción "Estándar - compatible con S7-300/400". Sólo se puede seleccionar esta opción al crear el FB. Para determinar si un FB existente se ha configurado como "Optimizado" o "Estándar - compatible con S7-300/400", en el árbol del proyecto haga clic con el botón derecho del ratón en el FB, elija "Propiedades" y seleccione luego "Atributos". La casilla "Acceso optimizado al bloque" indica si un bloque está optimizado cuando está seleccionada. De lo contrario, es estándar y compatible con las CPU S7-300/400.

- Variables de un bloque de datos global: El comportamiento de un DB global respecto a la asignación del estado remanente es similar al de un FB. En función del ajuste de acceso al bloque, es posible definir el estado remanente de algunas o todas las variables de un bloque de datos global.
 - Si se ha seleccionado "Optimizado" al crear el DB, se puede definir el estado remanente para cada variable.
 - Si se selecciona "Estándar - compatible con S7-300/400" al crear el DB, el ajuste de estado remanente se aplica a todas las variables del DB; tanto si todas las variables son remanentes como si ninguna es remanente.

Un total de 10240 bytes de datos pueden ser remanentes. Para ver cuánto espacio está disponible, haga clic en el botón "Remanencia" de la barra de herramientas de la tabla de variables PLC o del plano de ocupación. Aunque aquí se especifica el rango remanente para la memoria M, la segunda fila indica la memoria restante disponible en total para M y DB conjuntamente. Hay que tener en cuenta que para que este valor sea preciso, se deben compilar todos los bloques de datos con variables remanentes.

4.1.6.1 Marcas de sistema y de ciclo

Los bytes de "marcas de sistema" y "marcas de ciclo" se habilitan en las propiedades de la CPU. La lógica del programa puede referenciar los distintos bits de estas funciones por sus nombres de variable.

- Un byte del área de marcas (M) se puede asignar a las marcas de sistema. El byte de marcas de sistema pone a disposición los siguientes cuatro bits que puede referenciar el programa de usuario mediante los siguientes nombres de variables:
 - Primer ciclo: El bit (nombre de variable "FirstScan") se pone a 1 durante el primer ciclo tras finalizar el OB de arranque. (Una vez finalizada la ejecución del primer ciclo, el bit "Primer ciclo" se pone a 0.)
 - El estado de diagnóstico modificado (nombre de variable: "DiagStatusUpdate") se pone a 1 durante un ciclo, cuando la CPU registra un evento de diagnóstico. Puesto que la CPU no activa el bit "Diagrama de diagnóstico modificado" hasta el final de la primera ejecución de los OBs de ciclo de programa, el programa de usuario no puede detectar si ha cambiado el diagnóstico durante la ejecución de los OBs de arranque, o bien durante la primera ejecución de los OBs de ciclo de programa.
 - Siempre 1 (high): El bit (nombre de variable "AlwaysTRUE") está siempre puesto a 1.
 - Siempre 0 (low): El bit (nombre de variable "AlwaysFALSE") está siempre puesto a 0.
- Es posible asignar un byte de marcas de ciclo en el área de marcas. Todo bit del byte de marcas de ciclo genera un impulso de onda cuadrada. El byte de marcas de ciclo ofrece 8 frecuencias diferentes, comprendidas entre 0,5 Hz (lenta) hasta 10 Hz (rápida). Estos bits pueden utilizarse como bits de control para disparar acciones cíclicas en el programa de usuario, especialmente si se combinan con instrucciones de detección de flancos.

4.1 Ejecución del programa de usuario

La CPU inicializa estos bytes cuando el estado operativo cambia de STOP a ARRANQUE. Los bits de las marcas de ciclo cambian de forma síncrona al reloj de la CPU durante los estados operativos ARRANQUE y RUN.

⚠ PRECAUCIÓN

Si se sobrescriben los bits de marcas de sistema o de ciclo, se podrían corromper los datos en estas funciones. Debido a ello, el programa de usuario funcionará incorrectamente, lo que podría ocasionar daños materiales y lesiones corporales.

Puesto que las marcas de ciclo y de sistema forman no están reservadas en la memoria M, las instrucciones o comunicaciones pueden escribir en estas posiciones de memoria y corromper los datos.

Evite escribir datos en estas direcciones para garantizar el funcionamiento correcto de estas funciones y prevea siempre un circuito de parada de emergencia para el proceso o la máquina.

La marca de sistema configura un byte con bits que se activan (se ponen a 1) con un evento determinado.

Bits de marcas de sistema

Activar la utilización del byte de marcas de sistema

Dirección del byte de marcas de sistema (MBx):

Primer ciclo:

Diagrama de diagnóstico modificado:

Siempre 1 (high):

Siempre 0 (low):

Tabla 4- 6 Marcas de sistema

7	6	5	4	3	2	1	0
Reservado Valor 0				Siempre off Valor 0	Siempre ON Valor 1	Indicador de estado de diagnóstico • 1: Cambiar • 0: No cambiar	Indicador de primer ciclo • 1: Primer ciclo tras arranque • 0: No es primer ciclo

Las marcas de ciclo configuran un byte que activa y desactive los distintos bits en intervalos fijos. Cada bit de reloj genera un impulso de onda cuadrada en el bit correspondiente del área de marcas. Estos bits pueden utilizarse como bits de control para disparar acciones cíclicas en el programa de usuario, especialmente si se combinan con instrucciones de detección de flancos.

Bits de marcas de ciclo

Activar la utilización del byte de marcas de ciclo

Dirección del byte de marcas de ciclo (MBx):

Reloj 10 Hz:

Reloj 5 Hz:

Reloj 2,5 Hz:

Reloj 2 Hz:

Reloj 1,25 Hz:

Reloj 1 Hz:

Reloj 0,625 Hz:

Reloj 0,5 Hz:

Tabla 4- 7 Marcas de ciclo

Número de bit	7	6	5	4	3	2	1	0
Nombre de la variable								
Período (s)	2,0	1,6	1,0	0,8	0,5	0,4	0,2	0,1
Frecuencia (Hz)	0,5	0,625	1	1,25	2	2,5	5	10

Dado que la marca de ciclo es asíncrona respecto al ciclo de la CPU, el estado de la marca de ciclo puede cambiar varias veces durante un ciclo largo.

4.1.7 Búfer de diagnóstico

La CPU soporta un búfer de diagnóstico que contiene una entrada para cada evento de diagnóstico. Toda entrada incluye la fecha y hora del evento, así como su categoría y descripción. Las entradas se visualizan en orden cronológico. El evento más reciente aparece en primer lugar. En este búfer están disponibles los 50 eventos más recientes. Cuando se llena el búfer, un evento nuevo reemplaza al evento más antiguo. Cuando se corta la alimentación, se almacenan los eventos.

Los siguientes tipos de eventos se registran en el búfer de diagnóstico:

- Todo evento de diagnóstico del sistema, p. ej. errores de la CPU y de los módulos
- Todo cambio de estado de la CPU (todo arranque, toda transición a STOP, toda transición a RUN)

Para acceder al búfer de diagnóstico (Página 717) es preciso estar online. En la vista "Online y diagnóstico" el búfer de diagnóstico está en "Diagnóstico > Búfer de diagnóstico".

4.1.8 Reloj en tiempo real

La CPU soporta un reloj en tiempo real. Un condensador de alto rendimiento suministra la energía necesaria para que el reloj pueda seguir funcionando mientras está desconectada la alimentación de la CPU. El condensador de alto rendimiento se carga mientras está conectada la alimentación de la CPU. Tras haber estado conectada la alimentación de la CPU como mínimo 24 horas, la carga del condensador de alto rendimiento será suficiente para que el reloj pueda funcionar unos 20 días.

STEP 7 ajusta el reloj en tiempo real a la hora del sistema, que está a un valor predeterminado tras el primer encendido o tras un restablecimiento a los ajustes de fábrica. Para usar el reloj en tiempo real primero hay que ajustarlo. Los sellos de tiempo como los usados en las entradas del búfer de diagnóstico, los archivos de registro o las entradas de registros se basan en la hora del sistema. La hora se ajusta mediante la función "Ajustar la hora" (Página 714) de la vista "Online y diagnóstico" de la CPU online. STEP 7 calcula entonces la hora del sistema a partir de la hora ajustada y de la diferencia del sistema operativo Windows con el tiempo universal coordinado (UTC). El ajuste de la hora a la hora local actual genera una hora del sistema UTC si los ajustes de horario de verano y de zona horaria del sistema operativo Windows corresponden a la ubicación regional.

STEP 7 dispone de instrucciones (Página 259) para leer y escribir la hora del sistema (RD_SYS_T y WR_SYS_T), para leer la hora local (RD_LOC_T) y para ajustar la zona horaria (SET_TIMEZONE). La instrucción RD_LOC_T calcula la hora local usando las diferencias por zona horaria y por horario de verano ajustadas en la configuración "Hora" de las propiedades generales de la CPU (Página 127). Esos ajustes permiten establecer la zona horaria para la hora local, habilitar opcionalmente el horario de verano y especificar las fechas y horas iniciales y finales del horario de verano. También se puede usar la instrucción SET_TIMEZONE para configurar esos ajustes.

4.1.9 Configurar las salidas en una transición de RUN a STOP

Es posible configurar la reacción de las salidas digitales y analógicas cuando la CPU se encuentre en estado operativo STOP. Es posible congelar los valores de las salidas o aplicar un valor sustitutivo a cualquier salida de una CPU, SB o SM:

- Sustituir un valor de salida específico (ajuste predeterminado): Para cada salida (canal) de la CPU, de la SB o del SM se define un valor sustitutivo.

El valor sustitutivo predeterminado de los canales de salida digitales es OFF y el de los canales de salida analógicos es 0.

- Congelar las salidas a su último estado: Las salidas conservan su valor actual en el momento de la transición de RUN a STOP. Después del arranque, las salidas se ajustan al valor sustitutivo predeterminado.

La reacción de las salidas se configura en la "Configuración de dispositivos". Seleccione los dispositivos individuales y utilice la ficha "Propiedades" para configurar las salidas de cada dispositivo.

Cuando la CPU cambia de RUN a STOP, conserva la memoria imagen de proceso y escribe los valores correspondientes en las salidas digitales y analógicas según la configuración.

4.2 Almacenamiento de datos, áreas de memoria, E/S y direccionamiento

4.2.1 Acceder a los datos del S7-1200

STEP 7 facilita la programación simbólica. Se crean nombres simbólicos o "variables" para las direcciones de los datos, ya sea como variables PLC asignadas a direcciones de memoria y E/S o como variables locales utilizadas dentro de un bloque lógico. Para utilizar estas variables en el programa de usuario basta con introducir el nombre de variable para el parámetro de instrucción.

Para una mejor comprensión de cómo la CPU estructura y direcciona las áreas de memoria, los siguientes párrafos explican el direccionamiento "absoluto" al que se refieren las variables PLC. La CPU ofrece varias opciones para almacenar datos durante la ejecución del programa de usuario:

- Memoria global: La CPU ofrece distintas áreas de memoria, incluyendo entradas (I), salidas (Q) y marcas (M). Todos los bloques lógicos pueden acceder sin restricción alguna a esta memoria.
- Tabla de variables PLC: se pueden especificar nombres simbólicos en la tabla de variables PLC de STEP 7 para posiciones de memoria específicas. Esas variables son globales dentro del programa STEP 7 y permiten la programación con nombres significativos para la aplicación.
- Bloque de datos (DB): Es posible incluir DBs en el programa de usuario para almacenar los datos de los bloques lógicos. Los datos almacenados se conservan cuando finaliza la ejecución del bloque lógico asociado. Un DB "global" almacena datos que pueden ser utilizados por todos los bloques lógicos, mientras que un DB de instancia almacena datos para un bloque de función (FB) específico y está estructurado según los parámetros del FB.
- Memoria temporal: Cada vez que se llama un bloque lógico, el sistema operativo de la CPU asigna la memoria temporal o local (L) que debe utilizarse durante la ejecución del bloque. Cuando finaliza la ejecución del bloque lógico, la CPU reasigna la memoria local para la ejecución de otros bloques lógicos.

Toda posición de memoria diferente tiene una dirección unívoca. El programa de usuario utiliza estas direcciones para acceder a la información de la posición de memoria. Las referencias a las áreas de memoria de entrada (I) o salida (Q), como I0.3 o Q1.7, acceden a la memoria imagen del proceso. Para acceder inmediatamente a la entrada o salida física es preciso añadir ":P" a la dirección (p. ej. I0.3:P, Q1.7:P o "Stop:P").

Tabla 4- 8 Áreas de memoria

Área de memoria	Descripción	Forzado permanente	Remanente
I Memoria imagen de proceso de las entradas I_:P ¹ (entrada física)	Se copia de las entradas físicas al inicio del ciclo	No	No
	Lectura inmediata de las entradas físicas de la CPU, SB y SM	Sí	No

4.2 Almacenamiento de datos, áreas de memoria, E/S y direccionamiento

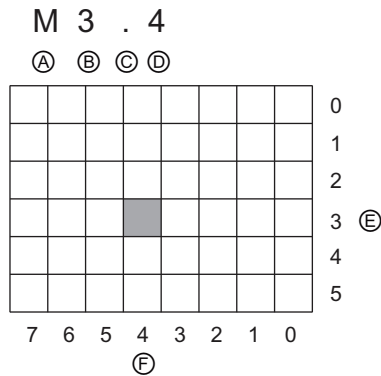
Área de memoria	Descripción	Forzado permanente	Remanente
Q Memoria imagen de proceso de las salidas Q_:P ¹ (salida física)	Se copia en las salidas físicas al inicio del ciclo	No	No
	Escritura inmediata en las salidas físicas de la CPU, SB y SM	Sí	No
M Área de marcas	Control y memoria de datos	No	Sí (opcional)
L Memoria temporal	Datos locales temporales de un bloque	No	No
DB Bloque de datos	Memoria de datos y de parámetros de FBs	No	Sí (opcional)

¹ Para acceder inmediatamente (leer o escribir) a las entradas o salidas físicas es preciso añadir ":P" a la dirección o variable (p. ej. I0.3:P, Q1.7:P o "Stop:P").

Toda posición de memoria diferente tiene una dirección unívoca. El programa de usuario utiliza estas direcciones para acceder a la información de la posición de memoria. La dirección absoluta consta de los elementos siguientes:

- Identificador de área de memoria (como I, Q o M)
- Tamaño de los datos a los que se va a acceder ("B" para Byte, "W" para Word, o "D" para DWord)
- Dirección inicial de los datos (como byte 3 o palabra 3)

Al acceder a un bit en la dirección para un valor booleano, no se introduce ningún nemónico para el tamaño. Sólo se introduce el área de memoria, la ubicación del byte y la ubicación del bit de los datos (como I0.0, Q0.1, o M3.4).



- | | | | |
|---|---------------------------|---|----------------------------|
| A | Identificador de área | E | Bytes del área de memoria |
| B | Dirección de byte: Byte 3 | F | Bits del byte seleccionado |
| C | Separador ("byte.bit") | | |
| D | Bit del byte (bit 4 de 8) | | |

En el ejemplo, el área de memoria y la dirección del byte (M = área de memoria de bit; y 3 = Byte 3) van seguidas de un punto ("."), que separa la dirección del bit (bit 4).

Acceder a los datos en las áreas de memoria de la CPU

STEP 7 facilita la programación simbólica. Normalmente, las variables se crean en variables PLC, en un bloque de datos o en la interfaz arriba de un OB, FC o FB. Estas variables incluyen un nombre, tipo de datos, offset y comentario. Además, es posible definir un valor inicial en un bloque de datos. Estas variables pueden utilizarse durante la programación, introduciendo el nombre de la variable en el parámetro de la instrucción. Opcionalmente se puede introducir el operando absoluto (área de memoria, tamaño y offset) en el parámetro de la instrucción. Los ejemplos de los apartados siguientes muestran cómo introducir operandos absolutos. El editor de programación antepone automáticamente el carácter % al operando absoluto. Es posible cambiar entre las siguientes vistas del editor de programación: simbólica, simbólica y absoluta o absoluta.

I (memoria imagen de proceso de las entradas): La CPU consulta las entradas de periferia (físicas) inmediatamente antes de ejecutar el OB de ciclo en cada ciclo y escribe estos valores en la memoria imagen de proceso de las entradas. A la memoria imagen de proceso de las entradas se puede acceder en formato de bit, byte, palabra o palabra doble. Aunque se permiten accesos de lectura y escritura, generalmente sólo se leen las entradas de la memoria imagen de proceso.

Tabla 4- 9 Direccionamiento absoluto para memoria I

Bit	I[dirección de byte].[dirección de bit]	I0.1
Byte, palabra o palabra doble	I[tamaño][dirección de byte inicial]	IB4, IW5 o ID12

Añadiendo una ":P" a la dirección es posible leer inmediatamente las entradas digitales y analógicas de la CPU, SB o SM. La diferencia entre un acceso que utiliza I_:P en vez de I es que los datos provienen directamente de las entradas direccionadas, en vez de la memoria imagen de proceso de las entradas. El acceso I_:P también se denomina "lectura inmediata", puesto que los datos se leen inmediatamente del origen y no de una copia creada la última vez que se actualizó la memoria imagen de proceso de las entradas.

Puesto que las entradas físicas reciben sus valores directamente de los aparatos de campo conectados a ellas, está prohibido escribir en estas entradas. Por tanto, los accesos I_:P son de sólo lectura, a diferencia de los accesos I que pueden ser de lectura o escritura.

Los accesos I_:P también están restringidos por el tamaño de las entradas que soporta una única CPU, SB o SM, redondeado al byte más próximo. Por ejemplo, si las entradas de una SB de 2 DI / 2 DQ se configuran de manera que comiencen en I4.0, las entradas se podrán direccionar como I4.0:P e I4.1:P, o bien IB4:P. Aunque no se rechazan los accesos a I4.2:P hasta I4.7:P, no tienen sentido ya que estas entradas no se utilizan. Los accesos a IW4:P y ID4:P están prohibidos, puesto que exceden el offset de bytes asociado a la SB.

Los accesos mediante I_:P no afectan el valor correspondiente almacenado en la memoria imagen de proceso de las entradas.

Tabla 4- 10 Direccionamiento absoluto para memoria I (inmediata)

Bit	I[dirección de byte].[dirección de bit]:P	I0.1:P
Byte, palabra o palabra doble	I[tamaño][dirección de byte inicial]:P	IB4:P, IW5:P o ID12:P

Q (memoria imagen de proceso de las salidas): La CPU copia los valores almacenados en la imagen de proceso de las salidas en las salidas físicas. A la memoria imagen de proceso de las salidas se puede acceder en formato de bit, byte, palabra o palabra doble. Se permiten accesos de lectura y escritura a la memoria imagen de proceso de las salidas.

Tabla 4- 11 Direccionamiento absoluto para memoria Q

Bit	Q[dirección de byte].[dirección de bit]	Q1.1
Byte, palabra o palabra doble	Q[tamaño][dirección de byte inicial]	QB5, QW10, QD40

Añadiendo una ":P" a la dirección es posible escribir inmediatamente en las salidas digitales y analógicas físicas de la CPU, SB o SM. La diferencia entre un acceso que utiliza Q_:P en vez de Q es que los datos se escriben directamente en las salidas direccionadas y también en la memoria imagen de proceso de las salidas. El acceso Q_:P se denomina a veces "escritura inmediata", puesto que los datos se escriben inmediatamente en la salida de destino. Por tanto, ésta no tiene que esperar hasta la siguiente actualización desde la memoria imagen de proceso de las salidas.

Puesto que las salidas físicas controlan directamente los aparatos de campo conectados a ellas, está prohibido leer de estas salidas. Por tanto, los accesos Q_:P son de sólo escritura, a diferencia de los accesos Q que pueden ser de lectura o escritura.

Los accesos Q_:P también están restringidos por el tamaño de las salidas que soporta una única CPU, SB o SM, redondeado al byte más próximo. Por ejemplo, si las salidas de una SB de 2 DI / 2 DQ se configuran de manera que comiencen en Q4.0, las salidas se podrán direccionar como Q4.0:P y Q4.1:P, o bien QB4:P. Aunque no se rechazan los accesos a QB4.2:P hasta QB4.7:P, no tienen sentido ya que estas salidas no se utilizan. Los accesos a QW4:P y QD4:P están prohibidos, puesto que exceden el offset de bytes asociado a la SB.

Los accesos mediante Q_:P afectan tanto la salida física como el valor correspondiente almacenado en la memoria imagen de proceso de las salidas.

Tabla 4- 12 Direccionamiento absoluto para memoria Q (inmediata)

Bit	Q[dirección de byte].[dirección de bit]:P	Q1.1:P
Byte, palabra o palabra doble	Q[tamaño][dirección de byte inicial]:P	QB5:P, QW10:P o QD40:P

M (área de marcas): El área de marcas (memoria M) puede utilizarse para relés de control y datos para almacenar el estado intermedio de una operación u otra información de control. Al área de marcas se puede acceder en formato de bit, byte, palabra o palabra doble. Se permiten accesos de lectura y escritura al área de marcas.

Tabla 4- 13 Direccionamiento absoluto para memoria M

Bit	M[dirección de byte].[dirección de bit]	M26.7
Byte, palabra o palabra doble	M[tamaño][dirección de byte inicial]	MB20, MW30, MD50

Temp (memoria temporal): La CPU asigna la memoria temporal según sea necesario. La CPU asigna la memoria temporal al bloque lógico cuando éste se inicia (en caso de un OB) o se llama (en caso de una FC o un FB). La asignación de la memoria temporal a un bloque lógico puede reutilizar las mismas posiciones de memoria temporal usadas anteriormente por un OB, FC o FB diferente. La CPU no inicializa la memoria temporal durante la asignación, por lo que esta memoria puede contener un valor cualquiera.

La memoria temporal es similar al área de marcas, con una excepción importante: el área de marcas tiene un alcance "global", en tanto que la memoria temporal tiene un alcance "local".

- Área de marcas: Cualquier OB, FC o FB puede acceder a los datos del área de marcas. Esto significa que los datos están disponibles globalmente para todos los elementos del programa de usuario.
- Memoria temporal: El acceso a los datos de la memoria temporal está restringido al OB, FC o FB que ha creado o declarado la posición de memoria temporal. Las posiciones de memoria temporal son siempre locales y no son compartidas por diferentes bloques lógicos, incluso si un bloque lógico llama otro bloque lógico. Ejemplo: Cuando un OB llama una FC, ésta no puede acceder a la memoria temporal del OB que ha efectuado la llamada.

La CPU pone a disposición memoria temporal (local) para cada una de las tres clases de prioridad de OBs:

- 16 KB para arranque y ciclo, incluyendo los FBs y FCs asociados
- 4 KB para eventos de alarma estándar, incluyendo FBs y FCs
- 4 KB para eventos de alarma de error, incluyendo FBs y FCs

A la memoria temporal se puede acceder sólo con direccionamiento simbólico.

DB (bloque de datos): Los bloques de datos se utilizan para almacenar diferentes tipos de datos, incluyendo el estado intermedio de una operación u otros parámetros de control de FBs, así como estructuras de datos requeridas para numerosas instrucciones, p. ej. temporizadores y contadores. A los bloques de datos se puede acceder en formato de bit, byte, palabra o palabra doble. A los bloques de datos que se pueden leer y escribir se permiten accesos de lectura y escritura. A los bloques de datos de sólo lectura se permiten sólo los accesos de lectura.

Tabla 4- 14 Direccionamiento absoluto para memoria DB

Bit	DB[número de bloque de datos].DBX[dirección de byte].[dirección de bit]	DB1.DBX2.3
Byte, palabra o palabra doble	DB[número de bloque de datos].DB [tamaño][dirección de byte inicial]	DB1.DBB4, DB10.DBW2, DB20.DBD8

Nota

Cuando especifica una dirección absoluta, STEP 7 coloca el carácter "%" antes de esta dirección para indicar que se trata de una dirección absoluta. Durante la programación, puede especificar una dirección absoluta con o sin el carácter "%" (por ejemplo: %I0.0 o bien I.0). Si no se especifica, STEP 7 incluye el carácter "%".

Configuración de las E/S de la CPU y los módulos de E/S



Al agregar una CPU y módulos de E/S en la ventana de configuración, se asignan automáticamente direcciones I y Q. El direccionamiento predeterminado puede cambiarse seleccionando el campo de dirección en la ventana de configuración y tecleando números nuevos.

- Las entradas y salidas digitales se asignan en grupos de 8 E/S (1 byte), sin importar si el módulo utiliza todas las E/S o no.
- Las entradas y salidas analógicas se asignan en grupos de 2 (4 bytes).

Vista general de dispositivos						
Módulo	Slot	Direcció.	Direcció.	Tipo	Refere	
	103					
	102					
RS485_1	101			CM 1241 (RS485)	6ES7	
PLC_1	1			CPU 1214C DC/DC	6ES7	
DI14/DO10	1.1	0...1	0...1	DI14/DO10		
AI2	1.2	64...67		AI2		
AO1 x 12bi	1.3		80...81	AO1 Signal Board	6ES7	
HSC_1	1.16	1000...		Contador rápido (-)		
HSC_2	1.17			Contador rápido (-)		
HSC_3	1.18			Contador rápido (-)		
HSC_4	1.19			Contador rápido (-)		
HSC_5	1.20			Contador rápido (-)		
HSC_6	1.21			Contador rápido (-)		
Pulse_1	1.32			Generador de Imp.		
Pulse_2	1.33			Generador de Imp.		
Interfaz PR. X1				Interfaz PROFINET		
DI8 x DC24V..	2	8		SM 1221 DI8 x DC.	6ES7	

La figura muestra un ejemplo de una CPU 1214C con dos SM y una SB. En este ejemplo, la dirección del módulo DI8 se podría cambiar a 2 en vez de 8. La herramienta cambia los rangos de direcciones cuyo tamaño sea incorrecto o que causen conflictos con otras direcciones.

4.3 Procesamiento de valores analógicos

Los módulos de señales analógicas proporcionan señales de entrada o esperan valores de salida que representen un rango de tensión o de corriente. Estos rangos son ±10V, ±5V, ±2,5V, o 0 - 20mA. Los valores que devuelven los módulos son valores enteros en los que 0 a 27648 representa el rango nominal de corriente, y -27648 a 27648 de tensión. Cualquier valor fuera del rango representa un rebase por exceso o por defecto. Véanse las tablas de representación de entradas analógicas (Página 814) y representación de salidas analógicas (Página 815) para más detalles.

En el programa de control puede ser necesario utilizar estos valores en unidades de ingeniería, por ejemplo, para representar un volumen, temperatura, peso o cualquier otro valor cuantitativo. En el caso de una entrada analógica, para hacerlo primero hay que normalizar el valor analógico a un valor real (coma flotante) de 0,0 a 1,0. A continuación hay que escalarlo a los valores mínimo y máximo de las unidades de ingeniería que representa. En el caso de valores de unidades de ingeniería que deben convertirse a valores de salida analógicos, primero hay que normalizar el valor en las unidades de ingeniería a un valor entre 0,0 y 1,0, y a continuación escalarlo entre 0 y 27648 o -27648 a 27648, dependiendo del rango del módulo analógico. Para este propósito, STEP 7 proporciona las instrucciones NORM_X y SCALE_X (Página 227). También se puede utilizar la instrucción CALCULATE (Página 205) para escalar los valores analógicos (Página 33).

4.4 Tipos de datos

Los tipos de datos se utilizan para determinar el tamaño de un elemento de datos y cómo deben interpretarse los datos. Todo parámetro de instrucción soporta como mínimo un tipo de datos. Algunos parámetros soportan varios tipos de datos. Sitúe el cursor sobre el campo de parámetro de una instrucción para ver qué tipos de datos soporta el parámetro en cuestión.

Un parámetro formal es el identificador en una instrucción que indica la ubicación de los datos que deben utilizarse (ejemplo: la entrada IN1 de una instrucción ADD). Un parámetro actual es la posición de memoria (precedida por el carácter "%") o constante que contiene los datos que debe utilizar la instrucción (ejemplo: %MD400 "Número_de_widgets"). El tipo de datos del parámetro actual definido por el usuario debe concordar con uno de los tipos de datos que soporta el parámetro formal especificado por la instrucción.

Al definir un parámetro actual es preciso indicar una variable (direccionamiento simbólico) o una dirección absoluta (direccionamiento directo). Las variables asocian un nombre simbólico (nombre de variable) con un tipo de datos, área de memoria, offset y comentario. Se pueden crear bien sea en el editor de variables PLC, o bien en la interfaz del bloque (OB, FC, FB y DB). Si se introduce una dirección absoluta que no tenga una variable asociada, es preciso utilizar un tamaño apropiado que coincida con el tipo de datos soportado. Al realizar la entrada se creará una variable predeterminada.

Todos los tipos de datos, excepto String, están disponibles en el editor de variables PLC y en la interfaz del bloque. String sólo está disponible en la interfaz del bloque. También es posible introducir un valor de constante para numerosos parámetros de entrada.

- Bit y secuencias de bit (Página 96): Bool (valor booleano o bit), Byte (valor byte de 8 bits), Word (valor de 16 bits), DWord (valor de 32 bits, doble palabra)
- Entero (Página 97)
 - USInt (entero de 8 bits sin signo), SInt (entero de 8 bits con signo),
 - UInt (entero de 16 bits sin signo), Int (entero de 16 bits con signo)
 - UDInt (entero de 32 bits sin signo), DInt (entero de 32 bits con signo)
- Real en coma flotante (Página 98): Real (real de 32 bits o valor en coma flotante), LReal (real de 64 bits o valor en coma flotante)
- Fecha y hora (Página 98): Time (valor de tiempo CEI de 32 bits), Date (valor de fecha de 16 bits), TOD (valor de hora de 32 bits), DT (valor de fecha y hora de 64 bits)

4.4 Tipos de datos

- Carácter y cadena (Página 100): Char (carácter único de 8 bits), String (cadena de longitud variable de hasta 254 caracteres)
- Matriz (Página 101)
- Estructura de los datos (Página 103): Struct
- Tipo de datos PLC (Página 103)
- Punteros (Página 104): Pointer, Any, Variant

Aunque no están disponibles como tipos de datos, las instrucciones de conversión soportan el siguiente formato numérico BCD.

Tabla 4- 15 Tamaño y rango del formato BCD

Formato	Tamaño (bits)	Rango numérico	Ejemplos de entrada de constantes
BCD16	16	-999 a 999	123, -123
BCD32	32	-9999999 a 9999999	1234567, -1234567

4.4.1 Tipos de datos Bool, Byte, Word y DWord

Tabla 4- 16 Tipos de datos bit y secuencia de bits

Tipo de datos	Tamaño o en bits	Tipo de número	Rango numérico	Ejemplos de constante	Ejemplos de dirección
Bool	1	Booleano	FALSE o TRUE	TRUE, 1,	I1.0 Q0.1 M50.7 DB1.DBX2.3 Nombre_variable
		Binario	0 ó 1	0, 2#0	
		Octal	8#0 ó 8#1	8#1	
		Hexadecimal	16#0 ó 16#1	16#1	
Byte	8	Binario	2#0 a 2#11111111	2#00001111	IB2 MB10 DB1.DBB4 Nombre_variable
		Entero sin signo	0 a 255	15	
		Octal	8#0 a 8#377	8#17	
		Hexadecimal	B#16#0 a B#16#FF	B#16#F, 16#F	
Word	16	Binario	2#0 a 2#1111111111111111	2#1111000011110000	MW10 DB1.DBW2 Nombre_variable
		Entero sin signo	0 a 65535	61680	
		Octal	8#0 a 8#177777	8#170360	
		Hexadecimal	W#16#0 a W#16#FFFF, 16#0 a 16#FFFF	W#16#F0F0, 16#F0F0	
DWord	32	Binario	2#0 a 2#11111111111111111111111111111111	2#111100001111111100001111	MD10 DB1.DBD8 Nombre_variable
		Entero sin signo	0 a 4294967295	15793935	
		Octal	8#0 a 8#3777777777	8#74177417	

Tipo de datos	Tamaño en bits	Tipo de número	Rango numérico	Ejemplos de constante	Ejemplos de dirección
		Hexadecimal	DW#16#0000_0000 a DW#16#FFFF_FFFF, 16#0000_0000 a 16#FFFF_FFFF	DW#16#F0FF0F, 16#F0FF0F	

4.4.2 Tipos de datos de entero

Tabla 4- 17 Tipos de datos de entero (U = sin signo, S = simple, D= doble)

Tipo de datos	Tamaño en bits	Rango numérico	Ejemplos de constante	Dirección Ejemplos
USInt	8	0 a 255	78, 2#01001110	MB0, DB1.DBB4,
SInt	8	128 a 127	+50, 16#50	Nombre_variable
UInt	16	0 a 65.535	65295, 0	MW2, DB1.DBW2,
Int	16	32.768 a 32.767	30000, +30000	Nombre_variable
UDInt	32	0 a 4.294.967.295	4042322160	MD6, DB1.DBD8,
DInt	32	-2.147.483.648 a 2.147.483.647	-2131754992	Nombre_variable

4.4.3 Tipos de datos de real en coma flotante

Los números reales (o en coma flotante) se representan como números de 32 bits de precisión simple (Real) o de 64 bits de precisión doble (LReal) según la norma ANSI/IEEE 7541985. Los números en coma flotante de precisión simple tienen una exactitud de hasta 6 dígitos significativos, en tanto que los de precisión doble tienen una exactitud de hasta 15 dígitos significativos. Al introducir una constante en coma flotante, pueden indicarse como máximo 6 (Real) o 15 (LReal) dígitos significativos para conservar la precisión.

Tabla 4- 18 Tipos de datos de real en coma flotante (L=largo)

Tipo de datos	Tamaño o en bits	Rango numérico	Ejemplos de constante	Ejemplos de dirección
Real	32	-3.402823e+38 a -1.175 495e-38, ±0, +1.175 495e-38 a +3.402823e+38	123.456, -3.4, 1.0e-5	MD100, DB1.DBD8, Nombre_variable
LReal	64	-1,7976931348623158e+308 a -2,2250738585072014e-308, ±0, +2,2250738585072014e-308 a +1,7976931348623158e+308	12345,123456789e40, 1.2E+40	Nombre_DB.nombre_var Reglas: <ul style="list-style-type: none"> No se soporta el direccionamiento directo Se puede asignar en una tabla de interfaz de OB, FB o FC

Los cálculos que comprenden una serie de valores prolongada, incluyendo números muy grandes y muy pequeños, pueden producir resultados inexactos. Esto puede suceder si los números difieren en 10 a la potencia de x, siendo $x > 6$ (Real) ó 15 (LReal). Por ejemplo (Real): $100\ 000\ 000 + 1 = 100\ 000\ 000$.

4.4.4 Tipos de datos de fecha y hora

Tabla 4- 19 Tipos de datos de fecha y hora

Tipo de datos	Tamaño	Rango	Ejemplos de entrada de constantes
Time	32 bits	T#-24d_20h_31m_23s_648ms a T#24d_20h_31m_23s_647ms Almacenado como: -2.147.483.648 ms hasta +2.147.483.647 ms	T#5m_30s T#1d_2h_15m_30s_45ms TIME#10d20h30m20s630ms 500h10000ms 10d20h30m20s630ms
Date	16 bits	D#1990-1-1 a D#2168-12-31	D#2009-12-31 DATE#2009-12-31 2009-12-31
Hora	32 bits	TOD#0:0:0.0 a TOD#23:59:59.999	TOD#10:20:30.400 TIME_OF_DAY#10:20:30.400 23:10:1
DTL (Fecha y hora largo)	12 bytes	Mín.: DTL#1970-01-01-00:00:00.0 Máx.: DTL#2554-12-31-23:59:59.999 999 999	DTL#2008-12-16-20:30:20.250

Time

El dato TIME se guarda como entero doble con signo y se interpreta como milisegundos. El formato del editor puede utilizar información para día (d), horas (h), minutos (m), segundos (s) y milisegundos (ms).

No es necesario especificar todas las unidades de tiempo. Son válidos por ejemplo T#5h10s y 500h.

El valor combinado de todos los valores de unidad especificados no puede superar los límites superior o inferior en milisegundos para el tipo de datos Time (-2.147.483.648 ms a +2.147.483.647 ms).

Date

DATE se guarda como valor entero sin signo y se interpreta como número de días agregados a la fecha patrón 01/01/1990 para obtener la fecha específica. El formato del editor debe especifica un año, un mes y un día.

TOD

TOD (TIME_OF_DAY) se guarda como entero doble sin signo y se interpreta como el número en milisegundos desde medianoche para obtener la hora específica del día (medianoche = 0 ms). Deben especificarse hora (24h/día), minuto y segundo. Las fracciones de segundo son opcionales.

DTL

El tipo de datos DTL (fecha y hora largo) utiliza una estructura de 12 bytes para guardar información sobre la fecha y la hora. DTL se puede definir en la memoria temporal de un bloque o en un DB. Debe indicarse un valor para todos los componentes en la columna "Valor inicial" del editor de DB.

Tabla 4- 20 Tamaño y rango para DTL

Longitud (bytes)	Formato	Rango de valores	Ejemplo de un valor de entrada
12	Reloj y calendario Año-Mes-Día:Hora:Minuto: Segundo.Nanosegundos	Mín.: DTL#1970-01-01-00:00:00.0 Máx.: DTL#2554-12-31-23:59:59.999 999 999	DTL#2008-12-16-20:30:20.250

Todo componente de DTL contiene un tipo de datos y un rango de valores diferentes. El tipo de datos de un valor especificado debe concordar con el tipo de datos de los componentes correspondientes.

Tabla 4- 21 Elementos de la estructura DTL

Byte	Componente	Tipo de datos	Rango de valores
0	Año	UINT	1970 a 2554
1			
2	Mes	USINT	1 a 12

4.4 Tipos de datos

Byte	Componente	Tipo de datos	Rango de valores
3	Día	USINT	1 a 31
4	Día de la semana ¹	USINT	1(domingo) a 7(sábado) ¹
5	Hora	USINT	0 a 23
6	Minuto	USINT	0 a 59
7	Segundo	USINT	0 a 59
8	Nanosegundos	UDINT	0 a 999 999 999
9			
10			
11			

¹ El día de la semana no se tiene en cuenta en la entrada del valor.

4.4.5 Tipos de datos Carácter y Cadena

Tabla 4- 22 Tipos de datos Carácter y Cadena

Tipo de datos	Tamaño	Rango	Ejemplos de entrada de constantes
Char	8 bits	Códigos de caracteres ASCII: 16#00 a 16#FF	'A', 't', '@'
String	n+ 2 bytes	n = (0 a 254 bytes de caracteres)	'ABC'

Char

Char ocupa un byte en la memoria y guarda un único carácter codificado en formato ASCII. La sintaxis del editor utiliza un carácter de comilla simple delante y detrás del carácter ASCII. Pueden usarse caracteres visibles y de control. En la descripción del tipo de datos String se recoge una tabla de caracteres de control válidos.

String

La CPU soporta el tipo de datos STRING para almacenar una secuencia de caracteres de un byte. El tipo de datos STRING contiene el número de caracteres total (número de caracteres de la cadena) y el número de caracteres actual. El tipo de datos STRING ofrece como máximo 256 bytes para almacenar el número máximo de caracteres total (1 byte), el número de caracteres actual (1 byte) y como máximo 254 caracteres. Cada carácter se almacena en 1 byte.

Es posible utilizar cadenas literales (constantes) para los parámetros de instrucción del tipo IN entre comillas sencillas. Por ejemplo, 'ABC' es una cadena de tres caracteres que podría utilizarse como entrada para el parámetro IN de la instrucción S_CONV. También es posible crear variables de cadena, seleccionando para ello el tipo de datos "String" en la interfaz de bloques OB, FC, FB y DB. En el editor de variables PLC no se pueden crear cadenas.

Se puede especificar el tamaño máximo de la cadena introduciendo corchetes después de la palabra clave "String" (una vez que el tipo de datos "String" se ha seleccionado de una lista desplegable de tipos de datos). Por ejemplo, "MyString String[10]" especificaría un tamaño máximo de 10 bytes para MyString. Si se omiten los corchetes con un indicador de tamaño máximo, se presupone que el tamaño máximo es de 254.

El ejemplo máximo define una cadena con un número máximo de 10 caracteres y un número de caracteres actual de 3. Esto significa que la cadena contiene actualmente 3 caracteres de un byte, pero que podría ampliarse de manera que contenga como máximo 10 caracteres de un byte.

Tabla 4- 23 Ejemplo de tipo de datos STRING

Número de caracteres total	Número de caracteres actual	Carácter 1	Carácter 2	Carácter 3	...	Carácter 10
10	3	'C' (16#43)	'A' (16#41)	'T' (16#54)	...	-
Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	...	Byte 11

Pueden usarse caracteres de control ASCII en datos Char y String. La tabla siguiente muestra ejemplos de sintaxis de caracteres de control.

Tabla 4- 24 Caracteres de control ASCII válidos

Caracteres de control	Valor hex ASCII	Función de control	Ejemplos
\$L o \$l	0A	Avance línea	'\$LText', '\$0AText'
\$N o \$n	0A y 0D	Salto de línea La línea nueva muestra dos caracteres en la cadena.	'\$NText', '\$0A\$0DText'
\$P o \$p	0C	Alimentación de página	'\$PText', '\$0CText'
\$R o \$r	0D	Retorno de carro (CR)	'\$RText', '\$0DText'
\$T o \$t	09	Tab	'\$TText', '\$09Text'
\$\$	24	Símbolo del dólar	'100\$\$', '100\$24'
\$'	27	Comilla simple	'\$'Text\$', '\$27Text\$27'

4.4.6 Tipo de datos ARRAY

Matrices

Se puede crear una matriz que contenga varios elementos del mismo tipo de datos. Las matrices pueden crearse en las interfaces de bloques OB, FC, FB y DB. En el editor de variables PLC no se pueden crear matrices.

4.4 Tipos de datos

Para crear una matriz en la interfaz del bloque, asigne un nombre a la matriz y seleccione el tipo de datos "Array [lo .. hi] of type", modifique luego "lo", "hi" y "type" como se indica a continuación:

- lo - el índice inicial (más bajo) de la matriz
- hi - el índice final (más alto) de la matriz
- type - uno de los tipos de datos, como BOOL, SINT, UDINT

Tabla 4- 25 Reglas para el tipo de datos ARRAY

Tipo de datos	Sintaxis de una matriz		
ARRAY	Nombre [index1_min..index1_max, index2_min..index2_max] de <tipo de datos>		
	<ul style="list-style-type: none"> • Todos los parámetros de la matriz deben tener el mismo tipo de datos. • El índice puede ser negativo, pero el límite inferior debe ser inferior o igual que el límite superior. • Las matrices pueden tener entre una y seis dimensiones. • Las declaraciones multidimensionales mín..máx están separadas por caracteres coma. • No se permiten matrices anidadas ni matrices de matrices. • El tamaño de memoria de una matriz = (tamaño de un elemento * número total de elementos de una matriz) 		
	Índice de matriz	Tipos de datos índice válidos	Reglas para índice de matriz
Constante o variable	USInt, SInt, UInt, Int, UDInt, DInt	<ul style="list-style-type: none"> • Límites de valores: -32768 a +32767 • Válido: Constantes y variables mezcladas • Válido: Expresiones constantes • No válido: Expresiones variables 	

Ejemplo:	ARRAY[1..20] of REAL	Una dimensión, 20 elementos
Declaraciones de matriz	ARRAY[-5..5] of INT	Una dimensión, 11 elementos
	ARRAY[1..2, 3..4] of CHAR	Dos dimensiones, 4 elementos
Ejemplo:	ARRAY1[0]	ARRAY1 elemento 0
Direcciones de matriz	ARRAY2[1,2]	ARRAY2 elemento [1,2]
	ARRAY3[i,j]	Si i =3 y j=4, entonces se direcciona ARRAY3 elemento [3, 4]

4.4.7 Tipo de datos Estructura de datos

Se puede utilizar el tipo de datos "Struct" para definir una estructura de datos formada por otros tipos de datos. El tipo de datos STRUCT puede emplearse para gestionar un grupo de datos de proceso relacionados como una unidad de datos simple. Se asigna un nombre a un tipo de datos STRUCT y la estructura de datos interna se declara en el editor de bloques de datos o un editor de interfaces de bloque.

Las matrices y estructuras también se pueden combinar en una estructura más grande. Se puede anidar una estructura hasta ocho niveles de profundidad. Por ejemplo, se puede crear una estructura de estructuras con matrices.

Una variable STRUCT comienza en una dirección de bytes pares y utiliza la memoria del siguiente grupo de palabras.

4.4.8 Tipo de datos PLC

El editor del tipo de datos PLC permite definir estructuras de datos, que pueden usarse varias veces en el programa. Para crear un tipo de datos PLC abra la rama "Tipos de datos PLC" del árbol del proyecto y haga doble clic en el elemento "Añadir nuevo tipo de datos". En el tipo de datos PLC recién creado, haga dos clics individuales para cambiar el nombre predeterminado y un doble clic para abrir el editor del tipo de datos PLC.

Para crear una estructura de tipo de datos PLC personalizada se utilizan los mismos métodos de edición que se utilizan en el editor de bloques de datos. Agregue nuevas filas para los tipos de datos que sean necesarios para crear la estructura de datos deseada.

Cuando se crea un nuevo tipo de datos PLC, el nombre del nuevo tipo PLC aparece en las listas de selección de tipo de datos del editor de DB y en el editor de interfaces de bloque lógico.

Usos potenciales de tipos de datos PLC:

- Los tipos de datos PLC pueden usarse directamente como tipo de datos en una interfaz de bloques lógicos o en bloques de datos.
- Los tipos de datos PLC pueden emplearse como plantilla para la creación de varios bloques de datos globales que usen la misma estructura de datos.

Por ejemplo, un tipo de datos PLC puede ser una receta de colores mezclados. Así, es posible asignar este tipo de datos PLC a varios bloques de datos. De ese modo, cada bloque de datos puede tener las variables ajustadas para crear un color específico.

4.4.9 Tipos de datos de puntero

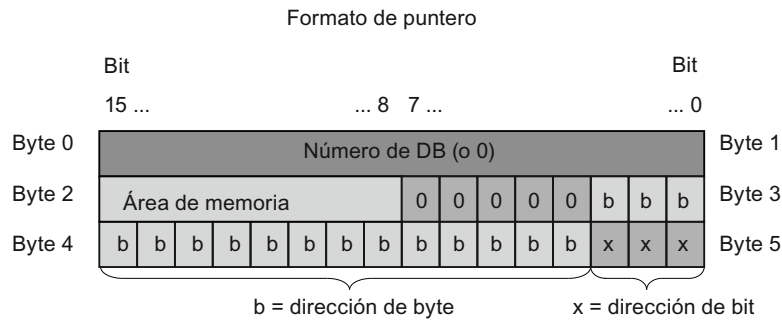
Los tipos de datos de puntero (Pointer, Any y Variant) pueden utilizarse en tablas de interfaz de bloque para bloques lógicos FB y FC. El tipo de datos de puntero se puede seleccionar de las listas desplegables de tipos de datos de interfaz de bloque.

El tipo de datos Variant también se utiliza para parámetros de instrucción.

4.4.9.1 Tipo de datos de puntero "Pointer"

El tipo de datos Pointer apunta a una variable en particular. Ocupa 6 bytes (48 bits) en la memoria y puede incluir la información siguiente:

- Número de DB o 0 si los datos no se guardan en un DB
- Área de almacenamiento en la CPU
- Dirección de la variable



Dependiendo de la instrucción, se pueden declarar los siguientes tres tipos de puntero:

- Puntero interno de área: Contiene información sobre la dirección de una variable
- Puntero inter-área: Contiene información sobre el área de memoria y la dirección de una variable
- Puntero DB: Contiene un número de bloque de datos y la dirección de una variable

Tabla 4- 26 Tipos de punteros:

Tipo	Formato	Ejemplo de entrada:
Puntero interno de área	P#Byte.Bit	P#20.0
Puntero inter-área	P#Área_memoria_Byte.Bit	P#M20.0
Puntero DB:	P#Bloque_datos.Elemento_datos	P#DB10.DBX20.0

Se puede introducir un parámetro del tipo Pointer sin el prefijo (P #). La entrada se convierte automáticamente al formato de puntero.

Tabla 4- 27 Encriptación de área de memoria en la información Pointer:

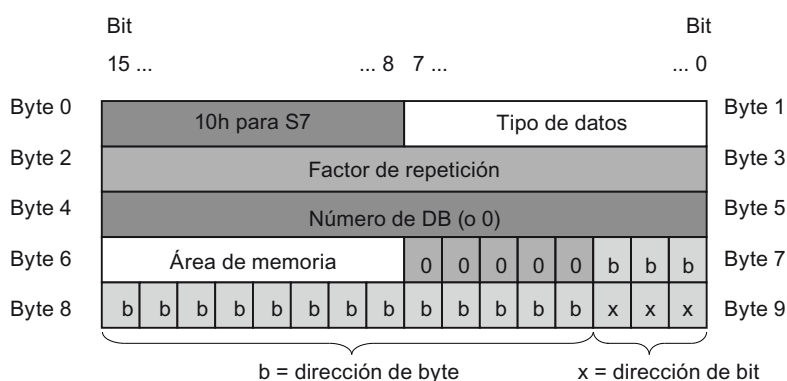
Código hexadecimal	Tipo de datos	Descripción
b#16#81	I	Área de memoria de las entradas
b#16#82	Q	Área de memoria de las salidas
b#16#83	M	Área de memoria de las marcas
b#16#84	DBX	Bloque de datos
b#16#85	DIX	Bloque de datos de instancia
b#16#86	L	Datos locales
b#16#87	V	Datos locales anteriores

4.4.9.2 Tipo de datos de puntero "Any"

El tipo de datos de puntero ANY ("Any") apunta al inicio de un área de datos y especifica su longitud. El puntero de ANY usa 10 bytes de la memoria y puede incluir la información siguiente:

- Tipo de datos: Tipo de datos de los elementos de datos
- Factor de repetición: Número de elementos de datos
- Número de DB: Bloque de datos en el que se guardan los elementos de datos
- Área de almacenamiento: Área de memoria de la CPU en la que se guardan los elementos de datos
- Dirección inicial: Dirección de inicio "Byte.Bit" de los datos

En la figura siguiente se muestra la estructura del puntero ANY:



Un puntero no puede detectar estructuras ANY. Sólo puede asignarse a variables locales.

Tabla 4- 28 Formato y ejemplos del puntero ANY:

Formato	Ejemplo de entrada	Descripción
P#Bloque_datos.Área_memoria Dirección_datos Tipo Número	P#DB 11.DBX 20.0 INT 10	10 palabras en DB 11 global comenzando por DBB 20.0
P#Área_memoria Dirección_datos Tipo Número	P#M 20.0 BYTE 10	10 bytes comenzando por MB 20.0
	P#I 1.0 BOOL 1	Entrada I1.0

Tabla 4- 29 Encriptación de tipo de datos en el puntero ANY

Código hexadecimal	Tipo de datos	Descripción
b#16#00	Null	Puntero NULL
b#16#01	Bool	Bits
b#16#02	Byte	Bytes, 8 bits
b#16#03	Char	Carácter de 8 bits
b#16#04	Word	Palabra de 16 bits
b#16#05	Int	Entero de 16 bits

4.4 Tipos de datos

Código hexadecimal	Tipo de datos	Descripción
b#16#37	SInt	Entero de 8 bits
b#16#35	UInt	Entero de 16 bits sin signo
b#16#34	USInt	Entero de 8 bits sin signo
b#16#06	DWord	Palabra doble de 32 bits
b#16#07	DInt	Entero doble de 32 bits
b#16#36	UDInt	Entero doble de 32 bits sin signo
b#16#08	Real	32 bits en coma flotante
b#16#0B	Time	Hora
b#16#13	String	Cadena de caracteres

Tabla 4- 30 Encriptación de área de memoria en el puntero ANY:

Código hexadecimal	Área de memoria	Descripción
b#16#81	I	Área de memoria de las entradas
b#16#82	Q	Área de memoria de las salidas
b#16#83	M	Área de memoria de las marcas
b#16#84	DBX	Bloque de datos
b#16#85	DIX	Bloque de datos de instancia
b#16#86	L	Datos locales
b#16#87	V	Datos locales anteriores

4.4.9.3 Tipo de datos de puntero "Variant"

El tipo de datos Variant puede apuntar a variables de diferentes tipos de datos o parámetros. El puntero Variant puede apuntar a estructuras y componentes estructurales individuales. El puntero Variant no ocupa ningún espacio en la memoria.

Tabla 4- 31 Propiedades del puntero Variant

Longitud (bytes)	Representación	Formato	Ejemplo de entrada:
0	Simbólica	Operando	MyTag
		Nombre_DB.Nombre_estruct.nombre_elemento	MiDB.Estruct1.presión1
	Absoluta	Operando	%MW10
		Número_DB.Operando Tipo Longitud	P#DB10.DBX10.0 INT 12

4.4.10 Acceder a un "segmento" de un tipo de datos de variable

Es posible acceder al nivel de bit, byte o palabra de las variables PLC y de bloque de datos en función de su tamaño. A continuación aparece la sintaxis para acceder a un segmento de datos de este tipo:

- "<nombre de la variable PLC>".xn (acceso a bit)
- "<nombre de la variable PLC>".bn (acceso a byte)
- "<nombre de la variable PLC>".wn (acceso a palabra)
- "<nombre del bloque de datos>".<nombre de la variable>.xn (acceso a bit)
- "<nombre del bloque de datos>".<nombre de la variable>.bn (acceso a byte)
- "<nombre del bloque de datos>".<nombre de la variable>.wn (acceso a palabra)

A una variable del tamaño de palabra doble se accede a través de los bits 0 - 31, bytes 0 - 3 o palabra 0 - 1. A una variable del tamaño de palabra se accede a través de los bits 0 - 15, bytes 0 - 2 o palabra 0. A una variable del tamaño de byte se accede a través de los bits 0 - 8 o byte 0. Los segmentos bit, byte y palabra se pueden utilizar dondequiera que bits, bytes y palabras estén previstos como operandos.

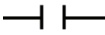

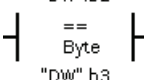
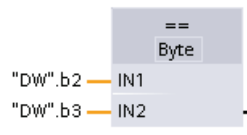
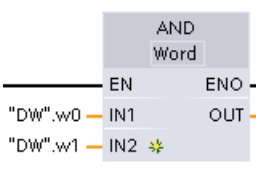
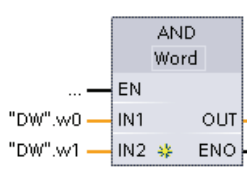
																BYTE															
																WORD															
DWORD																															
x31	x30	x29	x28	x27	x26	x25	x24	x23	x22	x21	x20	x19	x18	x17	x16	x15	x14	x13	x12	x11	x10	x9	x8	x7	x6	x5	x4	x3	x2	x1	x0
b3								b2								b1								b0							
w1																w0															

Nota

Los tipos de datos válidos a los que se puede acceder por segmento son Byte, Char, Conn_Any, Date, DInt, DWord, Event_Any, Event_Att, Hw_Any, Hw_Device, HW_Interface, Hw_Io, Hw_Pwm, Hw_SubModule, Int, OB_Any, OB_Att, OB_Cyclic, OB_Delay, OB_WHINT, OB_PCYCLE, OB_STARTUP, OB_TIMEERROR, OB_Tod, Port, Rtm, SInt, Time, Time_Of_Day, UDInt, UInt, USInt, y Word. A las variables PLC del tipo Real se puede acceder por segmento pero no a las variables de bloque de datos del tipo Real.

Ejemplos

En la tabla de variables PLC, "DW" es una variable declarada del tipo DWORD. Los ejemplos muestran el acceso al segmento bit, byte y palabra:

	KOP	FUP	SCL
Acceso a bit	<p>"DW".x11</p> 		<pre>IF "DW".x11 THEN ... END_IF;</pre>
Acceso a byte	<p>"DW".b2</p>  <p>"DW".b3</p>		<pre>IF "DW".b2 = "DW".b3 THEN ... END_IF;</pre>
Acceso a palabra			<pre>out:= "DW".w0 AND "DW".w1;</pre>

Consulte también

SCL (Página 160)

4.4.11 Acceso a una variable con una superposición de AT

La superposición de la variable AT permite acceder a una variable ya declarada de un bloque de acceso estándar con una declaración superpuesta de un tipo de datos diferente. Por ejemplo, se pueden direccionar los distintos bits de una variable de tipo Byte, Word o DWord con una matriz de boolean.

Declaración

Para superponer un parámetro, declare un parámetro adicional directamente después del parámetro que se va a superponer y seleccione el tipo de datos "AT". El editor crea la superposición y, a continuación, se puede elegir el tipo de datos, la estructura o la matriz que se desea utilizar para la superposición.

Ejemplo

En este ejemplo se muestran los parámetros de entrada de un FB de acceso estándar. La variable de byte B1 se superpone con una matriz de boolean:

■	B1	Byte
▼	AT	AT "B1" Array [0..7] of Bool
■	AT[0]	Bool
■	AT[1]	Bool
■	AT[2]	Bool
■	AT[3]	Bool
■	AT[4]	Bool
■	AT[5]	Bool
■	AT[6]	Bool
■	AT[7]	Bool

Tabla 4- 32 Superposición de un byte con una matriz de boolean

7	6	5	4	3	2	1	0
AT[0]	AT[1]	AT[2]	AT[3]	AT[4]	AT[5]	AT[6]	AT[7]

Otro ejemplo es una variable DWord superpuesta con una Struct:

■	DW1	DWord
▼	DW1_Struct	AT "DW1" Struct
■	S1	Word
■	S2	Byte
■	S3	Byte

Los tipos de superposición se pueden direccionar directamente en la lógica del programa:

KOP	FUP	SCL
		<pre>IF #AT[1] THEN ... END_IF;</pre>
		<pre>IF (#DW1_Struct.S1 = W#16#000C) THEN ... END_IF;</pre>
		<pre>out1 := #DW1_Struct.S2;</pre>

Reglas

- La superposición de variables sólo es posible en bloques FB y FC con acceso estándar.
- Se pueden superponer parámetros de todos los tipos de bloques y secciones de declaración.
- Un parámetro superpuesto se puede utilizar como cualquier otro parámetro de bloque.
- No se pueden superponer parámetros de tipo VARIANT.
- El tamaño del parámetro que se superpone debe ser menor o igual que el tamaño del parámetro superpuesto.
- La variable que se superpone se debe declarar inmediatamente después de la variable que queda superpuesta e identificar con la palabra clave "AT".

Consulte también

SCL (Página 160)

4.5 Utilizar una Memory Card

ATENCIÓN
La CPU sólo soporta la SIMATIC Memory Card (Página 872) preformateada. Antes de copiar cualquier programa en la Memory Card formateada, borre todo programa almacenado previamente en ella.

Utilice la Memory Card como tarjeta de transferencia o de programa. Todo programa que se copie en la Memory Card contendrá todos los bloques lógicos y de datos, los objetos tecnológicos y la configuración del dispositivo. Un programa copiado **no** contendrá los valores de forzado permanente.

- Utilice una tarjeta de transferencia (Página 114) para copiar un programa en la memoria de carga interna de la CPU sin usar STEP 7. Una vez insertada la tarjeta de transferencia, la CPU primero borra el programa de usuario y cualquier valor de forzado permanente de la memoria de carga interna y, después, copia el programa de la tarjeta de transferencia en la memoria de carga interna. Tras finalizar la transferencia es preciso extraer la tarjeta de transferencia.

Una tarjeta de transferencia vacía puede utilizarse para acceder a una CPU protegida por contraseña si se ha perdido u olvidado la contraseña (Página 122). Cuando se inserta una tarjeta de transferencia vacía, se borra el programa protegido por contraseña en la memoria de carga interna de la CPU. Luego es posible cargar un programa nuevo en la CPU.

- Utilice una tarjeta de programa (Página 116) como memoria de carga externa para la CPU. Cuando se inserta una tarjeta de programa, se borra toda la memoria de carga interna de la CPU (el programa de usuario y los valores de forzado permanente). La CPU ejecuta luego el programa en la memoria de carga externa (la tarjeta de programa). Si se realiza una carga en una CPU que tenga insertada una tarjeta de programa, se actualizará sólo la memoria de carga externa (la tarjeta de programa).

Puesto que la memoria de carga interna de la CPU se borró cuando se insertó la tarjeta de programa, ésta **debe** permanecer en la CPU. Si se extrae la tarjeta de programa, la CPU pasará a estado operativo STOP. (El LED de error parpadea para indicar que se ha extraído la tarjeta de programa.)

El programa copiado en una Memory Card incluye los bloques lógicos y de datos, los objetos tecnológicos y la configuración del dispositivo. La Memory Card **no** contiene valores de forzado permanente. Los valores de forzado permanente no forman parte del programa, pero se almacenan en la memoria de carga interna (en la CPU) o externa (en una tarjeta de programa). Si se inserta una tarjeta de programa en la CPU, STEP 7 aplicará los valores de forzado permanente sólo a la memoria de carga externa en la tarjeta de programa.

También se utiliza una Memory Card para descargar actualizaciones de firmware (Página 119).

4.5.1 Insertar una Memory Card en la CPU

PRECAUCIÓN

Las descargas electrostáticas pueden deteriorar la Memory Card o la ranura para tarjetas en la CPU.

Al manejar la Memory Card deberá estar en contacto con una superficie conductiva puesta a tierra y/o llevar una muñequera antiestática. Guarde la Memory Card en una caja conductiva.



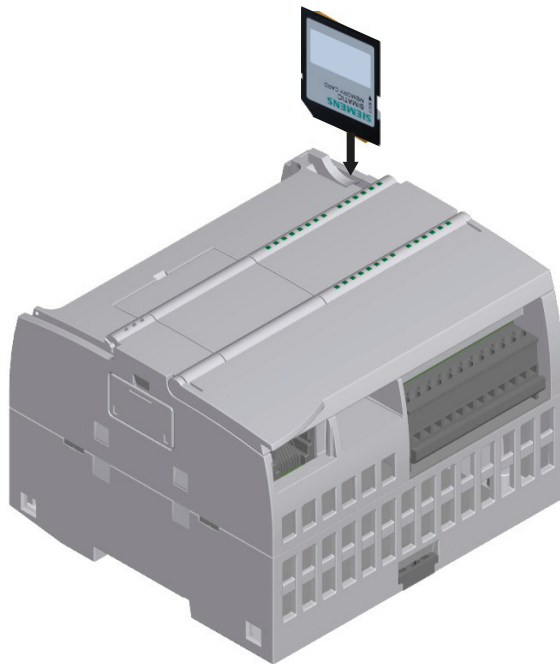
Asegúrese de que la Memory Card no está protegida contra escritura. Deslice el interruptor de protección fuera de la posición "Lock".

PRECAUCIÓN

Si se inserta una Memory Card (configurada como tarjeta de programa o como tarjeta de transferencia) en una CPU que esté funcionando, la CPU pasará inmediatamente a STOP, lo que podría causar daños en el equipo o en el proceso que se está controlando. Antes de insertar o extraer una Memory Card, asegúrese siempre de que la CPU no está controlando activamente una máquina o un proceso. Prevea siempre un circuito de parada de emergencia para la aplicación o proceso.

Nota

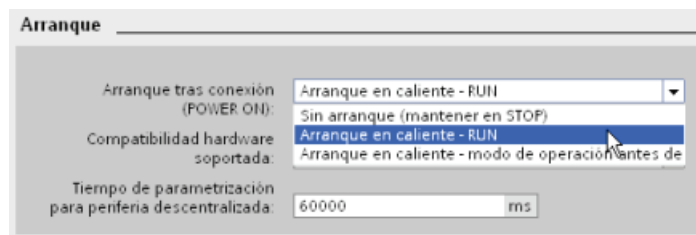
Si se inserta una Memory Card estando la CPU en estado operativo STOP, el búfer de diagnóstico mostrará un mensaje de que se ha iniciado la evaluación de la Memory Card. La CPU evaluará la Memory Card la próxima vez que la CPU conmute al estado operativo RUN, se realice un borrado total de la CPU (MRES) o se desconecte y vuelva a conectar la alimentación de la CPU.



Para insertar una Memory Card, abra la tapa superior de la CPU e inserte la Memory Card en la ranura. Un conector de trinquete facilita la inserción y extracción. La Memory Card está diseñada de manera que pueda insertarse en un único sentido.

4.5.2 Configurar los parámetros de arranque de la CPU antes de copiar el proyecto en la Memory Card

Cuando un programa se copia en una tarjeta de transferencia o de programa, incluye los parámetros de arranque de la CPU. Antes de copiar el programa en la Memory Card, asegúrese de que se ha configurado el estado operativo de la CPU posterior a la desconexión y conexión de la alimentación. Seleccione si la CPU debe arrancar en estado operativo STOP o RUN, o bien en el estado operativo anterior (a la desconexión y conexión de la alimentación).



4.5.3 Utilizar la Memory Card como tarjeta de "Transferencia"

PRECAUCIÓN

Las descargas electroestáticas pueden deteriorar la Memory Card o la ranura para tarjetas en la CPU.

Cuando maneje la Memory Card deberá estar en contacto con una superficie conductiva puesta a tierra o llevar una muñequera antiestática. Guarde la Memory Card en una caja conductiva.

Crear una tarjeta de transferencia

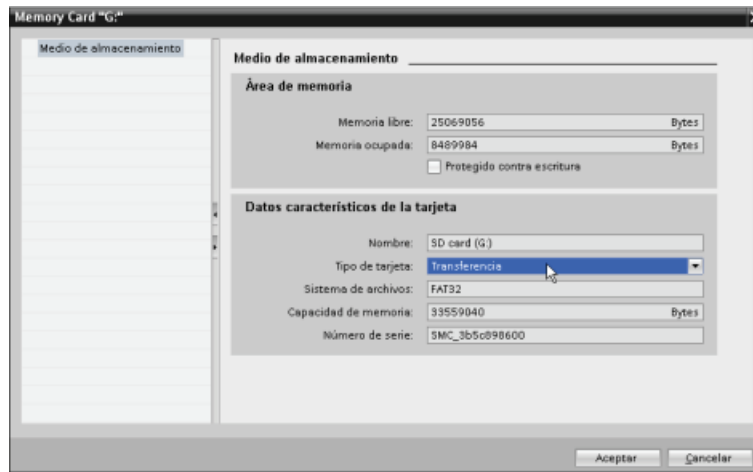
Recuerde siempre que es necesario configurar los parámetros de arranque de la CPU (Página 113) antes de copiar un programa en la tarjeta de transferencia. Para crear una tarjeta de transferencia, proceda del siguiente modo:

1. Inserte una Memory Card SIMATIC vacía en el lector/grabador de tarjetas conectado al PC.

Si se reutiliza una Memory Card SIMATIC que contiene un programa de usuario u otra actualización de firmware, **es necesario** borrar los archivos de programa antes de reutilizarla. Utilice el Explorador de Windows para visualizar el contenido de la Memory Card y borre el archivo "S7_JOB.S7S" y cualquier carpeta de registros de datos o carpeta (directorio) que haya (p. ej. "SIMATIC.S7S" o "FWUPDATE.S7S").
2. En el árbol del proyecto (vista del proyecto), expanda la carpeta "SIMATIC Card Reader" y seleccione el lector de tarjetas deseado.
3. Para abrir el cuadro de diálogo "Memory Card", haga clic con el botón derecho del ratón en la letra de unidad correspondiente a la Memory Card en el lector de tarjetas y elija el comando "Propiedades" del menú contextual.

4. En el cuadro de diálogo "Memory Card", seleccione "Transferencia" en el menú desplegable "Tipo de tarjeta".

A continuación, STEP 7 crea una tarjeta de transferencia vacía. Si está creando una tarjeta de transferencia vacía p. ej. para realizar una recuperación tras olvidar la contraseña de la CPU (Página 122), extraiga la tarjeta de transferencia del lector de tarjetas.



5. Agregue el programa seleccionando la CPU (p. ej. PLC_1 [CPU 1214 DC/DC/DC]) en el árbol del proyecto y arrastrándola hasta la Memory Card. (Como alternativa, copie la CPU e insértela en la Memory Card.) Cuando la CPU se copia en la Memory Card se abre el diálogo "Cargar vista preliminar".
6. En el diálogo "Cargar vista preliminar", haga clic en el botón "Cargar" para copiar la CPU en la Memory Card.
7. Cuando aparezca un mensaje indicando que la CPU (el programa) se ha cargado sin errores, haga clic en el botón "Finalizar".

Utilizar una tarjeta de transferencia

ADVERTENCIA

Verifique que la CPU no está ejecutando ningún proceso en ese momento antes de insertar la Memory Card.

La inserción de una Memory Card provocará el paso de la CPU a STOP, lo que podría afectar al funcionamiento de un proceso online o una máquina. El manejo inesperado de un proceso o una máquina podría provocar lesiones o incluso la muerte de personas y/o daños materiales.

Antes de insertar una Memory Card, asegúrese siempre de que la CPU está offline y en un estado seguro.

4.5 Utilizar una Memory Card

Para transferir el programa a una CPU, proceda del siguiente modo:

1. Inserte la tarjeta de transferencia en la CPU (Página 112). Si la CPU está en RUN, pasará a estado operativo STOP. El LED de mantenimiento (MAINT) parpadea para indicar que es necesario revisar la Memory Card.
2. Desconecte y vuelva a conectar la CPU para revisar la Memory Card. Otros métodos alternativos de rearmar la CPU consisten en hacer una transición de STOP a RUN o inicializar la memoria (MRES) desde STEP 7.
3. Tras rearmar y evaluar la Memory Card, la CPU copiará el programa en la memoria de carga interna de la CPU.

El LED RUN/STOP parpadea de forma alterna en verde y amarillo para indicar que el programa se está copiando. Cuando el LED RUN/STOP se enciende (en amarillo permanente) y el LED MAINT parpadea, el proceso de copia ha finalizado. Ahora ya se puede extraer la Memory Card.

4. Vuelva a arrancar la CPU (ya sea restableciendo la alimentación o con los métodos de rearmar alternativos) para evaluar el nuevo programa transferido a la memoria de carga interna.

La CPU pasa entonces al modo de arranque (RUN o STOP) configurado para el proyecto.

Nota

Extraiga la tarjeta de transferencia antes de cambiar la CPU a estado operativo RUN.

4.5.4 Utilizar la Memory Card como tarjeta de "Programa"

PRECAUCIÓN

Las descargas electrostáticas pueden deteriorar la Memory Card o la ranura para tarjetas en la CPU.

Al manejar la Memory Card deberá estar en contacto con una superficie conductiva puesta a tierra y/o llevar una muñequera antiestática. Guarde la Memory Card en una caja conductiva.



Asegúrese de que la Memory Card no está protegida contra escritura. Deslice el interruptor de protección fuera de la posición "Lock".

Antes de copiar elementos de programa en la tarjeta de programa, borre todo programa almacenado previamente en ella.

Crear una tarjeta de programa

Si se utiliza como tarjeta de programa, la Memory Card es la memoria de carga externa de la CPU. Si se extrae la tarjeta de programa, la memoria de carga interna de la CPU estará vacía.

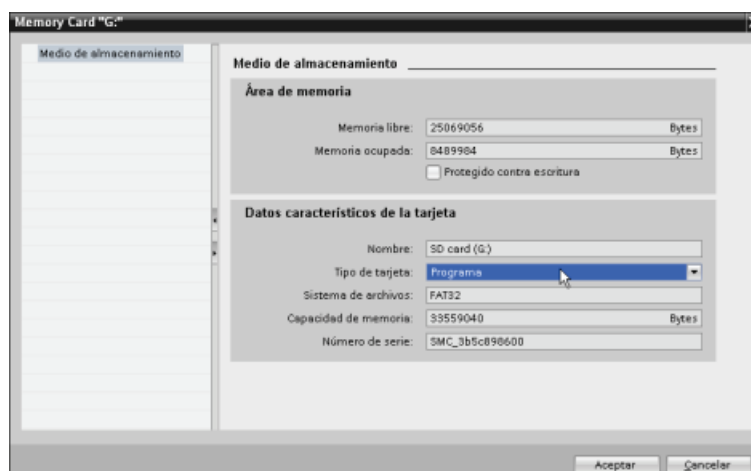
Nota

Si una Memory Card vacía se inserta en la CPU y se evalúa desconectando y conectando la alimentación de la CPU, cambiando el estado operativo de STOP a RUN o realizando un borrado total (MRES), el programa y los valores de forzado permanente contenidos en la memoria de carga interna de la CPU se copiarán en la Memory Card. (La Memory Card se convierte entonces en una tarjeta de programa.) Una vez finalizada la operación de copia, se borrará el programa en la memoria de carga interna de la CPU. La CPU pasa entonces al modo de arranque (RUN o STOP) configurado.

Recuerde siempre que es necesario configurar los parámetros de arranque de la CPU (Página 113) antes de copiar el proyecto en la tarjeta de programa. Para crear una tarjeta de programa, proceda del siguiente modo:

1. Inserte una Memory Card SIMATIC vacía en el lector/grabador de tarjetas conectado al PC.


Si se reutiliza una Memory Card SIMATIC que contiene un programa de usuario u otra actualización de firmware, **es necesario** borrar los archivos de programa antes de reutilizarla. Utilice el Explorador de Windows para visualizar el contenido de la Memory Card y borre el archivo "S7_JOB.S7S" y cualquier carpeta de registros de datos o carpeta (directorio) que haya (p. ej. "SIMATIC.S7S" o "FWUPDATE.S7S").
2. En el árbol del proyecto (vista del proyecto), expanda la carpeta "SIMATIC Card Reader" y seleccione el lector de tarjetas deseado.
3. Para abrir el cuadro de diálogo "Memory Card", haga clic con el botón derecho del ratón en la letra de unidad correspondiente a la Memory Card en el lector de tarjetas y elija el comando "Propiedades" del menú contextual.
4. En el diálogo "Memory Card", seleccione "Programa" en la lista desplegable.



4.5 Utilizar una Memory Card

5. Agregue el programa seleccionando la CPU (p. ej. PLC_1 [CPU 1214 DC/DC/DC]) en el árbol del proyecto y arrastrándola hasta la Memory Card. (Como alternativa, copie la CPU e insértela en la Memory Card.) Cuando la CPU se copia en la Memory Card se abre el diálogo "Cargar vista preliminar".
6. En el diálogo "Cargar vista preliminar", haga clic en el botón "Cargar" para copiar la CPU en la Memory Card.
7. Cuando aparezca un mensaje indicando que la CPU (el programa) se ha cargado sin errores, haga clic en el botón "Finalizar".

Utilizar una tarjeta de programa como memoria de carga para la CPU


 ADVERTENCIA
Verifique que la CPU no está ejecutando ningún proceso en ese momento antes de insertar la Memory Card.
La inserción de una Memory Card provocará el paso de la CPU a STOP, lo que podría afectar al funcionamiento de un proceso online o una máquina. El manejo inesperado de un proceso o una máquina podría provocar lesiones o incluso la muerte de personas y/o daños materiales.
Antes de insertar una Memory Card, asegúrese siempre de que la CPU está offline y en un estado seguro.

Para utilizar una tarjeta de programa en la CPU, proceda del siguiente modo:

1. Inserte la tarjeta de programa en la CPU. Si la CPU está en RUN, pasará a estado operativo STOP. El LED de mantenimiento (MAINT) parpadea para indicar que es necesario revisar la Memory Card.
2. Desconecte y vuelva a conectar la CPU para revisar la Memory Card. Otros métodos alternativos de reanunciar la CPU consisten en hacer una transición de STOP a RUN o inicializar la memoria (MRES) desde STEP 7.
3. Una vez que la CPU ha reanunciado y evaluado la tarjeta de programa, borra su memoria de carga interna.

La CPU pasa entonces al modo de arranque (RUN o STOP) configurado para ella.

La tarjeta de programa debe permanecer en la CPU. Si se extrae la tarjeta de programa, la memoria de carga interna de la CPU no contendrá ningún programa.

 ADVERTENCIA
Si se extrae la tarjeta de programa, la CPU perderá su memoria de carga externa y generará un error. La CPU pasa a estado operativo STOP y el LED de error parpadea.
Los dispositivos de control pueden fallar y provocar condiciones no seguras, causando a su vez reacciones inesperadas de los equipos controlados. Las reacciones inesperadas podrían producir la muerte, lesiones corporales graves y/o daños materiales.

4.5.5 Actualización de firmware

PRECAUCIÓN

Las descargas electrostáticas pueden deteriorar la Memory Card o la ranura para tarjetas en la CPU.

Cuando maneje la Memory Card deberá estar en contacto con una superficie conductiva puesta a tierra o llevar una muñequera antiestática. Guarde la Memory Card en una caja conductiva.

Utilice una Memory Card para descargar actualizaciones de firmware del Customer Support (<http://www.siemens.com/automation/>). Desde este sitio web, vaya a **Tecnología de automatización > Sistemas de automatización > Sistemas de automatización industrial SIMATIC > Controladores > Modular controllers SIMATIC S7 > SIMATIC S7-1200**. Desde aquí, siga navegando hasta llegar al tipo específico de módulo que debe actualizar. En "Support" (Soporte), haga clic en el link "Descarga de Software" para continuar.

Como alternativa, puede acceder directamente a la página web de descargas de S7-1200 (<http://support.automation.siemens.com/WW/view/es/34612486/133100>).

Nota

No puede actualizar una CPU S7-1200 V2.2 o anterior a S7-1200 V3.0 mediante la actualización de firmware.

PRECAUCIÓN

No utilice la utilidad de formateo de Windows o cualquier otra utilidad de formateo para reformatear la tarjeta de memoria.

Si se vuelve a formatear una Siemens Memory Card con la utilidad de formateo de Microsoft Windows, entonces la tarjeta de memoria ya no podrá utilizarse en una CPU S7-1200.

Para descargar la actualización de firmware en la Memory Card, siga los siguientes pasos:

1. Inserte una tarjeta SIMATIC MC 24 MB virgen en una unidad de lectura/escritura de tarjetas SD conectada al equipo.

Es posible reutilizar una SIMATIC Memory Card que contenga un programa de usuario u otra actualización de firmware, pero es necesario borrar algunos archivos de la Memory Card.

PRECAUCIÓN


NO borrar los archivos ocultos "__LOG__" y "crdinfo.bin" de la Memory Card.
--

Los archivos "__LOG__" y "crdinfo.bin" son necesarios para la Memory Card. Si borra estos archivos, no podrá utilizar la Memory Card con la CPU.
--

Para reutilizar una Memory Card, **es necesario** borrar el archivo "S7_JOB.S7S" y todas las carpetas de registro de datos o cualquier otra carpeta (como "SIMATIC.S7S" o "FWUPDATE.S7S") antes de descargar la actualización del firmware. No obstante, **no** borre los archivos "__LOG__" y "crdinfo.bin". (Estos archivos están ocultos y son necesarios.) Utilice Windows Explorer para visualizar el contenido de la Memory Card y borrar el archivo y las carpetas.

2. Seleccione el archivo autoextraíble (.exe) para actualizar el firmware que corresponde al módulo utilizado y descárguelo en su equipo. Haga doble clic en el archivo de actualización, indique la ruta de destino del archivo de tal modo que sea el directorio raíz de la SIMATIC Memory Card e inicie el proceso de extracción. Una vez finalizada la extracción, el directorio raíz (carpeta) de la Memory Card contendrá un directorio "FWUPDATE.S7S" y el archivo "S7_JOB.S7S".

Para instalar la actualización de firmware, proceda del siguiente modo:

 ADVERTENCIA
Antes de instalar la actualización de firmware, compruebe que la CPU no esté ejecutando activamente ningún proceso.
Al instalar la actualización de firmware, la CPU pasará a STOP, lo que puede afectar la operación de un proceso online o de una máquina. La operación inesperada de un proceso o de una máquina puede producir la muerte, lesiones corporales graves y/o daños materiales.
Antes de insertar la tarjeta de memoria, asegúrese de que la CPU esté offline y en un estado seguro.

1. Inserte la tarjeta de memoria en la CPU. Si la CPU está en RUN, pasará al estado operativo STOP. El LED de mantenimiento (MAINT) parpadea para indicar que debe evaluarse la tarjeta de memoria.
2. Apague y vuelva a encender la alimentación de la CPU para iniciar la actualización de firmware. Otras formas de rearrancar la CPU son ejecutando una transición de STOP a RUN o borrando la memoria (MRES) desde STEP 7.

ATENCIÓN
Para finalizar la actualización del firmware del módulo, asegúrese de que la alimentación externa de 24 DC del módulo permanece conectada.

Después de rearrancar la CPU, inicia la actualización del firmware. El LED RUN/STOP parpadea de forma alterna en verde y amarillo, indicando que la actualización se está copiando. Cuando el LED RUN/STOP se encienda de forma permanente (amarillo) y el LED MAINT parpadee, el proceso de copia habrá finalizado. A continuación se debe extraer la tarjeta de memoria.

3. Después de extraer la tarjeta de memoria, re arranque la CPU (restableciendo la alimentación o bien aplicando otro método) para cargar el nuevo firmware.

4.6 Recuperación si se olvida la contraseña

El programa de usuario y la configuración hardware no se ven afectados por la actualización de firmware. Al poner en marcha la CPU, ésta adopta el estado de arranque configurado. (Si el modo de arranque de la CPU estaba configurado como "Modo de re arranque en caliente antes de POWER OFF", la CPU se encontrará en el estado operativo STOP porque el último estado de la CPU era STOP).

Nota

Actualizar varios módulos conectados a la CPU

Si su configuración hardware contiene varios módulos que corresponden a un solo archivo de actualización de firmware en la Memory Card, la CPU aplica las actualizaciones a todos los módulos aplicables (CM, SM, SB) en el orden de configuración, es decir, en orden ascendente según la posición del módulo en la configuración de dispositivos de STEP 7.

Si ha descargado varias actualizaciones de firmware en la Memory Card para varios módulos, la CPU aplica las actualizaciones en el orden en que se han descargado en la Memory Card.

4.6 Recuperación si se olvida la contraseña

Si se ha olvidado la contraseña de una CPU protegida por contraseña, es preciso utilizar una tarjeta de transferencia vacía para borrar el programa protegido por contraseña. La tarjeta de transferencia vacía borra la memoria de carga interna de la CPU. Entonces es posible cargar un programa de usuario nuevo desde STEP 7 a la CPU.

Encontrará más información sobre cómo crear y utilizar una tarjeta de transferencia vacía en el apartado Tarjeta de transferencia (Página 114).

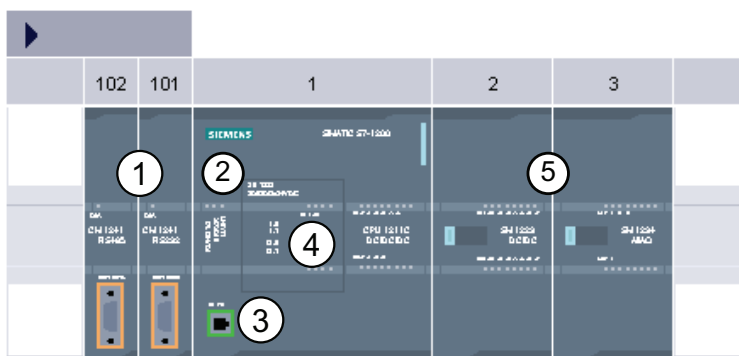
 **ADVERTENCIA**

Si se inserta una tarjeta de transferencia en una CPU que esté funcionando, la CPU pasará a STOP. Los dispositivos de control pueden fallar y provocar condiciones no seguras, causando a su vez reacciones inesperadas de los equipos controlados. Las reacciones inesperadas podrían producir la muerte, lesiones corporales graves y/o daños materiales.

Extraiga la tarjeta de transferencia antes de cambiar la CPU a estado operativo RUN.

Configuración de dispositivos

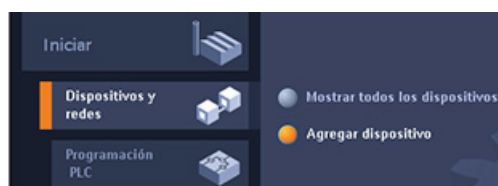
Para crear la configuración de dispositivos del PLC es preciso agregar una CPU y módulos adicionales al proyecto.



- ① Módulo de comunicación (CM) o procesador de comunicaciones (CP): máx. 3, insertados en los slots 101, 102 y 103
- ② CPU: Slot 1
- ③ Puerto Ethernet de la CPU
- ④ Signal Board (SB), Communication Board (CB) o Battery Board (BB): máx. 1, insertada en la CPU
- ⑤ Módulo de señales (SM) para E/S digitales o analógicas: máx. 8, insertados en los slots 2 a 9 (la CPU 1214C y la CPU 1215C permiten 8, la CPU 1212C permite 2, la CPU 1211C no permite ninguno)

Para crear la configuración de dispositivos, agregue un dispositivo al proyecto.

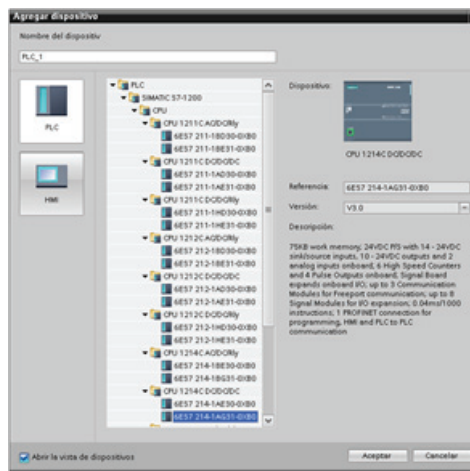
- En la vista de portal, seleccione "Dispositivos y redes" y haga clic en "Agregar dispositivo".
- En la vista del proyecto, bajo el nombre del proyecto, haga doble clic en "Agregar nuevo dispositivo".



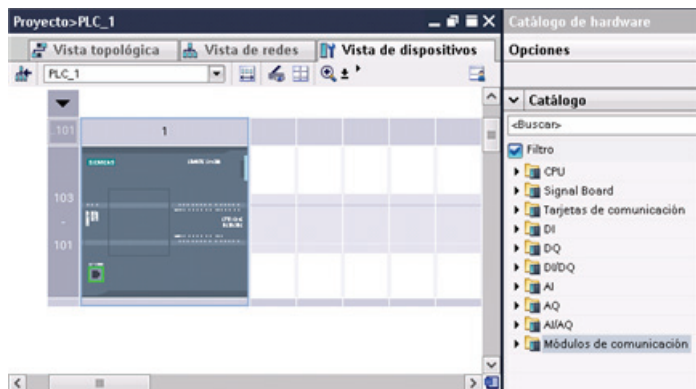
5.1 Insertar una CPU

La configuración de dispositivos se crea insertando una CPU en el proyecto. En la lista, asegúrese de insertar el modelo y la versión de firmware correctos. Al seleccionar la CPU en el diálogo "Agregar nuevo dispositivo" se crean el rack y la CPU.

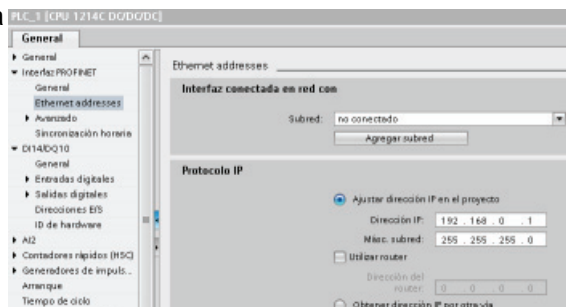
Diálogo "Agregar dispositivo"



Vista de dispositivos de la configuración de hardware



Al seleccionar la CPU en la vista de dispositivos se visualizan las propiedades de la CPU en la ventana de inspección.



Nota

La CPU no tiene una dirección IP preconfigurada. La dirección IP de la CPU se debe asignar manualmente durante la configuración de dispositivos. Si la CPU está conectada a un router de la red, también es preciso introducir la dirección IP del router.

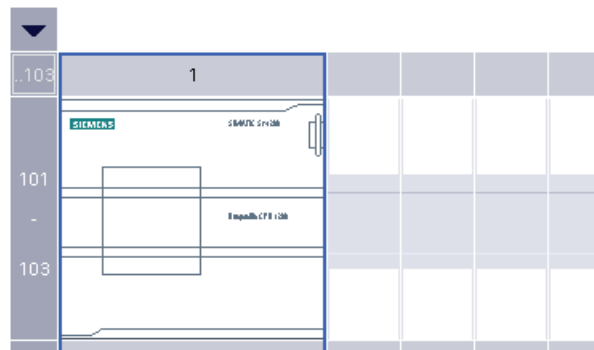
5.2 Detectar la configuración de una CPU sin especificar



Si existe una conexión con una CPU, es posible cargar su configuración en el módulo (incluidos los módulos). Tan sólo hay que crear un proyecto nuevo y seleccionar la "CPU sin especificar" en lugar de una específica. (También es posible omitir la configuración de dispositivo por completo seleccionando "Crear un programa PLC" en "Primeros pasos". Entonces, STEP 7 crea automáticamente una CPU sin especificar).

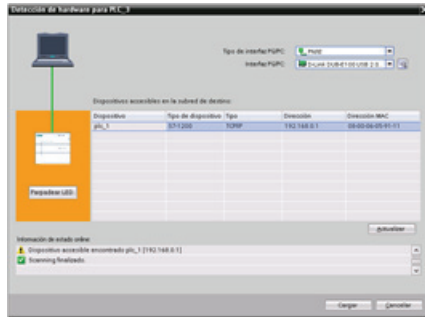
En el editor de programación, seleccione el comando "Detección de hardware" del menú "Online".

En el editor de configuración de dispositivos, seleccione la opción de detección del dispositivo conectado.



El dispositivo no está especificado.
 → Utilice el [catálogo de hardware](#) para especificar la CPU
 → o [determine](#) la configuración del dispositivo conectado.

Tras seleccionar la CPU en el cuadro de diálogo online y pulsar el botón de carga, STEP 7 carga la configuración de hardware de la CPU, incluidos todos los módulos (SM, SB o CM). Entonces pueden configurarse los parámetros de la CPU y de los módulos.





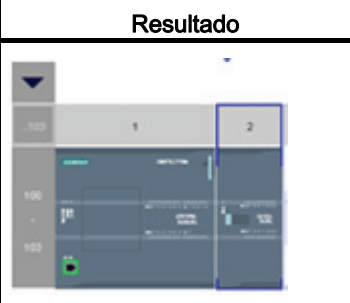



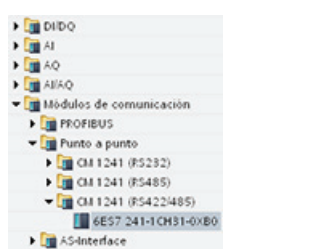

5.3 Agregar módulos a la configuración

El catálogo de hardware se utiliza para agregar módulos a la CPU:

- El módulo de señales (SM) ofrece E/S digitales o analógicas adicionales. Estos módulos se conectan a la derecha de la CPU.
- La Signal Board (SB) ofrece unas pocas E/S adicionales a la CPU. La SB se inserta en el frente de la CPU.
- La nueva BB 1297 ofrece respaldo a largo plazo del reloj en tiempo real. La BB se inserta en la parte frontal de la CPU.
- La placa de comunicación (CB) ofrece un puerto de comunicación adicional (como RS485). La CB se inserta en la parte frontal de la CPU.
- El módulo de comunicación (CM) y el procesador de comunicación (CP) ofrecen un puerto de comunicación adicional, como para PROFIBUS o GPRS. Estos módulos se conectan a la izquierda de la CPU.

Para insertar un módulo en la configuración del dispositivo, selecciónelo en el catálogo de hardware y haga doble clic en él, o bien arrástrelo hasta el slot resaltado. Debe agregar los módulos a la configuración del dispositivo y descargar la configuración de hardware en la CPU para que estén operativos.

Tabla 5- 1 Adición de un módulo a la configuración del dispositivo

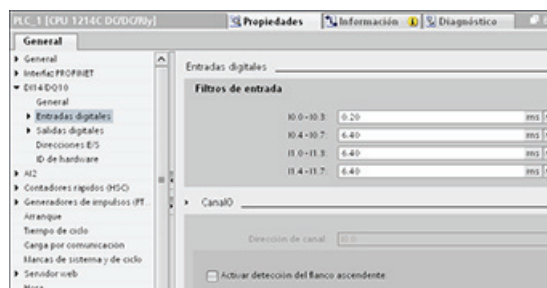
Módulo	Seleccionar el módulo	Insertar el módulo	Resultado
SM			
SB, BB o CB			
CM o CP			

5.4 Configurar el funcionamiento de la CPU

Para configurar los parámetros operativos de la CPU, selecciónela en la vista de dispositivos (recuadro azul alrededor de la CPU) y utilice la ficha "Propiedades" de la ventana de inspección.

Para configurar los tiempos de filtro de entrada, seleccione "Entradas digitales". El tiempo de filtro predeterminado para las entradas digitales es 6,4 ms.

Cada entrada tiene una sola configuración de filtro, aplicable a todos los usos: entradas de proceso, interrupciones, captura de impulsos y entradas de HSC.



Nota

Si un HSC no está configurado para usar una entrada, se aplica el ajuste de filtro seleccionado en esta pantalla. Si un HSC está configurado para usar una entrada, el filtro para esa entrada se ajusta automáticamente a 800 ns y no se ve afectado por la configuración en esta pantalla.

 **ADVERTENCIA**

Si el tiempo de filtro para un canal de entrada digital se reajusta, puede que sea necesario presentar un nuevo valor de entrada de nivel "0" durante un tiempo acumulado de 20 ms para que el filtro esté totalmente operativo ante nuevas entradas. Durante este tiempo, puede que no se detecten o no se cuenten los eventos de pulsación breve "0" cuya duración sea inferior a 20 ms.

Este cambio de los tiempos de filtro puede originar un funcionamiento inesperado de los equipos o del proceso, lo que puede causar la muerte o lesiones graves al personal y/o daños a los equipos.

Para asegurar que un tiempo de filtro nuevo tenga efecto inmediato, desconecte y vuelva a conectar la CPU.

Tabla 5- 2 Propiedades de la CPU

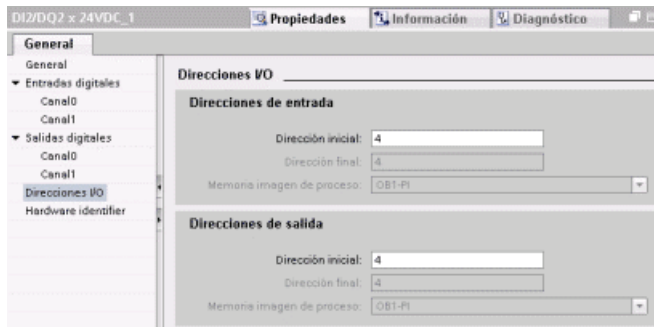
Propiedad	Descripción
Interfaz PROFINET	Permite configurar la dirección IP de la CPU y la sincronización horaria
DI, DO y AI	Permite configurar la reacción de las E/S locales (integradas) digitales y analógicas (por ejemplo, tiempos de filtro de entradas digitales y reacción de las salidas digitales ante una parada de la CPU).
Contadores rápidos (Página 353) y generadores de impulsos (Página 325)	Permite habilitar y configurar los contadores rápidos (HSC) y generadores de impulsos utilizados para las operaciones de tren de impulsos (PTO) y modulación del ancho de impulsos (PWM) Si las salidas de la CPU o Signal Board se configuran como generadores de impulsos (para su utilización con la PWM o con instrucciones de Motion Control), las direcciones de salida correspondientes (Q0.0, Q0.1, Q4.0 y Q4.1) se eliminarán de la memoria Q y no podrán utilizarse para ningún otro fin en el programa de usuario. Si el programa de usuario escribe un valor en una salida utilizada como generador de impulsos, la CPU no escribirá ese valor en la salida física.
Arranque (Página 70)	Arranque tras POWER ON: permite configurar la reacción de la CPU a una transición de OFF a ON, p. ej. el arranque en estado operativo STOP o la transición a RUN tras un arranque en caliente

Propiedad	Descripción
	<p>Compatibilidad de hardware soportada: Configura la estrategia de sustitución para todos los componentes del sistema (SM, SB, CM, CP y CPU):</p> <ul style="list-style-type: none"> • Permitir sustituto aceptable • Permitir cualquier sustituto (valor predeterminado) <p>Cada módulo contiene a nivel interno requisitos de compatibilidad de sustitución según el número de E/S, la compatibilidad eléctrica y otros puntos de comparación que procedan. Por ejemplo, un SM de 16 canales podría ser un sustituto aceptable de un SM de 8 canales, pero un SM de 8 canales no sería un sustituto aceptable para un SM de 16 canales. Si selecciona "Permitir sustituto aceptable", STEP 7 aplica las reglas de sustitución; en caso contrario, STEP 7 permite cualquier sustitución.</p> <p>Tiempo de asignación de los parámetros para E/S distribuidas: configura un tiempo máximo (valor predeterminado: 60000 ms) para las E/S distribuidas que deben conmutarse online. (Los CMs y CPs reciben la alimentación y los parámetros de comunicación de la CPU durante el arranque. El tiempo de asignación permite que las E/S conectadas al CM o CP cambien a online.)</p> <p>La CPU pasa a RUN en cuanto las E/S distribuidas están online, independientemente del tiempo de asignación. Si las E/S distribuidas no se han llevado online dentro de este tiempo, la CPU pasará igualmente a RUN sin las E/S distribuidas.</p> <p>Nota: si se utiliza un CM 1243-5 (maestro PROFIBUS) en la configuración, no debe ajustarse este parámetro por debajo de 15 segundos (15000 ms) con el fin de garantizar que el módulo sea llevado online.</p>
Ciclo (Página 81)	Permite definir un tiempo de ciclo máximo o un tiempo de ciclo mínimo fijo
Carga de comunicación	Permite asignar el porcentaje del tiempo de la CPU que debe dedicarse a las tareas de comunicación
Marcas de sistema y de ciclo (Página 85)	Habilita un byte para funciones de "memoria de sistema" y un byte para funciones de "marca de ciclo" (en las que cada bit cambia entre on y off con una frecuencia predefinida).
Servidor web (Página 523)	Habilita y configura la funcionalidad del servidor web.
Hora	Selecciona la zona horaria y configura el horario de verano/invierno
Protección (Página 168)	Permite ajustar la protección de lectura/escritura y la contraseña para acceder a la CPU
Recursos de conexión (Página 442)	Ofrece un resumen de las conexiones de comunicación que están disponibles para la CPU y del número de conexiones que se han configurado.
Vista general de las direcciones	Ofrece un resumen de las direcciones E/S que se han configurado para la CPU.

5.5 Configurar los parámetros de los módulos

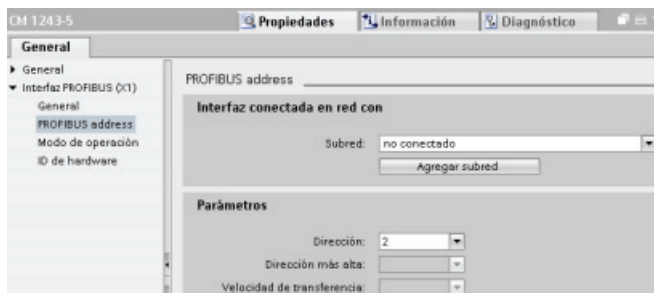
Para configurar los parámetros operativos de un módulo, selecciónelo en la vista de dispositivos y utilice la ficha "Propiedades" de la ventana de inspección.

Configurar un módulo de señales (SM) o una Signal Board (SB)



- E/S digitales: Las entradas pueden configurarse para detectar flancos ascendentes o descendentes (asociando cada una de ellas a un evento y una alarma de proceso) y para la "captura de impulsos" (para que permanezcan activadas tras un impulso momentáneo) hasta la siguiente actualización de la memoria imagen de proceso de las entradas. Las salidas pueden congelarse o utilizar un valor sustitutivo.
- E/S analógicas: Es posible configurar los parámetros de las distintas entradas, tales como el tipo de medición (tensión o intensidad), el rango y el alisamiento, así como habilitar el diagnóstico de rebase por defecto o por exceso. Las salidas analógicas ponen a disposición parámetros tales como el tipo de salida (tensión o intensidad) y de diagnóstico, p. ej. cortocircuito (para salidas de tensión) o diagnóstico de límite superior/inferior. En el diálogo Propiedades no se configuran rangos de entradas y salidas analógicas en unidades de ingeniería. Éstas deben procesarse en la lógica del programa de la forma descrita en "Procesamiento de valores analógicos (Página 94)".
- Direcciones E/S de diagnóstico: Permite configurar la dirección inicial de las entradas y salidas del módulo

Configurar una interfaz de comunicación (CM, CP o CB)





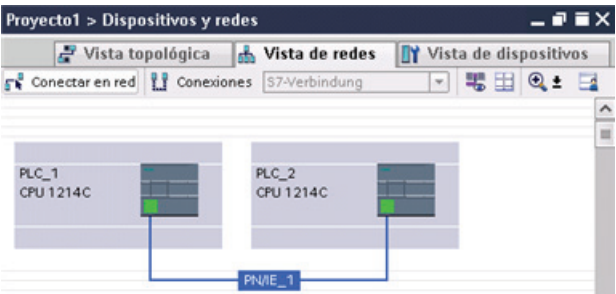
Dependiendo del tipo de interfaz de comunicación, se configuran los parámetros del segmento.

5.6 Configurar la CPU para la comunicación

5.6.1 Crear una conexión de red

Utilice la "Vista de red" de la "Configuración de dispositivos" para crear las conexiones de red entre los dispositivos del proyecto. Tras crear la conexión de red, utilice la ficha "Propiedades" de la ventana de inspección para configurar los parámetros de la red.

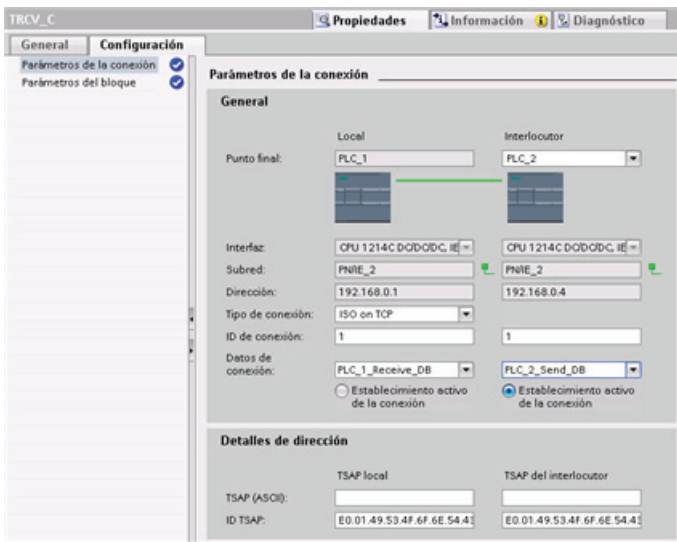
Tabla 5- 3 Creación de una conexión de red

Acción	Resultado
<p>Seleccione "Vista de red" para visualizar los dispositivos que deben conectarse.</p>	
<p>Seleccione el puerto de uno de los dispositivos y arrastre la conexión hasta el puerto del otro dispositivo.</p>	
<p>Suelte el botón del ratón para crear la conexión de red.</p>	

5.6.2 Configurar la vía de conexión local/de interlocutor

La ventana de inspección muestra las propiedades de la conexión cuando se selecciona cualquier parte de la instrucción. Los parámetros de comunicación se especifican en la ficha "Configuración" de "Propiedades" de la instrucción de comunicación.

Tabla 5- 4 Configurar la ruta de conexión (mediante las propiedades de la instrucción)

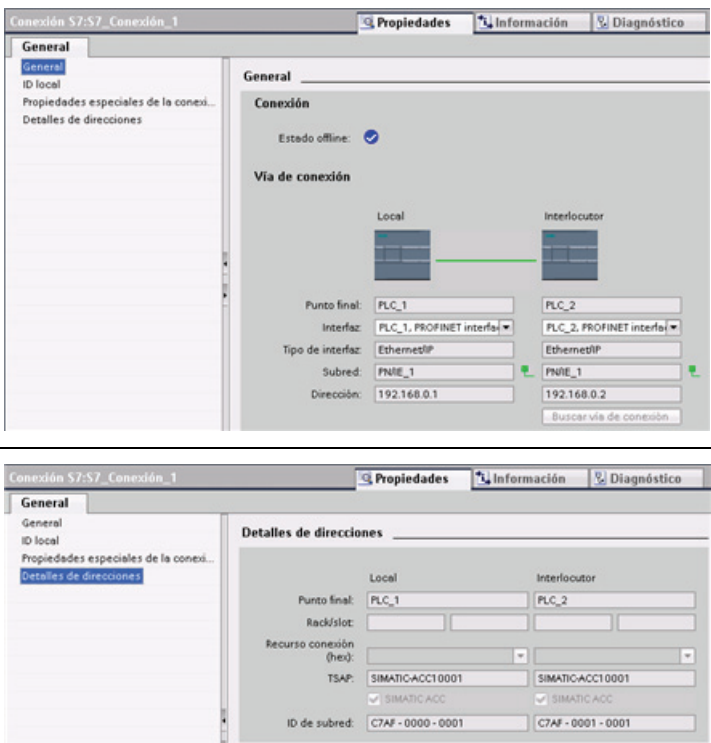
TCP, ISO on TCP y UDP	Propiedades de la conexión
<p>Para los protocolos de Ethernet TCP, ISO on TCP y UDP, utilice las "Propiedades" de la instrucción (TSEND_C, TRCV_C o TCON) para configurar las conexiones "local/interlocutor".</p> <p>La figura muestra las "Propiedades de conexión" de la ficha "Configuración" en el caso de una conexión ISO on TCP.</p>	

Nota

Cuando se configuran las propiedades de conexión de una CPU, STEP 7 permite seleccionar un DB de conexión específico en la CPU interlocutora (si hay alguno) o bien crear el DB de conexión para la CPU interlocutora. La CPU interlocutora ya debe estar creada en el proyecto y no puede ser una CPU "sin especificar".

Todavía hay que insertar una instrucción TSEND_C, TRCV_C o TCON en el programa de usuario de la CPU interlocutora. Al insertar la instrucción, seleccione el DB de conexión que se creó durante la configuración.

Tabla 5- 5 Configurar la ruta de conexión para la comunicación S7 (configuración del dispositivo)

Comunicación S7 (GET y PUT)	Propiedades de la conexión
<p>Para la comunicación S7, utilice el editor "Dispositivos y redes" de la red para configurar las conexiones local/interlocutor. Haga clic en el botón "Resaltado: conexión" para acceder a las "Propiedades".</p> <p>La ficha "General" ofrece varias propiedades:</p> <ul style="list-style-type: none"> • "General" (la que se muestra) • "ID local" • "Propiedades de conexión especiales" • "Detalles de direcciones" (la que se muestra) 	

Consulte los "Protocolos" (Página 448) de la sección "PROFINET" o "Crear una conexión S7" (Página 513) en la sección "Comunicación S7 para obtener más información y una lista de las instrucciones de comunicación disponibles.

Tabla 5- 6 Parámetros para la conexión múltiple de CPU

Parámetro	Definición																
Dirección	Direcciones IP asignadas																
General	<table border="1"> <tr> <td>Punto final</td> <td>Nombre asignado a la CPU interlocutora (receptora)</td> </tr> <tr> <td>Interfaz</td> <td>Nombre asignado a las interfaces</td> </tr> <tr> <td>Subred</td> <td>Nombre asignado a las subredes</td> </tr> <tr> <td>Tipo de interfaz</td> <td><i>Sólo comunicación S7:</i> Tipo de interfaz</td> </tr> <tr> <td>Tipo de conexión</td> <td>Tipo de protocolo Ethernet</td> </tr> <tr> <td>ID de conexión</td> <td>Número de ID</td> </tr> <tr> <td>Datos de conexión</td> <td>Ubicación de almacenamiento de datos de las CPUs local e interlocutora</td> </tr> <tr> <td>Establecer una conexión activa</td> <td>Botón de opción para seleccionar la CPU local o interlocutora como conexión activa</td> </tr> </table>	Punto final	Nombre asignado a la CPU interlocutora (receptora)	Interfaz	Nombre asignado a las interfaces	Subred	Nombre asignado a las subredes	Tipo de interfaz	<i>Sólo comunicación S7:</i> Tipo de interfaz	Tipo de conexión	Tipo de protocolo Ethernet	ID de conexión	Número de ID	Datos de conexión	Ubicación de almacenamiento de datos de las CPUs local e interlocutora	Establecer una conexión activa	Botón de opción para seleccionar la CPU local o interlocutora como conexión activa
Punto final	Nombre asignado a la CPU interlocutora (receptora)																
Interfaz	Nombre asignado a las interfaces																
Subred	Nombre asignado a las subredes																
Tipo de interfaz	<i>Sólo comunicación S7:</i> Tipo de interfaz																
Tipo de conexión	Tipo de protocolo Ethernet																
ID de conexión	Número de ID																
Datos de conexión	Ubicación de almacenamiento de datos de las CPUs local e interlocutora																
Establecer una conexión activa	Botón de opción para seleccionar la CPU local o interlocutora como conexión activa																
Detalles de dirección	<table border="1"> <tr> <td>Punto final</td> <td><i>Sólo comunicación S7:</i> Nombre asignado a la CPU interlocutora (receptora)</td> </tr> <tr> <td>Rack/slot</td> <td><i>Sólo comunicación S7:</i> Ubicación de rack y slot</td> </tr> <tr> <td>Recurso de conexión</td> <td><i>Sólo comunicación S7:</i> Componente de TSAP que se utiliza en la configuración de una conexión S7 con una CPU S7-300 ó S7-400</td> </tr> <tr> <td>Puerto (decimal):</td> <td>TCP y UDP: Puerto de la CPU interlocutora en formato decimal</td> </tr> </table>	Punto final	<i>Sólo comunicación S7:</i> Nombre asignado a la CPU interlocutora (receptora)	Rack/slot	<i>Sólo comunicación S7:</i> Ubicación de rack y slot	Recurso de conexión	<i>Sólo comunicación S7:</i> Componente de TSAP que se utiliza en la configuración de una conexión S7 con una CPU S7-300 ó S7-400	Puerto (decimal):	TCP y UDP: Puerto de la CPU interlocutora en formato decimal								
Punto final	<i>Sólo comunicación S7:</i> Nombre asignado a la CPU interlocutora (receptora)																
Rack/slot	<i>Sólo comunicación S7:</i> Ubicación de rack y slot																
Recurso de conexión	<i>Sólo comunicación S7:</i> Componente de TSAP que se utiliza en la configuración de una conexión S7 con una CPU S7-300 ó S7-400																
Puerto (decimal):	TCP y UDP: Puerto de la CPU interlocutora en formato decimal																

Parámetro		Definición
	TSAP ¹ e ID de subred:	ISO on TCP (RFC 1006) y comunicación S7: TSAPs de las CPUs local e interlocutora en formato ASCII y hexadecimal

¹ Al configurar una conexión con una CPU S7-1200 para ISO on TCP, utilice sólo caracteres ASCII en la extensión TSAP para los interlocutores pasivos.

Transport Service Access Points (TSAPs)

El uso de TSAPs, el protocolo ISO on TCP y la comunicación S7 permite conexiones múltiples con una dirección IP única (conexiones hasta 64K). Los TSAPs identifican unívocamente estas conexiones de puntos finales de comunicación a una dirección IP.

En el área "Detalles de dirección" del diálogo "Parámetros de la conexión" se definen los TSAPs que deben utilizarse. El TSAP de una conexión en la CPU se introduce en el campo "TSAP local". El TSAP asignado a la conexión en la CPU interlocutora se introduce en el campo "TSAP del interlocutor".

Números de puerto

Con los protocolos TCP y UDP, la configuración de los parámetros de conexión de la CPU local (activa) debe especificar la dirección IP remota y el número de puerto de la CPU interlocutora (pasiva).

En el área "Detalles de dirección" del diálogo "Parámetros de la conexión" se definen los puertos que deben utilizarse. El puerto de una conexión en la CPU se introduce en el campo "Puerto local". El puerto asignado a la conexión en la CPU interlocutora se introduce en el campo "Puerto del interlocutor".

5.6.3 Parámetros de la conexión PROFINET

Para las instrucciones TSEND_C, TRCV_C y TCON es necesario especificar los parámetros relacionados con la conexión para poder conectarse con el dispositivo interlocutor. Dichos parámetros están especificados por la estructura TCON_Param para los protocolos TCP, ISO on TCP y UDP. Por norma general, para especificar estos parámetros se utiliza la ficha "Configuración" de las "Propiedades" de la instrucción. Si la ficha "Configuración" no está disponible, hay que especificar la estructura TCON_Param por medio de programación.

Tabla 5-7 Estructura de la descripción de la conexión (TCON_Param)

Byte	Parámetro y tipo de datos		Descripción
0 ... 1	block_length	UInt	Longitud: 64 bytes (fijos)
2 ... 3	id	CONN_OUC (Word)	Referencia a esta conexión: Rango de valores: de 1 (predeterminado) a 4095. Especifique el valor de este parámetro para la instrucción TSEND_C, TRCV_C o TCON en ID.
4	connection_type	USInt	Tipo de conexión: <ul style="list-style-type: none"> • 17: TCP (predeterminado) • 18: ISO on TCP • 19: UDP

Byte	Parámetro y tipo de datos		Descripción
5	active_est	Bool	ID del tipo de conexión: <ul style="list-style-type: none"> • TCP e ISO on TCP: <ul style="list-style-type: none"> – FALSE: conexión pasiva – TRUE: conexión activa (predeterminado) • UDP: FALSE
6	local_device_id	USInt	ID de la interfaz Industrial Ethernet o PROFINET local: 1 (predeterminado)
7	local_tsap_id_len	USInt	Longitud del parámetro local_tsap_id utilizado, en bytes; valores posibles: <ul style="list-style-type: none"> • TCP: 0 (activa, predeterminado) o 2 (pasiva) • ISO on TCP: de 2 a 16 • UDP: 2
8	rem_subnet_id_len	USInt	Este parámetro no se utiliza.
9	rem_staddr_len	USInt	Longitud de la dirección del punto final del interlocutor, en bytes: <ul style="list-style-type: none"> • 0: sin especificar (el parámetro rem_staddr es irrelevante) • 4 (predeterminado): dirección IP válida en el parámetro rem_staddr (sólo para TCP e ISO on TCP)
10	rem_tsap_id_len	USInt	Longitud del parámetro rem_tsap_id utilizado, en bytes; valores posibles: <ul style="list-style-type: none"> • TCP: 0 (pasiva) o 2 (activa, predeterminado) • ISO on TCP: de 2 a 16 • UDP: 0
11	next_staddr_len	USInt	Este parámetro no se utiliza.
12 ... 27	local_tsap_id	Array [1..16] of Byte	Componente de conexión para la dirección local: <ul style="list-style-type: none"> • TCP e ISO on TCP: n.º de puerto local (valores posibles: de 1 a 49151; valores recomendados: 2000...5000): <ul style="list-style-type: none"> – local_tsap_id[1] = byte High del número de puerto en notación hexadecimal; – local_tsap_id[2] = byte Low del número de puerto en notación hexadecimal; – local_tsap_id[3-16] = irrelevante • ISO on TCP: ID de TSAP local: <ul style="list-style-type: none"> – local_tsap_id[1] = B#16#E0; – local_tsap_id[2] = rack y slot de los puntos finales locales (bits 0 a 4: número de slot, bits 5 a 7: número de rack); – local_tsap_id[3-16] = extensión de TSAP, opcional • UDP: Este parámetro no se utiliza. <p>Nota: asegúrese de que todos los valores de local_tsap_id son unívocos dentro de la CPU.</p>
28 ... 33	rem_subnet_id	Array [1..6] of USInt	Este parámetro no se utiliza.

5.6 Configurar la CPU para la comunicación

Byte	Parámetro y tipo de datos		Descripción
34 ... 39	rem_staddr	Array [1..6] of USInt	Sólo TCP e ISO on TCP: dirección IP del punto final del interlocutor. (No relevante para las conexiones pasivas.) Por ejemplo, la dirección IP 192.168.002.003 se guarda en los elementos siguientes de la matriz: rem_staddr[1] = 192 rem_staddr[2] = 168 rem_staddr[3] = 002 rem_staddr[4] = 003 rem_staddr[5-6]= irrelevante
40 ... 55	rem_tsap_id	Array [1..16] of Byte	Componente de conexión para la dirección del interlocutor: <ul style="list-style-type: none"> • TCP: número de puerto del interlocutor. Rango: de 1 a 49151; valores recomendados: de 2000 a 5000): <ul style="list-style-type: none"> – rem_tsap_id[1] = byte High del número de puerto en notación hexadecimal – rem_tsap_id[2] = byte Low del número de puerto en notación hexadecimal; – rem_tsap_id[3-16] = irrelevante • ISO on TCP: ID de TSAP del interlocutor: <ul style="list-style-type: none"> – rem_tsap_id[1] = B#16#E0 – rem_tsap_id[2] = rack y slot del punto final del interlocutor (bits 0 a 4: número de slot, bits 5 a 7: número de rack) – rem_tsap_id[3-16] = extensión de TSAP, opcional • UDP: este parámetro no se utiliza.
56 ... 61	next_staddr	Array [1..6] of Byte	Este parámetro no se utiliza.
62 ... 63	spare	Word	Reservado: W#16#0000

Consulte también

Configurar la vía de conexión local/de interlocutor (Página 132)

5.6.4 Asignar direcciones IP (Internet Protocol)

5.6.4.1 Asignar direcciones IP a los dispositivos de programación y red

Si la programadora incorpora una tarjeta adaptadora conectada a la LAN de la instalación (y posiblemente a Internet), la ID de red de la dirección IP y la máscara de subred de la CPU y la tarjeta adaptadora integrada en la programadora deberán ser idénticas. La ID de red es la primera parte de la dirección IP (los tres primeros octetos) (p. ej. **211.154.184.16**) y determina la red IP utilizada. Normalmente, la máscara de subred tiene el valor **255.255.255.0**. No obstante, puesto que el equipo está integrado en una LAN corporativa, la máscara de subred puede tener distintos valores (p. ej. **255.255.254.0**) para configurar subredes unívocas. Al combinar la máscara de subred con la dirección IP del dispositivo en una operación Y matemática se definen los límites de la subred IP.

Nota

En Internet, puesto que las programadoras, dispositivos de red y routers IP se comunican con el mundo entero, es preciso asignar direcciones IP unívocas para evitar conflictos con otros usuarios de la red. Contacte con los especialistas del departamento IT de su empresa, que están familiarizados con la red corporativa, para asignar las direcciones IP.

Si la programadora utiliza una tarjeta adaptadora Ethernet-USB conectada a una red aislada, la ID de red de la dirección IP y la máscara de subred de la CPU y la tarjeta adaptadora Ethernet-USB integrada en la programadora deberán ser exactamente iguales. La ID de red es la primera parte de la dirección IP (los tres primeros octetos) (p. ej. **211.154.184.16**) y determina la red IP utilizada. Normalmente, la máscara de subred tiene el valor **255.255.255.0**. Al combinar la máscara de subred con la dirección IP del dispositivo en una operación Y matemática se definen los límites de la subred IP.

Nota

Una tarjeta adaptadora Ethernet-USB es apropiada si la CPU no debe integrarse en la LAN corporativa. Esta opción es especialmente útil durante la comprobación inicial o los tests de puesta en marcha.

Tabla 5- 8 Asignación de direcciones Ethernet

Tarjeta adaptadora de la programadora	Tipo de red	Dirección IP (Internet Protocol)	Máscara de subred
Tarjeta adaptadora integrada	Conectada a la LAN corporativa (y posiblemente a Internet)	La ID de red de la CPU y la tarjeta adaptadora integrada en la programadora deben ser exactamente iguales. La ID de red es la primera parte de la dirección IP (los tres primeros octetos) (p. ej. 211.154.184 .16) y determina la red IP utilizada.	La máscara de subred de la CPU y la tarjeta adaptadora integrada deben ser exactamente iguales. Normalmente, la máscara de subred tiene el valor 255.255.255.0 . No obstante, puesto que el equipo está integrado en una LAN corporativa, la máscara de subred puede tener distintos valores (p. ej. 255.255.254.0) para configurar subredes unívocas. Al combinar la máscara de subred con la dirección IP del dispositivo en una operación Y matemática se definen los límites de la subred IP.
Tarjeta adaptadora Ethernet-USB	Conectada a una red aislada	La ID de red de la CPU y la tarjeta adaptadora Ethernet-USB de la programadora deben ser exactamente iguales. La ID de red es la primera parte de la dirección IP (los tres primeros octetos) (p. ej. 211.154.184 .16) y determina la red IP utilizada.	La máscara de subred de la CPU y la tarjeta adaptadora Ethernet-USB deben ser exactamente iguales. Normalmente, la máscara de subred tiene el valor 255.255.255.0 . Al combinar la máscara de subred con la dirección IP del dispositivo en una operación Y matemática se definen los límites de la subred IP.

Asignar o comprobar la dirección IP de la programadora utilizando "Mis sitios de red" (en el Escritorio)

La dirección IP de la programadora se puede asignar o comprobar mediante los siguientes comandos de menú:

- (Clic con el botón derecho del ratón en) "Mis sitios de red"
- "Propiedades"
- (Clic con el botón derecho del ratón en) "Conexión de área local"
- "Propiedades"

En el diálogo "Propiedades de conexión de área local", campo "Esta conexión utiliza los siguientes elementos:" desplácese hasta "Protocolo Internet (TCP/IP)". Haga clic en "Protocolo Internet (TCP/IP)" y luego en el botón "Propiedades". Seleccione "Obtener una dirección IP automáticamente (DHCP)" o "Usar la siguiente dirección IP" (para introducir una dirección IP estática).

Nota

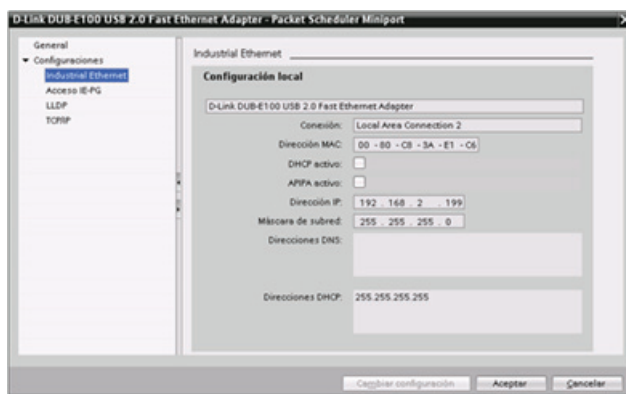
El "Dynamic Host Configuration Protocol" (DHCP o protocolo de configuración dinámica de host) asigna automáticamente una dirección IP a la programadora después del arranque desde el servidor DHCP.

5.6.4.2 Comprobar la dirección IP de la programadora

Las direcciones MAC e IP de la programadora pueden consultarse con los siguientes comandos de menú:

1. En el "Árbol del proyecto", expanda "Acceso online".
2. Haga clic con el botón derecho en la red en cuestión y seleccione "Propiedades".
3. En el diálogo de la red, expanda "Configuraciones" y seleccione "Industrial Ethernet".

Se muestran las direcciones MAC e IP de la programadora.

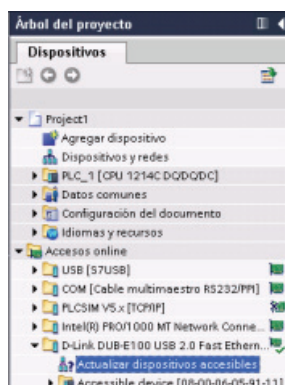


5.6.4.3 Asignar una dirección IP a una CPU online

Es posible asignar una dirección IP a un dispositivo de red online. Esto es especialmente útil al configurar los dispositivos por primera vez.

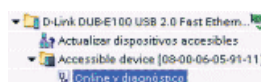
1. En el "Árbol del proyecto", verifique que la CPU no tiene asignada ninguna dirección IP. Utilice para ello los comandos de menú siguientes:

- "Accesos online"
- <Tarjeta adaptadora para la red en la que se encuentra el dispositivo>
- "Actualizar dispositivos accesibles"



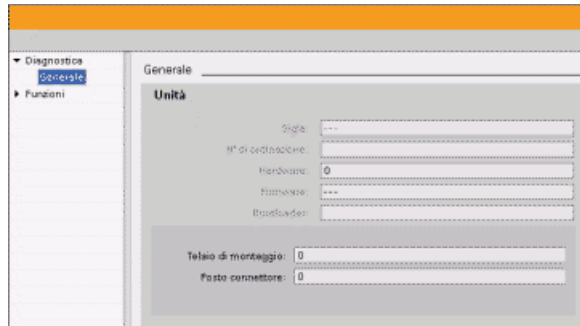
NOTA: Si se muestra una dirección MAC en lugar de una dirección IP, no se ha asignado ninguna dirección IP.

2. En el dispositivo accesible en cuestión, haga doble clic en "Online y diagnóstico"

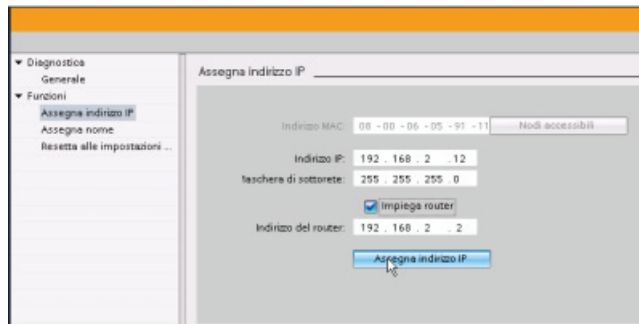


3. Seleccione los siguientes comandos de menú en el diálogo "Online y diagnóstico":

- "Funciones"
- "Asignar dirección IP"

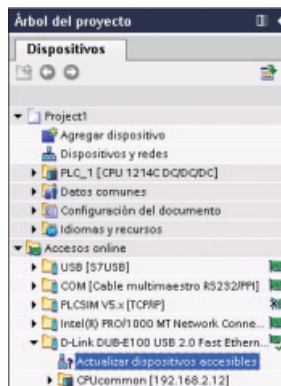


4. En el campo "Dirección IP", introduzca la nueva dirección IP y haga clic en el botón "Asignar dirección IP".



5. En el "Árbol del proyecto", verifique que la nueva dirección IP se ha asignado a la CPU. Utilice para ello los comandos de menú siguientes:

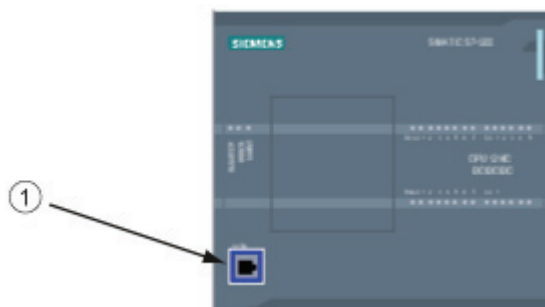
- "Accesos online"
- <Adaptador para la red en la que se encuentra el dispositivo>
- "Actualizar dispositivos accesibles"



5.6.4.4 Configurar una dirección IP para una CPU en el proyecto

Configurar la interfaz PROFINET

Para configurar parámetros para la interfaz PROFINET, seleccione la casilla PROFINET verde en la CPU. La ficha "Propiedades" de la ventana de inspección muestra el puerto PROFINET.



① Puerto PROFINET

Configurar la dirección IP

Dirección Ethernet (MAC): Todo dispositivo de una red PROFINET recibe una dirección MAC (Media Access Control o control de acceso al medio) del fabricante para su identificación. Una dirección MAC consta de seis grupos de dos dígitos hexadecimales, separados por guiones (-) o dos puntos (:), en orden de transmisión (p. ej. 01-23-45-67-89-AB ó 01:23:45:67:89:AB).

Dirección IP: Todo dispositivo debe tener también una dirección IP (Internet Protocol o Protocolo Internet). Esta dirección permite al dispositivo transferir datos a través de una red enrutada y más compleja.

Toda dirección IP se divide en cuatro segmentos de ocho bits (octetos) y se expresa en formato decimal separado por puntos (p. ej. 211.154.184.16). La primera parte de la dirección IP se utiliza para la ID de red (¿en qué red se encuentra?) y, la segunda, para la ID del host (unívoca para cada dispositivo de la red). Una dirección IP 192.168.x.y es una designación estándar reconocida como parte de una red privada que no se enruta vía Internet.

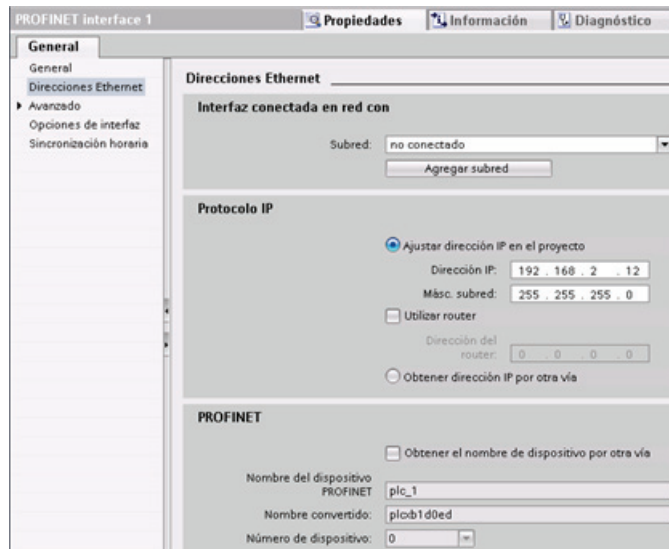
Máscara de subred: Una subred es una agrupación lógica de dispositivos de red conectados. Generalmente, los nodos de una subred están próximos físicamente en una red de área local (LAN). Una máscara (denominada "máscara de subred" o "máscara de red") define los límites de una subred IP.

Generalmente, una máscara de subred 255.255.255.0 se adecúa para una red local pequeña. Esto significa que los 3 primeros octetos de todas las direcciones IP de esta red deberían ser iguales. Los diferentes dispositivos de la red se identifican mediante el último octeto (campo de 8 bits). Por ejemplo, es posible asignar la máscara de subred 255.255.255.0 y direcciones IP comprendidas entre 192.168.2.0 y 192.168.2.255 a los dispositivos de una red local pequeña.

La única conexión entre las diferentes subredes se realiza a través de un router. Si se utilizan subredes, es preciso utilizar un router IP.

Router IP: Los routers interconectan las distintas LANs. Si se utiliza un router, un equipo de una LAN puede enviar mensajes a otras redes que, a su vez, pertenezcan a otras LANs. Si el destino de los datos se encuentra fuera de la LAN, el router reenvía los datos a otra red o grupo de redes desde donde pueden transferirse a su destino.

Los routers necesitan direcciones IP para poder transferir y recibir paquetes de datos.



Propiedades de direcciones IP: En la ventana de propiedades, seleccione la entrada de configuración "Direcciones Ethernet". STEP 7 muestra el diálogo de configuración de direcciones Ethernet, en el que el proyecto de software se asocia a la dirección IP de la CPU que lo recibirá.

Tabla 5-9 Parámetros para la dirección IP

Parámetro	Descripción	
Subred	Nombre de la subred a la que está conectada el dispositivo. Haga clic en el botón "Agregar nueva subred" para crear una subred nueva. La opción predeterminada es "No conectado". Son posibles dos tipos de conexión: <ul style="list-style-type: none"> • El ajuste predeterminado "no conectado" ofrece una conexión local. • Una subred se requiere cuando la red comprende dos o más dispositivos. 	
Protocolo IP	Dirección IP	Dirección IP asignada a la CPU
	Máscara de subred	Máscara de subred asignada
	Utilizar router IP	Haga clic en esta casilla de verificación para indicar el uso de un router IP
	Dirección del router	Dirección IP asignada al router (si es aplicable)

Nota

Todas las direcciones IP se configuran al cargar el proyecto en el dispositivo. Dado que la CPU no dispone de una dirección IP preconfigurada, es necesario asociar el proyecto con la dirección MAC del dispositivo de destino. Si la CPU está conectada a un router de la red, también es preciso introducir la dirección IP del router.

El botón de opción "Asignar dirección IP por otra vía" permite cambiar la dirección IP online o mediante la instrucción "T_CONFIG" después de descargar el programa. Este método de asignación de la dirección IP sólo sirve para la CPU.

 **ADVERTENCIA**

Tras descargar una configuración de hardware con la opción "Asignar dirección IP por otra vía" habilitada, no se puede conmutar el estado operativo de la CPU de RUN a STOP o de STOP a RUN.

El equipo del usuario seguirá funcionando en estas condiciones y puede causar operaciones imprevistas de la máquina o el proceso, provocando como resultado daños materiales, lesiones corporales graves o incluso la muerte si no se toman las precauciones adecuadas.

Asegúrese de que las direcciones IP de la CPU se han establecido anteriormente utilizando la CPU en un entorno de automatización actual. Esto puede hacerse utilizando el paquete de programación STEP 7, la herramienta S7-1200 o un dispositivo HMI conectado conjuntamente con la instrucción T_CONFIG.

 **ADVERTENCIA**

Al modificar la dirección IP de una CPU online o desde el programa de usuario, es posible crear una condición en la que se detenga la red PROFINET.

Si la dirección IP de una CPU cambia a una dirección IP fuera de la subred, la red PROFINET perderá la comunicación y se detendrá todo intercambio de datos. Existe la posibilidad de configurar el equipo del usuario para que siga funcionando en estas condiciones. La pérdida de la comunicación PROFINET puede causar operaciones imprevistas de la máquina o el proceso con el resultado de muerte, lesiones corporales graves o daños materiales si no se toman las precauciones adecuadas.

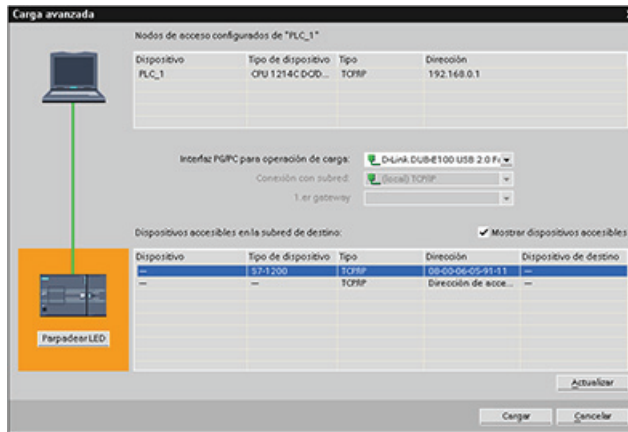
Si hay que cambiar manualmente la dirección IP, asegúrese de que la nueva dirección IP está dentro de la subred.

Consulte también

T_CONFIG (Página 470)

5.6.5 Comprobar la red PROFINET

Tras finalizar la configuración, cargue el proyecto (Página 172) en la CPU. Todas las direcciones IP se configuran al cargar el proyecto en el dispositivo.



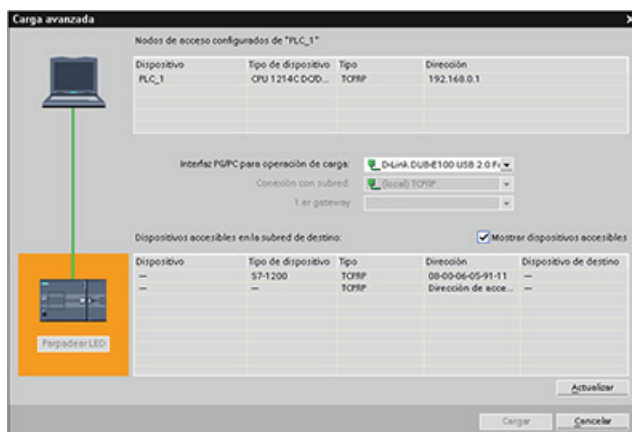
Asignar una dirección IP a un dispositivo online

La CPU S7-1200 no tiene dirección IP preconfigurada. La dirección IP de la CPU se debe asignar manualmente:

- Para asignar una dirección IP a un dispositivo online, véase "Configuración de dispositivos: Asignar una dirección IP a una CPU online" (Página 139) para conocer el procedimiento paso a paso.
- Para asignar una dirección IP en el proyecto, es preciso configurarla en la "Configuración de dispositivos", guardar la configuración y cargarla en el PLC. Véase "Configuración de dispositivos: Configurar una dirección IP para una CPU en el proyecto" (Página 141) para más información.

Utilizar el diálogo "Carga avanzada" para comprobar los dispositivos de red conectados

La función de la CPU S7-1200 "Cargar en dispositivo" y su diálogo "Carga avanzada" permiten visualizar todos los dispositivos de red accesibles y verificar si se han asignado direcciones IP unívocas a todos ellos. Para visualizar todos los dispositivos accesibles y disponibles con sus respectivas direcciones MAC o IP asignadas, active la casilla de verificación "Mostrar dispositivos accesibles".



Si el dispositivo de red deseado no se encuentra en esta lista, la comunicación con ese dispositivo se habrá interrumpido por algún motivo. En este caso es preciso examinar el dispositivo y la red para buscar errores de hardware y/o configuración.

5.6.6 Localizar la dirección Ethernet (MAC) en la CPU

En las redes PROFINET, una dirección MAC (Media Access Control o control de acceso al medio) es un identificador que el fabricante asigna a la interfaz de red para su identificación. Generalmente, una dirección MAC codifica el número de identificación registrado del fabricante.

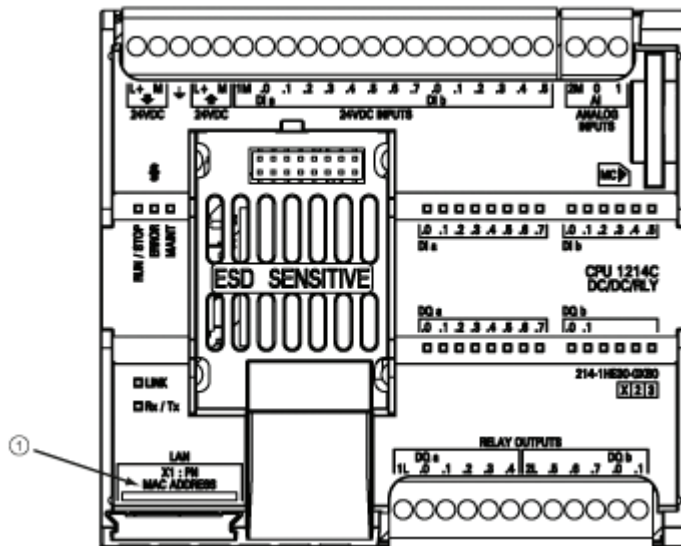
El formato estándar (IEEE 802.3) permite imprimir direcciones MAC de forma amigable y consta de seis grupos de dos dígitos hexadecimales, separados por guiones (-) o dos puntos (:), en orden de transmisión (p. ej. 01-23-45-67-89-ab ó 01:23:45:67:89:ab).

Nota

Toda CPU se suministra de fábrica con una dirección MAC unívoca y permanente. La dirección MAC de la CPU no se puede cambiar.

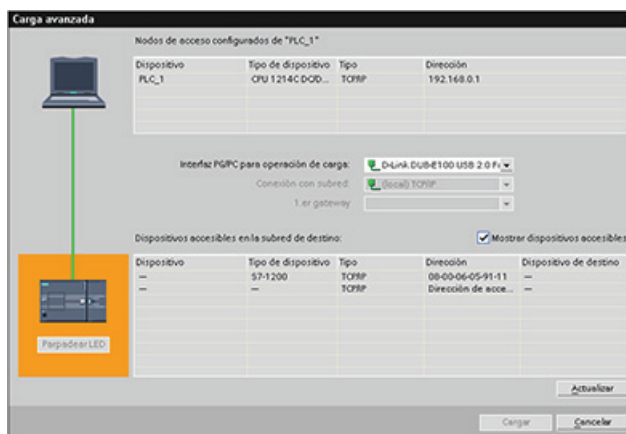
5.6 Configurar la CPU para la comunicación

La dirección MAC está impresa en la esquina inferior izquierda en el frente de la CPU. Para ver la información de la dirección MAC es necesario abrir la tapa inferior.



① Dirección MAC

Inicialmente, la CPU no tiene dirección IP, sino sólo una dirección MAC ajustada de fábrica. Para la comunicación PROFINET es necesario que todos los dispositivos tengan asignada una dirección IP unívoca.



La función de la CPU "Cargar en dispositivo" y el diálogo "Carga avanzada en dispositivo" permiten visualizar todos los dispositivos de red accesibles y asegurar que se han asignado direcciones IP unívocas a todos ellos. Este diálogo muestra todos los dispositivos accesibles y disponibles con sus respectivas direcciones MAC o IP asignadas. Las direcciones MAC son especialmente importantes para identificar dispositivos que no disponen de la dirección IP unívoca necesaria.

5.6.7 Configurar la sincronización del Network Time Protocol (NTP)

El Network Time Protocol (NTP) es un protocolo ampliamente utilizado para sincronizar los relojes de los sistemas informáticos con los servidores de hora de Internet. En el modo NTP el CP envía consultas de hora en intervalos regulares (en el modo de cliente) al servidor NTP de la subred (LAN). A partir de las respuestas del servidor, se calcula la hora más probable y precisa y se sincroniza la hora en la estación.

La ventaja de este método radica en que permite sincronizar la hora entre subredes.

Deben configurarse las direcciones IP de hasta cuatro servidores NTP. El intervalo de actualización define el tiempo entre las consultas de hora (en segundos). El valor del intervalo puede variar entre 10 segundos y un día.

En el modo NTP generalmente se transfiere la hora UTC (Universal Time Coordinated o tiempo universal combinado), que corresponde a GMT (Greenwich Mean Time o tiempo promedio del Observatorio de Greenwich).

En la ventana de propiedades, seleccione la entrada de configuración "Sincronización horaria". STEP 7 muestra el diálogo de configuración de la sincronización horaria:

Nota

Todas las direcciones IP se configuran al cargar el proyecto en el dispositivo.

Tabla 5- 10 Parámetros de la sincronización horaria

Parámetro	Definición
Activar la sincronización horaria vía servidores NTP (Network Time Protocol)	Haga clic en la casilla de verificación para activar la sincronización horaria vía servidores NTP.
Servidor 1	Dirección IP asignada al servidor de hora de red 1
Servidor 2	Dirección IP asignada al servidor de hora de red 2
Servidor 3	Dirección IP asignada al servidor de hora de red 3
Servidor 4	Dirección IP asignada al servidor de hora de red 4
Intervalo de sincronización horaria	Valor del intervalo (seg)

5.6.8 Tiempo de arranque de dispositivo, asignación de nombre y de dirección en PROFINET

PROFINET IO puede ampliar el tiempo de arranque del sistema (timeout configurable). Un número mayor de dispositivos o los dispositivos lentos influyen en el tiempo de arranque necesario para conmutar a RUN.

En una red S7-1200 PROFINET se puede tener el siguiente número máximo de dispositivos PROFINET IO:

- En V3.0 puede haber un máximo de 16 dispositivos IO.
- En V2.2 puede haber un máximo de 8 dispositivos IO.

Cada estación (o dispositivo IO) arranca independientemente en el arranque, y esto afecta al tiempo global de arranque de la CPU. Si para el timeout configurable se ajusta un valor muy bajo, el tiempo global de arranque de la CPU puede no ser suficiente para que todas las estaciones completen su arranque. Si eso ocurre, aparecerán errores de estación no reales.

En "Propiedades de la CPU", bajo "Arranque", encontrará el "Tiempo de asignación de los parámetros para E/S distribuidas" (timeout). El timeout predeterminado configurable es 60.000 ms (1 minuto), pero el usuario puede modificarlo.

Asignación de nombre y dirección de dispositivo PROFINET en STEP 7

Todos los dispositivos PROFINET **deben** tener un nombre de dispositivo y una dirección IP. Utilice STEP 7 para especificar los nombres de dispositivo y configurar las direcciones IP. Los nombres de dispositivo se descargan a los dispositivos IO por medio de PROFINET DCP (Discovery and Configuration Protocol).

Asignación de dirección PROFINET en el arranque del sistema

El controlador transmite los nombres de los dispositivos a la red, y éstos responden con sus direcciones MAC. Entonces el controlador asigna una dirección IP al dispositivo utilizando el protocolo PROFINET DCP:

- Si la dirección MAC tiene una dirección IP configurada, la estación arranca.
- Si la dirección MAC no tiene una dirección IP configurada, STEP 7 asigna la dirección configurada en el proyecto y, entonces, la estación arranca.
- Si se produce un problema durante este proceso, se produce un error de estación y no hay arranque. La situación provoca un rebase del timeout configurable.

Principios básicos de programación

6.1 Directrices para diseñar un sistema PLC

Al diseñar un sistema PLC es posible seleccionar entre diferentes métodos y criterios. Las directrices generales siguientes pueden aplicarse a un gran número de proyectos. Por supuesto que es necesario respetar las directrices corporativas y las prácticas usuales aprendidas y aplicadas.

Tabla 6- 1 Directrices para diseñar un sistema PLC

Pasos recomendados	Tareas
Dividir el proceso o máquina	Divida el proceso o máquina en secciones independientes. Estas secciones determinan los límites entre los controladores e influyen en las especificaciones funcionales y la asignación de recursos.
Crear las especificaciones funcionales	Describa el funcionamiento de cada una de las secciones del proceso o máquina, tales como las entradas y salidas, la descripción funcional de la operación, los estados que deben adoptarse antes de que puedan entrar en acción los actuadores (como p. ej. electroválvulas, motores o accionamientos), la descripción de la interfaz de operador y cualquier interfaz con otras secciones del proceso o máquina.
Diseñar los circuitos de seguridad	<p>Determine los equipos que puedan requerir cableado fijo por motivos de seguridad. Recuerde que los dispositivos de control pueden fallar y provocar condiciones no seguras, causando a su vez un arranque inesperado o cambios de funcionamiento de la maquinaria. El funcionamiento inesperado o incorrecto de la maquinaria puede causar lesiones corporales o daños materiales considerables. Por tanto, prevea dispositivos de protección electromecánicos (que funcionen independientemente del PLC) para evitar las condiciones no seguras. Las siguientes tareas deben incluirse en el diseño de circuitos de seguridad:</p> <ul style="list-style-type: none"> • Definir el funcionamiento erróneo o inesperado de los actuadores que pudiera resultar peligroso. • Definir las condiciones que garanticen un funcionamiento seguro y determinar cómo detectar estas condiciones, independientemente del PLC. • Definir cómo el PLC y los módulos de ampliación deben influir en el proceso al conectarse y desconectarse la alimentación eléctrica, así como al detectarse errores. Utilice esta información sólo para proyectar el funcionamiento normal y el funcionamiento anormal esperado. Por motivos de seguridad, no conviene fiarse del supuesto más favorable. • Prever dispositivos de parada de emergencia manual o dispositivos de protección electromecánicos que impidan el funcionamiento peligroso, independientemente del PLC. • Proporcionar información de estado apropiada desde los circuitos independientes al PLC para que el programa y las interfaces de operador dispongan de la información necesaria. • Definir otros requisitos adicionales de seguridad para el funcionamiento seguro del proceso.
Planificar la seguridad del sistema	Determine qué nivel de protección (Página 168) necesita para el acceso al proceso. Puede proteger por contraseña las CPU y los bloques de programa frente al acceso no autorizado.

Pasos recomendados	Tareas
Determinar las estaciones de operador	Según los requisitos de las especificaciones funcionales, cree los siguientes dibujos de las estaciones de operador: <ul style="list-style-type: none"> • Dibujo general de la ubicación de todas las estaciones de operador con respecto al proceso o máquina • Dibujo de la disposición mecánica de los dispositivos de la estación de operador, p. ej. display, interruptores y lámparas • Esquemas eléctricos con las E/S asociadas del PLC y los módulos de señales
Crear los dibujos de configuración	Según los requisitos de las especificaciones funcionales, cree dibujos de configuración de los equipos de control: <ul style="list-style-type: none"> • Dibujo general de la ubicación de todos los PLCs con respecto al proceso o máquina • Dibujo de la disposición mecánica de todos los PLCs y módulos de E/S, incluyendo los armarios y otros equipos. • Esquemas eléctricos de todos los PLCs y módulos de E/S, incluyendo los números de referencia de los dispositivos, las direcciones de comunicación y las direcciones de E/S.
Crear una lista de nombres simbólicos	Cree una lista de los nombres simbólicos correspondientes a las direcciones absolutas. Incluya no sólo las E/S físicas, sino también los demás elementos (p. ej. los nombres de variables) que se utilizarán en el programa.

6.2 Estructurar el programa de usuario

Al crear el programa de usuario para las tareas de automatización, las instrucciones del programa se insertan en bloques lógicos:

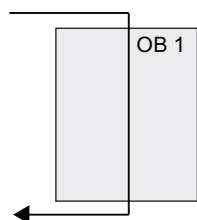
- Un bloque de organización (OB) reacciona a un evento específico en la CPU y puede interrumpir la ejecución del programa de usuario. El bloque predeterminado para la ejecución cíclica del programa de usuario (OB 1) ofrece la estructura básica y es el único bloque lógico que se requiere para el programa de usuario. Si se incluyen otros OBs en el programa, éstos interrumpen la ejecución del OB 1. Los demás OBs ejecutan funciones específicas, tales como tareas de arranque, procesamiento de alarmas y tratamiento de errores, o ejecución de un código de programa específico en determinados intervalos.
- Un bloque de función (FB) es una subrutina que se ejecuta cuando se llama desde otro bloque lógico (OB, FB o FC). El bloque que efectúa la llamada transfiere parámetros al FB e identifica un bloque de datos determinado (DB) que almacena los datos de la llamada o instancia específica de este FB. La modificación del DB instancia permite a un FB genérico controlar el funcionamiento de un conjunto de dispositivos. Por ejemplo, un solo FB puede controlar varias bombas o válvulas con diferentes DBs de instancia que contienen los parámetros operativos específicos de cada bomba o válvula.
- Una función (FC) es una subrutina que se ejecuta cuando se llama desde otro bloque lógico (OB, FB o FC). La FC no tiene un DB instancia asociado. El bloque que efectúa la llamada transfiere los parámetros a la FC. Los valores de salida de la FC deben escribirse en una dirección de la memoria o en un DB global.

Seleccionar el tipo de estructura del programa de usuario

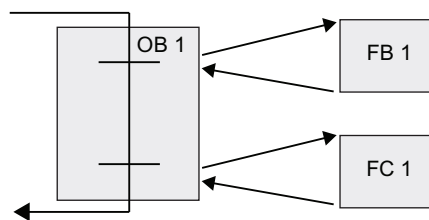
Según los requisitos de la aplicación, es posible seleccionar una estructura lineal o modular para crear el programa de usuario:

- Un programa lineal ejecuta todas las instrucciones de la tarea de automatización de forma secuencial, es decir, una tras otra. Generalmente, el programa lineal deposita todas las instrucciones del programa en el OB encargado de la ejecución cíclica del programa (OB 1).
- Un programa modular llama bloques de función específicos que ejecutan determinadas tareas. Para crear una estructura modular, la tarea de automatización compleja se divide en tareas subordinadas más pequeñas, correspondientes a las funciones tecnológicas del proceso. Cada bloque lógico provee el segmento del programa para cada tarea subordinada. El programa se estructura llamando uno de los bloques lógicos desde otro bloque.

Estructura lineal:



Estructura modular:



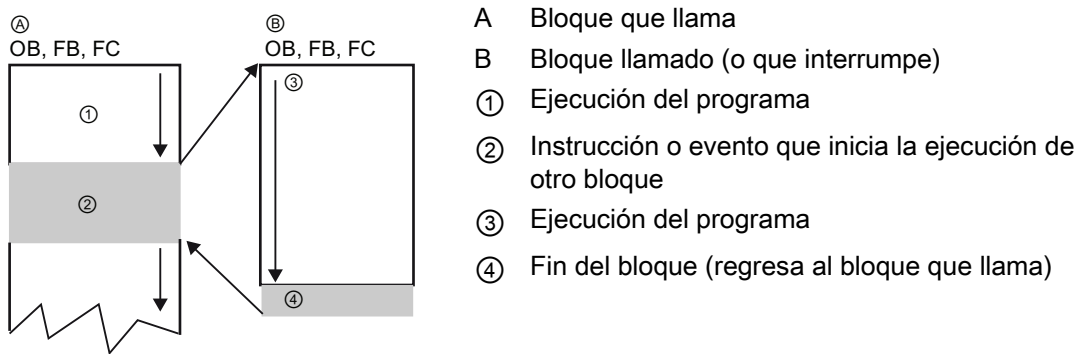
Creando bloques lógicos genéricos que pueden reutilizarse en el programa de usuario, es posible simplificar el diseño y la implementación del programa de usuario. La utilización de bloques lógicos genéricos ofrece numerosas ventajas:

- Es posible crear bloques lógicos reutilizables para tareas estándar, tales como el control de una bomba o motor. También es posible almacenar estos bloques lógicos genéricos en una librería, de manera que puedan ser utilizados por diferentes aplicaciones o soluciones.
- El programa de usuario puede dividirse en componentes modulares para las tareas funcionales, facilitando así su comprensión y gestión. Los componentes modulares ayudan no sólo a estandarizar el diseño del programa, sino que también pueden facilitar y agilizar la actualización o modificación del código del programa.
- La creación de componentes modular simplifica la depuración del programa. Dividiendo el programa completo en segmentos de programa modulares, es posible comprobar las funciones de cada bloque lógico a medida que se va desarrollando.
- La creación de componentes modulares para las distintas funciones tecnológicas permite simplificar y reducir el tiempo de puesta en marcha de la aplicación.

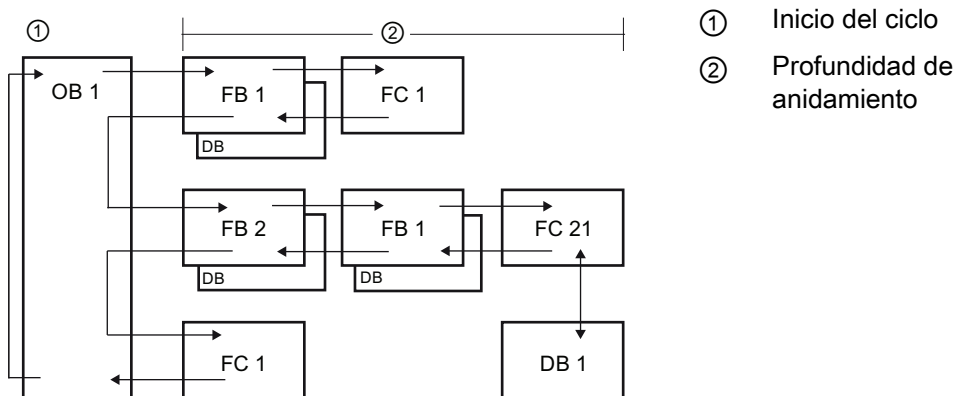
6.3 Utilizar bloques para estructurar el programa

Diseñando FBs y FCs que ejecuten tareas genéricas, se crean bloques lógicos modulares. El programa se estructura luego, de manera que otros bloques lógicos llamen estos bloques modulares reutilizables. El bloque que efectúa la llamada transfiere los parámetros específicos del dispositivo al bloque llamado.

Cuando un bloque lógico llama otro bloque lógico, la CPU ejecuta el código del programa en el bloque llamado. Una vez finalizada la ejecución del bloque llamado, la CPU reanuda la ejecución del bloque que ha efectuado la llamada. El procesamiento continúa con la ejecución de la instrucción siguiente a la llamada de bloque.



Las llamadas de bloque pueden anidarse para crear una estructura más modular. En el ejemplo siguiente, la profundidad de anidamiento es 4: El OB de ciclo de programa más 3 niveles de llamadas de bloques lógicos.



6.3.1 Bloque de organización (OB)

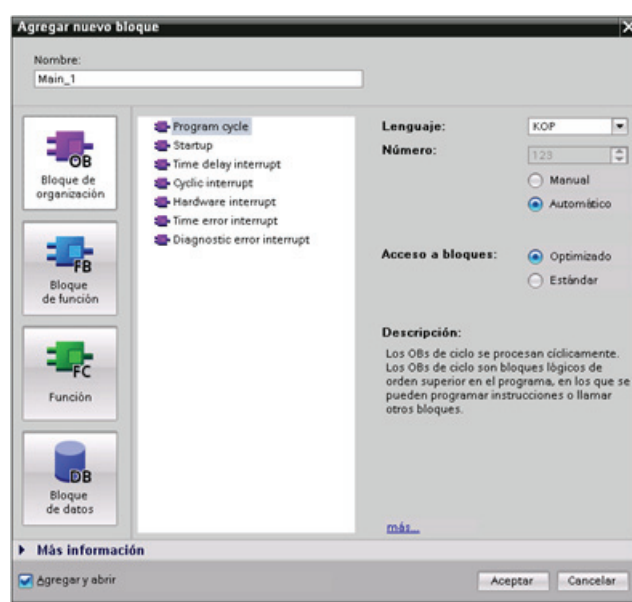
Los bloques de organización permiten estructurar el programa. Estos bloques sirven de interfaz entre el sistema operativo y el programa de usuario. Los OBs son controlados por eventos. Un evento, p. ej. una alarma de diagnóstico o un intervalo, hace que la CPU ejecute un OB. Algunos OBs tienen eventos de arranque y comportamiento en arranque predefinidos.

El OB de ciclo contiene el programa principal. Es posible incluir más de un OB de ciclo en el programa de usuario. En estado operativo RUN, los OBs de ciclo se ejecutan en el nivel de prioridad más bajo y pueden ser interrumpidos por todos los demás tipos de procesamiento del programa. El OB de arranque no interrumpe el OB de ciclo, puesto que la CPU ejecuta el OB de arranque antes de pasar al estado operativo RUN.

Tras finalizar el procesamiento de los OBs de ciclo, la CPU vuelve a ejecutarlos inmediatamente. Esta ejecución cíclica es el tipo de procesamiento "normal" que se utiliza para los controladores lógicos programables. En numerosas aplicaciones, el programa de usuario entero está contenido en un solo OB de ciclo.

Es posible crear otros OBs para ejecutar funciones específicas, tales como el procesamiento de alarmas y el tratamiento de errores, o la ejecución de un código de programa específico en determinados intervalos. Estos OBs interrumpen la ejecución de los OBs de ciclo.

Utilice el diálogo "Agregar nuevo bloque" para crear OBs nuevos en el programa de usuario.



El procesamiento de alarmas siempre está controlado por eventos. Cuando ocurre un evento, la CPU interrumpe la ejecución del programa de usuario y llama el OB configurado para procesar ese evento. Una vez finalizada la ejecución del OB de alarma, la CPU reanuda la ejecución del programa de usuario en el punto de interrupción.

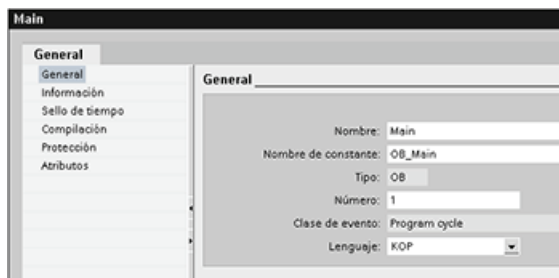
La CPU determina el orden de procesamiento de eventos de alarma según la prioridad asignada a cada OB. Todo evento tiene una prioridad de procesamiento propia. El nivel de prioridad correspondiente dentro de una clase de prioridad determina el orden en que se ejecutan los OB. Varios eventos de alarma pueden combinarse en clases de prioridad. Para más información, consulte la sección relativa a la ejecución del programa de usuario en el capítulo "Principios básicos del PLC" (Página 67).

Crear un OB adicional en una clase de OB

Es posible crear varios OBs para el programa de usuario, incluso para las clases de OB correspondientes a los OBs de ciclo y de arranque. Utilice el cuadro de diálogo "Agregar bloque" para crear un OB. Introduzca el nombre del OB y un número mayor o igual a 200.

Si se crean varios OBs de ciclo de programa, la CPU ejecutará cada uno de ellos en una secuencia numérica, comenzando con el OB con el número menor (p. ej. OB 1): Ejemplo: Tras finalizar el primer OB de ciclo de programa (p. ej. OB1), la CPU ejecuta el siguiente OB de ciclo de programa más alto (p. ej. OB 200).

Configurar el funcionamiento de un OB



Los parámetros operativos de un OB se pueden modificar. Por ejemplo, es posible configurar el parámetro de tiempo de un OB de alarma de retardo o de ciclo.

6.3.2 Función (FC)

Una función (FC) es un bloque lógico que, por lo general, realiza una operación específica en un conjunto de valores de entrada. La FC almacena los resultados de esta operación en posiciones de memoria. Por ejemplo, las FC se utilizan para ejecutar operaciones estándar y reutilizables (como cálculos matemáticos) o funciones tecnológicas (como para controles individuales que utilizan lógica de bits). Una FC también se puede llamar varias veces en diferentes puntos de un programa. Esto facilita la programación de tareas que se repiten con frecuencia.

Una FC no tiene ningún bloque de datos instancia asociado (DB). La FC usa la pila de datos locales para los datos temporales utilizados para calcular la operación. Los datos temporales no se almacenan. Para almacenar los datos de forma permanente es preciso asignar el valor de salida a una posición de memoria global, p. ej. el área de marcas o un DB global.

6.3.3 Bloque de función (FB)

Un bloque de función (FB) es un bloque lógico que utiliza un bloque de datos instancia para sus parámetros y datos estáticos. Los FBs tienen una memoria variable ubicada en un bloque de datos (DB) o DB "instancia". El DB instancia ofrece un bloque de memoria asociado a esa instancia (o llamada) del FB y almacena datos una vez que haya finalizado el FB. Es posible asociar distintos DBs de instancia a diferentes llamadas del FB. Los DBs instancia permiten utilizar un FB genérico para controlar varios dispositivos. El programa se estructura de manera que un bloque lógico llame un FB y un DB instancia. La CPU ejecuta luego el código del programa en ese FB y almacena los parámetros del bloque y los datos locales estáticos en el DB instancia. Cuando finaliza la ejecución del FB, la CPU regresa al bloque lógico que ha llamado el FB. El DB instancia conserva los valores de esa instancia del FB. Estos valores están disponibles para las llamadas posteriores al bloque de función, bien sea en el mismo ciclo o en otros ciclos.

Bloques lógicos reutilizables con memoria asociada

Por lo general, los FBs se utilizan para controlar tareas o dispositivos cuya operación no finaliza dentro de un ciclo. Para almacenar los parámetros operativos de manera que sea posible acceder rápidamente a ellos de un ciclo a otro, todo FB del programa de usuario tiene uno o más DBs instancia. Cuando se llama un FB, se especifica también un DB instancia que contiene los parámetros del bloque y los datos locales estáticos de esa llamada o "instancia" del FB. El DB instancia conserva estos valores una vez finalizada la ejecución del FB.

Si el FB se diseña para realizar tareas de control genéricas, es posible reutilizarlo para varios dispositivos, seleccionando diferentes DB instancia para las distintas llamadas del FB.

Un FB guarda los parámetros de entrada, salida, entrada y salida y estáticos en un DB de instancia.

Asignar el valor inicial en el DB instancia

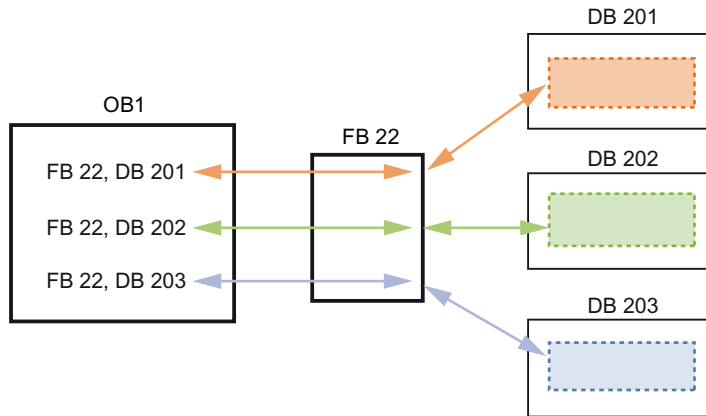
El DB instancia almacena un valor predeterminado y un valor inicial para cada parámetro. El valor inicial proporciona el valor que debe utilizarse cuando se ejecuta el FB.

Posteriormente, el valor inicial puede modificarse durante la ejecución del programa de usuario.

La interfaz del FB también dispone de una columna "Valor predeterminado" que permite asignar un nuevo valor inicial a los parámetros mientras se escribe el código del programa. Este valor predeterminado en el FB se transfiere posteriormente al valor inicial en el DB instancia asociado. Si no se asigna ningún valor inicial nuevo a un parámetro en la interfaz del FB, el valor predeterminado del DB instancia se copia en el valor inicial.

Utilizar un solo FB con DBs

La figura siguiente muestra un OB que llama un FB tres veces, utilizando un bloque de datos diferente para cada llamada. Esta estructura permite que un FB genérico controle varios dispositivos similares (p. ej. motores), asignando un bloque de datos instancia diferente a cada llamada de los distintos dispositivos. Cada DB instancia almacena los datos (p. ej. velocidad, tiempo de aceleración y tiempo de operación total) de un dispositivo en particular.



En este ejemplo, el FB 22 controla tres dispositivos diferentes. El DB 201 almacena los datos operativos del primer dispositivo, el DB 202, los del segundo y, el DB 203, los del tercero.

6.3.4 Bloque de datos (DB)

Los bloques de datos (DB) se crean en el programa de usuario para almacenar los datos de los bloques lógicos. Todos los bloques del programa de usuario pueden acceder a los datos en un DB global. En cambio, un DB instancia almacena los datos de un bloque de función (FB) específico.

Los datos almacenados en un DB no se borran cuando finaliza la ejecución del bloque lógico asociado. Hay dos tipos de DBs, a saber:

- Un DB global almacena los datos de los bloques lógicos en el programa. Cualquier OB, FB o FC puede acceder a los datos en un DB global.
- Un DB instancia almacena los datos de un FB específico. La estructura de los datos en un DB instancia refleja los parámetros (Input, Output e InOut) y los datos estáticos del FB. (La memoria temporal del FB no se almacena en el DB instancia.)

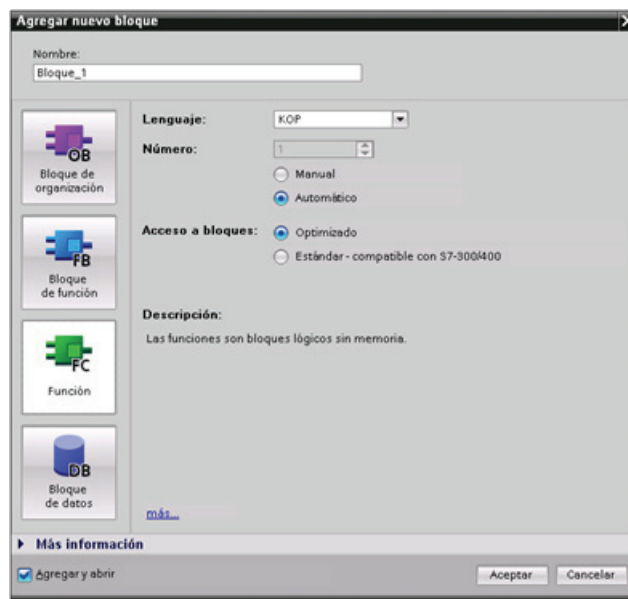
Nota

Aunque el DB instancia refleja los datos de un FB específico, cualquier bloque lógico puede acceder a los datos en un DB instancia.

Un DB se puede configurar de manera que sea de sólo lectura:

1. Haga clic con el botón derecho del ratón en el DB en el árbol del proyecto y seleccione "Propiedades" del menú contextual.
2. En el cuadro de diálogo "Propiedades", seleccione "Atributos".
3. Seleccione la opción "Bloque de datos protegido contra escritura en el dispositivo" y haga clic en "Aceptar".

Crear bloques lógicos reutilizables



Utilice el diálogo "Agregar nuevo bloque" en "Bloques de programa" en el árbol del proyecto para crear OBs, FBs, FCs y DBs globales.

Al crear un bloque lógico se selecciona el lenguaje de programación para el bloque. El lenguaje de un DB no se selecciona, puesto que éste sólo almacena datos.

6.4 Principios básicos de la coherencia de datos

La CPU conserva la coherencia de datos de todos los tipos de datos simples (p. ej. Word o DWord) y de todas las estructuras definidas por el sistema (p. ej. IEC_TIMERS o DTL). La lectura o escritura de valores no se puede interrumpir. (Por ejemplo, la CPU protege el acceso a un valor de palabra doble (DWord) hasta que se hayan leído o escrito los cuatro bytes de la palabra doble.) Para garantizar que los OBs de ciclo y de alarma no puedan escribir simultáneamente en la misma posición de memoria, la CPU no ejecuta un OB de alarma hasta que no haya finalizado la operación de lectura o escritura en el OB de ciclo.

Si el programa de usuario comparte varios valores entre un OB de ciclo y un OB de alarma en la memoria, dicho programa debe garantizar asimismo que estos valores se modifiquen o lean de forma coherente. Utilice las instrucciones DIS_AIRT (deshabilitar tratamiento de eventos de alarma) y EN_AIRT (habilitar tratamiento de eventos de alarma) del OB de ciclo de programa para proteger cualquier acceso a los valores compartidos.

- Inserte una instrucción DIS_AIRT en el bloque lógico para garantizar que un OB de alarma no pueda ejecutarse durante la operación de lectura o escritura.
- Inserte las instrucciones que leen o escriben los valores que podrían ser modificados por un OB de alarma.
- Inserte una instrucción EN_AIRT al final de la secuencia para cancelar la instrucción DIS_AIRT y permitir la ejecución del OB de alarma.

Una petición de comunicación de un dispositivo HMI o de otra CPU puede interrumpir asimismo la ejecución del OB de ciclo de programa. Las peticiones de comunicación también pueden causar problemas relacionados con la coherencia de datos. La CPU garantiza que las instrucciones del programa de usuario lean y escriban coherentemente los tipos de datos simples. Puesto que las comunicaciones interrumpen el programa de usuario de forma periódica, no es posible garantizar que el dispositivo HMI actualice simultáneamente varios valores en la CPU. Por ejemplo, los valores visualizados en la pantalla de un HMI podrían provenir de diferentes ciclos de la CPU.

Las instrucciones PtP (punto a punto), PROFINET (p. ej. TSEND_C y TRCV_C), PROFINET E/S descentralizadas y PROFIBUS E/S descentralizadas (Página 285) transfieren búferes de datos que podrían interrumpirse. La coherencia de datos de los búferes debe asegurarse evitando operaciones de lectura y escritura en los búferes, tanto en el OB de ciclo como en un OB de alarma. Si es necesario modificar los valores de los búferes para estas instrucciones en un OB de alarma, utilice una instrucción DIS_AIRT para retardar las alarmas (un OB de alarma o una alarma de comunicación de un HMI u otra CPU) hasta que se ejecute una instrucción EN_AIRT.

Nota

La utilización de la instrucción DIS_AIRT retarda el procesamiento de los OBs de alarma hasta que se ejecuta la instrucción EN_AIRT, lo que afecta la latencia de alarmas (tiempo que transcurre desde un evento hasta que se ejecuta el OB de alarma) del programa de usuario.

6.5 Lenguaje de programación

STEP 7 ofrece los lenguajes de programación estándar siguientes para S7-1200:

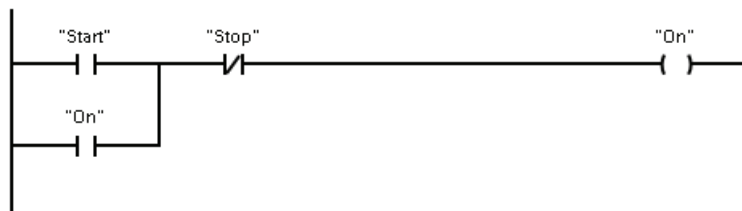
- KOP (esquema de contactos) es un lenguaje de programación gráfico. Su representación se basa en esquemas (Página 159) de circuitos.
- FUP (diagrama de funciones) es un lenguaje de programación que se basa en los símbolos lógicos gráficos empleados en el álgebra (Página 160) booleana.
- SCL (structured control language) es un lenguaje de programación de alto nivel basado en texto (Página 160).

Al crear un bloque lógico, se debe seleccionar el lenguaje de programación que empleará dicho bloque.

El programa de usuario puede emplear bloques lógicos creados con cualquiera de los lenguajes de programación.

6.5.1 Esquema de contactos (KOP)

Los elementos de un esquema de circuitos, tales como los contactos normalmente cerrados y normalmente abiertos, así como las bobinas, se combinan para formar segmentos.



Para crear la lógica de operaciones complejas, es posible insertar ramas para los circuitos paralelos. Las ramas paralelas se abren hacia abajo o se conectan directamente a la barra de alimentación. Las ramas se terminan hacia arriba.

KOP ofrece instrucciones con cuadros para numerosas funciones, p. ej. matemáticas, temporizadores, contadores y transferencia.

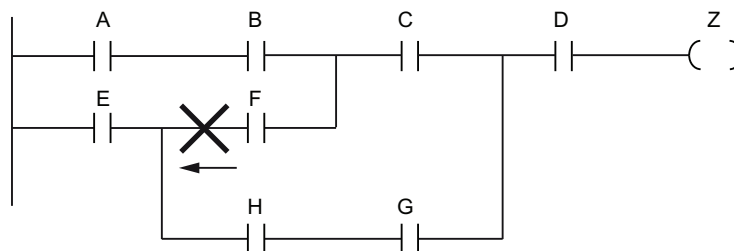
STEP 7 no limita el número de instrucciones (filas y columnas) de un segmento KOP.

Nota

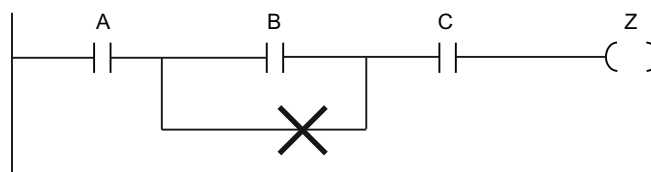
Todo segmento KOP debe terminar con una bobina o cuadro.

Tenga en cuenta las reglas siguientes al crear segmentos KOP:

- No se permite programar ramas que puedan ocasionar un flujo invertido de la corriente.

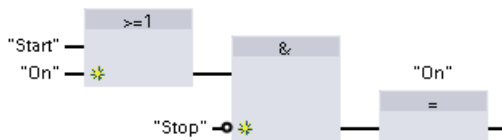


- No se permite programar ramas que causen cortocircuitos.



6.5.2 Diagrama de funciones (FUP)

Al igual que KOP, FUP es un lenguaje de programación gráfico. La representación de la lógica se basa en los símbolos lógicos gráficos del álgebra booleana.



Para crear la lógica de operaciones complejas, inserte ramas paralelas entre los cuadros.

Las funciones matemáticas y otras operaciones complejas pueden representarse directamente en combinación con los cuadros lógicos.

STEP 7 no limita el número de instrucciones (filas y columnas) de un segmento FUP.

6.5.3 SCL

El lenguaje de control estructurado (SCL, Structured Control Language) es un lenguaje de programación de alto nivel basado en PASCAL para las CPU de SIMATIC S7. SCL soporta la estructura de bloques de STEP 7 (Página 152). También es posible incluir bloques de programa escritos en SCL con bloques de programa escritos en KOP y FUP.

Las instrucciones de SCL emplean operadores de programación estándar, p. ej. para asignación (:=) o para funciones matemáticas (+ para la suma, - para la resta, * para la multiplicación y / para la división). SCL también utiliza operaciones de control de programa PASCAL estándar, tales como IF-THEN-ELSE, CASE, REPEAT-UNTIL, GOTO y RETURN. Es posible utilizar cualquier referencia PASCAL para elementos sintácticos del lenguaje de programación SCL. Muchas otras instrucciones de SCL, como los temporizadores y contadores, se corresponden con las instrucciones de KOP y FUP. Para más información sobre determinadas instrucciones, véanse las instrucciones específicas de las secciones correspondientes a Instrucciones básicas (Página 179) e Instrucciones avanzadas (Página 257).

Es posible designar cualquier tipo de bloque (OB, FB o FC) para utilizar el lenguaje de programación en el momento de crear el bloque. STEP 7 cuenta con un editor de programas SCL que incluye los elementos siguientes:

- Sección de interface para definir los parámetros del bloque lógico
- Sección de código para el código del programa
- Árbol de instrucciones que contiene las instrucciones SCL soportadas por la CPU

El código SCL para la instrucción se introduce directamente en la sección de código. Para obtener instrucciones más complejas, basta con arrastrar las instrucciones SCL del árbol de instrucciones y soltarlas en el programa. También se puede emplear cualquier editor de texto para crear un programa SCL y, a continuación, importar el archivo a STEP 7.

Interfaz			
	Nombre	Tipo de datos	Comentario
1	Input		
2	StartStopSwitch	Bool	
3	Output		
4	RunYesNo	Bool	
5	InOut		
6	<agregar>		
7	Temp		
8	<agregar>		
9	Return		
10	Ret_Val	Void	


```

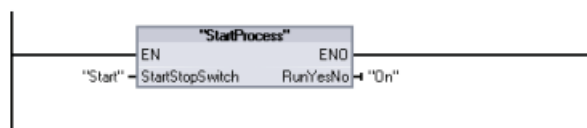
1 IF condition THEN
2   // Statement section IF
3   ;
4 END_IF;

```

En la sección del bloque lógico SCL se pueden declarar los tipos de parámetros siguientes:

- Input, Output, InOut y Ret_Val: estos parámetros definen las variables de entrada, las variables de salida y el valor de retorno del bloque lógico. El nombre de la variable introducida en este punto se emplea de forma local durante la ejecución del bloque lógico. Normalmente, no se emplea el nombre de variables globales en la tabla de variables.
- Estáticos (solo FBs; la figura de arriba corresponde a un FC): las variables estáticas se utilizan para almacenar resultados intermedios estáticos en el bloque de datos instancia. Los datos estáticos se retienen hasta que se sobrescriben, lo cual puede ocurrir después de varios ciclos. Los nombres de los bloques, los cuales se llaman en este bloque lógico como multiinstancia, también se almacenan en los datos locales estáticos.
- Temp: estos parámetros son variables temporales que se emplean durante la ejecución del bloque lógico.

Si se llama el bloque lógico SCL desde otro bloque lógico, los parámetros del bloque lógico SCL aparecen como entradas o salidas.



En este ejemplo, las variables de "Start" y "On" (de la tabla de variables de proyecto) se corresponden con "StartStopSwitch" y "RunYesNo" en la tabla de declaración del programa SCL.

Construir una expresión SCL

Una expresión SCL es una fórmula para calcular un valor. La expresión consiste en operandos y operadores (p. ej. *, /, + o -). Los operandos pueden ser variables, constantes o expresiones.

La evaluación de la expresión se desarrolla en un orden determinado, que está definido por los siguientes factores:

- Cada operador tiene una prioridad predefinida y la operación con mayor prioridad se ejecuta en primer lugar.
- Si tienen la misma prioridad, los operadores se procesan en una secuencia de izquierda a derecha.
- Se utilizan paréntesis para designar una serie de operadores que deben evaluarse conjuntamente.

El resultado de una expresión se puede utilizar para asignar un valor a una variable utilizada por el programa, como una condición que debe ser considerada en una instrucción de control, o como parámetros para otra instrucción SCL o para llamar un bloque lógico.

Tabla 6- 2 Operadores en SCL

Tipo	Operación	Operador	Prioridad
Paréntesis	(<i>Expresión</i>)	(,)	1
Matemáticas	Alimentación	**	2
	Signo (más unario)	+	3
	Signo (menos unario)	-	3
	Multipliación	*	4
	División	/	4
	Modulo	MOD	4
	Suma	+	5
	Resta	-	5
Comparación	Menor	<	6
	Menor o igual	<=	6
	Mayor	>	6
	Mayor o igual	>=	6
	Igual	=	7
	Diferente	<>	7
Lógica con bits	Negación (unaria)	NOT	3
	Combinación Y lógica	Y o &	8
	Combinación lógica O-exclusiva	XOR	9
	Combinación lógica O	O	10
Parametrización	Parametrización	:=	11

Como lenguaje de programación de alto nivel, SCL utiliza instrucciones estándar para tareas básicas:

- Instrucción de asignación: :=
- Funciones matemáticas: +, -, * y /
- Direccinamiento de variables globales (variables): "<nombre de variable>" (nombre de la variable o nombre del bloque de datos encerrado entre comillas dobles)
- Direccinamiento de variables locales: #<nombre de variable> (nombre de la variable precedido por el símbolo "#")

Los ejemplos siguientes muestran diferentes expresiones para diversos usos.

<code>"C" := #A+#B;</code>	Asigna la suma de dos variables locales a una variable
<code>"Data_block_1".Tag := #A;</code>	Asignación a una variable de bloque de datos
<code>IF #A > #B THEN "C" := #A;</code>	Condición de la instrucción IF-THEN
<code>"C" := SQRT (SQR (#A) + SQR (#B));</code>	Parámetros de la instrucción SQRT

Los operadores aritméticos pueden procesar varios tipos de datos numéricos. El tipo de datos resultante lo determina el tipo de datos de los operandos más significativos. Por ejemplo, una operación de multiplicación que emplea un operando INT y un operando REAL da como resultado un valor REAL.

Instrucciones de control

Una instrucción de control es un tipo de expresión SCL especializada, que realiza las siguientes tareas:

- Ramificación del programa
- Repetición de secciones del código del programa SCL
- Salto a otras partes del programa SCL
- Ejecución condicionada

Las instrucciones de control de SCL incluyen IF-THEN, CASE-OF, FOR-TO-DO, WHILE-DO, REPEAT-UNTIL, CONTINUE, GOTO y RETURN.

Una instrucción única generalmente ocupa una línea de código. Es posible introducir varias instrucciones en una línea o dividir una instrucción en varias líneas de código para que el código resulte más comprensible. Los separadores (como tabulaciones, saltos de línea o espacios adicionales) se ignoran durante la comprobación de sintaxis. Una instrucción END termina la instrucción de control.

Los siguientes ejemplos muestran una instrucción de control FOR-TO-DO. (Ambas formas de codificación son válidas sintácticamente.)

```
FOR x := 0 TO max DO sum := sum + value(x); END_FOR;
FOR x := 0 TO max DO
    sum := sum + value(x);
END_FOR;
```

Una instrucción de control también puede llevar una etiqueta. Una etiqueta se inserta colocando dos puntos (:) al comienzo de la instrucción:

```
Etiqueta: <Instrucción>;
```

La ayuda en línea de STEP 7 ofrece una completa referencia del lenguaje de programación SCL.

Condiciones

Una condición es una expresión de comparación o una expresión lógica cuyo resultado es del tipo BOOL (con el valor TRUE o bien FALSE). El siguiente ejemplo muestra condiciones de varios tipos.

<code>#Temperatura > 50</code>	Expresión de relación
<code>#Contador <= 100</code>	
<code>#CHAR1 < 'S'</code>	
<code>(#Alpha <> 12) AND NOT #Beta</code>	Comparación y expresión lógica
<code>5 + #Alpha</code>	Expresión aritmética

Una condición puede utilizar expresiones aritméticas:

- La condición de la expresión es TRUE si el resultado es cualquier valor diferente a cero.
- La condición de la expresión es FALSE si el resultado es igual a cero.

Direccionamiento

Igual que en el caso de KOP y FUP, SCL permite utilizar variables (direccionamiento simbólico) o direcciones absolutas en el programa de usuario. SCL también permite utilizar una variable como índice de matriz.

Direccionamiento absoluto

`I0.0`
`MB100`

Direccionamiento simbólico

<code>"PLC_Tag_1"</code>	Variable en la tabla de variables PLC
<code>"Data_block_1".Tag_1</code>	Variable en un bloque de datos
<code>"Data_block_1".MyArray[#i]</code>	Elemento de una matriz de bloque de datos

Direccionamiento indexado con instrucciones PEEK y POKE

SCL ofrece instrucciones PEEK y POKE que permiten leer de bloques de datos, E/S o memoria, o escribir en ellos. Hay que proporcionar parámetros de offsets de byte, o de bit, específicos para la operación.

Nota

Para usar las instrucciones PEEK y POKE con bloques de datos deben usarse bloques de datos estándar (no optimizados). Nótese que las instrucciones PEEK y POKE solo transfieren datos. Desconocen los tipos de datos que hay en las direcciones.

```
PEEK(area:=_in_,
      dbNumber:=_in_,
      byteOffset:=_in_);
```

Lee el byte referenciado por byteOffset del bloque de datos, E/S o área de memoria referenciados.

Ejemplo de referencia al bloque de datos:

```
%MB100 := PEEK(area:=16#84,
               dbNumber:=1, byteOffset:=#i);
```

Ejemplo de referencia a la entrada IB3:

```
%MB100 := PEEK(area:=16#81,
               dbNumber:=0, byteOffset:=#i); // when
#i = 3
```

```
PEEK_WORD(area:=_in_,
           dbNumber:=_in_,
           byteOffset:=_in_);
```

Lee la palabra referenciada por byteOffset del bloque de datos, E/S o área de memoria referenciados.

Ejemplo:

```
%MW200 := PEEK_WORD(area:=16#84,
                    dbNumber:=1, byteOffset:=#i);
```

```
PEEK_DWORD(area:=_in_,
            dbNumber:=_in_,
            byteOffset:=_in_);
```

Lee la doble palabra referenciada por byteOffset del bloque de datos, E/S o área de memoria referenciados.

Ejemplo:

```
%MD300 := PEEK_DWORD(area:=16#84,
                     dbNumber:=1, byteOffset:=#i);
```

```
PEEK_BOOL(area:=_in_,
           dbNumber:=_in_,
           byteOffset:=_in_,
           bitOffset:=_in_);
```

Lee un booleano referenciado por el bitOffset y el byteOffset del bloque de datos, E/S o área de memoria referenciados.

Ejemplo:

```
%MB100.0 := PEEK_BOOL(area:=16#84,
                      dbNumber:=1, byteOffset:=#ii,
                      bitOffset:=#j);
```

```
POKE(area:=_in_,
      dbNumber:=_in_,
      byteOffset:=_in_,
      value:=_in_);
```

Escribe el valor (Byte, Word o DWord) en el byteOffset referenciado del bloque de datos, E/S o área de memoria referenciados.

Ejemplo de referencia al bloque de datos:

```
POKE(area:=16#84, dbNumber:=2,
      byteOffset:=3, value:="Tag_1");
```

Ejemplo de referencia a la salida QB3:

```
POKE(area:=16#82, dbNumber:=0,
      byteOffset:=3, value:="Tag_1");
```

```
POKE_BOOL(area:=_in_,
           dbNumber:=_in_,
           byteOffset:=_in_,
           bitOffset:=_in_,
           value:=_in_);

POKE_BLK(area_src:=_in_,
          dbNumber_src:=_in_,
          byteOffset_src:=_in_,
          area_dest:=_in_,
          dbNumber_dest:=_in_,
          byteOffset_dest:=_in_,
          count:=_in_);
```

Escribe el valor booleano en el bitOffset y byteOffset referenciados del bloque de datos, E/S o área de memoria referenciados.

Ejemplo:

```
POKE_BOOL(area:=16#84, dbNumber:=2,
           byteOffset:=3, bitOffset:=5,
           value:=0);
```

Escribe el número de "recuento" de bytes empezando por el offset de bytes referenciado del bloque de datos, E/S o área de memoria de origen referenciados en el byteOffset referenciado del bloque de datos, E/S o área de memoria de destino referenciados.

Ejemplo:

```
POKE_BLK(area_src:=16#84,
          dbNumber_src:=#src_db,
          byteOffset_src:=#src_byte,
          area_dest:=16#84,
          dbNumber_dest:=#src_db,
          byteOffset_dest:=#src_byte,
          count:=10);
```

Para las instrucciones PEEK y POKE, se aplican los valores siguientes de los parámetros "area", "area_src" y "area_dest". Para otras áreas distintas de los bloques de datos, el parámetro dbNumber debe ser 0.

16#81	I
16#82	Q
16#83	M
16#84	DB

Llamar otros bloques lógicos desde el programa SCL

Para llamar otro bloque lógico en el programa de usuario, basta con introducir el nombre (o dirección absoluta) del FB o la FC que contiene los parámetros. En el caso de un FB, también hay que indicar un DB de instancia para llamarlo con el FB.

<Nombre de DB> (lista de parámetros) Llamar como instancia única
 <#Nombre de instancia> (lista de parámetros) Llamar como multiinstancia

```
"MyDB" (MyInput:=10, MyInOut:="Tag1");
```

<Nombre de FC> (Lista de parámetros) Llamada estándar
 <Operando>:=<Nombre de FC> (Lista de parámetros) Llamar en una expresión

```
"MyFC" (MyInput:=10, MyInOut:="Tag1") ;
```

También se pueden arrastrar bloques desde el árbol de navegación al editor del programa SCL y completar la asignación de parámetros.

6.5.4 EN y ENO para KOP, FUP y SCL

Determinar el "flujo de corriente" (EN y ENO) para una instrucción

Algunas instrucciones (p. ej. matemáticas y de transferencia) proporcionan parámetros para EN y ENO. Estos parámetros se refieren al flujo de corriente en KOP y FUP y determinan si la instrucción se ejecuta en ese ciclo. SCL también permite activar el parámetro ENO para un bloque lógico.

- EN (Enable In) es una entrada booleana. Debe haber flujo de corriente (EN = 1) en esta entrada para que la instrucción de cuadro pueda ejecutarse. Si la entrada EN de un cuadro KOP se conecta directamente a la barra de potencia izquierda, la instrucción se ejecutará siempre.
- ENO (Enable Out) es una salida booleana. Si el cuadro tiene flujo de corriente en la entrada EN y ejecuta su función sin errores, la salida ENO transfiere el flujo de corriente (ENO = 1) al siguiente elemento. Si se detecta un error en la ejecución del cuadro, el flujo de corriente se interrumpe (ENO = 0) en la instrucción de cuadro que ha generado el error.

Tabla 6- 3 Operandos para EN y ENO

Editor de programas	Entradas/salidas	Operandos	Tipo de datos
KOP	EN, ENO	Flujo de corriente	Bool
FUP	EN	I, I:P, Q, M, DB, Temp, flujo de corriente	Bool
	ENO	Flujo de corriente	Bool
SCL	EN ¹	TRUE, FALSE	Bool
	ENO ²	TRUE, FALSE	Bool

¹ El uso de EN sólo está disponible para bloques FB.

² El uso de ENO con el bloque lógico SCL es opcional. Hay que configurar el compilador SCL de forma que active ENO cuando acabe el bloque lógico.

Configuración de SCL para definir ENO

Para configurar el compilador SCL para que active ENO proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
2. Expanda las propiedades "Programación PLC" y seleccione "SCL (Structured Control Language)".
3. Seleccione la opción "Activar ENO automáticamente".

Efecto de los parámetros Ret_Val o Status sobre ENO

Algunas instrucciones, como las de comunicación o de conversión de cadena, ofrecen un parámetro de salida que incluye información relacionada con el procesamiento de la instrucción. Así, por ejemplo, algunas instrucciones ofrecen un parámetro Ret_Val (valor de retorno), que suele ser un tipo de datos Int que incluye información de estado en un rango de -32768 a +32767. Otras instrucciones disponen de un parámetro Status que suele ser un tipo de datos Word que almacena información de estado en un rango de valores hexadecimales de 16#0000 a 16#FFFF. El valor numérico almacenado en un parámetro Ret_Val o Status determina el estado de ENO para dicha instrucción.

- Ret_Val: un valor de 0 a 32767 pone típicamente ENO = 1 (o TRUE). Un valor de -32768 a -1 pone típicamente ENO = 0 (o FALSE). Para evaluar Ret_Val, cambie la representación a hexadecimal.
- Status: un valor de 16#0000 a 16#7FFF pone típicamente ENO = 1 (o TRUE). Un valor de 16#8000 a 16#FFFF pone típicamente ENO = 0 (o FALSE).

Las instrucciones cuya ejecución dura más de un ciclo suelen disponer de un parámetro Busy (Bool) para indicar que la instrucción está activa pero aún no se ha completado su ejecución. Dichas instrucciones también suelen incluir un parámetro Done (Bool) y un parámetro Error (Bool). Done indica que la instrucción se ha completado sin errores y Error indica que la instrucción se ha completado con una condición de error.

- Cuando Busy = 1 (o TRUE), ENO = 1 (o TRUE).
- Cuando Done = 1 (o TRUE), ENO = 1 (o TRUE).
- Cuando Error = 1 (o TRUE), ENO = 0 (o FALSE).

Consulte también

Instrucciones "Comprobar validez" y "Comprobar invalidez" (Página 204)

6.6 Protección

6.6.1 Protección de acceso a la CPU

La CPU ofrece tres niveles de seguridad para restringir el acceso a determinadas funciones. Al configurar el nivel de protección y la contraseña de una CPU, se limitan las funciones y áreas de memoria accesibles sin introducir una contraseña.

La contraseña distingue entre mayúsculas y minúsculas.

Para configurar la contraseña, proceda del siguiente modo:

1. Seleccione la CPU en la "Configuración de dispositivos".
2. Seleccione la ficha "Propiedades" en la ventana de inspección.
3. Elija la propiedad "Protección" para seleccionar el nivel de protección e introducir una contraseña.

Todo nivel permite acceder a ciertas funciones sin introducir una contraseña. El ajuste predeterminado de la CPU es "sin restricción" y "sin protección por contraseña". Para restringir el acceso a una CPU, es preciso configurar sus propiedades e introducir la contraseña.

Si la contraseña se introduce a través de una red, esto no afecta la protección por contraseña de la CPU. La protección por contraseña no es aplicable a la ejecución de las instrucciones del programa de usuario incluyendo las funciones de comunicación. Si se introduce la contraseña correcta es posible acceder a todas las funciones.

El nivel de protección de la CPU no restringe la comunicación entre PLCs (mediante instrucciones de comunicación en los bloques lógicos). Tampoco se restringen las funciones HMI.

Tabla 6- 4 Niveles de protección de la CPU

Nivel de protección	Restricciones de acceso
Sin protección	Permite el acceso completo sin protección por contraseña.
Protección contra escritura	Ofrece acceso a los dispositivos HMI y permite toda la comunicación entre PLCs sin protección por contraseña. La contraseña se requiere para modificar (escribir en) la CPU y cambiar su estado operativo (RUN/STOP).
Protección contra lectura/escritura	Ofrece acceso a los dispositivos HMI y permite toda la comunicación entre PLCs sin protección por contraseña. La contraseña se requiere para leer los datos de la CPU, modificar (escribir en) la CPU y cambiar su estado operativo (RUN/STOP).

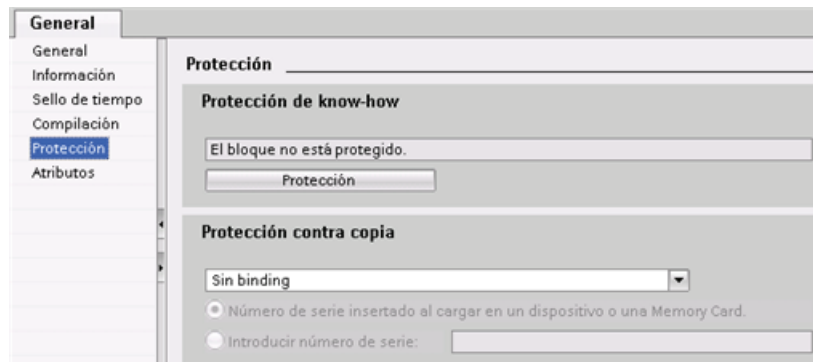
6.6.2 Protección de know-how

La protección de know-how impide el acceso no autorizado a uno o más bloques lógicos (OB, FB, FC o DB) del programa. Es posible crear una contraseña para limitar el acceso al bloque lógico. La protección por contraseña impide que el bloque lógico sea leído o modificado sin autorización. Si no se introduce la contraseña, sólo es posible leer la siguiente información del bloque lógico:

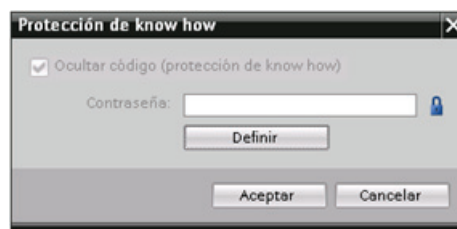
- Título, comentario y propiedades del bloque
- Parámetros de transferencia (IN, OUT, IN_OUT, Return)
- Estructura de llamadas del programa
- Variables globales en las referencias cruzadas (sin información acerca de la ubicación); las variables locales se ocultan

Si un bloque se configura para que tenga protección de "know-how", no será posible acceder al código del bloque si no se introduce la contraseña.

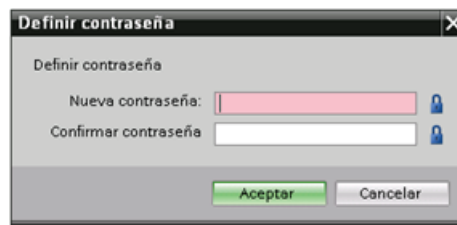
Utilice la Task Card "Propiedades" del bloque lógico para configurar la protección de know-how de dicho bloque. Después de abrir el bloque lógico, seleccione "Protección" en "Propiedades".



1. En las propiedades del bloque lógico, haga clic en el botón "Protección" para que aparezca el cuadro de diálogo "Protección de know-how".
2. Haga clic en el botón "Definir" para introducir la contraseña.



Después de introducir y confirmar la contraseña, haga clic en "Aceptar".

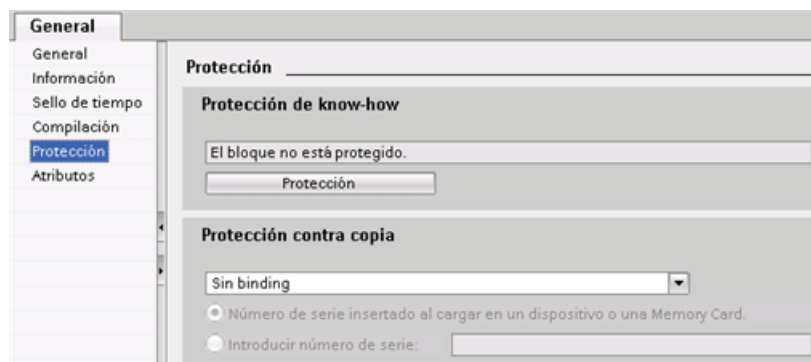


6.6.3 Protección anticopia

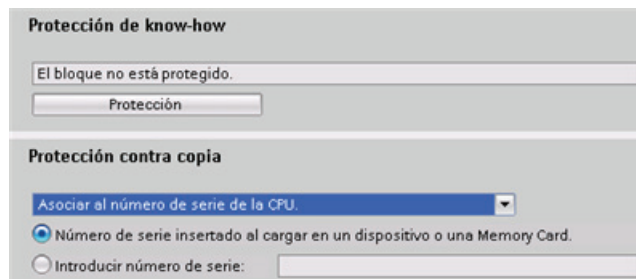
Una función de protección adicional permite enlazar el programa o los bloques lógicos para usarlos con una CPU o Memory Card determinada. Esta función se usa especialmente para proteger la propiedad intelectual. Al enlazar un programa o bloque a un dispositivo específico, sólo se permite usar dicho programa o bloque lógico con una CPU o Memory Card determinada. Esta función permite distribuir un programa o bloque lógico de forma electrónica (por ejemplo, a través de Internet o correo electrónico) o a través del envío de un cartucho de memoria.

Utilice la Task Card "Propiedades" del bloque lógico para enlazar el bloque con una CPU o Memory Card determinada.

1. Después de abrir el bloque lógico, seleccione "Protección".



2. En la opción "Protección contra copia" de la lista desplegable, seleccione la opción para enlazar el bloque lógico con una Memory Card o con una CPU determinada.



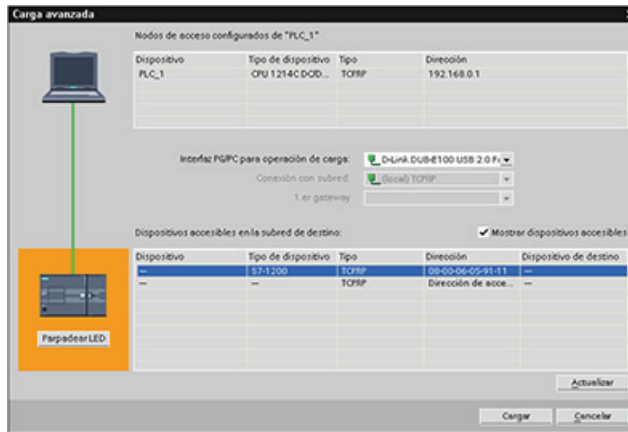
3. Seleccione el tipo de protección anticopia e introduzca el número de serie de la CPU o Memory Card.

Nota

El número de serie distingue entre mayúsculas y minúsculas.

6.7 Cargar los elementos del programa en la CPU

Los elementos del proyecto se pueden cargar desde la programadora a la CPU. Al cargar un proyecto en la CPU, el programa de usuario (OBs, FCs, FBs y DBs) se almacena en la memoria no volátil de la CPU.



El proyecto se puede cargar de la programadora en la CPU desde cualquiera de las ubicaciones siguientes:

- "Árbol del proyecto": Haga clic con el botón derecho del ratón en el elemento de programa deseado y elija el comando "Cargar en dispositivo" del menú contextual.
- Menú "Online": Elija el comando "Cargar en dispositivo".
- Barra de herramientas: Haga clic en el botón "Cargar en dispositivo".

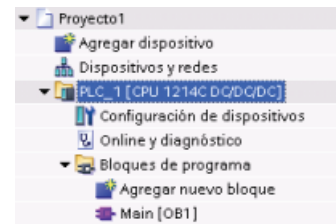
6.8 Cargar desde la CPU

6.8.1 Copia de elementos del proyecto

También se pueden copiar bloques de programa de una CPU online o una Memory Card conectada a la programadora.

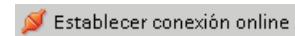
Prepare el proyecto offline para los bloques de programa copiados:

1. Agregue un dispositivo CPU que coincida con la CPU online.
2. Expanda el nodo de la CPU una vez de manera que se vea la carpeta "Bloques de programa".



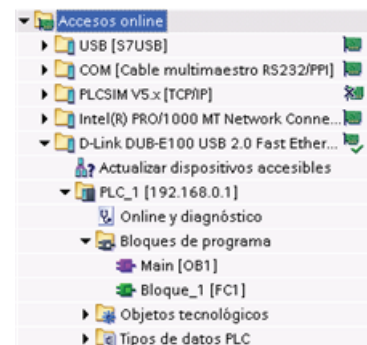
Para cargar los bloques de programa desde la CPU online al proyecto offline, siga estos pasos:

1. Haga clic en la carpeta "Bloques de programa" del proyecto offline.
2. Haga clic en el botón "Establecer conexión online".
3. Haga clic en el botón "Cargar".
4. Confirme la decisión en el cuadro de diálogo Cargar (Página 711).

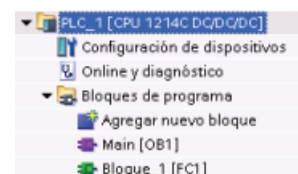


Como alternativa al método anterior, siga estos pasos:

1. Desde el árbol del proyecto, expanda el nodo "Accesos online" para seleccionar los bloques de programa de la CPU online:
2. Expanda el nodo de la red y haga doble clic en "Actualizar dispositivos accesibles".
3. Expanda el nodo de la CPU.
4. Arrastre la carpeta "Bloques de programa" desde la CPU online y suéltela en la carpeta "Bloques de programa" del proyecto offline.
5. En el cuadro de diálogo "Vista preliminar para cargar del dispositivo", seleccione el cuadro para continuar y, a continuación, haga clic en el botón "Cargar de dispositivo".



Cuando finaliza la carga, todos los bloques de programa, bloques tecnológicos y variables se muestran en el área offline.



Nota

Se pueden copiar los bloques de programa de la CPU online en un programa existente. La carpeta "Bloques de programa" del proyecto offline no tiene que estar vacía. No obstante, el programa existente se elimina y se sustituye por el programa de usuario de la CPU online.

6.8.2 Utilización de la función de comparación

El editor de comparación (Página 718) de STEP 7 permite buscar diferencias entre los proyectos online y offline. Esto resulta útil antes de cargar desde la CPU.

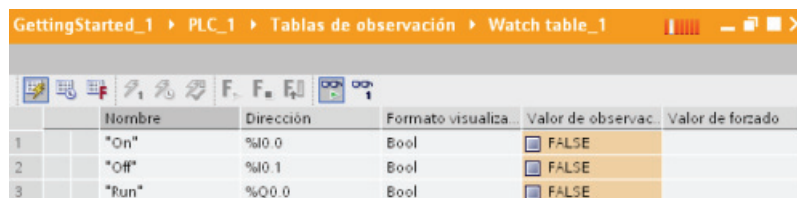
6.9 Depurar y comprobar el programa

6.9.1 Vigilar y modificar datos de la CPU

Tal y como muestra la tabla siguiente, los valores de la CPU online pueden vigilarse y modificarse.

Tabla 6- 5 Vigilar y modificar datos con STEP 7

Editor	Vigilar	Modificar	Forzado permanente
Tabla de observación	Sí	Sí	No
Tabla de forzado permanente	Sí	No	Sí
Editor de programas	Sí	Sí	No
Tabla de variables	Sí	No	No
Editor de DB	Sí	No	No



Vigilar con una tabla de observación



Vigilar con el editor KOP.

Véase el capítulo "Online y diagnóstico" para más información sobre cómo vigilar y modificar datos de la CPU (Página 719).

6.9.2 Tablas de observación y tablas de forzado

Las "tablas de observación" se utilizan para observar y forzar los valores del programa de usuario que se está ejecutando en la CPU online. Es posible crear y guardar diferentes tablas de observación en el programa para soportar distintos entornos de test. Esto permite reproducir los tests durante la puesta en marcha, o bien para fines de servicio y mantenimiento.

Una tabla de observación permite observar e interactuar con la CPU mientras ésta ejecuta el programa de usuario. Es posible ver o cambiar los valores no sólo de las variables de los bloques lógicos y bloques de datos, sino también de las áreas de memoria de la CPU, incluyendo las entradas y salidas (I y Q), entradas de periferia (I:P), marcas (M) y bloques de datos (DB).

La tabla de observación permite habilitar las salidas físicas (Q:P) de una CPU en estado operativo STOP. Por ejemplo, es posible asignar valores específicos a las salidas al comprobar el cableado de la CPU.

STEP 7 también dispone de una tabla para "forzar permanentemente" una variable a un valor concreto. Para más información sobre el forzado permanente, consulte el apartado Forzado permanente de valores en la CPU (Página 726) del capítulo "Online y diagnóstico".

Nota

Los valores de forzado permanente se guardan en la CPU y no en la tabla de visualización.

No se puede forzar permanentemente una entrada (o dirección "I"). No obstante, sí que es posible forzar permanentemente una entrada de periferia. Para forzar permanentemente una entrada de periferia, agregue ":P" a la dirección (por ejemplo: "On:P").

6.9.3 Referencia cruzada para mostrar la utilización

La ventana de inspección muestra referencias cruzadas sobre cómo un objeto seleccionado se utiliza en todo el proyecto, p. ej. en el programa de usuario, la CPU y los dispositivos HMI. La ficha "Referencias cruzadas" muestra las instancias en las que se está utilizando un objeto seleccionado y los objetos que las utilizan. La ventana de inspección también incluye bloques que sólo están disponibles online en las referencias cruzadas. Para mostrar las referencias cruzadas, elija el comando "Mostrar referencias cruzadas". (En la vista del proyecto encontrará las referencias cruzadas en el menú "Herramientas".)

Nota

No es necesario cerrar el editor para ver la información de las referencias cruzadas.

Las entradas de la tabla de referencias cruzadas pueden clasificarse. La lista de referencias cruzadas proporciona una vista general del uso de direcciones de memoria y variables en el programa de usuario.

- Al crear y cambiar un programa se genera una vista general de los operandos, variables y llamadas de bloque utilizados.
- Desde las referencias cruzadas se puede saltar directamente a la ubicación de los operandos y variables.
- Durante un test de programa o el tratamiento de errores se indica qué parte de la memoria se está procesando por qué comando y en qué bloque, qué variable se está utilizando en qué pantalla y qué bloque es llamado por qué otro bloque.

Tabla 6- 6 Elementos de la referencia cruzada

Columna	Descripción
Objeto	Nombre del objeto que utiliza los objetos del nivel inferior o que es utilizado por éstos.
Cantidad	Número de utilizaciones
Ubicación	Cada una de las ubicaciones, por ejemplo, un segmento
Propiedad	Propiedades especiales de objetos referenciados, por ejemplo, los nombres de variables en declaraciones multiinstancia.
como	Muestra información adicional sobre el objeto, como p. ej. si un DB instancia se utiliza como plantilla o como multiinstancia
Acceso	Tipo de acceso: el acceso al operando puede ser un acceso de lectura (R) y/o de escritura (W)
Dirección	Dirección del operando
Tipo	Información del tipo y el lenguaje utilizados para crear el objeto
Ruta	Ruta del objeto en el árbol del proyecto

6.9.4 Estructura de llamadas para ver la jerarquía de llamadas

La estructura de llamadas describe la jerarquía de llamadas del bloque dentro del programa de usuario. Proporciona una vista general de los bloques utilizados, las llamadas a otros bloques, la relación entre bloques, los datos necesarios para cada bloque y el estado de los bloques. Es posible abrir el editor de programación y editar bloques desde la estructura de llamadas.

Al visualizar la estructura de llamadas se dispone de una lista de los bloques utilizados en el programa de usuario. STEP 7 resalta el primer nivel de la estructura de llamadas y muestra los bloques que no son llamados por ningún otro bloque del programa. El primer nivel de la estructura de llamadas muestra los OBs y todas las FCs, los FBs y DBs que no son llamados por ningún OB. Si un bloque lógico llama otro bloque, el bloque llamado se muestra en forma de sangría debajo del bloque invocante. La estructura de llamadas sólo muestra aquellos bloques que son llamados por un bloque lógico.

Dentro de la estructura de llamadas se pueden visualizar selectivamente sólo bloques que causan conflictos. Los conflictos se dan en las siguientes condiciones:

- Bloques que ejecutan llamadas con sello de tiempo anterior o posterior
- Bloques que llaman un bloque con interfaz forzada
- Bloques que utilizan una variable con dirección y/o tipo de datos forzado
- Bloques que no son llamados ni directa ni indirectamente por ningún OB
- Bloques que llaman un bloque inexistente o no disponible

Se pueden agrupar varias llamadas de bloque y bloques de datos. Una lista desplegable muestra los enlaces a las diferentes ubicaciones de llamada.

También es posible realizar una comprobación de coherencia para ver conflictos de sello de tiempo. Cambiando el sello de tiempo de un bloque durante o tras la generación del programa se pueden provocar conflictos, lo que, a su vez, provoca incoherencias en los bloques que llaman y que son llamados.

- La mayoría de los conflictos de sello de tiempo y de interfaz pueden corregirse volviendo a compilar los bloques lógicos.
- Si la compilación no depura las incoherencias, utilice el enlace de la columna "Detalles" para ir a la fuente del problema en el editor de programación. De ese modo se pueden eliminar las incoherencias manualmente.
- Los bloques marcados en rojo deben volver a compilarse.

Instrucciones básicas

7.1 Instrucciones lógicas con bits

7.1.1 Contactos y bobinas en instrucciones lógicas con bits

KOP y FUP resultan muy efectivos para procesar lógica booleana. Por otro lado, aunque SCL resulta especialmente efectivo para la computación matemática compleja y para estructuras de control de proyectos, se puede utilizar para la lógica booleana.

Contactos KOP

Tabla 7- 1 Contactos normalmente abiertos y normalmente cerrados



KOP	SCL	Descripción
"IN" 	<pre>IF in THEN Statement; ELSE Statement; END IF;</pre>	Contactos normalmente abiertos y normalmente cerrados: Los contactos se pueden conectar a otros contactos, creando así una lógica combinacional propia. Si el bit de entrada indicado utiliza el identificador de memoria I (entrada) o Q (salida), el valor de bit se lee de la memoria imagen de proceso. Las señales de los contactos físicos del proceso controlado se cablean con los bornes de entrada del PLC. La CPU consulta las señales de entrada cableadas y actualiza continuamente los valores de estado correspondientes en la memoria imagen de proceso de las entradas.
"IN" 	<pre>IF NOT (in) THEN Statement; ELSE Statement; END_IF;</pre>	La lectura inmediata de una entrada física se indica introduciendo ":P" después del offset I (p. ej. "%I3.4:P"). En una lectura inmediata, los valores de datos de bit se leen directamente de la entrada física y no de la memoria imagen de proceso. La lectura inmediata no actualiza la memoria imagen de proceso.

Tabla 7- 2 Tipos de datos para los parámetros

Parámetro	Tipo de datos	Descripción
IN	Bool	Bit asignado

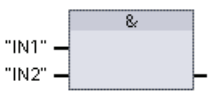
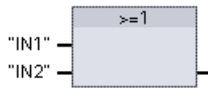
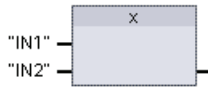
- El contacto normalmente abierto se cierra (ON) cuando el valor de bit asignado es igual a 1.
- El contacto normalmente cerrado se cierra (ON) cuando el valor de bit asignado es igual a 0.
- Los contactos conectados en serie crean segmentos lógicos Y.
- Los contactos conectados en paralelo crean segmentos lógicos O.

Cuadros Y, O y O-exclusiva en FUP

En la programación FUP, los segmentos de los contactos KOP se transforman en segmentos de cuadros Y (&), O (>=1) y O-exclusiva OR (x), en los que pueden indicarse valores de bit para las entradas y salidas de los cuadros. También es posible interconectar cuadros lógicos y crear combinaciones lógicas propias. Tras colocar un cuadro en el segmento, es posible arrastrar la función "Insertar entrada" desde la barra de herramientas "Favoritos" o desde el árbol de instrucciones y soltarla en el lado de entrada del cuadro para agregar entradas adicionales. También se puede hacer clic con el botón derecho del ratón en el conector de entrada del cuadro y seleccionar "Insertar entrada".

Es posible conectar las entradas y salidas de los cuadros con un cuadro lógico diferente, o bien introducir una dirección de bit o un nombre simbólico de bit para una entrada no conectada. Cuando se ejecuta el cuadro, los estados actuales de las entradas se aplican a la lógica del cuadro binario y, si se cumplen, la salida del cuadro será verdadera.

Tabla 7- 3 Cuadros Y, O y O-exclusiva

FUP	SCL ¹	Descripción
	<pre>out := in1 AND in2;</pre>	Todas las entradas de un cuadro Y tienen que cumplirse para que la salida sea TRUE (verdadera).
	<pre>out := in1 OR in2;</pre>	Una entrada cualquiera de un cuadro O tiene que cumplirse para que la salida sea TRUE (verdadera).
	<pre>out := in1 XOR in2;</pre>	Un número impar de entradas de un cuadro O-exclusiva tiene que cumplirse para que la salida sea TRUE (verdadera).

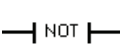
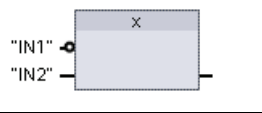
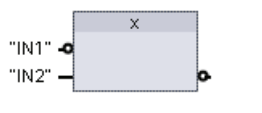
¹ En SCL: El resultado de la operación debe asignarse a una variable para que pueda usarse en otra instrucción.

Tabla 7- 4 Tipos de datos para los parámetros

Parámetro	Tipo de datos	Descripción
IN1, IN2	Bool	Bit de entrada

Invertir resultado lógico (NOT)



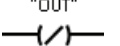
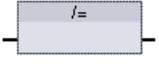
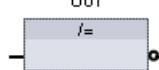
Tabla 7- 5 Invertir resultado lógico (NOT)

KOP	FUP	SCL	Descripción
	 	NOT	<p>En la programación FUP es posible arrastrar la función "Negar valor binario" desde la barra de herramientas "Favoritos" o desde el árbol de instrucciones y soltarla en una entrada o salida para crear un inversor lógico en ese conector del cuadro.</p> <p>El contacto NOT KOP invierte el estado lógico de la entrada de flujo de corriente.</p> <ul style="list-style-type: none"> • Si no fluye corriente al contacto NOT, hay flujo de corriente en la salida. • Si fluye corriente al contacto NOT, no hay flujo de corriente en la salida.

Bobina de relé y cuadro de asignación

La instrucción "Bobina de salida, relé" escribe un valor en un bit de salida. Si el bit de salida indicado utiliza el identificador de memoria Q, la CPU activa o desactiva el bit de salida en la memoria imagen de proceso, poniendo el bit especificado al correspondiente estado de flujo de corriente. Las señales de salida de los actuadores de control se cablean con los terminales Q de la CPU. En el modo RUN, el sistema de la CPU explora las señales de entrada continuamente, procesa los estados de entrada acorde con la lógica del programa de usuario, y a continuación reacciona aplicando nuevos estados lógicos de salida en la memoria imagen de proceso de las salidas. Tras cada ciclo del programa, la CPU transfiere el nuevo estado de las salidas almacenado en la memoria imagen de proceso a los bornes de salida cableados.

Tabla 7- 6 Bobina de relé (KOP) y cuadro de asignación de salida (FUP)

KOP	FUP	SCL	Descripción
		<code>out := <Expresión booleana>;</code>	<p>En la programación FUP, las bobinas KOP se transforman en cuadros de asignación (= y /=), en los que se indica una dirección de bit para la salida del cuadro. Es posible conectar las entradas y salidas del cuadro con otros cuadros lógicos, o bien introducir una dirección de bit.</p> <p>La escritura inmediata en una salida física se indica introduciendo ":P" después del offset Q (p. ej. "%Q3.4:P"). En una escritura inmediata, los valores de datos de bit se escriben en la memoria imagen de proceso de las salidas y directamente en la salida física.</p>
		<code>out := NOT <Expresión booleana>;</code>	
			

7.1 Instrucciones lógicas con bits

Tabla 7- 7 Tipos de datos para los parámetros

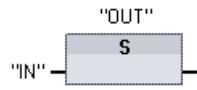
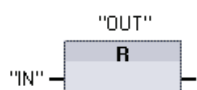
Parámetro	Tipo de datos	Descripción
OUT	Bool	Bit asignado

- Si fluye corriente a través de una bobina de salida o se habilita un cuadro FUP "=", el bit de salida se pone a 1.
- Si no fluye corriente a través de una bobina de salida o no se habilita un cuadro de asignación FUP "=", el bit de salida se pone a 0.
- Si fluye corriente a través de una bobina de salida invertida o se habilita un cuadro FUP "/=", el bit de salida se pone a 0.
- Si no fluye corriente a través de una bobina de salida invertida o no se habilita un cuadro FUP "/=", el bit de salida se pone a 1.

7.1.2 Instrucciones "Activar salida" y "Desactivar salida"

Activar y desactivar 1 bit

Tabla 7- 8 Instrucciones S y R

KOP	FUP	SCL	Descripción
"OUT" —(S)—		No disponible	Si se activa S (Set) el valor de datos de la dirección OUT se pone a 1. Si S no es está activado, OUT no cambia.
"OUT" —(R)—		No disponible	Si se activa R (Reset), el valor de datos de la dirección de salida OUT se pone a 0. Si no se activa R, no se modifica OUT.

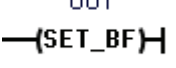
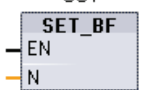
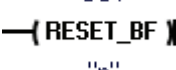

- 1 En KOP y FUP: Estas instrucciones pueden disponerse en cualquier posición del segmento.
- 2 En SCL: Es necesario escribir código para duplicar esta función en la aplicación.

Tabla 7- 9 Tipos de datos para los parámetros

Parámetro	Tipo de datos	Descripción
IN (o conectar a contacto/lógica de puerta)	Bool	Dirección de bit que debe vigilarse
OUT	Bool	Dirección de bit que se debe activar o desactivar

Activar y desactivar mapa de bits

Tabla 7- 10 Instrucciones SET_BF y RESET_BF

KOP ¹	FUP	SCL	Descripción
"OUT"  "n"	"OUT" 	No disponible	Cuando se activa SET_BF, el valor de datos 1 se asigna a "n" bits, comenzando en la dirección OUT. Si SET_BF no se activa, OUT no cambia.
"OUT"  "n"	"OUT" 	No disponible	RESET_BF escribe el valor de datos 0 en "n" bits, comenzando en la dirección OUT. Si RESET_BF no se activa, OUT no cambia.

¹ En KOP y FUP: Estas instrucciones sólo se pueden disponer en el extremo derecho de una rama.

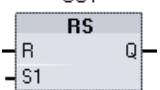
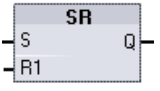
² En SCL: Es necesario escribir código para duplicar esta función en la aplicación.

Tabla 7- 11 Tipos de datos para los parámetros

Parámetro	Tipo de datos	Descripción
OUT	Bool	Elemento inicial de un mapa de bits que se debe activar o desactivar (ejemplo: #MyArray[3])
n	Constante (UInt)	Número de bits que deben escribirse

Flipflop de activación/desactivación y flipflop de desactivación/activación

Tabla 7- 12 Instrucciones RS y SR

KOP / FUP	SCL	Descripción
"OUT" 	No disponible	RS es un flipflop en el que domina la activación. Si las señales de activación (S1) y desactivación (R) son verdaderas, la dirección de salida OUT se pone a 1.
"OUT" 	No disponible	SR es un flipflop en el que domina la desactivación. Si las señales de activación (S) y desactivación (R1) son verdaderas, la dirección de salida OUT se pone a 0.

¹ En KOP y FUP: Estas instrucciones sólo se pueden disponer en el extremo derecho de una rama.

² En SCL: Es necesario escribir código para duplicar esta función en la aplicación.

Tabla 7- 13 Tipos de datos para los parámetros

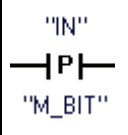
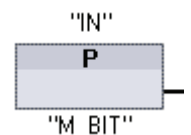
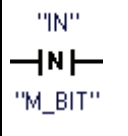
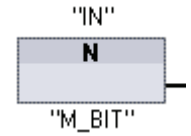
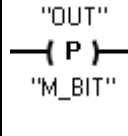
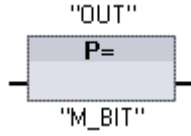
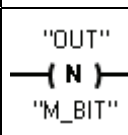
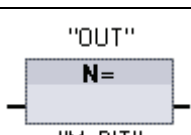
Parámetro	Tipo de datos	Descripción
S, S1	Bool	Activar entrada; 1 indica dominancia
R, R1	Bool	Desactivar entrada; 1 indica dominancia
OUT	Bool	Salida de bit asignada "OUT"
Q	Bool	Corresponde al estado del bit "OUT"

El parámetro OUT indica la dirección de bit que se activa o desactiva. La salida opcional OUT Q refleja el estado lógico de la dirección "OUT".

Instrucción	S1	R	Bit "OUT"
RS	0	0	Estado anterior
	0	1	0
	1	0	1
	1	1	1
SR	S	R1	
	0	0	Estado anterior
	0	1	0
	1	0	1
	1	1	0

7.1.3 Instrucciones "Consultar flanco de señal ascendente de un operando" y "Consultar flanco de señal descendente de un operando"



Tabla 7- 14 Detección de flanco ascendente y descendente

KOP	FUP	SCL	Descripción
 <p>"IN" "M_BIT"</p>	 <p>"IN" "M_BIT"</p>	No disponible	<p>KOP: El estado de este contacto es TRUE cuando se detecta un flanco ascendente (OFF a ON) en el bit "IN" asignado. El estado lógico del contacto se combina entonces con el estado de entrada del flujo de corriente para activar el estado de salida del flujo de corriente. El contacto P puede disponerse en cualquier posición del segmento, excepto al final de una rama.</p> <p>FUP: El estado lógico de la salida es TRUE (verdadero) cuando se detecta un flanco ascendente (OFF a ON) en el bit de entrada asignado. El cuadro P sólo se puede disponer al comienzo de una rama.</p>
 <p>"IN" "M_BIT"</p>	 <p>"IN" "M_BIT"</p>	No disponible	<p>KOP: El estado de este contacto es TRUE (verdadero) cuando se detecta un flanco descendente (ON a OFF) en el bit de entrada asignado. El estado lógico del contacto se combina entonces con el estado de entrada del flujo de corriente para activar el estado de salida del flujo de corriente. El contacto N puede disponerse en cualquier posición del segmento, excepto al final de una rama.</p> <p>FUP: El estado lógico de la salida es TRUE (verdadero) cuando se detecta un flanco descendente (ON a OFF) en el bit de entrada asignado. El cuadro N sólo se puede disponer al comienzo de una rama.</p>
 <p>"OUT" "M_BIT"</p>	 <p>"OUT" "M_BIT"</p>	No disponible	<p>KOP: El bit asignado "OUT" es TRUE (verdadero) cuando se detecta un flanco ascendente (OFF a ON) en el flujo de corriente que entra a la bobina. El estado de entrada del flujo de corriente atraviesa la bobina como el estado de salida del flujo de corriente. La bobina P puede disponerse en cualquier posición del segmento.</p> <p>FUP: El bit asignado "OUT" es TRUE (verdadero) cuando se detecta un flanco ascendente (OFF a ON) en el estado lógico de la conexión de entrada del cuadro, o bien en la asignación del bit de entrada si el cuadro está ubicado al comienzo de una rama. El estado lógico de la entrada atraviesa el cuadro como el estado lógico de la salida. El cuadro P= puede disponerse en cualquier posición de la rama.</p>
 <p>"OUT" "M_BIT"</p>	 <p>"OUT" "M_BIT"</p>	No disponible	<p>KOP: El bit asignado "OUT" es TRUE (verdadero) cuando se detecta un flanco descendente (ON a OFF) en el flujo de corriente que entra a la bobina. El estado de entrada del flujo de corriente atraviesa la bobina como el estado de salida del flujo de corriente. La bobina N puede disponerse en cualquier posición del segmento.</p> <p>FUP: El bit asignado "OUT" es TRUE cuando se detecta un flanco descendente (ON a OFF) en el estado lógico de la conexión de entrada del cuadro, o bien en la asignación del bit de entrada si el cuadro está ubicado al comienzo de una rama. El estado lógico de la entrada atraviesa el cuadro como el estado lógico de la salida. El cuadro N= puede disponerse en cualquier posición de la rama.</p>

¹ En SCL: Es necesario escribir código para duplicar esta función en la aplicación.

7.1 Instrucciones lógicas con bits

Tabla 7- 15 Instrucciones P_TRIG y N_TRIG

KOP / FUP	SCL	Descripción
	No disponible	<p>El flujo de corriente o estado lógico de la salida Q es TRUE cuando se detecta un flanco ascendente (OFF a ON) en el estado lógico de CLK (en FUP) o en el flujo de corriente de CLK (en KOP).</p> <p>En KOP, la instrucción P_TRIG no se puede disponer ni al comienzo ni al final de un segmento. En FUP, la instrucción P_TRIG puede disponerse en cualquier posición de la rama, excepto al final.</p>
	No disponible	<p>El flujo de corriente o estado lógico de la salida Q es TRUE cuando se detecta un flanco descendente (ON a OFF) en el estado lógico de CLK (en FUP) o en el flujo de corriente de CLK (en KOP).</p> <p>En KOP, la instrucción N_TRIG no se puede disponer ni al comienzo ni al final de un segmento. En FUP, la instrucción N_TRIG puede disponerse en cualquier posición de la rama, excepto al final.</p>

¹ En SCL: Es necesario escribir código para duplicar esta función en la aplicación.

Tabla 7- 16 Tipos de datos para los parámetros (P y N contactos/bobinas, P=, N=, P_TRIG y N_TRIG)

Parámetro	Tipo de datos	Descripción
M_BIT	Bool	Marca en la que se almacena el estado anterior de la entrada
IN	Bool	Bit de entrada cuyo flanco debe detectarse
OUT	Bool	Bit de salida que indica que se ha detectado un flanco
CLK	Bool	Flujo de corriente o bit de entrada cuyo flanco debe detectarse
Q	Bool	Salida que indica que se ha detectado un flanco

Todas las instrucciones de detección de flancos utilizan una marca (M_BIT) para almacenar el estado anterior de la señal de entrada que se está vigilando. Un flanco se detecta comparando el estado de la entrada con el estado de la marca. Si los estados indican un cambio de la entrada en el sentido deseado, se notifica un flanco activando la salida (TRUE). De lo contrario, se desactivará la salida (FALSE).

Nota

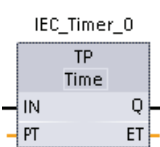
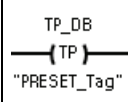
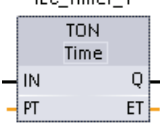
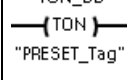
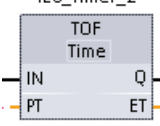
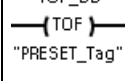
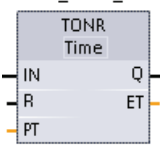
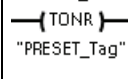
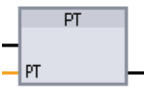
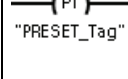
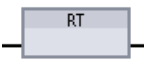

Las instrucciones de detección de flancos evalúan los valores de la entrada y de la marca cada vez que se ejecutan, incluyendo la primera ejecución. Los estados iniciales de la entrada y de la marca deben considerarse al diseñar el programa, con objeto de permitir o impedir la detección de flancos en el primer ciclo.

Puesto que la marca debe conservarse desde una ejecución hasta la siguiente, es preciso utilizar un bit unívoco para cada instrucción de detección de flancos. Este bit no se puede utilizar en ninguna otra ubicación del programa. También se debe evitar la memoria temporal y la memoria que pueda ser modificada por otras funciones de sistema, p. ej. una actualización de E/S. Utilice sólo el área de marcas (M), DB global o memoria estática (en un DB de instancia) para las asignaciones de memoria de M_BIT.

7.2 Temporizadores

Las instrucciones con temporizadores se utilizan para crear retardos programados. El número de temporizadores que pueden utilizarse en el programa de usuario está limitado sólo por la cantidad de memoria disponible en la CPU. Cada temporizador utiliza una estructura de DB del tipo de datos IEC_Timer de 16 bytes para guardar la información del temporizador especificada encima de la instrucción de cuadro o bobina. STEP 7 crea automáticamente el DB al introducir la instrucción.

Tabla 7- 17 Instrucciones con temporizadores

Cuadros KOP / FUP	Bobinas KOP	SCL	Descripción
		<pre>"IEC_Timer_0_DB".TP (IN:=_bool_in_, PT:=_time_in_, Q=>_bool_out_, ET=>_time_out_);</pre>	El temporizador TP genera un impulso con una duración predeterminada.
		<pre>"IEC_Timer_0_DB".TON (IN:=_bool_in_, PT:=_time_in_, Q=>_bool_out_, ET=>_time_out_);</pre>	El temporizador TON pone la salida Q a ON tras un tiempo de retardo predeterminado.
		<pre>"IEC_Timer_0_DB".TOF (IN:=_bool_in_, PT:=_time_in_, Q=>_bool_out_, ET=>_time_out_);</pre>	El temporizador TOF pone la salida Q a OFF tras un tiempo de retardo predeterminado.
		<pre>"IEC_Timer_0_DB".TONR (IN:=_bool_in_, R:=_bool_in_, PT:=_time_in_, Q=>_bool_out_, ET=>_time_out_);</pre>	El temporizador TONR pone la salida Q a ON tras un tiempo de retardo predeterminado. El tiempo transcurrido se acumula a lo largo de varios periodos de temporización hasta que la entrada R inicializa el tiempo transcurrido.
Sólo FUP: 		(Sin equivalencia en SCL)	La bobina PT (Cargar temporizador) carga un nuevo valor de tiempo PRESET (predeterminado) en el IEC_Timer especificado.
Sólo FUP: 		(Sin equivalencia en SCL)	La bobina RT (Inicializar temporizador) inicializa el IEC_Timer especificado.

- STEP 7 crea el DB automáticamente al insertar la instrucción.
- En los ejemplos SCL, "IEC_Timer_0_DB" es el nombre del DB de instancia.

Tabla 7- 18 Tipos de datos para los parámetros

Parámetro	Tipo de datos	Descripción
Cuadro: IN Bobina: Flujo de corriente	Bool	TP, TON, y TONR: cuadro: 0=deshabilitar temporizador, 1=habilitar temporizador Bobina: Sin flujo de corriente=deshabilitar temporizador, flujo de corriente=habilitar temporizador TOF: Cuadro: 0=habilitar temporizador, 1=deshabilitar temporizador Bobina: Sin flujo de corriente=habilitar temporizador, flujo de corriente=deshabilitar temporizador
R	Bool	Sólo cuadro TONR: 0=Sin inicialización 1= Inicializar el tiempo transcurrido y el bit Q a 0
Cuadro: PT Bobina: "PRESET_variable"	Time	Cuadro o bobina de temporizador: Entrada de tiempo predeterminado
Cuadro: Q Bobina: DBdata.Q	Bool	Cuadro de temporizador: salida de cuadro Q o bit Q en los datos del DB de temporizador Bobina de temporizador: sólo se puede direccionar el bit Q en los datos del DB de temporizador
Cuadro: ET Bobina: DBdata.ET	Time	Cuadro de temporizador: salida de cuadro ET (tiempo transcurrido) o valor de tiempo ET en los datos del DB de temporizador Bobina de temporizador: sólo se puede direccionar el valor de tiempo ET en los datos del DB de temporizador.

Tabla 7- 19 Efecto de los cambios de valores en los parámetros PT e IN

Temporizador	Cambios en los parámetros de cuadro PT e IN y en los parámetros de bobina correspondientes
TP	<ul style="list-style-type: none"> Un cambio de PT no tiene efecto alguno durante el funcionamiento del temporizador. Un cambio de IN no tiene efecto alguno durante el funcionamiento del temporizador.
TON	<ul style="list-style-type: none"> Un cambio de PT no tiene efecto alguno durante el funcionamiento del temporizador. Si IN cambia a FALSE durante el funcionamiento del temporizador, éste se inicializará y se detendrá.
TOF	<ul style="list-style-type: none"> Un cambio de PT no tiene efecto alguno durante el funcionamiento del temporizador. Si IN cambia a TRUE durante el funcionamiento del temporizador, éste se inicializará y se detendrá.
TONR	<ul style="list-style-type: none"> Un cambio de PT no tiene efecto alguno durante el funcionamiento del temporizador, pero sí cuando reanuda el contaje. Si IN cambia a FALSE durante el funcionamiento del temporizador, éste se detendrá pero no se inicializará. Si IN vuelve a cambiar a TRUE, el temporizador comenzará a contar desde el valor de tiempo acumulado.

Los valores PT (tiempo predeterminado) y ET (tiempo transcurrido) se almacenan en los datos de DB IEC_TIMER como enteros dobles con signo que representan milisegundos. Los datos TIME utilizan el identificador T# y pueden introducirse como unidad de tiempo simple ("T#200ms ó 200) y como unidades de tiempo compuestas "T#2s_200ms".

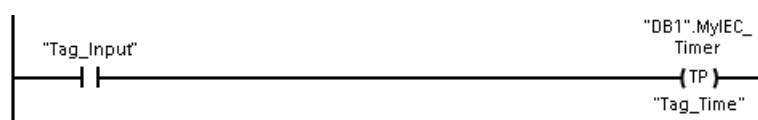
Tabla 7- 20 Tamaño y rango del tipo de datos TIME

Tipo de datos	Tamaño	Rangos válidos ¹
TIME	32 bits, almacenados como datos DInt	T#-24d_20h_31m_23s_648ms hasta T#24d_20h_31m_23s_647ms Almacenado como -2.147.483.648 ms hasta +2.147.483.647 ms

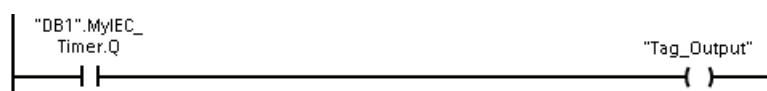
- ¹ El rango negativo del tipo de datos TIME indicado arriba no puede utilizarse con las instrucciones de temporizador. Los valores PT (tiempo predeterminado) negativos se ponen a cero cuando se ejecuta la instrucción de temporización. ET (tiempo transcurrido) es siempre un valor positivo.

Ejemplo de bobina de temporizador

Las bobinas de temporizador -(TP)-, -(TON)-, -(TOF)- y -(TONR)- deben ser la última instrucción de un segmento KOP. Como se indica en el ejemplo de temporizador, una instrucción de contacto de un segmento posterior evalúa el bit Q en los datos de DB IEC_Timer de una bobina de temporizador. Del mismo modo, hay que direccionar el elemento ELAPSED en los datos de DB IEC_timer si se desea emplear el valor de tiempo transcurrido en el programa.



El temporizador como impulso arranca en una transición de 0 a 1 del valor de bit Tag_Input. El temporizador se ejecuta durante el tiempo especificado por el valor de tiempo Tag_Time.



Mientras el temporizador se ejecute, el estado de DB1.MyIEC_Timer.Q es 1 y el valor de Tag_Output es 1. Una vez que el valor de Tag_Time ha transcurrido, DB1.MyIEC_Timer.Q es 0 y el valor de Tag_Output es 0.

Bobinas Inicializar temporizador -(RT)- y predeterminar temporizador -(PT)-

Estas instrucciones de bobina pueden utilizarse con temporizadores de cuadro o bobina y pueden colocarse en una posición central. El estado del flujo de salida de la bobina siempre es el mismo que el estado de entrada de la bobina. Cuando la bobina -(RT)- se activa, el elemento de tiempo ELAPSED de los datos de DB IEC_Timer especificado se reinicia a 0. Cuando la bobina -(PT)- se activa, el elemento de tiempo PRESET de los datos de DB IEC_Timer especificado se reinicia a 0.

Nota

Si se insertan instrucciones de temporizador en un FB, se puede seleccionar la opción "Bloque de datos multiinstancia". Los nombres de estructura de temporizador pueden ser diferentes con diferentes estructuras de datos, pero los datos del temporizador están contenidos como un bloque de datos único y no requiere un bloque de datos separado para cada temporizador. Esto reduce el tiempo de procesamiento y la memoria de datos necesaria para gestionar los temporizadores. No hay interacción entre las estructuras de datos de los temporizadores en el DB multiinstancia compartido.

Manejo de temporizadores

Tabla 7- 21 Tipos de temporizadores IEC

Temporizador	Cronograma
<p>TP: Temporizador como impulso El temporizador TP genera un impulso con una duración predeterminada.</p>	
<p>TON: Temporizador como retardo a la conexión El temporizador TON pone la salida Q a ON tras un tiempo de retardo predeterminado.</p>	

Temporizador	Cronograma
<p>TOF: Temporizador como retardo a la desconexión</p> <p>El temporizador TOF pone la salida Q a OFF tras un tiempo de retardo predeterminado.</p>	
<p>TONR: Temporizador como retado a la conexión</p> <p>El temporizador TONR pone la salida Q a ON tras un tiempo de retardo predeterminado. El tiempo transcurrido se acumula a lo largo de varios periodos de temporización hasta que la entrada R inicializa el tiempo transcurrido.</p>	

Nota

En la CPU no se asigna ningún recurso dedicado a ninguna instrucción de temporizador específica. En lugar de eso, cada temporizador utiliza su estructura de temporizador propia en la memoria de DB y un temporizador de funcionamiento continuo interno de la CPU para la temporización.

Cuando se arranca un temporizador debido a un flanco en la entrada de una instrucción TP, TON, TOF o TONR, el valor del temporizador de funcionamiento continuo interno de la CPU se copia en el miembro START de la estructura del DB asignado a esta instrucción de temporizador. Este valor de arranque permanece inalterado mientras el temporizador continúa funcionando, y se vuelve a utilizar cada vez que el temporizador se actualiza. Cada vez que el temporizador arranca se carga un nuevo valor en la estructura de temporizador del temporizador interno de la CPU.

Cuando se actualiza un temporizador, el valor de arranque arriba descrito se resta del valor actual del temporizador interno de la CPU para determinar el tiempo transcurrido. El tiempo transcurrido se compara entonces con el predeterminado para determinar el estado del bit Q del temporizador. Los miembros ELAPSED y Q se actualizan en la estructura de DB asignada a este temporizador. Recuerde que el tiempo transcurrido se detiene en el valor predeterminado (el temporizador no continúa acumulando tiempo transcurrido una vez que se ha alcanzado el valor predeterminado).

Una actualización de temporizador se realiza única y exclusivamente cuando:

- Se ejecuta una instrucción de temporizador (TP, TON, TOF o TONR)
- El miembro "ELAPSED" de la estructura del DB se referencia directamente con una instrucción
- El miembro "Q" de la estructura de temporizador del DB se referencia directamente con una instrucción

Programación de temporizadores

A la hora de planificar y crear el programa de usuario deben considerarse las siguientes consecuencias del manejo de temporizadores:

- Pueden producirse múltiples actualizaciones de un temporizador en el mismo ciclo. El temporizador se actualiza cada vez que la instrucción de temporizador (TP, TON, TOF, TONR) se ejecuta y cada vez que el miembro ELAPSED o Q de la estructura de temporizador se utiliza como parámetro para otra instrucción ejecutada. Esto es una ventaja si se desea tener el último dato de tiempo (en esencia, una lectura inmediata del temporizador). No obstante, si desea disponer de valores coherentes durante un ciclo del programa, inserte la instrucción de temporizador antes de todas las instrucciones restantes que requieran estos valores, y utilice las variables de las salidas Q y ET de la instrucción de temporizador en lugar de los miembros ELAPSED y Q de la estructura de DB de temporizador.
- Pueden producirse ciclos durante los cuales no se actualice ningún temporizador. Es posible arrancar el temporizador en una función y dejar de llamar la función durante uno o más ciclos. Si no se ejecuta ninguna otra instrucción que referencie los miembros ELAPSED o Q de la estructura de temporizador, el temporizador no se actualizará. No se produce una nueva actualización hasta que la instrucción de temporizador se ejecuta de nuevo o se ejecuta alguna otra instrucción utilizando ELAPSED o Q de la estructura del temporizador como un parámetro.
- Aunque no es común, se puede asignar la misma estructura de DB de temporizador a varias instrucciones de temporizador. En general, para evitar una interacción inesperada, debería utilizarse sólo una instrucción de temporizador (TP, TON, TOF, TONR) por estructura de temporizador de DB.

- Las inicializaciones propias de los temporizadores resultan útiles para lanzar acciones que deben producirse periódicamente. Generalmente, los temporizadores autoinicializables se crean colocando un contacto normalmente cerrado que referencie el bit de temporizador situado frente a la instrucción de temporizador. Este segmento de temporizador se suele colocar sobre uno o más segmentos dependientes, que utilizan el bit del temporizador para lanzar acciones. Cuando el temporizador finaliza (el tiempo transcurrido llega al valor predeterminado), el bit de temporizador permanece en ON durante un ciclo, permitiendo que se ejecute la lógica de segmento dependiente controlada por el bit de temporizador. Con la siguiente ejecución del segmento de temporizador, el contacto normalmente cerrado está en desconexión, con lo que el temporizador se reinicia y el bit de temporizador se desactiva. En el siguiente ciclo, el contacto normalmente cerrado está en conexión, de modo que el contacto se inicializa. Cuando cree temporizadores autoinicializables de este manera, no utilice el miembro "Q" de la estructura de DB de temporizador, ya que el parámetro para el contacto normalmente cerrado está delante de la instrucción de temporizador. En lugar de ello, utilice la variable asociada a la salida "Q" de la instrucción de temporizador. La razón de evitar el acceso al miembro Q de la estructura de DB de temporizador es que ello provoca una actualización del temporizador, y si éste se actualiza debido al contacto normalmente cerrado, éste inicializará la instrucción de temporizador inmediatamente. La salida Q de la instrucción de temporizador no estará ON para el ciclo en cuestión, y los segmentos dependientes no se ejecutarán.

Retención de los datos de tiempo tras una transición RUN-STOP-RUN o una desconexión y nueva conexión de la CPU

Si una sesión en modo RUN finaliza con el modo STOP o una desconexión y nueva conexión de la CPU y se inicia una nueva sesión en modo RUN, los datos de temporizador guardados en la sesión anterior se pierden, a no ser que la estructura de datos se haya definido como remanente (temporizadores TP, TON, TOF y TONR).

Si se aceptan los ajustes predeterminados del diálogo de opciones de llamada una vez insertada la instrucción en el editor de programas, automáticamente se asignará un DB de instancia **que no puede definirse como remanente**. Para que los datos de temporizador puedan ser remanentes, hay que usar un DB global o un DB multiinstancia.

Asignar un DB global para guardar los datos de temporizador como datos remanentes

Esta opción funciona independientemente de dónde se encuentre el temporizador (OB, FC o FB).

1. Crear un DB global:
 - Haga doble clic en "Agregar nuevo bloque" en el árbol del proyecto
 - Haga clic en el icono del bloque de datos (DB)
 - Elija DB global como tipo
 - Si desea poder definir elementos de datos individuales en este DB como remanentes, asegúrese de que está activada la casilla "Optimizado". La otra opción para el tipo de DB "Estándar - compatible con S7-300/400" sólo permite definir todos los elementos del DB como remanentes o no remanentes.
 - Haga clic en "Aceptar".
2. Agregar estructura(s) de temporizador al DB:
 - En el nuevo DB global, agregue una nueva variable estática utilizando el tipo de datos IEC_Timer.
 - En la columna "Remanencia" compruebe la casilla correspondiente para que la estructura sea remanente.
 - Repita este procedimiento para crear estructuras para todos los temporizadores que desee guardar en ese DB. Es posible insertar cada estructura de temporizador en un DB global único, o insertar múltiples estructuras de temporizador en el mismo DB global. También es posible asignar al DB global otras variables estáticas aparte de los temporizadores. La colocación de múltiples estructuras de temporizador en el mismo DB global permite reducir el número general de bloques.
 - Cambie el nombre de las estructuras de temporizador si lo desea.
3. Abra el bloque de programa para editar la posición en la que desea colocar el temporizador remanente (OB, FC o FB).
4. Sitúe la instrucción de temporizador en la posición deseada.
5. Cuando aparezca el diálogo de opciones de llamada, haga clic en el botón Cancelar.
6. En la parte superior de la nueva instrucción de temporizador escriba el nombre (no utilice la función de ayuda para examinar) del DB global y de la estructura de temporizador que ha creado antes (ejemplo: "Bloque_datos_3.Estático_1").

Asignar un DB multiinstancia para guardar los datos de temporizador como datos remanentes

Esta opción sólo funciona si el temporizador se coloca en un FB

Esta opción depende de si el FB se creó con acceso a bloques "Optimizado" (sólo permite el direccionamiento simbólico). Una vez que el FB se ha creado ya no se puede cambiar la casilla de verificación de "Optimizado"; hay que seleccionarla correctamente en el momento de la creación del FB, en la primera pantalla que aparece tras seleccionar "Agregar nuevo bloque" del árbol. Para comprobar cómo está configurado el atributo de acceso para un FB existente, haga clic con el botón derecho sobre el FB en el árbol del proyecto, seleccione Propiedades y a continuación Atributos.

Si el FB se creó con la casilla "Optimizado" seleccionada (sólo permite el direccionamiento simbólico):

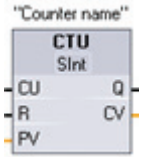
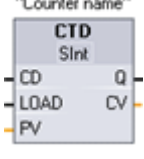
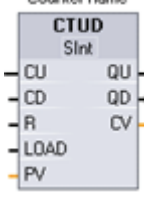
1. Abra el FB para editarlo.
2. Sitúe la instrucción de temporizador en la posición deseada dentro del FB.
3. Cuando aparezca el diálogo de opciones de llamada, haga clic en el botón de multiinstancia. La opción de multiinstancia sólo está disponible si la instrucción se está colocando en un FB.
4. En el diálogo de opciones de llamada, cambie el nombre del temporizador si lo desea.
5. Haga clic en "Aceptar". La instrucción de temporizador aparece en el editor, y la estructura IEC_TIMER aparece en la interfaz de FB, bajo Estático.
6. En caso necesario, abra el editor de interfaz de FB (puede ser necesario hacer clic en la flecha pequeña para expandir la vista).
7. En Estático, localice la estructura de temporizador que se acaba de crear.
8. En la columna Remanencia correspondiente a dicha estructura, cambie la selección a "Remanencia". Cada vez que este FB se llame posteriormente desde otro bloque de programa se creará un DB de instancia con esta definición de interfaz, que contiene la estructura de temporizador marcada como remanente.

Si el FB se creó con la casilla "Estándar - Compatible con S7-300/400" seleccionada (permite el direccionamiento simbólico y directo):

1. Abra el FB para editarlo.
2. Sitúe la instrucción de temporizador en la posición deseada dentro del FB.
3. Cuando aparezca el diálogo de opciones de llamada, haga clic en el botón de multiinstancia. La opción de multiinstancia sólo está disponible si la instrucción se está colocando en un FB.
4. En el diálogo de opciones de llamada, cambie el nombre del temporizador si lo desea.
5. Haga clic en "Aceptar". La instrucción de temporizador aparece en el editor, y la estructura IEC_TIMER aparece en la interfaz de FB, bajo Estático.
6. Abra el bloque que usará este FB.
7. Sitúe el FB en la posición deseada. De este modo se crea un bloque de datos de instancia para este FB.
8. Abra el bloque de datos de instancia creado al colocar el FB en el editor.
9. En Estático, coloque la estructura de temporizador en cuestión. En la columna Remanencia correspondiente a dicha estructura, cambie la casilla de verificación de modo que la estructura sea remanente.

7.3 Contadores

Tabla 7- 22 Instrucciones con contadores

KOP / FUP	SCL	Descripción
	<pre>"IEC_Counter_0_DB".CTU(CU:=_bool_in, R:=_bool_in, PV:=_int_in, Q=>_bool_out, CV=>_int_out);</pre>	<p>Las instrucciones con contadores se utilizan para contar eventos del programa internos y eventos del proceso externos. Todo contador utiliza una estructura almacenada en un bloque de datos para conservar sus datos. El bloque de datos se asigna al colocar la instrucción de contaje en el editor.</p> <ul style="list-style-type: none"> • CTU es un contador ascendente • CTD es un contador descendente • CTUD es un contador ascendente/descendente
	<pre>"IEC_Counter_0_DB".CTD(CD:=_bool_in, LD:=_bool_in, PV:=_int_in, Q=>_bool_out, CV=>_int_out);</pre>	
	<pre>"IEC_Counter_0_DB".CTUD(CU:=_bool_in, CD:=_bool_in, R:=_bool_in, LD:=_bool_in, PV:=_int_in, QU=>_bool_out, QD=>_bool_out, CV=>_int_out);</pre>	

- 1 En KOP y FUP: Seleccione el tipo de datos del valor de contaje en la lista desplegable situada debajo del nombre de la instrucción.
- 2 STEP 7 crea el DB automáticamente al insertar la instrucción.
- 3 En los ejemplos SCL, "IEC_Counter_0_DB" es el nombre del DB de instancia.

Tabla 7- 23 Tipos de datos para los parámetros

Parámetro	Tipo de datos ¹	Descripción
CU, CD	Bool	Contaje ascendente o descendente, en incrementos de uno
R (CTU, CTUD)	Bool	Poner a cero el valor del contador
LD (CTD, CTUD)	Bool	Control de carga del valor predeterminado
PV	SInt, Int, DInt, USInt, UInt, UInt	Valor de contaje predeterminado
Q, QU	Bool	Es verdadero si CV >= PV
QD	Bool	Es verdadero si CV <= 0
CV	SInt, Int, DInt, USInt, UInt, UInt	Valor de contaje actual

¹ El rango numérico de valores de contaje depende del tipo de datos seleccionado. Si el valor de contaje es un entero sin signo, es posible contar hacia atrás hasta cero o hacia delante hasta el límite del rango. Si el valor de contaje es un entero con signo, es posible contar hacia atrás hasta el límite de entero negativo y contar hacia delante hasta el límite de entero positivo.

El número de contadores que pueden utilizarse en el programa de usuario está limitado sólo por la cantidad de memoria disponible en la CPU. Los contadores utilizan la siguiente cantidad de memoria:

- En los tipos de datos SInt o USInt, la instrucción de contaje utiliza 3 bytes.
- En los tipos de datos nt o UInt, la instrucción de contaje utiliza 6 bytes.
- En los tipos de datos DInt o UDInt, la instrucción de contaje utiliza 12 bytes.

Estas instrucciones utilizan contadores por software cuya frecuencia de contaje máxima está limitada por la frecuencia de ejecución del OB en el que están contenidas. El OB en el que se depositan las instrucciones debe ejecutarse con suficiente frecuencia para detectar todas las transiciones de las entradas CU o CD. Para operaciones de contaje rápido, consulte la instrucción CTRL_HSC (Página 353).

Nota

Al colocar instrucciones con contadores en un FB es posible seleccionar la opción de DB multiinstancia. Los nombres de estructura de los contadores pueden diferir en las distintas estructuras, pero los datos de los contadores se encuentran en un DB individual y no requieren un DB propio para cada contador. Esto reduce el tiempo de procesamiento y la memoria de datos necesaria para los contadores. No hay interacción entre las estructuras de datos de los contadores en el DB multiinstancia compartido.

Manejo de los contadores

Tabla 7- 24 Manejo del contador CTU

Contador	Operación
<p>El contador CTU incrementa en 1 cuando el valor del parámetro CU cambia de 0 a 1. El cronograma de CTU muestra el manejo con un valor de contaje de entero sin signo (donde PV = 3).</p> <ul style="list-style-type: none"> • Si el valor del parámetro CV (valor de contaje actual) es superior o igual que el del parámetro PV (valor de contaje predeterminado), el parámetro de salida del contador Q = 1. • Si el valor del parámetro de desactivación R cambia de 0 a 1, el valor de contaje actual se pone a 0. 	<p>The diagram shows the relationship between the counter's inputs and outputs. CU (Count Up) is a square wave with four rising edges. R (Reset) is a pulse that occurs after the third rising edge of CU. CV (Current Value) is a staircase function that increases by 1 at each rising edge of CU, reaching 4 at the fourth edge, and then resets to 0 when R transitions from 0 to 1. Q (Output) is a pulse that becomes 1 when CV reaches 3 and returns to 0 when R transitions from 0 to 1.</p>

7.3 Contadores

Tabla 7- 25 Manejo del contador CTD

Contador	Operación
<p>El contador CTD decrementa en 1 cuando el valor del parámetro CD cambia de 0 a 1. El cronograma de CTD muestra el manejo con un valor de contaje de entero sin signo (donde PV = 3).</p> <ul style="list-style-type: none"> • Si el valor del parámetro CV (valor de contaje actual) es inferior o igual a 0, el parámetro de salida del contador Q = 1. • Si el valor del parámetro LOAD cambia de 0 a 1, el valor del parámetro PV (valor predeterminado) se carga en el contador como nuevo CV (valor de contaje actual). 	

Tabla 7- 26 Manejo del contador CTUD

Contador	Operación
<p>El contador CTUD incrementa o decrementa en 1 en una transición de 0 a 1 de las entradas de contaje ascendente (CU) o descendente (CD). El cronograma muestra el funcionamiento de un contador CTUD con un valor de contaje de entero sin signo (donde PV = 4).</p> <ul style="list-style-type: none"> • Si el valor del parámetro CV es superior o igual que el parámetro PV, el parámetro de salida del contador QU = 1. • Si el valor del parámetro CV es inferior o igual a 0, el parámetro de salida del contador QD = 1. • Si el valor del parámetro LOAD cambia de 0 a 1, el valor del parámetro PV se carga en el contador como nuevo CV. • Si el valor del parámetro de reset R cambia de 0 a 1, el valor de contaje actual se pone a 0. 	

Remanencia de los datos de contador tras una transición RUN-STOP-RUN o una desconexión y nueva conexión de la CPU

Si una sesión en modo RUN finaliza con el modo STOP o una desconexión y nueva conexión de la CPU y se inicia una nueva sesión en modo RUN, los datos de contador guardados en la sesión anterior se pierden, a no ser que la estructura de datos se haya definido como remanente (contadores CTU, CTD y CTUD).

Si se aceptan los ajustes predeterminados del diálogo de opciones de llamada una vez insertada la instrucción de contador en el editor de programas, automáticamente se asignará un DB de instancia **que no puede definirse como remanente**. Para que los datos de contador puedan ser remanentes, hay que usar un DB global o un DB multiinstancia.

Asignar un DB global para guardar los datos de contador como datos remanentes

Esta opción funciona independientemente de dónde se encuentre el contador (OB, FC o FB).

1. Crear un DB global:

- Haga doble clic en "Agregar nuevo bloque" en el árbol del proyecto
- Haga clic en el icono del bloque de datos (DB)
- Elija DB global como tipo
- Si desea poder definir elementos individuales en este DB como remanentes, asegúrese de que está activada la casilla que limita el acceso a sólo simbólico.
- Haga clic en "Aceptar".

2. Agregar estructura(s) de contador al DB:

- En el nuevo DB global, agregue una nueva variable estática utilizando uno de los tipos de datos de contador. Asegúrese de elegir el tipo que desea utilizar para los valores predeterminado y de contaje.

Tipo de datos de contador	Tipo correspondiente para los valores predeterminado y de contaje
IEC_Counter	INT
IEC_SCounter	SINT
IEC_DCounter	DINT
IEC_UCounter	UINT
IEC_USCounter	USINT
IEC_UDCounter	UDINT

1. En la columna "Remanencia" compruebe la casilla correspondiente para que la estructura sea remanente.
 - Repita este procedimiento para crear estructuras para todos los contadores que desee guardar en ese DB. Es posible insertar cada estructura de contador en un DB global único, o insertar múltiples estructuras de contador en el mismo DB global. También es posible asignar al DB global otras variables estáticas aparte de los contadores. La colocación de múltiples estructuras de contador en el mismo DB global permite reducir el número general de bloques.
 - Cambie el nombre de las estructuras de temporizador si lo desea.
2. Abra el bloque de programa para editar la posición en la que desea colocar el contador remanente (OB, FC o FB).
3. Sitúe la instrucción de contador en la posición deseada.
4. Cuando aparezca el diálogo de opciones de llamada, haga clic en el botón Cancelar. Ahora debería ver una nueva instrucción de contador con "???" justo arriba y abajo del nombre de la instrucción.
5. En la parte superior de la nueva instrucción de contador escriba el nombre (no utilice la función de ayuda para examinar) del DB global y de la estructura de contador que ha creado antes (ejemplo: "Bloque_datos_3.Estático_1"). Con esa acción se rellenan los tipos de valores predeterminado y de contaje correspondientes (ejemplo: UInt para una estructura IEC_UCounter).

Asignar un DB multiinstancia para guardar los datos de contador como datos remanentes

Esta opción sólo funciona si el contador se coloca en un FB

Esta opción depende de si el FB se ha creado sólo con acceso simbólico. Una vez que el FB se ha creado ya no se puede cambiar la casilla de verificación que limita el direccionamiento a sólo simbólico; hay que seleccionarla correctamente en el momento de la creación del FB, en la primera pantalla que aparece tras seleccionar "Agregar nuevo bloque" del árbol. Para comprobar cómo está configurada esta casilla para un FB existente, haga clic con el botón derecho sobre el FB en el árbol del proyecto, seleccione Propiedades y a continuación Atributos.

Si el FB se creó con la casilla de sólo direccionamiento simbólico seleccionada:

1. Abra el FB para editarlo.
2. Sitúe la instrucción de contador en la posición deseada dentro del FB.
3. Cuando aparezca el diálogo de opciones de llamada, haga clic en el botón de multiinstancia. La opción de multiinstancia sólo está disponible si la instrucción se está colocando en un FB.
4. En el diálogo de opciones de llamada, cambie el nombre del contador si lo desea.
5. Haga clic en "Aceptar". La instrucción de contador aparece en el editor con el tipo INT para los valores predeterminado y de contaje, y la estructura IEC_COUNTER aparece en la interfaz de FB, bajo Estático.
6. Si lo desea, cambie el tipo de en la instrucción de contador de INT a uno de los tipos restantes. La estructura de contador cambiará de la forma correspondiente.

Tipo indicado en la instrucción de contador (para valores predeterminado y de contaje)	Tipo de estructura correspondiente indicado en la interfaz de FB
INT	IEC_Counter
SINT	IEC_SCounter
DINT	IEC_DCounter
UINT	IEC_UCounter
USINT	IEC_USCounter
UDINT	IEC_UDCounter

1. En caso necesario, abra el editor de interfaz de FB (puede ser necesario hacer clic en la flecha pequeña para expandir la vista).
2. En Estático, localice la estructura de contador que se acaba de crear.
3. En la columna Remanencia correspondiente a dicha estructura, cambie la selección a "Remanencia". Cada vez que este FB se llame posteriormente desde otro bloque de programa se creará un DB de instancia con esta definición de interfaz con la estructura de contador marcada como remanente.

Si el FB se creó con la casilla "Sólo direccionamiento simbólico" *no* seleccionada:

1. Abra el FB para editarlo.
2. Sitúe la instrucción de contador en la posición deseada dentro del FB.
3. Cuando aparezca el diálogo de opciones de llamada, haga clic en el botón de multiinstancia. La opción de multiinstancia sólo está disponible si la instrucción se está colocando en un FB.
4. En el diálogo de opciones de llamada, cambie el nombre del contador si lo desea.
5. Haga clic en "Aceptar". La instrucción de contador aparece en el editor con el tipo INT para los valores predeterminado y de contaje, y la estructura IEC_COUNTER aparece en la interfaz de FB, bajo Estático.
6. Si lo desea, cambie el tipo de en la instrucción de contador de INT a uno de los tipos restantes. La estructura de contador cambiará de la forma correspondiente.

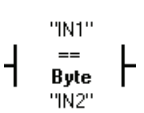

Tipo indicado en la instrucción de contador (para valores predeterminado y de contaje)	Tipo de estructura correspondiente indicado en la interfaz de FB
INT	IEC_Counter
SINT	IEC_SCounter
DINT	IEC_DCounter
UINT	IEC_UCounter
USINT	IEC_USCounter
UDINT	IEC_UDCounter

1. Abra el bloque que usará este FB.
2. Sitúe el FB en la posición deseada. De este modo se crea un bloque de datos de instancia para este FB.
3. Abra el bloque de datos de instancia creado al colocar el FB en el editor.
4. En Estático, localice la estructura de contador en cuestión. En la columna Remanencia correspondiente a dicha estructura, cambie la casilla de verificación de modo que la estructura sea remanente.

7.4 Comparación

7.4.1 Comparación

Tabla 7- 27 Instrucciones de comparación

KOP	FUP	SCL	Descripción
		<pre> out := in1 = in2; or IF in1 = in2 THEN out := 1; ELSE out := 0; END IF; </pre>	<p>Compara varios elementos del mismo tipo de datos. Si la comparación de contactos KOP es TRUE (verdadera), se activa el contacto. Si la comparación de cuadros FUP es TRUE (verdadera), la salida del cuadro es TRUE.</p>

¹ En KOP y FUP: haga clic en el nombre de la instrucción (p. ej. "==") para cambiar el tipo de comparación en la lista desplegable. Haga clic en "???" y seleccione un tipo de datos en la lista desplegable.

Tabla 7- 28 Tipos de datos para los parámetros

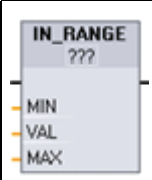

Parámetro	Tipo de datos	Descripción
IN1, IN2	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, String, Char, Time, DTL, constante	Valores que deben compararse

Tabla 7- 29 Descripciones de comparaciones

Tipo de relación	La comparación se cumple si ...
=	IN1 es igual a IN2
<>	IN1 es diferente de IN2
>=	IN1 es mayor o igual a IN2
<=	IN1 es menor o igual a IN2
>	IN1 es mayor que IN2
<	IN1 es menor que IN2

7.4.2 Instrucciones "Valor dentro del rango" y "Valor fuera del rango"

Tabla 7- 30 Valor dentro del rango y Valor fuera del rango

KOP / FUP	SCL	Descripción
	<pre>out := IN_RANGE (min, val, max) ;</pre>	<p>Comprueba si un valor de entrada está dentro o fuera de un rango de valores específico.</p> <p>Si la comparación es TRUE (verdadera), la salida del cuadro es TRUE.</p>
	<pre>out := OUT_RANGE (min, val, max) ;</pre>	

¹ En KOP y FUP: haga clic en "???" y seleccione el tipo de datos de la lista desplegable.

Tabla 7- 31 Tipos de datos para los parámetros

Parámetro	Tipo de datos ¹	Descripción
MIN, VAL, MAX	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, Constante	Entradas de comparación

¹ Los parámetros de entrada MIN, VAL y MAX deben tener un mismo tipo de datos.

- La comparación IN_RANGE se cumple si: MIN <= VAL <= MAX
- La comparación OUT_RANGE se cumple si: VAL < MIN o VAL > MAX

7.4.3 Instrucciones "Comprobar validez" y "Comprobar invalidez"

Tabla 7- 32 Instrucciones "Comprobar validez" y "Comprobar invalidez"

KOP	FUP	SCL	Descripción
		No disponible	Comprueba si una referencia de datos de entrada es un número real válido según la especificación IEEE 754.
		No disponible	

¹ En KOP y FUP: Si el contacto KOP es TRUE, se activa el contacto éste y conduce corriente. Si el cuadro FUP es TRUE (verdadero), la salida del cuadro es TRUE.

Tabla 7- 33 Tipos de datos para los parámetros

Parámetro	Tipo de datos	Descripción
IN	Real, LReal	Datos de entrada

Tabla 7- 34 Funcionamiento

Instrucción	La comprobación del número Real es TRUE (verdadera) si:
OK	El valor de entrada es un número real válido ¹
NOT_OK	El valor de entrada no es un número real válido ¹

¹ Un valor Real o LReal no es válido si es +/- INF (infinito), NaN (no es un número) o si es un valor desnormalizado. Un valor desnormalizado es un número muy próximo a cero. La CPU sustituye un valor desnormalizado por cero en los cálculos.

Consulte también

EN y ENO para KOP, FUP y SCL (Página 167)

7.5 Funciones matemáticas

7.5.1 Instrucción Calcular

Tabla 7- 35 Instrucción CALCULATE

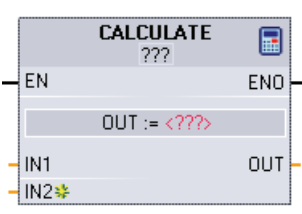
KOP / FUP	SCL	Descripción
	<p>Utilice las expresiones matemáticas SCL estándar para crear la ecuación.</p>	<p>La instrucción CALCULATE permite crear una función matemática que funciona con entradas (IN1, IN2, ... INn) y genera el resultado en OUT, según la ecuación definida.</p> <ul style="list-style-type: none"> En primer lugar, seleccione un tipo de datos. Todas las entradas y la salida deben tener un mismo tipo de datos. Para agregar otra entrada, haga clic en el icono de la última entrada.

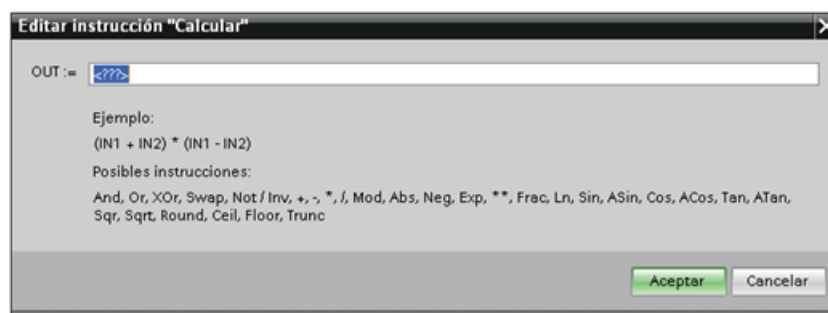
Tabla 7- 36 Tipos de datos para los parámetros

Parámetro	Tipo de datos ¹
IN1, IN2, ..INn	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, Byte, Word, DWord
OUT	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, Byte, Word, DWord

¹ Los parámetros IN y OUT deben tener un mismo tipo de datos (con conversiones implícitas de los parámetros de entrada). Ejemplo: un valor SINT para una entrada se convertiría a un valor INT o REAL si OUT fuera un INT o REAL.

Haga clic en el icono de la calculadora para abrir el cuadro de diálogo y definir la función matemática. La ecuación se introduce como entradas (p. ej. IN1 y IN2) y operaciones. Cuando se hace clic en "Aceptar" para guardar la función, el cuadro de diálogo crea automáticamente las entradas de la instrucción CALCULATE.

En la parte inferior del editor, se muestra un ejemplo y una lista de las operaciones matemáticas que se pueden incluir.



Nota

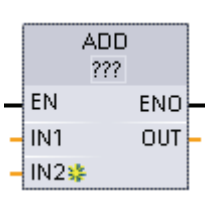
También es necesario crear una entrada para las constantes que pudiera haber en la función. En este caso, el valor constante se introduciría en la entrada asociada de la instrucción CALCULATE.

Si se introducen constantes como entradas, es posible copiar la instrucción CALCULATE a otras ubicaciones del programa de usuario sin tener que cambiar la función. Entonces, pueden modificarse los valores o las variables de las entradas de la instrucción sin modificar la función.

Cuando se ejecuta CALCULATE y todas las operaciones individuales del cálculo se realizan de forma correcta, entonces ENO = 1. Si no es así, ENO = 0.

7.5.2 Instrucciones "Sumar", "Restar", "Multiplicar" y "Dividir"

Tabla 7- 37 Instrucciones "Sumar", "Restar", "Multiplicar" y "Dividir"

KOP / FUP	SCL	Descripción
	<pre> out := in1 + in2; out := in1 - in2; out := in1 * in2; out := in1 / in2; </pre>	<ul style="list-style-type: none"> • ADD: Sumar (IN1 + IN2 = OUT) • SUB: Restar (IN1 - IN2 = OUT) • MUL: Multiplicar (IN1 * IN2 = OUT) • DIV: Dividir (IN1 / IN2 = OUT) <p>Una operación de división de enteros trunca la parte fraccionaria del cociente y produce un valor de salida entero.</p>

¹ En KOP y FUP: haga clic en "???" y seleccione un tipo de datos en el menú desplegable.

Tabla 7- 38 Tipos de datos para los parámetros (KOP y FUP)

Parámetro	Tipo de datos ¹	Descripción
IN1, IN2	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, constante	Entradas de la operación matemática
OUT	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal	Salida de la operación matemática

¹ Los parámetros IN1, IN2 y OUT deben ser del mismo tipo de datos.



Para agregar una entrada ADD o MUL, haga clic en el icono "Crear" o haga clic con el botón derecho del ratón en el conector de entrada del parámetro IN existente y seleccione el comando "Insertar entrada".

Para quitar una entrada, haga clic con el botón derecho del ratón en el conector de entrada de uno de los parámetros IN existentes (si hay más entradas además de las dos originales) y seleccione el comando "Borrar".

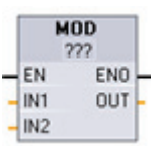
Si está habilitada (EN = 1), la instrucción matemática realiza la operación indicada en los valores de entrada (IN1 e IN2) y almacena el resultado en la dirección de memoria que indica el parámetro de salida (OUT). Una vez finalizada correctamente la operación, la instrucción pone ENO a 1.

Tabla 7- 39 Estado de ENO

ENO	Descripción
1	Sin error
0	El resultado de la operación matemática quedaría fuera del rango numérico válido del tipo de datos seleccionado. Se devuelve la parte menos significativa del resultado que quepa en el tamaño de destino.
0	División por 0 (IN2 = 0): El resultado es indefinido y se devuelve cero.
0	Real/LReal: Si uno de los valores de entrada es NaN (no es un número), se devuelve NaN.
0	ADD Real/LReal: Si ambos valores IN son INF con signos diferentes, la operación no está permitida y se devuelve NaN.
0	SUB Real/LReal: Si ambos valores IN son INF con signos iguales, la operación no está permitida y se devuelve NaN.
0	MUL Real/LReal: Si un valor IN es cero y el otro es INF, la operación no está permitida y se devuelve NaN.
0	DIV Real/LReal: Si ambos valores IN son cero o INF, la operación no está permitida y se devuelve NaN.

7.5.3 Instrucción Modulo

Tabla 7- 40 Instrucción "Obtener resto de división"

KOP / FUP	SCL	Descripción
	<pre>out := in1 MOD in2;</pre>	La instrucción MOD se puede utilizar para obtener el resto de una operación de división de enteros. El valor de la entrada IN1 se divide por el valor de la entrada IN2 y el producto se deposita en la salida OUT.

¹ En KOP y FUP: haga clic en "???" y seleccione un tipo de datos en el menú desplegable.

Tabla 7- 41 Tipos de datos para parámetros

Parámetro	Tipo de datos ¹	Descripción
IN1 y IN2	SInt, Int, DInt, USInt, UInt, UDIInt, constante	Entradas modulo
OUT	SInt, Int, DInt, USInt, UInt, UDIInt	Salida modulo


¹ Los parámetros IN1y IN2y OUTdeben ser del mismo tipo de datos.

Tabla 7- 42 Valores ENO

ENO	Descripción
1	Sin error
0	Valor IN2 = 0, a OUT se le asigna el valor cero

7.5.4 Instrucción de negación

Tabla 7- 43 Instrucción NEG

KOP / FUP	SCL	Descripción
	- (i.n) ;	La instrucción NEG invierte el signo aritmético del valor del parámetro IN y deposita el resultado en el parámetro OUT.

¹ En KOP y FUP: haga clic en "???" y seleccione un tipo de datos en el menú desplegable.

Tabla 7- 44 Tipos de datos para parámetros

Parámetro	Tipo de datos ¹	Descripción
IN	SInt, Int, DInt, Real, LReal, constante	Entrada de la operación matemática
OUT	SInt, Int, DInt, Real, LReal	Salida de la operación matemática

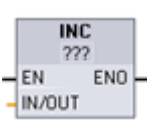
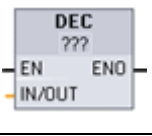
¹ Los parámetros IN y OUT deben ser del mismo tipo de datos.

Tabla 7- 45 Estado de ENO

ENO	Descripción
1	Sin error
0	El resultado está fuera del rango numérico válido del tipo de datos seleccionado. Ejemplo de SInt: NEG (-128) arroja el resultado +128 que excede el límite máximo del tipo de datos.

7.5.5 Instrucciones "Incrementar" y "Decrementar"

Tabla 7- 46 Instrucciones INC y DEC

KOP / FUP	SCL	Descripción
	<code>in_out := in_out + 1;</code>	Incrementa un valor de número entero con o sin signo: Valor IN_OUT +1 = valor IN_OUT
	<code>in_out := in_out - 1;</code>	Decrementa un valor de número entero con o sin signo: Valor IN_OUT - 1 = valor IN_OUT

¹ En KOP y FUP: haga clic en "???" y seleccione un tipo de datos en el menú desplegable.

Tabla 7- 47 Tipos de datos para parámetros

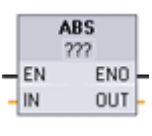
Parámetro	Tipo de datos	Descripción
IN/OUT	SInt, Int, DInt, USInt, UInt, UDIInt	Entrada/salida de la operación matemática

Tabla 7- 48 Estado de ENO

ENO	Descripción
1	Sin error
0	El resultado está fuera del rango numérico válido del tipo de datos seleccionado. Ejemplo de SInt: INC (+127) arroja el resultado +128, que excede el límite máximo del tipo de datos.

7.5.6 Instrucción "Calcular valor absoluto"

Tabla 7- 49 Instrucción ABS

KOP / FUP	SCL	Descripción
	<code>out := ABS(in);</code>	Calcula el valor absoluto de un entero con signo o número real indicado en el parámetro IN y deposita el resultado en el parámetro OUT.

¹ En KOP y FUP: haga clic en "???" y seleccione un tipo de datos en el menú desplegable.

Tabla 7- 50 Tipos de datos para parámetros

Parámetro	Tipo de datos ¹	Descripción
IN	SInt, Int, DInt, Real, LReal	Entrada de la operación matemática
OUT	SInt, Int, DInt, Real, LReal	Salida de la operación matemática

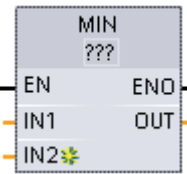
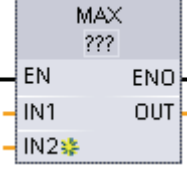
¹ Los parámetros IN y OUT deben tener un mismo tipo de datos.

Tabla 7- 51 Estado de ENO

ENO	Descripción
1	Sin error
0	El resultado de la operación aritmética está fuera del rango numérico válido del tipo de datos seleccionado. Ejemplo de SInt: ABS (-128) arroja el resultado +128 que excede el límite máximo del tipo de datos.

7.5.7 Instrucciones de mínimo y máximo

Tabla 7- 52 Instrucciones MIN y MAX

KOP / FUP	SCL	Descripción
	<pre>out:= MIN(in1:=_variant_in_, in2:=_variant_in_ [...in32]);</pre>	La instrucción MIN compara el valor de dos parámetros IN1 y IN2 y asigna el valor mínimo (menor) al parámetro OUT.
	<pre>out:= MAX(in1:=_variant_in_, in2:=_variant_in_ [...in32]);</pre>	La instrucción MAX compara el valor de dos parámetros IN1 y IN2 y asigna el valor máximo (mayor) al parámetro OUT.

¹ En KOP y FUP: haga clic en "???" y seleccione un tipo de datos en el menú desplegable.

Tabla 7- 53 Tipos de datos para los parámetros

Parámetro	Tipo de datos ¹	Descripción
IN1, IN2 [...IN32]	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, constante	Entradas de la operación matemática (hasta 32 entradas)
OUT	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal	Salida de la operación matemática

¹ Los parámetros IN1, IN2 y OUT deben tener un mismo tipo de datos.



Para agregar una entrada, haga clic en el icono "Crear" o haga clic con el botón derecho del ratón en el conector de entrada del parámetro IN existente y seleccione el comando "Insertar entrada".

Para quitar una entrada, haga clic con el botón derecho del ratón en el conector de entrada de uno de los parámetros IN existentes (si hay más entradas además de las dos originales) y seleccione el comando "Borrar".

Tabla 7- 54 Estado de ENO

ENO	Descripción
1	Sin error
0	Sólo para el tipo de datos Real: <ul style="list-style-type: none"> • Una o más entradas no son un número real (NaN). • La salida OUT resultante es +/- INF (infinito).

7.5.8 Instrucción Limit

Tabla 7- 55 Instrucción "Ajustar valor límite"

KOP / FUP	SCL	Descripción
	LIMIT (MN:= _variant_in_, IN:= _variant_in_, MX:= _variant_in_, OUT:= _variant_out_);	La instrucción Limit comprueba si el valor del parámetro IN se encuentra dentro del rango de valores especificado por los parámetros MIN y MAX and if not, clamps the value at MIN or MAX.

¹ En KOP y FUP: haga clic en "???" y seleccione un tipo de datos en el menú desplegable.

Tabla 7- 56 Tipos de datos para los parámetros

Parámetro	Tipo de datos ¹	Descripción
MN, INy MX	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, constante	Entradas de operación matemática
OUT	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal	Salida de operación matemática

¹ Los parámetros MN, IN, MXy OUTdeben ser del mismo tipo de datos.

Si el valor del parámetro IN está dentro del rango indicado, el valor de IN se deposita en el parámetro OUT. Si el valor del parámetro IN está fuera del rango indicado, el valor OUT es entonces el valor del parámetro MIN (si el valor IN es menor que el valor MIN) o del parámetro MAX (si el valor IN es mayor que el valor MAX).

Tabla 7- 57 Estado de ENO

ENO	Descripción
1	No hay error
0	Real: Si uno o varios de los valores de MIN, IN y MAX es NaN (no es un número), se devuelve NaN.
0	Si MIN es mayor que MAX, el valor IN se asigna a OUT.

Ejemplos de SCL:

- MyVal := LIMIT(MN:=10,IN:=53, MX:=40); //Resultado: MyVal = 40
- MyVal := LIMIT(MN:=10,IN:=37, MX:=40); //Resultado: MyVal = 37
- MyVal := LIMIT(MN:=10,IN:=8, MX:=40); //Resultado: MyVal = 10

7.5.9 Instrucciones matemáticas en coma flotante



Las instrucciones en coma flotante sirven para programar operaciones matemáticas utilizando los tipos de datos Real o LReal:

- SQR: Calcular cuadrado ($IN^2 = OUT$)
- SQRT: Calcular raíz cuadrada ($\sqrt{IN} = OUT$)
- LN: Calcular logaritmo natural ($LN(IN) = OUT$)
- EXP: Calcular valor exponencial ($e^{IN} = OUT$), donde la base e = 2.71828182845904523536
- EXPT: Elevar a potencia ($IN1^{IN2} = OUT$)

Los parámetros de EXPT IN1 y OUT siempre son del mismo tipo de datos, que puede ser Real o LReal. Para el parámetro del exponente IN2 se puede elegir entre varios tipos de datos.

- FRAC: Determinar decimales (parte fraccionaria del número en coma flotante $IN = OUT$)
- SIN: Calcular valor de seno ($\text{seno}(IN \text{ radianes}) = OUT$)
ASIN: Calcular valor de arcoseno ($\text{arcoseno}(IN) = OUT \text{ radianes}$), donde $\text{seno}(OUT \text{ radianes}) = IN$
- COS: Calcular valor de coseno ($\text{coseno}(IN \text{ radianes}) = OUT$)
ACOS: Calcular valor de arcocoseno ($\text{arcocoseno}(IN) = OUT \text{ radianes}$), donde $\text{coseno}(OUT \text{ radianes}) = IN$
- TAN: Calcular valor de tangente ($\text{tangente}(IN \text{ radianes}) = OUT$)
ATAN: Calcular valor de arcotangente ($\text{arcotangente}(IN) = OUT \text{ radianes}$), donde $\text{tangente}(OUT \text{ radianes}) = IN$

Tabla 7- 58 Ejemplos de instrucciones matemáticas en coma flotante

KOP / FUP	SCL	Descripción
	<pre>out := SQR(in); 0 out := in * in;</pre>	Cuadrado: $IN^2 = OUT$ Ejemplo: Si $IN = 9$, $OUT = 81$.
	<pre>out := in1 ** in2;</pre>	Exponente general: $IN1^{IN2} = OUT$ Ejemplo: Si $IN1 = 3$ y $IN2 = 2$, $OUT = 9$.

- 1 En KOP y FUP: haga clic en "???" (junto al nombre de la instrucción) y seleccione un tipo de datos en el menú desplegable.
- 2 En SCL: También es posible utilizar los operadores matemáticos básicos de SCL para crear expresiones matemáticas.

Tabla 7- 59 Tipos de datos para parámetros

Parámetro	Tipo de datos	Descripción
IN, IN1	Real, LReal, constante	Entradas
IN2	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, constante	Entrada exponencial EXPT
OUT	Real, LReal	Salidas

Tabla 7- 60 Estado de ENO

ENO	Instrucción	Condición	Resultado (OUT)
1	Todas	No hay error	Resultado válido
0	SQR	El resultado excede el rango Real/LReal válido	+INF
		IN es +/- NaN (no es un número)	+NaN
	SQRT	IN es negativo	-NaN
		IN es +/- INF (infinito) o +/- NaN	+/- INF o +/- NaN
	LN	IN es 0,0, negativo, -INF o -NaN	-NaN
		IN es +INF o +NaN	+INF o +NaN
	EXP	El resultado excede el rango Real/LReal válido	+INF
		IN es +/- NaN	+/- NaN
	SIN, COS, TAN	IN es +/- INF o +/- NaN	+/- INF o +/- NaN
	ASIN, ACOS	IN está fuera del rango válido de -1,0 a +1,0	+NaN
		IN es +/- NaN	+/- NaN
	ATAN	IN es +/- NaN	+/- NaN
	FRAC	IN es +/- INF o +/- NaN	+NaN
	EXPT	IN1 es +INF e IN2 no es -INF	+INF
IN1 es negativo o -INF		+NaN si IN2 es Real/LReal, -INF en caso contrario	
IN1 o IN2 es +/- NaN		+NaN	

ENO	Instrucción	Condición	Resultado (OUT)
		IN1 es 0,0 e IN2 es Real/LReal (sólo)	+NaN

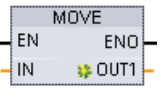
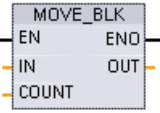
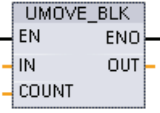
7.6 Desplazamiento

7.6.1 Instrucciones "Copiar valor" y "Copiar área"

Las instrucciones de desplazamiento permiten copiar elementos de datos en otra dirección de memoria y convertir un tipo de datos en otro. El proceso de desplazamiento no modifica los datos de origen.

- La instrucción MOVE copia un elemento de datos individual de la dirección de origen que indica el parámetro IN en la dirección de destino que indica el parámetro OUT.
- Las instrucciones MOVE_BLK y UMOVE_BLK tienen un parámetro COUNT adicional. COUNT especifica cuántos elementos de datos se copian. El número de bytes por elemento copiado depende del tipo de datos asignado a los nombres de variables de los parámetros IN y OUT en la tabla de variables PLC.

Tabla 7- 61 Instrucciones MOVE, MOVE_BLK y UMOVE_BLK

KOP / FUP	SCL	Descripción
	<pre>out1 := in;</pre>	Copia un elemento de datos almacenado en una dirección indicada en una o varias direcciones diferentes. ¹
	<pre>MOVE_BLK (in:=_variant_in, count:=_uint_in, out=>_variant_out);</pre>	Desplazamiento con interrupciones que copia un bloque de elementos de datos en otra dirección.
	<pre>UMOVE_BLK (in:=_variant_in, count:=_uint_in, out=>_variant_out);</pre>	Desplazamiento sin interrupciones que copia un bloque de elementos de datos en otra dirección.

¹ Instrucción MOVE: para agregar otra salida en KOP o FUP, haga clic en el icono "Crear" situado junto al parámetro de salida. Para SCL, utilice múltiples instrucciones de asignación. También se puede usar una de las construcciones de bucle.

Tabla 7- 62 Tipos de datos para la instrucción MOVE

Parámetro	Tipo de datos	Descripción
IN	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, Byte, Word, DWord, Char, Array, Struct, DTL, Time	Dirección de origen
OUT	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, Byte, Word, DWord, Char, Array, Struct, DTL, Time	Dirección de destino



Para agregar salidas MOVE, haga clic en el icono "Crear" o haga clic con el botón derecho en el conector de salida de uno de los parámetros OUT existentes y seleccione el comando "Insertar salida".

Para eliminar una salida, haga clic con el botón derecho del ratón en el conector de salida de uno de los parámetros OUT existentes (si hay más salidas además de las dos originales) y seleccione el comando "Borrar".

Tabla 7- 63 Tipos de datos para las instrucciones MOVE_BLK y UMOVE_BLK

Parámetro	Tipo de datos	Descripción
IN	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal Byte, Word, DWord	Dirección de origen inicial
COUNT	UInt	Número de elementos de datos que deben copiarse
OUT	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, Byte, Word, DWord	Dirección de destino inicial

Nota

Reglas para las operaciones de copia de datos

- Para copiar el tipo de datos Bool utilice SET_BF, RESET_BF, R, S o bobina de salida (KOP) (Página 182)
- Para copiar un solo tipo de datos simple, utilice MOVE
- Para copiar una matriz de un tipo de datos simple, utilice MOVE_BLK o UMOVE_BLK
- Para copiar una estructura, utilice MOVE
- Para copiar una cadena, utilice S_MOVE (Página 265)
- Para copiar un solo carácter en una cadena, utilice MOVE
- Las instrucciones MOVE_BLK y UMOVE_BLK no pueden utilizarse para copiar matrices o estructuras en las áreas de memoria I, Q o M.

Las instrucciones MOVE_BLK y UMOVE_BLK se diferencian en la forma de procesar las alarmas:

- Los eventos de alarma **se ponen en cola de espera y se procesan** durante la ejecución de MOVE_BLK. Utilice la instrucción MOVE_BLK si los datos contenidos en la dirección de destino del desplazamiento no se utilizan en un OB de alarma. Si se utilizan, los datos de destino no tienen que ser coherentes. Si se interrumpe una instrucción MOVE_BLK, el último elemento de datos desplazado estará completo y será coherente en la dirección de destino. La instrucción MOVE_BLK se reanuda una vez finalizada la ejecución del OB de alarma.
- Los eventos de alarma **se ponen en cola de espera pero no se procesarán** hasta que no finalice la ejecución de UMOVE_BLK. Utilice la instrucción UMOVE_BLK si la operación de desplazamiento debe finalizarse y los datos de destino deben ser coherentes antes de la ejecución de un OB de alarma. Encontrará más información en el apartado Coherencia de datos (Página 157).

ENO siempre es verdadero tras ejecutarse la instrucción MOVE.

Tabla 7- 64 Estado de ENO

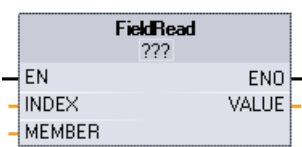

ENO	Condición	Resultado
1	No hay error	Todos los elementos de COUNT se han copiado correctamente.
0	El rango de origen (IN) o destino (OUT) excede el área de memoria disponible.	Se copian los elementos que quepan. No se copian elementos parciales.

7.6.2 Instrucciones FieldRead y FieldWrite

Nota

STEP 7 V10.5 **no soportaba** una referencia variable como índice de matriz o matrices multidimensionales. Las instrucciones FieldRead y FieldWrite se utilizaban para proporcionar operaciones de índice de matriz variables para una matriz unidimensional. STEP 7 V11 **sí que soporta** una variable como índice de matriz y matrices multidimensionales. FieldRead y FieldWrite se han incluido en STEP 7 V11 para disponer de una compatibilidad retroactiva con programas que utilizaban estas instrucciones.

Tabla 7- 65 Instrucciones FieldRead y FieldWrite

KOP / FUP	SCL	Descripción
	<pre>value := member[index];</pre>	FieldRead lee el elemento de matriz con el valor de índice INDEX de la matriz cuyo primer elemento está especificado por el parámetro MEMBER. El valor del elemento de matriz se transfiere a la posición especificada en el parámetro VALUE.
	<pre>member[index] := value;</pre>	WriteField transfiere el valor de la posición especificada por el parámetro VALUE a la matriz cuyo primer elemento se especifica en el parámetro MEMBER. El valor se transfiere al elemento de la matriz cuyo índice de matriz se especifica en el parámetro INDEX.

¹ En KOP y FUP: haga clic en "???" y seleccione un tipo de datos en el menú desplegable.

Tabla 7- 66 Tipos de datos para parámetros

Parámetro y tipo		Tipo de datos	Descripción
Índice alfabético	Entrada	DInt	El número de índice del elemento de matriz que debe leerse o en el que debe escribirse
Miembro ¹	Entrada	Tipos de elementos de matriz: Bool, Byte, Word, DWord, Char, SInt, Int, Dint, USInt, UInt, UDInt, Real, LReal	Posición del primer elemento en una matriz unidimensional definida en un bloque de datos global o una interfaz de bloque. Ejemplo: Si el índice de matriz se especifica como [-2..4], el índice del primer elemento es -2 y no 0.
Valor ¹	Out	Bool, Byte, Word, DWord, Char, SInt, Int, Dint, USInt, UInt, UDInt, Real, LReal	Posición en la que se copia el elemento de matriz especificado (FieldRead) Posición del valor que se copia en el elemento de matriz especificado (FieldWrite)

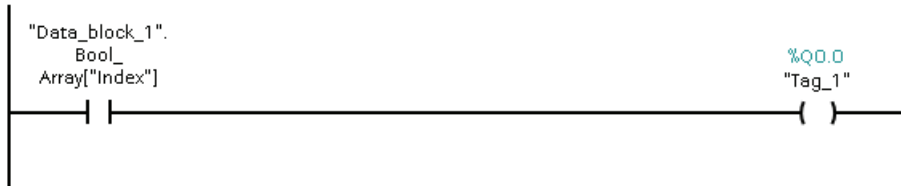
¹ El tipo de datos del elemento de matriz especificado por los parámetros MEMBER y VALUE debe tener el mismo tipo de datos.

La salida de habilitación ENO es 0 si se cumple una de las siguientes condiciones:

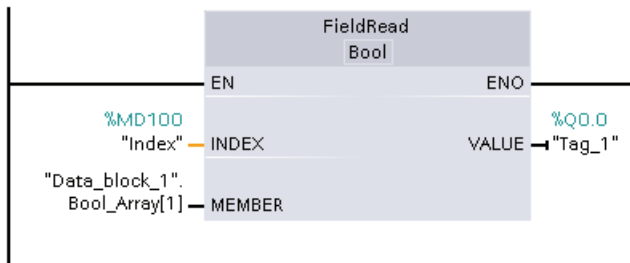
- La entrada EN tiene el estado lógico "0"
- El elemento de matriz especificado en el parámetro INDEX no está definido en la matriz referenciada en el parámetro MEMBER
- Durante el procesamiento se produce un error, p. ej. un rebosamiento

Acceso a datos mediante indexación de matriz

Para acceder a los elementos de una matriz con una variable, basta con utilizar la variable como índice de matriz en la lógica del programa. Por ejemplo, el segmento siguiente activa una salida basada en el valor booleano de una matriz de booleanos en "Data_block_1" referenciada por la variable PLC "Índice".



La lógica del índice de matriz variable es equivalente al método antiguo utilizando la instrucción FieldRead:



Las instrucciones FieldWrite y FieldRead se pueden sustituir por la lógica de índice de matriz variable.

SCL no tiene instrucciones FieldRead ni FieldWrite, pero admite el direccionamiento indirecto de una matriz con una variable:

```
#Tag_1 := "Data_block_1".Bool_Array[#Index];
```

7.6.3 Instrucciones de relleno

Tabla 7- 67 Instrucciones FILL_BLK y UFILL_BLK

KOP / FUP	SCL	Descripción
	<pre>FILL_BLK (in:=_variant_in, count:=int, out=>_variant_out);</pre>	Instrucción "Rellenar área sin interrupciones": rellena un área de direcciones con copias de un elemento de datos específico
	<pre>UFILL_BLK (in:=_variant_in, count:=int out=>_variant_out);</pre>	Instrucción "Rellenar área sin interrupciones": Rellena un área de direcciones con copias de un elemento de datos específico

Tabla 7- 68 Tipos de datos para parámetros

Parámetro	Tipo de datos	Descripción
IN	SInt, Int, DIntT, USInt, UInt, UDIInt, Real, LReal, Byte, Word, DWord	Dirección de origen de los datos
COUNT	USInt, UInt	Número de elementos de datos que deben copiarse
OUT	SInt, Int, DIntT, USInt, UInt, UDIInt, Real, LReal, Byte, Word, DWord	Dirección de destino de los datos

Nota**Reglas para las operaciones de rellenar área**

- Para rellenar el tipo de datos BOOL utilice SET_BF, RESET_BF, R, S, o bobina de salida (KOP)
- Para rellenar con un solo tipo de datos simple, utilice MOVE
- Para rellenar una matriz con un tipo de datos simple, utilice FILL_BLK o UFILL_BLK
- Para rellenar un solo carácter en una cadena, utilice MOVE
- Las instrucciones FILL_BLK y UFILL_BLK no pueden utilizarse para rellenar matrices en las áreas de memoria I, Q o M.

Las instrucciones FILL_BLK y UFILL_BLK copian el elemento de datos de origen IN en el destino, cuya dirección inicial se indica en el parámetro OUT. El proceso de copia se repite y un área de direcciones adyacentes se rellena hasta que el número de copias sea igual al parámetro COUNT.

Las instrucciones FILL_BLK y UFILL_BLK se diferencian en la forma de procesar las alarmas:

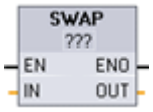
- Los eventos de alarma **se ponen en cola de espera y se procesan** durante la ejecución de FILL_BLK. Utilice la instrucción FILL_BLK si los datos contenidos en la dirección de destino del desplazamiento no se utilizan en un OB de alarma. Si se utilizan, los datos de destino no tienen que ser coherentes.
- Los eventos de alarma **se ponen en cola de espera pero no se procesarán** hasta que no finalice la ejecución de UFILL_BLK. Utilice la instrucción UFILL_BLK si la operación de desplazamiento debe finalizarse y los datos de destino deben ser coherentes antes de la ejecución de un OB de alarma.

Tabla 7- 69 Estado de ENO

ENO	Condición	Resultado
1	No hay error	El elemento IN se ha copiado correctamente a todos los destinos de COUNT.
0	El rango de destino (OUT) excede el área de memoria disponible	Se copian los elementos que quepan. No se copian elementos parciales.

7.6.4 Instrucción "Cambiar disposición"

Tabla 7- 70 Instrucción SWAP

KOP / FUP	SCL	Descripción
	<pre>out := SWAP(in);</pre>	Invierte el orden de los bytes para elementos de datos de dos bytes y cuatro bytes. El orden de los bits no se modifica dentro de los distintos bytes. ENO es siempre TRUE (verdadero) tras ejecutarse la instrucción SWAP.

¹ En KOP y FUP: haga clic en "???" y seleccione un tipo de datos en el menú desplegable.

Tabla 7- 71 Tipos de datos para los parámetros

Parámetro	Tipo de datos	Descripción
IN	Word, DWord	Bytes de datos ordenados en IN
OUT	Word, DWord	Bytes de datos en orden inverso en OUT

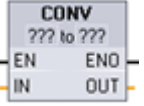
Ejemplo 1	Parámetro IN = MB0 (antes de la ejecución)		Parámetro OUT = MB4 (tras la ejecución)	
Dirección	MB0	MB1	MB4	MB5
W#16#1234	12	34	34	12
WORD	MSB	LSB	MSB	LSB

Ejemplo 2	Parámetro IN = MB0 (antes de la ejecución)				Parámetro OUT = MB4 (tras la ejecución)			
Dirección	MB0	MB1	MB2	MB3	MB4	MB5	MB6	MB7
DW#16# 12345678	12	34	56	78	78	56	34	12
DWORD	MSB			LSB	MSB		LSB	

7.7 Convertir

7.7.1 Instrucción CONV

Tabla 7- 72 Instrucción Convertir valor (CONV)

KOP / FUP	SCL	Descripción
	<pre>out := <data type in>_TO_<data type out>(in);</pre>	Convierte un elemento de datos de un tipo de datos a otro tipo de datos.

- 1 En KOP y FUP: haga clic en "???" y seleccione los tipos de datos del menú desplegable.
- 2 En SCL: Defina la instrucción de conversión identificando el tipo de datos del parámetro de entrada (in) y el parámetro de salida (out). Por ejemplo, DWORD_TO_REAL convierte un valor DWord en un valor Real.

Tabla 7- 73 Tipos de datos para los parámetros

Parámetro	Tipo de datos	Descripción
IN	Cadena de bits ¹ , SInt, USInt, Int, UInt, DInt, UDInt, Real, LReal, BCD16, BCD32	Valor de entrada
OUT	Cadena de bits ¹ , SInt, USInt, Int, UInt, DInt, UDInt, Real, LReal, BCD16, BCE32	Valor de entrada convertido a un nuevo tipo de datos

- 1 La instrucción no le permite seleccionar cadenas de bits (Byte, Word, DWord). Para introducir como parámetro de la instrucción un operando del tipo de datos Byte, Word o DWord, seleccione un entero sin signo con la misma longitud en bits. Por ejemplo, USInt para un Byte, UInt para un Word o UDInt para un DWord.

Tras haber seleccionado el tipo de datos que se desea convertir, las conversiones posibles aparecen en la lista desplegable (convertir a). Las conversiones de y a BCD16 están limitadas al tipo de datos Int. Las conversiones de y a BCD32 están limitadas al tipo de datos DInt.

Tabla 7- 74 Estado de ENO

ENO	Descripción	Resultado de OUT
1	Sin error	Resultado válido
0	IN es +/- INF o +/- NaN	+/- INF o +/- NaN
0	El resultado excede el rango válido del tipo de datos de OUT	OUT se ajusta a los bytes menos significativos de IN

7.7.2 Instrucciones de conversión de SCL

Instrucciones de conversión de SCL

Tabla 7- 75 Conversión de un Bool, Byte, Word o DWord

Tipo de datos	Instrucción	Resultado
Bool	BOOL_TO_BYTE, BOOL_TO_WORD, BOOL_TO_DWORD, BOOL_TO_INT, BOOL_TO_DINT	El valor se transfiere al bit menos significativo del tipo de datos de destino.
Byte	BYTE_TO_BOOL	El bit menos significativo se transfiere al tipo de datos de destino.
	BYTE_TO_WORD, BYTE_TO_DWORD	El valor se transfiere al byte low del tipo de datos de destino.
	BYTE_TO_SINT, BYTE_TO_USINT	El valor se transfiere al tipo de datos de destino.
	BYTE_TO_INT, BYTE_TO_UINT, BYTE_TO_DINT, BYTE_TO_UDINT	El valor se transfiere al byte menos significativo del tipo de datos de destino.
Word	WORD_TO_BOOL	El bit menos significativo se transfiere al tipo de datos de destino.
	WORD_TO_BYTE	El byte low del valor fuente se transfiere al tipo de datos de destino.
	WORD_TO_DWORD	El valor se transfiere al byte low del tipo de datos de destino.
	WORD_TO_SINT, WORD_TO_USINT	El byte low del valor fuente se transfiere al tipo de datos de destino.
	WORD_TO_INT, WORD_TO_UINT	El valor se transfiere al tipo de datos de destino.
	WORD_TO_DINT, WORD_TO_UDINT	El valor se transfiere al byte low del tipo de datos de destino.
DWord	DWORD_TO_BOOL	El bit menos significativo se transfiere al tipo de datos de destino.
	DWORD_TO_BYTE, DWORD_TO_WORD, DWORD_TO_SINT, DWORD_TO_USINT, DWORD_TO_INT, DWORD_TO_UINT	El byte low del valor fuente se transfiere al tipo de datos de destino.
	DWORD_TO_DINT, DWORD_TO_UDINT, DWORD_TO_REAL	El valor se transfiere al tipo de datos de destino.

Tabla 7- 76 Conversión de un entero corto (SInt o USInt)

Tipo de datos	Instrucción	Resultado
SInt	SINT_TO_BOOL	El bit menos significativo se transfiere al tipo de datos de destino.
	SINT_TO_BYTE	El valor se transfiere al tipo de datos de destino.
	SINT_TO_WORD, SINT_TO_DWORD, SINT_TO_INT, SINT_TO_DINT	El valor se transfiere al byte low del tipo de datos de destino.

Tipo de datos	Instrucción	Resultado
	SINT_TO_USINT, SINT_TO_UINT, SINT_TO_UDINT, SINT_TO_REAL, SINT_TO_LREAL, SINT_TO_CHAR, SINT_TO_STRING	El valor está convertido.
USInt	USINT_TO_BOOL	El bit menos significativo se transfiere al tipo de datos de destino.
	USINT_TO_BYTE	El valor se transfiere al tipo de datos de destino.
	USINT_TO_WORD, USINT_TO_DWORD, USINT_TO_INT, USINT_TO_UINT, USINT_TO_DINT, USINT_TO_UDINT	El valor se transfiere al byte low del tipo de datos de destino.
	USINT_TO_SINT, USINT_TO_REAL, USINT_TO_LREAL, USINT_TO_CHAR, USINT_TO_STRING	El valor está convertido.

Tabla 7-77 Conversión de un entero (Int o UInt)

Tipo de datos	Instrucción	Resultado
Int	INT_TO_BOOL	El bit menos significativo se transfiere al tipo de datos de destino.
	INT_TO_BYTE, INT_TO_DWORD, INT_TO_SINT, INT_TO_USINT, INT_TO_UINT, INT_TO_UDINT, INT_TO_REAL, INT_TO_LREAL, INT_TO_CHAR, INT_TO_STRING	El valor está convertido.
	INT_TO_WORD	El valor se transfiere al tipo de datos de destino.
	INT_TO_DINT	El valor se transfiere al byte low del tipo de datos de destino.
UInt	UINT_TO_BOOL	El bit menos significativo se transfiere al tipo de datos de destino.
	UINT_TO_BYTE, UINT_TO_SINT, UINT_TO_USINT, UINT_TO_INT, UINT_TO_REAL, UINT_TO_LREAL, UINT_TO_CHAR, UINT_TO_STRING	El valor está convertido.
	UINT_TO_WORD, UINT_TO_DATE	El valor se transfiere al tipo de datos de destino.
	UINT_TO_DWORD, UINT_TO_DINT, UINT_TO_UDINT	El valor se transfiere al byte low del tipo de datos de destino.

Tabla 7-78 Conversión de un entero doble (Dint o UInt)

Tipo de datos	Instrucción	Resultado
DInt	DINT_TO_BOOL	El bit menos significativo se transfiere al tipo de datos de destino.
	DINT_TO_BYTE, DINT_TO_WORD, DINT_TO_SINT, DINT_TO_USINT, DINT_TO_INT, DINT_TO_UINT, DINT_TO_UDINT, DINT_TO_REAL, DINT_TO_LREAL, DINT_TO_CHAR, DINT_TO_STRING	El valor está convertido.

7.7 Convertir

Tipo de datos	Instrucción	Resultado
	DINT_TO_DWORD, DINT_TO_TIME	El valor se transfiere al tipo de datos de destino.
UDInt	UDINT_TO_BOOL	El bit menos significativo se transfiere al tipo de datos de destino.
	UDINT_TO_BYTE, UDINT_TO_WORD, UDINT_TO_SINT, UDINT_TO_USINT, UDINT_TO_INT, UDINT_TO_UINT, UDINT_TO_DINT, UDINT_TO_REAL, UDINT_TO_LREAL, UDINT_TO_CHAR, UDINT_TO_STRING	El valor está convertido.
	UDINT_TO_DWORD, UDINT_TO_TOD	El valor se transfiere al tipo de datos de destino.

Tabla 7- 79 Conversión de un número real (Real o LReal)

Tipo de datos	Instrucción	Resultado
Real	REAL_TO_DWORD, REAL_TO_LREAL	El valor se transfiere al tipo de datos de destino.
	REAL_TO_SINT, REAL_TO_USINT, REAL_TO_INT, REAL_TO_UINT, REAL_TO_DINT, REAL_TO_UDINT, REAL_TO_STRING	El valor está convertido.
LReal	LREAL_TO_SINT, LREAL_TO_USINT, LREAL_TO_INT, LREAL_TO_UINT, LREAL_TO_DINT, LREAL_TO_UDINT, LREAL_TO_REAL, LREAL_TO_STRING	El valor está convertido.

Tabla 7- 80 Conversión de Time, DTL, TOD o Date



Tipo de datos	Instrucción	Resultado
Time	TIME_TO_DINT	El valor se transfiere al tipo de datos de destino.
DTL	DTL_TO_DATE, DTL_TO_TOD	El valor está convertido.
TOD	TOD_TO_UDINT	El valor está convertido.
Date	DATE_TO_UINT	El valor está convertido.

Tabla 7- 81 Conversión de un Char o String

Tipo de datos	Instrucción	Resultado
Char	CHAR_TO_SINT, CHAR_TO_USINT, CHAR_TO_INT, CHAR_TO_UINT, CHAR_TO_DINT, CHAR_TO_UDINT	El valor está convertido.
	CHAR_TO_STRING	El valor se transfiere al primer carácter de la cadena.
String	STRING_TO_SINT, STRING_TO_USINT, STRING_TO_INT, STRING_TO_UINT, STRING_TO_DINT, STRING_TO_UDINT, STRING_TO_REAL, STRING_TO_LREAL	El valor está convertido.
	STRING_TO_CHAR	El primer carácter de la cadena se copia en Char.

7.7.3 Instrucciones "Redondear número" y "Truncar a entero doble"

Tabla 7- 82 Instrucciones ROUND y TRUNC

KOP / FUP	SCL	Descripción
	<code>out := ROUND (in);</code>	<p>Convierte un número real en un entero. El tipo de datos predeterminado es DINT. Cuando la salida es un tipo de datos válido distinto de DINT, debe declararse de forma explícita; por ejemplo, ROUND_REAL o ROUND_LREAL.</p> <p>La fracción del número real se redondea al número entero más cercano (IEEE - redondear al número más cercano). Si el número se encuentra exactamente entre dos enteros (p. ej. 10,5), el número se redondeará al entero par. Ejemplo:</p> <ul style="list-style-type: none"> • ROUND (10.5) = 10 • ROUND (11.5) = 12
	<code>out := TRUNC (in);</code>	<p>TRUNC convierte un número real en un entero. La parte fraccionaria del número real se trunca a cero (IEEE - redondear hacia cero).</p>

- 1 En KOP y FUP: haga clic en "???" (junto al nombre de la instrucción) y seleccione un tipo de datos en el menú desplegable.

Tabla 7- 83 Tipos de datos para los parámetros

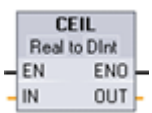

Parámetro	Tipo de datos	Descripción
IN	Real, LReal	Número en coma flotante en la entrada
OUT	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal	Salida redondeada o truncada

Tabla 7- 84 Estado de ENO

ENO	Descripción	Resultado de OUT
1	No hay error	Resultado válido
0	IN es +/- INF o +/- NaN	+/- INF o +/- NaN

7.7.4 Instrucciones "Crear el siguiente número entero superior a partir del número en coma flotante" y "Crear el siguiente número entero inferior a partir del número en coma flotante"

Tabla 7- 85 Instrucciones CEIL y FLOOR

KOP / FUP	SCL	Descripción
	<code>out := CEIL(in);</code>	Convierte un número real (Real o LReal) en el siguiente entero mayor o igual a ese número real (IEEE - redondear hacia el infinito positivo).
	<code>out := FLOOR(in);</code>	Convierte un número real (Real o LReal) en el siguiente entero menor o igual a ese número real (IEEE - redondear hacia el infinito negativo).

1 En KOP y FUP: haga clic en "???" (junto al nombre de la instrucción) y seleccione un tipo de datos en el menú desplegable.

Tabla 7- 86 Tipos de datos para los parámetros

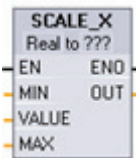
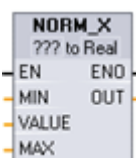
Parámetro	Tipo de datos	Descripción
IN	Real, LReal	Número en coma flotante en la entrada
OUT	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal	Salida convertida

Tabla 7- 87 Estado de ENO

ENO	Descripción	Resultado de OUT
1	Sin error	Resultado válido
0	IN es +/- INF o +/- NaN	+/- INF o +/- NaN

7.7.5 Instrucciones "Escalar" y "Normalizar"

Tabla 7- 88 Instrucciones SCALE_X y NORM_X

KOP / FUP	SCL	Descripción
	<pre>out :=SCALE_X(min:=_in_, value:=_in_, max:=_in_);</pre>	<p>Escala el parámetro VALUE real normalizado (donde $0,0 \leq \text{VALUE} \leq 1,0$) al tipo de datos y rango de valores especificados por los parámetros MIN y MAX:</p> $\text{OUT} = \text{VALUE} (\text{MAX} - \text{MIN}) + \text{MIN}$
	<pre>out :=NORM_X(min:=_in_, value:=_in_, max:=_in_);</pre>	<p>Normaliza el parámetro VALUE dentro del rango de valores especificado por los parámetros MIN y MAX:</p> $\text{OUT} = (\text{VALUE} - \text{MIN}) / (\text{MAX} - \text{MIN}),$ <p>donde $(0,0 \leq \text{OUT} \leq 1,0)$</p>

¹ En KOP y FUP: haga clic en "???" y seleccione un tipo de datos en el menú desplegable.

Tabla 7- 89 Tipos de datos para los parámetros

Parámetro	Tipo de datos ¹	Descripción
MIN	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal	Entrada que indica el valor mínimo del rango
VALUE	SCALE_X: Real, LReal NORM_X: SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal	Valor de entrada que se debe escalar o normalizar
MAX	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal	Entrada que indica el valor máximo del rango
OUT	SCALE_X: SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal NORM_X: Real, LReal	Valor de salida escalado o normalizado

¹ Para SCALE_X: Los parámetros MIN, MAXy OUTdeben tener el mismo tipo de datos.
Para NORM_X: Los parámetros MIN, VALUEy MAXdeben tener el mismo tipo de datos.

Nota

SCALE_X parámetro **VALUE** debe restringirse a ($0.0 \leq VALUE \leq 1.0$)

Si el parámetro **VALUE** es menos que 0.0 o mayor que 1.0:

- La operación de escala lineal puede producir valores **OUT** menores que el parámetro **MIN** o mayores que el valor del parámetro **MAX** de valores **OUT** comprendidos dentro del rango de valores del tipo de datos **OUT**. La ejecución de **SCALE_X** pone **ENO = TRUE** para estos casos.
- Es posible generar números escalados no comprendidos en el rango del tipo de datos de **OUT**. En estos casos, el parámetro **OUT** se ajusta a un valor intermedio igual a la parte menos significativa del número real escalado antes de la conversión final al tipo de datos de **OUT**. En ese caso, la ejecución de **SCALE_X** pone **ENO = FALSE**.

NORM_X parámetro **VALUE** debe restringirse a ($MIN \leq VALUE \leq MAX$)

Si el parámetro **VALUE** es menor que **MIN** o mayor que **MAX**, la operación de escala lineal puede producir valores **OUT** normalizados menores que 0,0 o mayores que 1,0. La ejecución de **NORM_X** pone **ENO = TRUE** en este caso.

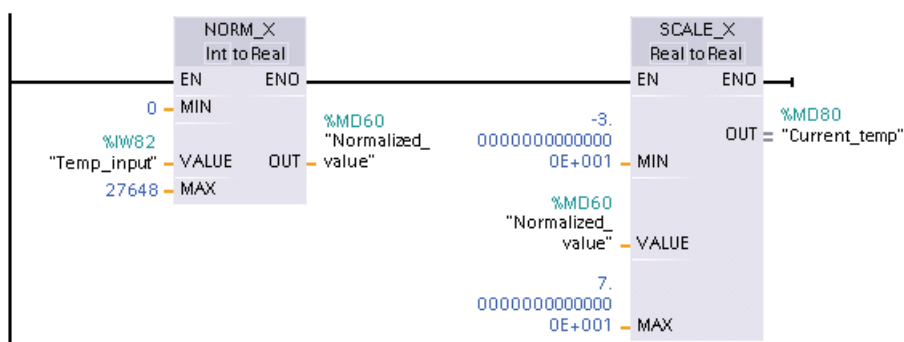
Tabla 7- 90 Estado de ENO

ENO	Condición	Resultado de OUT
1	No hay error	Resultado válido
0	El resultado excede el rango válido del tipo de datos de OUT	Resultado intermedio: La parte menos significativa de un número real antes de la conversión final al tipo de datos de OUT .
0	Parámetros $MAX \leq MIN$	SCALE_X : la parte menos significativa del número real VALUE que debe rellenar el tamaño de OUT . NORM_X : VALUE en el tipo de datos VALUE ampliado para rellenar una palabra doble.
0	Parámetro VALUE = +/- INF o +/- NaN	VALUE se escribe en OUT

Ejemplo (KOP): Normalización y escalado de un valor de entrada analógica

Una entrada analógica de un módulo de señales analógicas o Signal Board que usa entrada de intensidad se encuentra en el rango de valores válidos entre 0 y 27648. Suponiendo que una entrada analógica representa una temperatura en la que el valor 0 de la entrada analógica representa -30,0 grados C y 27648 representa 70,0 grados C.

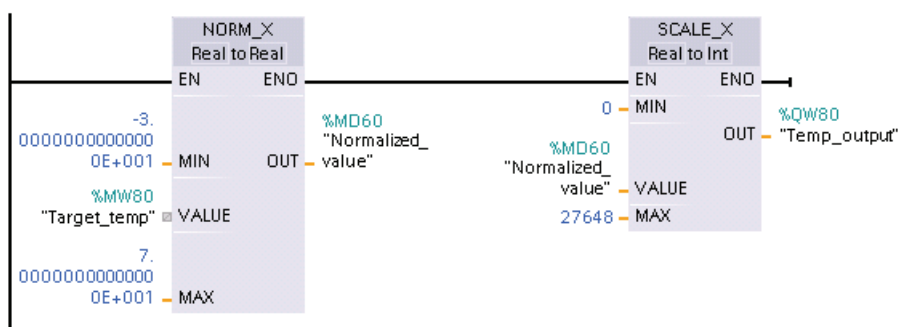
Para transformar el valor analógico en las correspondientes unidades de ingeniería, normalice la entrada a un valor entre 0,0 y 1,0 y a continuación escálelo entre -30,0 y 70,0. El valor resultante es la temperatura representada por la entrada analógica en grados Celsius:



Recuerde que si la entrada analógica provenía de un módulo de señales analógicas o una Signal Board con tensión, el valor MIN para la instrucción NORM_X sería -27648 en lugar de 0.

Ejemplo (KOP): Normalización y escalado de un valor de salida analógico

Una salida analógica de un módulo de señales analógicas o Signal Board que utiliza una salida de intensidad debe estar en el rango de valores válidos entre 0 y 27648. Supongamos que una salida analógica representa un ajuste de temperatura en el que el valor 0 de la entrada analógica representa -30,0 grados Celsius y 27648 representa 70,0 grados Celsius. Para convertir un valor de temperatura guardado que se encuentra entre -30,0 y 70,0 en un valor para la salida analógica dentro del rango entre 0 y 27648, es necesario normalizar el valor correspondiente a unidades de ingeniería en un valor entre 0,0 y 1,0, y a continuación escalarlo al rango de la salida analógica, de 0 a 27648:



Recuerde que si la salida analógica estaba destinada a un módulo de señales analógicas o una Signal Board con tensión, el valor MIN para la instrucción SCALE_X sería -27648 en lugar de 0.

En los Datos técnicos encontrará información adicional sobre representaciones de entradas analógica (Página 814) y representaciones de salidas analógicas (Página 815) en intensidad y tensión.

7.8 Control del programa

7.8.1 Vista general de las instrucciones de control del programa de SCL

Structured Control Language (SCL) proporciona tres tipos de instrucción de control de programa para estructurar el programa de usuario:

- **Instrucciones selectivas:** Una instrucción selectiva permite dirigir la ejecución del programa a secuencias de instrucciones alternativas.
- **Bucles:** La ejecución de bucles puede controlarse utilizando instrucciones de iteración. Una instrucción de iteración especifica qué partes de un programa deberían iterarse en función de determinadas condiciones.
- **Salto del programa:** Un salto del programa es un salto inmediato a una meta de salto específica y, con ello, a una instrucción diferente dentro del mismo bloque.

Estas instrucciones de control de programa utilizan la sintaxis del lenguaje de programación PASCAL.

Tabla 7- 91 Tipos de instrucciones de control en SCL

Instrucción de control del programa	Descripción	
Selectivo	Instrucción IF-THEN (Página 231)	Permite dirigir la ejecución del programa a una de las dos ramas alternativas, en función de que una condición sea TRUE o FALSE
	Instrucción CASE (Página 232)	Permite la ejecución selectiva en 1 de n ramas alternativas, en función del valor de una variable
Bucle	Instrucción FOR (Página 233)	Repite una secuencia de instrucciones mientras la variable de control permanezca dentro del rango de valores especificado
	Instrucción WHILE-DO (Página 234)	Repite una secuencia de instrucciones mientras se siga cumpliendo una condición de ejecución
	Instrucción REPEAT-UNTIL (Página 235)	Repite una secuencia de instrucciones hasta que se cumpla la condición de fin
Salto del programa	Instrucción CONTINUE (Página 236)	Detiene la ejecución de la iteración de bucle actual
	Instrucción EXIT (Página 236)	Abandona un bucle en cualquier momento independientemente de si se cumple o no la condición de fin
	Instrucción GOTO (Página 237)	Provoca un salto inmediato del programa a una etiqueta específica
	Instrucción IF-THEN (Página 231)	Hace que el programa abandone el bloque que se está ejecutando y regrese al bloque invocante

Consulte también

Instrucción RETURN (Página 238)

7.8.2 Instrucción IF-THEN

La instrucción IF-THEN es una instrucción condicional que controla el flujo del programa ejecutando una serie de instrucciones basándose en la evaluación de un valor Bool de una expresión lógica. También es posible utilizar paréntesis para anidar o estructurar la ejecución de instrucciones IF-THEN múltiples.

Tabla 7- 92 Elementos de la instrucción IF-THEN

SCL	Descripción
<pre>IF "condición" THEN instrucción_A; instrucción_B; instrucción_C; ;</pre>	<p>Si "condición" es TRUE o 1, entonces ejecuta las siguientes instrucciones hasta que aparezca la instrucción END_IF.</p> <p>Si "condición" es FALSE o 0, salta a la instrucción END_IF (a no ser que el programa incluya instrucciones ELSIF o ELSE adicionales).</p>
<pre>[ELSIF "condición-n" THEN instrucción_N; ;]</pre>	<p>La condición ELSEIF¹ opcional aporta condiciones adicionales que deben evaluarse. Ejemplo: Si "condición" de la instrucción IF-THEN es FALSE, entonces el programa evalúa "condición-n". Si "condición-n" es TRUE, ejecuta "instrucción_N".</p>
<pre>[ELSE instrucción_X; ;]</pre>	<p>La instrucción ELSE opcional aporta instrucciones que deben ejecutarse si la "condición" de la instrucción IF-THEN es FALSE.</p>
<pre>END_IF;</pre>	<p>La instrucción END_IF finaliza la instrucción IF-THEN.</p>

¹ Se pueden incluir varias instrucciones ELSIF dentro de la instrucción IF-THEN.

Tabla 7- 93 Variables de la instrucción IF-THEN

Variables	Descripción
"condición"	Requerida. La expresión lógica puede ser TRUE (1) o FALSE (0).
"instrucción_A"	Opcional. Una o más instrucciones deben ejecutarse si "condición" es TRUE.
"condición-n"	Opcional. La expresión lógica que debe ser evaluada por la instrucción opcional ELSIF.
"instrucción_N"	Opcional. Una o más instrucciones deben ejecutarse si "condición-n" de la instrucción ELSIF es TRUE.
"instrucción_X"	Opcional. Una o más instrucciones deben ejecutarse si "condición" de la instrucción IF-THEN es TRUE.

Las instrucciones IF se ejecutan de acuerdo con las siguientes normas:

- Se ejecuta la primera secuencia de las instrucciones cuya expresión lógica es TRUE. Las secuencias restantes de las instrucciones no se ejecutan.
- Si ninguna expresión booleana es TRUE, se ejecuta la secuencia de instrucciones correspondiente a ELSE (o ninguna secuencia si la rama ELSE no existe).
- Puede haber un número ilimitado de instrucciones ELSIF.

Nota

El uso de una o más ramas tiene la ventaja de que las expresiones lógicas que siguen a una expresión válida ya no se evalúan, al contrario que una secuencia de una instrucción IF. De ese modo es posible reducir el tiempo de ejecución de un programa.

7.8.3 Instrucción CASE

Tabla 7- 94 Elementos de la instrucción CASE

SCL	Descripción
<pre> CASE "Valor_test" OF "ListaValores": Instrucción[; Instrucción, ...] "ListaValores": Instrucción[; Instrucción, ...] [ELSE Instrucción Else[; Instrucción Else, ...]] END CASE;</pre>	<p>La instrucción CASE ejecuta uno de varios grupos de instrucciones en función del valor de una expresión.</p>

Tabla 7- 95 Parámetros

Parámetro	Descripción
"Valor_Test"	Requerida. Cualquier expresión numérica del tipo de datos Int
"ListaValores"	Requerida. Un valor único o una lista de valores o rangos de valores separados por coma. (Utilice dos periodos para definir un rango de valores: 2..8) El siguiente ejemplo ilustra las diferentes variantes de la lista de valores: 1: Instrucción_A; 2, 4: Instrucción _B; 3, 5..7,9: Instrucción _C;
Instrucción	Requerida. Se ejecutan una o más instrucciones cuando "Valor_test" coincide con cualquier valor de la lista de valores
Instrucción Else	Opcional. Una o más instrucciones que se ejecutan si no hay ninguna concordancia con un valor de "ListaValores"

La instrucción CASE se ejecuta de acuerdo con las siguientes normas:

- La expresión de selección debe devolver un valor del tipo Int.
- Si se procesa una instrucción CASE, el programa comprueba si el valor de la expresión de selección está contenido dentro de una lista de valores determinada. Si se encuentra una coincidencia, se ejecuta el componente de la instrucción asignado a la lista.
- Si no se encuentra ninguna coincidencia, se ejecuta la sección del programa posterior a ELSE o no se ejecuta ninguna instrucción si la rama ELSE no existe.

Las instrucciones CASE pueden anidarse. Cada instrucción CASE anidada debe tener asociada una instrucción END_CASE.

```

CASE var1 OF
    1 : var2 := "A";
    2 : var2 := "B";
ELSE
    CASE var3 OF
        65..90: var2 := "UpperCase";
        97..122: var2 := "LowerCase";
    ELSE
        var2:= "SpecialCharacter";
    END_CASE;
END_CASE;

```

7.8.4 Instrucción FOR

Tabla 7- 96 Elementos de la instrucción FOR

SCL	Descripción
<pre> FOR "variable_control" := "inicio" TO "fin" [BY "incremento"] DO instrucción; ; END_FOR; </pre>	<p>Una instrucción FOR se utiliza para repetir una secuencia de instrucciones mientras la variable de control se encuentre dentro del rango de valores especificado. La definición de un bucle con FOR incluye la especificación de un valor inicial y otro final. Ambos valores deben ser del mismo tipo de datos que la variable de control.</p>

Tabla 7- 97 Parámetros

Parámetro	Descripción
"variable_control"	Requerida. Un entero (Int o DInt) que sirve como contador de bucles
"inicio"	Requerida. Expresión simple que especifica el valor inicial de las variables de control

Parámetro	Descripción
"fin"	Requerida. Expresión simple que determina el valor final de las variables de control
"Incremento"	Opcional. Cantidad con la que una "variable de control" incrementa después de cada bucle. El "incremento" debe tener el mismo tipo de datos que la "variable de control". Si el valor de "incremento" no está especificado, el valor de las variables de ejecución se incrementará en 1 después de cada bucle. No es posible cambiar el "incremento" mientras se ejecuta la instrucción FOR.

La instrucción FOR se ejecuta de la siguiente manera:

- Al comienzo del bucle, la variable de control se pone al valor inicial (asignación inicial) y cada vez que el bucle itera se incrementa con el incremento especificado (incremento positivo) o se decrementa (incremento negativo) hasta que se alcanza el valor final.
- Después de la ejecución de cada bucle se comprueba la condición (alcanzado valor final) para saber si se cumple o no. Si se cumple, se ejecuta la secuencia de instrucciones, de lo contrario se salta el bucle y con él la secuencia de instrucciones.

Reglas para la formulación de instrucciones FOR:

- La variable de control sólo puede ser del tipo de datos Int o DInt.
- Se puede omitir la instrucción BY [incremento]. Si no se especifica ningún incremento, se asume automáticamente que es +1.

Para finalizar el bucle independientemente del estado de la expresión "condición" utilice la Instrucción EXIT (Página 236). La instrucción EXIT ejecuta la instrucción que sigue inmediatamente a la instrucción END_FOR.

Utilice la Instrucción CONTINUE (Página 236) para omitir las siguientes instrucciones de un bucle FOR y continuar con la comprobación de si se cumple o no la condición de fin.

7.8.5 Instrucción WHILE-DO

Tabla 7- 98 Instrucción WHILE

SCL	Descripción
<pre>WHILE "condición" DO Instrucción; Instrucción; ...; END WHILE;</pre>	<p>La instrucción WHILE realiza una serie de instrucciones hasta que una condición determinada es TRUE.</p> <p>Los bucles WHILE se pueden anidar. La instrucción END_WHILE se refiere a la última instrucción WHILE ejecutada.</p>

Tabla 7- 99 Parámetros

Parámetro	Descripción
"condición"	Requerida. Una expresión lógica que evalúa si el estado es TRUE o FALSE. (Una condición "null" se interpreta como FALSE.)
Instrucción	Opcional. Una o más instrucciones que se ejecutan hasta que la comprobación de la condición sea TRUE.

Nota

La instrucción WHILE evalúa el estado de "condición" antes de ejecutar cualquier instrucción. Para ejecutar las instrucciones como mínimo una vez independientemente del estado de "condición" utilice la instrucción REPEAT.

La instrucción WHILE se ejecuta de acuerdo con las siguientes normas:

- Antes de cada iteración del cuerpo del bucle se evalúa la condición de ejecución.
- El cuerpo del bucle que sigue a DO va iterando mientras la condición de ejecución tenga el valor TRUE.
- Cuando se da el valor FALSE, el bucle se salta y se ejecuta la instrucción que lo sigue.

Para finalizar el bucle independientemente del estado de la expresión "condición" utilice la Instrucción EXIT (Página 236). La instrucción EXIT ejecuta la instrucción que sigue inmediatamente a la instrucción END_WHILE

Utilice la instrucción CONTINUE para saltar las siguientes instrucciones de un bucle WHILE y continuar con la comprobación de si se cumple o no la condición de fin.

7.8.6 Instrucción REPEAT-UNTIL

Tabla 7- 100 Instrucción REPEAT

SCL	Descripción
REPEAT Instrucción; ; UNTIL "condition" END_REPEAT;	La instrucción REPEAT ejecuta una serie de instrucciones hasta que una condición determinada es TRUE. Los bucles REPEAT se pueden anidar. La instrucción END_REPEAT se refiere a la última instrucción REPEAT ejecutada.

Tabla 7- 101 Parámetros

Parámetro	Descripción
Instrucción	Opcional. Una o más instrucciones que se ejecutan hasta que la condición sea TRUE.
"condition"	Requerida. Una o más expresiones del siguiente modo: Una expresión numérica o de cadena que evalúa si el estado es TRUE o FALSE. Una condición "null" se interpreta como FALSE.

Nota

Antes de evaluar el estado de "condición", la instrucción REPEAT ejecuta las instrucciones durante la primera iteración del bucle (incluso si "condición" es FALSE). Para comprobar el estado de "condición" antes de ejecutar las instrucciones utilice la instrucción WHILE.

Para finalizar el bucle independientemente del estado de la expresión "condición" utilice la Instrucción EXIT (Página 236). La instrucción EXIT ejecuta la instrucción inmediatamente después de la instrucción END_REPEAT

Utilice la Instrucción CONTINUE (Página 236) para omitir las siguientes instrucciones de un bucle REPEAT y continuar con la comprobación de si se cumple o no la condición de fin.

7.8.7 Instrucción CONTINUE

Tabla 7- 102 Instrucción CONTINUE

SCL	Descripción
CONTINUE Instrucción; ;	La instrucción CONTINUE salta las instrucciones siguientes de un bucle de programa (FOR, WHILE, REPEAT) y continúa el bucle comprobando si se da la condición de fin. Si no es así, el bucle continúa.

La instrucción CONTINUE se ejecuta de acuerdo con las siguientes normas:

- Esta instrucción finaliza inmediatamente la ejecución de un cuerpo de bucle.
- Dependiendo de si la condición de repetición del bucle se cumple o no, el bucle vuelve a ejecutarse o se interrumpe la iteración y se ejecuta la instrucción inmediatamente posterior.
- En una instrucción FOR la variable de control se incrementa con el incremento especificado inmediatamente después de una instrucción CONTINUE.

Utilice la instrucción CONTINUE exclusivamente dentro de un bucle. En bucles anidados CONTINUE siempre se refiere al bucle que lo incluye de forma inmediata. Generalmente, CONTINUE se utiliza en combinación con una instrucción IF.

Si el bucle debe persistir independientemente de la comprobación de fin, utilice la instrucción EXIT.

El siguiente ejemplo muestra el uso de la instrucción CONTINUE para evitar un error de división entre 0 al calcular el porcentaje de un valor:

```

FOR x = 0 TO 10 DO
  IF valor [i] = 0 THEN CONTINUE; END_IF;
  p := parte / valor[i] * 100;
  s := INT_TO_STRING(p);
  porcentaje=CONCAT(IN1:=s, IN2:="%");
END_FOR;
    
```

7.8.8 Instrucción EXIT

Tabla 7- 103 Instrucción EXIT

SCL	Descripción
EXIT;	La instrucción EXIT se utiliza para finalizar un bucle (FOR, WHILE o REPEAT) en cualquier punto, independientemente de si se cumple o no la condición de fin.

La instrucción EXIT se ejecuta de acuerdo con las siguientes normas:

- Esta instrucción provoca la finalización inmediata de la instrucción de repetición situada directamente junto a la instrucción EXIT.
- La ejecución del programa continúa tras la interrupción del bucle (por ejemplo después de END_FOR).

Utilice la instrucción EXIT dentro de un bucle. En bucles anidados, la instrucción EXIT devuelve el procesamiento al siguiente nivel de anidamiento superior.

```
FOR i = 0 TO 10 DO
CASE valor[i, 0] OF
  1..10: valor [i, 1]:="A";
  11..40: valor [i, 1]:="B";
  41..100: valor [i, 1]:="C";
ELSE
EXIT;
END_CASE;
END_FOR;
```

7.8.9 Instrucción GOTO

Tabla 7- 104 Instrucción GOTO

SCL	Descripción
GOTO JumpLabel; Instrucción ; ... ; JumpLabel: Instrucción;	La instrucción GOTO omite instrucciones y salta a una etiqueta del mismo bloque. La etiqueta ("JumpLabel") y la instrucción GOTO deben estar en el mismo bloque. El nombre de una etiqueta sólo puede asignarse una vez dentro de un bloque. Cada etiqueta puede ser el destino de varias instrucciones GOTO.

No es posible saltar a una sección de bucle (FOR, WHILE o REPEAT). Sí que es posible saltar desde dentro de un bucle.

En el ejemplo siguiente: en función del valor del operando "Tag_value", la ejecución del programa se reanuda en el punto definido por la etiqueta correspondiente. Si "Tag_value" = 2, la ejecución del programa se reanuda en la etiqueta "MyLabel2" y omite "MyLabel1".

```
CASE "Tag_value" OF
  1 : GOTO MiEtiqueta1;
  2 : GOTO MiEtiqueta2;
ELSE GOTO MiEtiqueta3;
END_CASE;
MiEtiqueta1: "Tag_1" := 1;
MiEtiqueta2: "Tag_2" := 1;
MiEtiqueta3: "Tag_4" := 1;
```

7.8.10 Instrucción RETURN

Tabla 7- 105 Instrucción RETURN

SCL	Descripción
RETURN ;	La instrucción RETURN abandona sin condiciones el bloque lógico que se está ejecutando. La ejecución del programa regresa al bloque invocante o al sistema operativo (si abandona un OB).

Ejemplo de una instrucción RETURN:

```
IF "Error" <> 0 THEN
RETURN ;
END_IF ;
```

Nota

Una vez ejecutada la última instrucción, el bloque lógico regresa automáticamente al bloque invocante. No inserte una instrucción RETURN al final de un bloque lógico.

7.8.11 Instrucciones de salto y etiquetas

Tabla 7- 106 Instrucciones JMP, JMPN y LABEL

KOP	FUP	SCL	Descripción
Label_name —{JMP}—	Label_name JMP	Véase la instrucción GOTO (Página 237).	Si fluye corriente a una bobina JMP (KOP) o si se cumple el cuadro JMP (FUP), la ejecución del programa continúa con la primera instrucción que le sigue a la etiqueta indicada.
Label_name —{JMPN}—	Label_name JMPN		Si no fluye corriente a una bobina JMPN (KOP) o si el cuadro JMPN no se cumple (FUP), la ejecución del programa continúa con la primera instrucción que le sigue a la etiqueta indicada.
Label_name	Label_name		Etiqueta de destino de una instrucción de salto JMP o JMPN.

¹ Los nombres de las etiquetas se crean tecleando directamente en la instrucción LABEL. Utilice el icono de ayuda para seleccionar los nombres de etiqueta disponibles para el campo de nombre de etiqueta JMP y JMPN. El nombre de la etiqueta también puede teclearse directamente en la instrucción JMP o JMPN.

Tabla 7- 107 Tipos de datos para los parámetros

Parámetro	Tipo de datos	Descripción
Label_name	Identificador de la etiqueta	Identificador de las instrucciones de salto y etiqueta correspondiente al destino de salto

- Cada etiqueta debe ser única dentro de un bloque lógico.
- Se puede saltar dentro de un bloque lógico, pero no se puede saltar de un bloque lógico a otro.

- Se puede saltar hacia delante o hacia atrás.
- Se puede saltar a la misma etiqueta desde más de un punto del mismo bloque lógico.

7.8.12 Instrucción JMP_LIST

Tabla 7- 108 Instrucción JMP_LIST

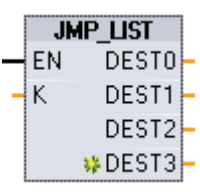
KOP / FUP	SCL	Descripción
	<pre> CASE k OF 0: GOTO dest0; 1: GOTO dest1; 2: GOTO dest2; [n: GOTO destn;] END_CASE; </pre>	<p>La instrucción JMP_LIST actúa como un distribuidor de saltos de programa para controlar la ejecución de secciones de un programa. Dependiendo del valor de la entrada K, se produce un salto a la correspondiente etiqueta del programa. La ejecución del programa continúa con las instrucciones que siguen a la etiqueta de destino. Si el valor de la entrada K excede el número de etiquetas - 1, no se produce ningún salto y el procesamiento continúa con el siguiente segmento del programa.</p>

Tabla 7- 109 Tipos de datos para parámetros

Parámetro	Tipo de datos	Descripción
K	UInt	Valor de control del distribuidor de saltos
DEST0, DEST1, ..., DESTn.	Etiquetas de programa	Etiquetas de destino correspondientes a valores específicos del parámetro K: Si el valor de K es 0, se produce un salto a la etiqueta del programa asignada a la salida DEST0. Si el valor de K es 1, se produce un salto a la etiqueta del programa asignada a la salida DEST1, etc. Si el valor de la entrada K excede el (número de etiquetas - 1), no se produce ningún salto y el procesamiento continúa con el siguiente segmento del programa.

En KOP y FUP: Primero se coloca el cuadro JMP_LIST en el programa; hay dos salidas de etiqueta. Los destinos de saltos se pueden agregar o borrar.



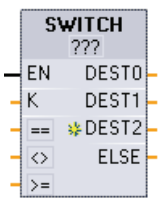
Haga clic en el icono "Crear" dentro del cuadro (en la parte izquierda del último parámetro DEST) para agregar nuevas salidas de etiquetas.



- Haga clic con el botón derecho sobre un conector de salida y seleccione el comando "Insertar salida".
- Haga clic con el botón derecho sobre un conector de salida y seleccione el comando "Borrar".

7.8.13 Instrucción SWITCH

Tabla 7- 110 Instrucción SWITCH

KOP / FUP	SCL	Descripción
	No disponible	La instrucción SWITCH actúa como un distribuidor de salto de programa para controlar la ejecución de secciones de un programa. Dependiendo del resultado de la comparación entre el valor de la entrada K y los valores asignados a las entradas de comparación específicas, se produce un salto a la etiqueta del programa que corresponde al primer test de comparación que sea TRUE. Si ninguna de las comparaciones es TRUE, se produce un salto asignado a la etiqueta asignada a ELSE. La ejecución del programa continúa con las instrucciones que siguen a la etiqueta de destino.

- 1 En KOP y FUP: Haga clic debajo del nombre del cuadro y seleccione un tipo de datos en el menú desplegable.
- 2 En SCL: utilice una serie de comparaciones IF-THEN.

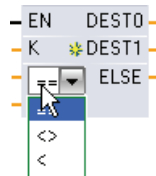
Tabla 7- 111 Tipos de datos para parámetros

Parámetro	Tipo de datos ¹	Descripción
K	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, Byte, Word, DWord, Time, TOD, Date	Entrada de valor de comparación común
==, <>, <, <=, >, >=	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, Byte, Word, DWord, Time, TOD, Date	Entradas de valor de comparación separadas para tipos de comparación específicos
DEST0, DEST1, ..., DESTn. ELSE	Etiquetas de programa	Etiquetas de destino correspondientes a comparaciones determinadas: Primero se procesa la entrada de comparación bajo la entrada K y siguiente a ella, lo que provoca un salto a la etiqueta asignada a DEST0 si la comparación entre el valor K y esta entrada es TRUE. El siguiente test de comparación utiliza la siguiente entrada situada debajo y provoca un salto a la etiqueta asignada a DEST1 si la comparación es TRUE; las comparaciones restantes se procesan de forma similar y si ninguna de ellas es TRUE, se produce un salto asignado a la etiqueta asignada a la salida ELSE.

¹ La entrada K y las entradas de comparación (==, <>, <, <=, >, >=) deben tener el mismo tipo de datos.

Agregar y borrar entradas y especificar tipos de comparación

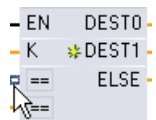
Si el cuadro SWITCH de KOP o FUP se coloca en el programa en primer lugar, hay dos entradas de comparación. Se pueden asignar tipos de comparación y agregar entradas/destinos de salto de la forma indicada a continuación.



Haga clic sobre un operador de comparación dentro del cuadro y seleccione un nuevo operador de la lista desplegable.



Haga clic en el icono "Crear" dentro del cuadro (a la izquierda del último parámetro DEST) para agregar nuevos destinos de comparación.



- Haga clic con el botón derecho sobre un conector de entrada y seleccione el comando "Insertar entrada".
- Haga clic con el botón derecho en un conector de entrada y seleccione el comando "Borrar".

Tabla 7- 112 Selección del tipo de datos del cuadro SWITCH y operaciones de comparación permitidas

Tipo de datos	Comparación	Sintaxis de operadores
Byte, Word, DWord	Igual a	==
	Diferente	<>
SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, Time, TOD, Date	Igual a	==
	Diferente	<>
	Mayor o igual	>=
	Menor o igual	<=
	Mayor	>
	Menor	<

Reglas de posicionamiento del cuadro SWITCH

- No se permite una conexión de instrucción KOP/FUP en frente de la entrada de comparación.
- No hay salida ENO por lo que se permite una instrucción SWITCH en un segmento y debe ser la última operación del segmento.

7.8.14 Instrucción de control de ejecución "Retroceder"

La instrucción RET opcional sirve para finalizar la ejecución del bloque actual. Sólo si fluye corriente a la bobina RET (LAD) o si se cumple la entrada del cuadro RET (FUP), la ejecución del programa del bloque actual finalizará en ese punto y las instrucciones posteriores a la instrucción RET no se ejecutarán. Si el bloque actual es un OB, se ignora el parámetro "Return_Value". Si el bloque actual es un FB o FC, el valor del parámetro "Return_Value" se devuelve a la rutina que efectúa la llamada como valor ENO del cuadro llamado.

No es necesario introducir manualmente la instrucción RET como última instrucción de un bloque. Esto se efectúa automáticamente. Un solo bloque puede comprender varias instrucciones RET.

Para SCL, véase la instrucción RETURN (Página 238).

Tabla 7- 113 Instrucción de control de ejecución "Retroceder" (RET)

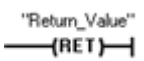

KOP	FUP	SCL	Descripción
		<pre>RETURN;</pre>	Finaliza la ejecución del bloque actual

Tabla 7- 114 Tipos de datos para los parámetros

Parámetro	Tipo de datos	Descripción
Return_Value	Bool	El parámetro "Return_value" de la instrucción RET se asigna a la salida ENO del cuadro de llamada de bloque en el bloque que efectúa la llamada.

Procedimiento para utilizar la instrucción RET en un bloque lógico FC (ejemplo):


1. Cree un proyecto nuevo y agregue una FC:
2. Edite la FC:
 - Agregue instrucciones del árbol de instrucciones.
 - Agregue una instrucción RET, incluyendo uno de los siguientes valores para el parámetro "Return_Value":
TRUE, FALSE o una posición de memoria que indique el valor de retorno requerido.
 - Agregue más instrucciones.
3. Llame la FC desde el bloque MAIN [OB1].

La entrada EN del cuadro FC del bloque lógico MAIN se debe cumplir para comenzar la ejecución de la FC.

El valor que indica la instrucción RET en la FC se encontrará en la salida ENO del cuadro FC del bloque lógico tras ejecutarse la FC para la que fluye corriente a la instrucción RET.

7.8.15 Instrucción "Reiniciar tiempo de vigilancia del ciclo"

Tabla 7- 115 Instrucción RE_TRIGR

KOP / FUP	SCL	Descripción
	RE_TRIGR () ;	La instrucción RE_TRIGR (Reiniciar la vigilancia del tiempo de ciclo) sirve para prolongar el tiempo máximo permitido antes de que el temporizador de vigilancia del ciclo genere un error.

La instrucción RE_TRIGR se utiliza para reiniciar el temporizador de vigilancia del ciclo durante un único ciclo. De esta manera, el tiempo de ciclo máximo se prolonga un periodo de tiempo de ciclo máximo desde la última ejecución de la función RE_TRIGR.

Nota

Antes de la versión 2.2 del firmware de la CPU S7-1200, RE_TRIGR estaba limitado a la ejecución desde un OB de ciclo de programa y podía utilizarse para prolongar de forma indefinida el tiempo de ciclo del PLC. ENO = FALSE y el temporizador de vigilancia no se inicializa si RE_TRIGR se ejecuta desde un OB de arranque, de alarma o de error.

Para la versión 2.2 y posteriores del firmware, RE_TRIGR puede ejecutarse desde cualquier OB (incluidos OB de arranque, de interrupción y de error). Sin embargo, el ciclo del PLC sólo puede prolongarse como máximo 10 veces el tiempo de ciclo máximo configurado.

Ajustar el tiempo de ciclo máximo del PLC

Configure el valor para el tiempo máximo de ciclo en la Configuración de dispositivos para "Tiempo de ciclo".

Tabla 7- 116 Valores de tiempo de ciclo

Vigilancia del tiempo de ciclo	Valor mínimo	Valor máximo	Valor predeterminado
Tiempo de ciclo máximo	1 ms	6000 ms	150 ms

Timeout de vigilancia

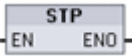
Si el tiempo de ciclo máximo finaliza antes de haberse completado el ciclo, se generará un error. Si se incluye un bloque lógico de tratamiento de errores OB 80 en el programa de usuario, la CPU lo ejecuta, y se puede agregar lógica del programa para crear una reacción especial. Si no se incluye el OB 80, se ignora la primera condición de timeout y la CPU pasa a estado operativo STOP.

Si ocurre un segundo timeout de tiempo de ciclo máximo en ese mismo ciclo del programa (valor del tiempo de ciclo máximo multiplicado por 2), se disparará un error y la CPU cambiará al estado operativo STOP.

En el estado operativo STOP se detiene la ejecución del programa mientras continúan las comunicaciones y los diagnósticos de sistema de la CPU.

7.8.16 Instrucción "Parar ciclo del PLC"

Tabla 7- 117 Instrucción STP

KOP / FUP	SCL	Descripción
	<code>STP () ;</code>	La instrucción STP (Finalizar programa) pone la CPU en el modo STOP. Cuando la CPU está en STOP, se detienen la ejecución del programa de usuario y las actualizaciones físicas desde la memoria imagen de proceso.

Encontrará más información en: Configurar las salidas en una transición de RUN a STOP (Página 88).

Si EN = TRUE, la CPU pasa a estado operativo STOP, se detiene la ejecución del programa y el estado de ENO carece de importancia. De lo contrario, EN = ENO = 0.

7.8.17 Instrucciones GET_ERROR

Las instrucciones GET_ERROR proporcionan información acerca de errores de ejecución de bloques de programa. Si se ha insertado una instrucción GetError o GetErrorID en el bloque lógico, los errores del programa podrán tratarse en el bloque de programa.

GetError

Tabla 7- 118 Instrucción GetError

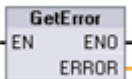
KOP / FUP	SCL	Descripción
	<code>GET_ERROR (_o ut_) ;</code>	Indica que ha ocurrido un error de ejecución de un bloque y rellena una estructura de datos de error predefinida con información detallada acerca del error.

Tabla 7- 119 Tipos de datos para los parámetros

Parámetro	Tipo de datos	Descripción
ERROR	ErrorStruct	Estructura de datos del error: Es posible cambiar el nombre de la estructura pero no sus elementos.

Tabla 7- 120 Elementos de la estructura de datos ErrorStruct

Componentes de la estructura	Tipo de datos	Descripción
ERROR_ID	Word	ID de error
FLAGS	Byte	Indica si se ha producido un error durante una llamada de bloque. <ul style="list-style-type: none"> • 16#01: Error durante una llamada de bloque. • 16#00: Ningún error durante una llamada de bloque.

Componentes de la estructura		Tipo de datos	Descripción					
REACTION		Byte	Respuesta predeterminada: <ul style="list-style-type: none"> • 0: Ignorar (error de escritura), • 1: Continuar con el valor de sustitución "0" (error de lectura), • 2: Omitir instrucción (error de sistema) 					
CODE_ADDRESS		CREF	Información sobre la dirección y el tipo de bloque					
	BLOCK_TYPE	Byte	Tipo de bloque en el que ha ocurrido el error: <ul style="list-style-type: none"> • 1: OB • 2: FC • 3: FB 					
	CB_NUMBER	UInt	Número del bloque lógico					
	OFFSET	UDInt	Referencia a la memoria interna					
MODE		Byte	Modo de acceso: Dependiendo del tipo de acceso, puede obtenerse la siguiente información:					
			Modo	(A)	(B)	(C)	(D)	(E)
			0					
			1					Offset
			2			Área		
			3	Ubicación	Alcance		Número	
			4			Área		Offset
			5			Área	N.º DB	Offset
			6	N.º Ptr. /Acc.		Área	N.º DB	Offset
7	N.º Ptr. /Acc.	N.º slot/Alcance	Área	N.º DB	Offset			
OPERAND_NUMBER		UInt	Número de operandos del comando de la máquina					
POINTER_NUMBER_LOCATION		UInt	(A) Puntero interno					
SLOT_NUMBER_SCOPE		UInt	(B) Área de almacenamiento dentro de la memoria interna					
DATA_ADDRESS		NREF	Información sobre la dirección de un operando					
	AREA	Byte	(C) Área de memoria: <ul style="list-style-type: none"> • L: 16#40 – 4E, 86, 87, 8E, 8F, C0 – CE • I: 16#81 • Q: 16#82 • M: 16#83 • DB: 16#84, 85, 8A, 8B 					
	DB_NUMBER	UInt	(D) Número de bloque de datos					
	OFFSET	UDInt	(E) Dirección relativa del operando					

GetErrorID

Tabla 7- 121 Instrucción GetErrorID


KOP / FUP	SCL	Descripción
	<pre>GET_ERR_ID () ;</pre>	Indica que ha ocurrido un error de ejecución de bloque de programa y notifica la ID (identificación) del error.

Tabla 7- 122 Tipos de datos para los parámetros

Parámetro	Tipo de datos	Descripción
ID	Word	Valores de identificador del error del elemento ERROR_ID de ErrorStruct

Tabla 7- 123 Valores de Error_ID

ERROR_ID hexadecimal	ERROR_ID decimal	Error de ejecución de bloque de programa
0	0	No hay error
2503	9475	Error de puntero no inicializado
2522	9506	Error de lectura de operando fuera de rango
2523	9507	Error de escritura de operando fuera de rango
2524	9508	Error de lectura de área no válida
2525	9509	Error de escritura de área no válida
2528	9512	Error de lectura de alineación de datos (alineación de bit incorrecta)
2529	9513	Error de escritura de alineación de datos (alineación de bit incorrecta)
2530	9520	DB protegido contra escritura
253A	9530	El DB global no existe
253C	9532	Versión incorrecta o la FC no existe
253D	9533	La instrucción no existe
253E	9534	Versión incorrecta o el FB no existe
253F	9535	La instrucción no existe
2575	9589	Error de profundidad de anidamiento del programa
2576	9590	Error de asignación de datos locales
2942	10562	La entrada física no existe
2943	10563	La salida física no existe

Funcionamiento

De forma predeterminada, la CPU reacciona a un error de ejecución de bloque registrando un error en el búfer de diagnóstico. No obstante, si se insertan una o más instrucciones GetError o GetErrorID en un bloque lógico, éste puede tratar los errores en el bloque. En este caso, la CPU no registra un error en el búfer de diagnóstico. En su lugar, la información de error se deposita en la salida de la instrucción GetError o GetErrorID. Es posible leer la información de error detallada con la instrucción GetError, o bien sólo el identificador del error con la instrucción GetErrorID. Normalmente, el primer error es el más importante; los errores siguientes son sólo consecuencias del primer error.

La primera ejecución de una instrucción GetError o GetErrorID en un bloque devuelve el primer error detectado durante la ejecución del bloque. Este error puede haberse producido en cualquier punto entre el inicio del bloque y la ejecución de GetError o GetErrorID. Las siguientes ejecuciones de GetError o GetErrorID devuelven el primer error desde la ejecución previa de GetError o GetErrorID. El historial de errores no se guarda y la ejecución de la instrucción en cuestión hará que el sistema PLC capture el siguiente error.

El tipo de datos ErrorStruct que utiliza la instrucción GetError puede agregarse en el editor del bloque de datos y editores de interfaz de bloque, de manera que la lógica del programa pueda acceder a estos valores. Seleccione ErrorStruct en la lista desplegable de tipos de datos para agregar esta estructura. Es posible crear varios ErrorStruct utilizando nombres unívocos. No es posible cambiar el nombre de los elementos de un ErrorStruct.

Condición de error indicada por ENO

Si EN = TRUE y se ejecuta GetError o GetErrorID, entonces:

- ENO = TRUE indica que ha ocurrido un error de ejecución del bloque lógico y que hay un error de datos
- ENO = FALSE indica que no ha ocurrido ningún error de ejecución del bloque lógico

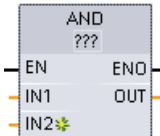
Es posible conectar la lógica del programa de reacción a errores a ENO que se activa cuando ocurre un error. Si existe un error, sus datos se almacenan en el parámetro de salida, donde el programa puede acceder a ellos.

GetError y GetErrorID pueden utilizarse para enviar información de error desde el bloque que se está ejecutando (bloque llamado) a un bloque invocante. Coloque la instrucción en el último segmento del bloque de programa llamado para notificar el estado de ejecución final del bloque llamado.

7.9 Operaciones lógicas con palabras

7.9.1 Instrucciones Y, O y O-exclusiva

Tabla 7- 124 Instrucciones Y, O y O-exclusiva

KOP / FUP	SCL	Descripción
	<code>out := in1 AND in2;</code>	Y: Y lógica
	<code>out := in1 OR in2;</code>	O: O lógica
	<code>out := in1 XOR in2;</code>	XOR: O-exclusiva lógica

¹ En KOP y FUP: haga clic en "???" y seleccione un tipo de datos en el menú desplegable.



Para agregar una entrada, haga clic en el icono "Crear" o haga clic con el botón derecho del ratón en el conector de entrada del parámetro IN existente y seleccione el comando "Insertar entrada".

Para quitar una entrada, haga clic con el botón derecho del ratón en el conector de entrada de uno de los parámetros IN existentes (si hay más entradas además de las dos originales) y seleccione el comando "Borrar".

Tabla 7- 125 Tipos de datos para los parámetros


Parámetro	Tipo de datos	Descripción
IN1, IN2	Byte, Word, DWord	Entradas lógicas
OUT	Byte, Word, DWord	Salida lógica

¹ La selección del tipo de datos ajusta los parámetros IN1, IN2 y OUT a un mismo tipo de datos.

Los valores de bit correspondientes de IN1 y IN2 se combinan para producir un resultado lógico binario en el parámetro OUT. ENO es siempre TRUE (verdadero) tras ejecutarse estas instrucciones.

7.9.2 Instrucción "Complemento a uno"

Tabla 7- 126 Instrucción INV

KOP / FUP	SCL	Descripción
	No disponible	Calcula el complemento binario a uno del parámetro IN. El complemento a uno se obtiene invirtiendo cada valor de bit del parámetro IN (cambiando cada 0 a 1 y cada 1 a 0). ENO es siempre TRUE (verdadero) tras ejecutarse esta instrucción.

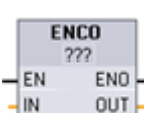

¹ En KOP y FUP: haga clic en "???" y seleccione un tipo de datos en el menú desplegable.

Tabla 7- 127 Tipos de datos para los parámetros

Parámetro	Tipo de datos	Descripción
IN	SInt, Int, DInt, USInt, UInt, UDInt, Byte, Word, DWord	Elemento que debe invertirse
OUT	SInt, Int, DInt, USInt, UInt, UDInt, Byte, Word, DWord	Salida invertida

7.9.3 Instrucciones "Codificar" y "Decodificar"

Tabla 7- 128 Instrucciones ENCO y DECO

KOP / FUP	SCL	Descripción
	<code>out := ENCO(in_);</code>	Codifica un patrón de bits en un número binario La instrucción ENCO convierte el parámetro IN al número binario correspondiente al bit activado menos significativo del parámetro IN y deposita el resultado en el parámetro OUT. Si el parámetro IN es 0000 0001 ó 0000 0000, el valor 0 se deposita en el parámetro OUT. Si el valor del parámetro IN es 0000 0000, ENO adopta el estado lógico FALSE.
	<code>out := DECO(in_);</code>	Descodifica un número binario a un patrón de bits La instrucción DECO descodifica un número binario del parámetro IN poniendo a "1" el bit correspondiente en el parámetro OUT (todos los demás bits se ponen a 0). ENO es siempre TRUE (verdadero) tras ejecutarse la instrucción DECO. Nota: El tipo de datos predeterminado para la instrucción DECO es DWORD. En SCL, cambie el nombre de instrucción a DECO_BYTE o DECO_WORD descodificar un valor de byte o de palabra y asígnelo a una dirección o variable de byte o palabra.

¹ En KOP y FUP: haga clic en "???" y seleccione un tipo de datos en el menú desplegable.

Tabla 7- 129 Tipos de datos para los parámetros

Parámetro	Tipo de datos	Descripción
IN	ENCO: Byte, Word, DWord DECO: UInt	ENCO: Patrón de bits que debe codificarse DECO: Valor que debe descodificarse
OUT	ENCO: Int DECO: Byte, Word, DWord	ENCO: Valor codificado DECO: Patrón de bits descodificado

Tabla 7- 130 Parámetro OUT para ENCO

ENO	Condición	Resultado (OUT)
1	No hay error	Número de bit válido
0	IN es cero	OUT se pone a cero

La selección del tipo de datos del parámetro OUT de DECO como Byte, Word o DWord delimita el rango útil del parámetro IN. Si el valor del parámetro IN supera el rango útil, se realiza una operación modulo para obtener los bits menos significativos, tal y como se indica a continuación.

Rango del parámetro IN de DECO:

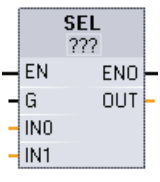
- 3 bits (valores 0-7) IN se utilizan para activar 1 posición de bit en Byte OUT
- 4 bits (valores 0-15) IN se utilizan para activar 1 posición de bit en Word OUT
- 5 bits (valores 0-31) IN se utilizan para activar 1 posición de bit en DWord OUT

Tabla 7- 131 Ejemplos

Valor IN de DECO		Valor OUT de DECO (descodificar un bit)	
Byte OUT 8 bits	Mín. IN	0	00000001
	Máx. IN	7	10000000
Word OUT 16 bits	Mín. IN	0	0000000000000001
	Máx. IN	15	1000000000000000
DWord OUT 32 bits	Mín. IN	0	00000000000000000000000000000001
	Máx. IN	31	10000000000000000000000000000000

7.9.4 Instrucciones Seleccionar, Multiplexar y Demultiplexar

Tabla 7- 132 Instrucción SEL (seleccionar)

KOP / FUP	SCL	Descripción
	<pre>out := SEL(g:=_bool_in, in0:=_variant_in, in1:=_variant_in);</pre>	La instrucción SEL (Seleccionar) asigna uno de dos valores de entrada al parámetro OUT, dependiendo del valor del parámetro G.

¹ En KOP y FUP: haga clic en "???" y seleccione un tipo de datos en el menú desplegable.

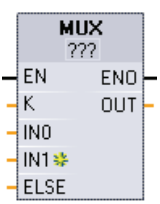
Tabla 7- 133 Tipos de datos para la instrucción SEL

Parámetro	Tipo de datos ¹	Descripción
G	Bool	<ul style="list-style-type: none"> • 0 selecciona IN0 • 1 selecciona IN1
IN0, IN1	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, Byte, Word, DWord, Time, Char	Entradas
OUT	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, Byte, Word, DWord, Time, Char	Salida

¹ Las variables de entrada y la de salida deben tener el mismo tipo de datos.

Códigos de condición: ENO siempre es TRUE tras la ejecución de la instrucción SEL.

Tabla 7- 134 Instrucción MUX (multiplexar)

KOP / FUP	SCL	Descripción
	<pre>out := MUX(k:=_unit_in, in1:=variant_in, in2:=variant_in, [...in32:=variant_in,] inelse:=variant_in);</pre>	La instrucción MUX copia uno de varios valores de entrada en el parámetro OUT, dependiendo del valor del parámetro K. Si el valor del parámetro K supera (INn - 1), el valor del parámetro ELSE se copia en el parámetro OUT.

¹ En KOP y FUP: haga clic en "???" y seleccione un tipo de datos en el menú desplegable.



Para agregar una entrada, haga clic en el icono "Crear" o haga clic con el botón derecho del ratón en el conector de entrada del parámetro IN existente y seleccione el comando "Insertar entrada".

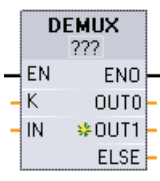
Para eliminar una entrada, haga clic con el botón derecho del ratón en el conector de entrada de uno de los parámetros IN existentes (si hay más entradas además de las dos originales) y seleccione el comando "Borrar".

Tabla 7- 135 Tipos de datos para la instrucción MUX

Parámetro	Tipo de datos	Descripción
K	UInt	<ul style="list-style-type: none"> • 0 selecciona IN1 • 1 selecciona IN2 • n selecciona INn
IN0, IN1, .. INn	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, Byte, Word, DWord, Time, Char	Entradas
ELSE	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, Byte, Word, DWord, Time, Char	Entrada del valor sustitutivo (opcional)
OUT	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, Byte, Word, DWord, Time, Char	Salida

¹ Las variables de entrada y la de salida deben tener el mismo tipo de datos.

Tabla 7- 136 Instrucción DEMUX (desmultiplexar)

KOP / FUP	SCL	Descripción
	<pre> DEMUX (k:=_unit_in, in:=variant_in, out1:=variant_in, out2:=variant_in, [...out32:=variant_in,] outelse:=variant_in); </pre>	DEMUX copia el valor del punto asignado al parámetro IN en una o varias salidas. El valor del parámetro K selecciona la salida asignada como destino del valor IN. Si el valor de K es mayor que el número (OUTn - 1) el valor IN se copia en el punto asignado al parámetro ELSE.

¹ En KOP y FUP: haga clic en "???" y seleccione un tipo de datos en el menú desplegable.

Para agregar una salida, haga clic en el icono "Crear" o haga clic con el botón derecho del ratón en un conector de salida del parámetro OUT existente y seleccione el comando "Insertar salida". Para eliminar una salida, haga clic con el botón derecho del ratón en el conector de salida de uno de los parámetros OUT existentes (si hay más salidas además de las dos originales) y seleccione el comando "Borrar".



Para agregar una salida, haga clic en el icono "Crear" o haga clic con el botón derecho del ratón en un conector de salida del parámetro OUT existente y seleccione el comando "Insertar salida".

Para eliminar una salida, haga clic con el botón derecho del ratón en el conector de salida de uno de los parámetros OUT existentes (si hay más salidas además de las dos originales) y seleccione el comando "Borrar".

Tabla 7- 137 Tipos de datos para la instrucción DEMUX

Parámetro	Tipo de datos ¹	Descripción
K	UInt	Valor selector: <ul style="list-style-type: none"> • 0 selecciona OUT1 • 1 selecciona OUT2 • n selecciona OUTn
IN	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, Byte, Word, DWord, Time, Char	Entrada
OUT0, OUT1, .. OUTn	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, Byte, Word, DWord, Time, Char	Salidas
ELSE	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, Byte, Word, DWord, Time, Char	Sustituir salida si K es mayor que (OUTn - 1)

¹ La variable de entrada y las de salida deben tener el mismo tipo de datos.

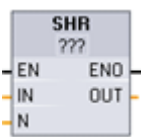
Tabla 7- 138 Estado de ENO para las instrucciones MUX y DEMUX

ENO	Condición	Resultado de OUT
1	No hay error	MUX: El valor de IN seleccionado se copia en OUT DEMUX: el valor de IN se copia en el parámetro OUT seleccionado
0	MUX: K es mayor que el número de entradas -1	<ul style="list-style-type: none"> • No hay ELSE: OUT permanece invariable, • Hay ELSE, valor de ELSE asignado a OUT
	DEMUX: K es mayor que el número de salidas -1	<ul style="list-style-type: none"> • No hay ELSE: Las salidas permanecen invariables, • Hay ELSE, el valor de IN se copia en ELSE

7.10 Instrucciones de desplazamiento y rotación

7.10.1 Instrucciones de desplazamiento

Tabla 7- 139 Instrucciones SHR y SHL

KOP / FUP	SCL	Descripción
	<pre> out := SHR(in:=_variant_in_, n:=_uint_in_); out := SHL(in:=_variant_in_, n:=_uint_in_); </pre>	Utilice las instrucciones de desplazamiento (SHL y SHR) para desplazar el patrón de bits del parámetro IN. El resultado se asigna al parámetro OUT. El parámetro N especifica el número de posiciones de bit desplazadas: <ul style="list-style-type: none"> • SHR: Desplazar patrón de bits hacia la derecha • SHL: Desplazar patrón de bits hacia la izquierda

¹ En KOP y FUP: haga clic en "???" y seleccione los tipos de datos del menú desplegable.

7.10 Instrucciones de desplazamiento y rotación

Tabla 7- 140 Tipos de datos para los parámetros

Parámetro	Tipo de datos	Descripción
IN	Byte, Word, DWord	Patrón de bits que debe desplazarse
N	UInt	Número de bits que deben desplazarse
OUT	Byte, Word, DWord	Patrón de bits después del desplazamiento

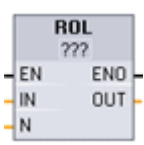
- Con N=0 no hay desplazamiento. El valor IN se asigna a OUT.
- Los ceros se desplazan a los bits que quedan vacíos tras el desplazamiento.
- Si el número de posiciones que deben desplazarse (N) excede el número de bits en el valor de destino (8 para Byte, 16 para Word, 32 para DWord), todos los valores de bit originales se desplazarán hacia fuera y se reemplazarán por ceros (cero se asigna a OUT).
- ENO es siempre TRUE (verdadero) en las operaciones de desplazamiento.

Tabla 7- 141 Ejemplo de SHL para datos del tipo Word (palabra):

Desplazar los bits de una palabra a la izquierda insertando ceros en la derecha (N = 1)			
IN	1110 0010 1010 1101	Valor de OUT antes del primer desplazamiento:	1110 0010 1010 1101
		Después del primer desplazamiento a la izquierda:	1100 0101 0101 1010
		Después del segundo desplazamiento a la izquierda:	1000 1010 1011 0100
		Después del tercer desplazamiento a la izquierda:	0001 0101 0110 1000

7.10.2 Instrucciones de rotación

Tabla 7- 142 Instrucciones ROR y ROL

KOP / FUP	SCL	Descripción
	<pre> out := ROL(in:=_variant_in_, n:=_uint_in_); out := ROR(in:=_variant_in_, n:=_uint_in_); </pre>	Utilice las instrucciones de rotación (ROR y ROL) para rotar el patrón de bits del parámetro IN. El resultado se deposita en el parámetro OUT. El parámetro N define el número de bits rotados. <ul style="list-style-type: none"> • ROR: Rotar patrón de bits hacia la derecha • ROL: Rotar patrón de bits hacia la izquierda

¹ En KOP y FUP: haga clic en "???" y seleccione los tipos de datos del menú desplegable.

Tabla 7- 143 Tipos de datos para los parámetros

Parámetro	Tipo de datos	Descripción
IN	Byte, Word, DWord	Patrón de bits que debe rotarse
N	UInt	Número de bits que deben rotarse
OUT	Byte, Word, DWord	Patrón de bits después de la rotación

- Con N=0 no hay rotación. El valor IN se asigna a OUT.
- Los bits rotados hacia fuera en un lado del valor de destino se rotan hacia el otro lado, por lo que no se pierden los valores de bit originales.
- Si el número de bits que deben rotarse (N) excede el número de bits en el valor de destino (8 para Byte, 16 para Word, 32 para DWord), la rotación se efectúa de todas maneras.
- ENO es siempre TRUE (verdadero) tras ejecutarse las instrucciones de rotación.

Tabla 7- 144 Ejemplo de ROR para datos del tipo Word (palabra):

Rotar bits desde la derecha a la izquierda (N = 1)			
IN	0100 0000 0000 0001	Valor de OUT antes de la primera rotación:	0100 0000 0000 0001
		Después de la primera rotación a la derecha:	1010 0000 0000 0000
		Después de la segunda rotación a la derecha:	0101 0000 0000 0000

Instrucciones avanzadas

8.1 Fecha y hora

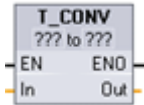
8.1.1 Instrucciones de fecha y hora

Utilice las instrucciones de fecha y hora para programar cálculos de calendario y hora.

- T_CONV convierte el tipo de datos de un valor de tiempo: (Time a DInt) o (DInt a Time)
- T_ADD suma los valores Time y DTL: (Time + Time = Time) o (DTL + Time = DTL)
- T_SUB resta los valores Time y DTL: (Time - Time = Time) o (DTL - Time = DTL)
- T_DIFF proporciona la diferencia entre dos valores DTL como valor Time: DTL - DTL = Time
- T_COMBINE combina un valor Date y un valor Time_and_Date para crear un valor DTL.

Para más información sobre la estructura de los datos DTL y Time, véase el apartado sobre los tipos de datos Fecha y hora (Página 98).

Tabla 8- 1 Instrucción T_CONV (Convertir y extraer tiempos)

KOP / FUP	SCL	Descripción
	<pre>out := T_CONV(in:=_variant_in);</pre>	T_CONV convierte un tipo de datos Time en un tipo de datos DInt, o a la inversa, el tipo de datos DInt en el tipo de datos Time.


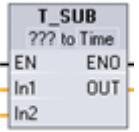
¹ En KOP y FUP: haga clic en "???" y seleccione los tipos de datos del menú desplegable.

Tabla 8- 2 Tipos de datos para los parámetros T_CONV

Parámetro y tipo	Tipo de datos	Descripción
IN	DInt, Time	Valor de entrada Time o valor DInt
OUT	DInt, Time	Valor convertido DInt o valor Time

8.1 Fecha y hora

Tabla 8-3 Instrucciones T_ADD (sumar tiempos) y T_SUB (restar tiempos)

KOP / FUP	SCL	Descripción
	<pre>out := T_ADD(in1:=_variant_in, in2:=_time_in);</pre>	<p>T_ADD suma el valor de la entrada IN1 (tipos de datos DTL o Time) con el valor de la entrada IN2 Time. El parámetro OUT proporciona el resultado DTL o Time. Son posibles dos operaciones con estos tipos de datos, a saber:</p> <ul style="list-style-type: none"> • Time + Time = Time • DTL + Time = DTL
	<pre>out := T_SUB(in1:=_variant_in, in2:=_time_in);</pre>	<p>T_SUB resta el valor IN2 Time del valor IN1 (DTL o Time). En el parámetro OUT se deposita el valor de diferencia como tipo de datos DTL o Time. Son posibles dos operaciones con estos tipos de datos, a saber:</p> <ul style="list-style-type: none"> • Time - Time = Time • DTL - Time = DTL

1 En KOP y FUP: haga clic en "???" y seleccione los tipos de datos del menú desplegable.

Tabla 8-4 Tipos de datos para los parámetros T_ADD y T_SUB

Parámetro y tipo	Tipo de datos	Descripción	
IN1 ¹	IN	DTL, Time	Valor DTL o Time
IN2	IN	Time	Valor Time que debe sumarse o restarse
OUT	OUT	DTL, Time	DTL o suma o diferencia Time

1 Seleccione el tipo de datos de IN1 en la lista desplegable debajo del nombre de la instrucción. La selección del tipo de datos de IN1 ajusta también el tipo de datos del parámetro OUT.

Tabla 8-5 Instrucción T_DIFF (Diferencia de tiempo)


KOP / FUP	SCL	Descripción
	<pre>out := T_DIFF(in1:=_DTL_in, in2:=_DTL_in);</pre>	<p>T_DIFF resta el valor DTL (IN2) del valor DTL (IN1). En el parámetro OUT se deposita el valor de diferencia como tipo de datos Time.</p> <ul style="list-style-type: none"> • DTL - DTL = Time

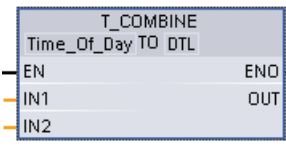
Tabla 8-6 Tipos de datos para los parámetros T_DIFF

Parámetro y tipo	Tipo de datos	Descripción	
IN1	IN	DTL	Valor DTL
IN2	IN	DTL	Valor DTL que se debe restar
OUT	OUT	Time	Diferencia Time

Códigos de condición: ENO = 1 significa que no se ha producido ningún error. ENO = 0 y parámetro OUT = 0 errores:

- El valor DTL no es válido
- El valor Time no es válido

Tabla 8- 7 Instrucción T_COMBINE (combinar valores de tiempo)

KOP / FUP	SCL	Descripción
	<pre>out := CONCAT_DATE_TOD(In1 := _date_in, In2 := _tod_in);</pre>	T_COMBINE combina un valor Date y un valor Time_of_Day para crear un valor DTL.

¹ Observe que la instrucción T_COMBINE de las Instrucciones avanzadas se corresponde con la función CONCAT_DATE_TOD en SCL.


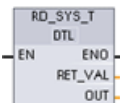

Tabla 8- 8 Tipos de datos para los parámetros T_COMBINE

Parámetro y tipo		Tipo de datos	Descripción
IN1	IN	Date	El valor Date que debe combinarse debe estar entre DATE#1990-01-01 y DATE#2089-12-31
IN2	IN	Time_of_Day	Valores Time_of_Day que deben combinarse
OUT	OUT	DTL	Valor DTL

8.1.2 Ajustar y leer el reloj del sistema

Las instrucciones de reloj se utilizan para ajustar y leer el reloj del sistema de la CPU. El tipo de datos DTL (Página 98) se utiliza para proporcionar valores de fecha y hora.

Tabla 8- 9 Instrucciones para la hora del sistema

KOP / FUP	SCL	Descripción
	<pre>ret_val := WR_SYS_T(in:=_DTL_in);</pre>	WR_SYS_T (leer la hora) ajusta la hora de la CPU del reloj diario con un valor DTL en el parámetro IN. Este valor de hora no incluye la diferencia con respecto a la hora local ni tampoco al horario de verano.
	<pre>ret_val := RD_SYS_T(out=>_DTL_out);</pre>	RD_SYS_T (leer hora del sistema) lee la hora actual del sistema de la CPU. Este valor de hora no incluye la diferencia con respecto a la hora local ni tampoco al horario de verano.
	<pre>ret_val := RD_LOC_T(out=>_DTL_out);</pre>	RD_LOC_T (leer hora local) proporciona la hora local actual de la CPU como tipo de datos DTL. Este valor de tiempo refleja la zona horaria local ajustada correctamente según el cambio de horario de verano/invierno (si está configurado).

8.1 Fecha y hora

Tabla 8- 10 Tipos de datos para los parámetros

Parámetro y tipo		Tipo de datos	Descripción
IN	IN	DTL	Hora que debe ajustarse en el reloj del sistema de la CPU
RET_VAL	OUT	Int	Código de condición de ejecución
OUT	OUT	DTL	RD_SYS_T: Hora actual del sistema de la CPU RD_LOC_T: Hora local actual incluido cualquier ajuste correspondiente al cambio de horario de verano/invierno, si está configurado

- La hora local se calcula aplicando la zona horaria y las variaciones correspondientes al horario de verano/invierno ajustadas en los parámetros de la ficha de configuración del dispositivo "Hora".
- La configuración de la zona horaria es un offset respecto a la hora UTC o GTM.
- La configuración del horario de verano especifica el mes, semana, día y hora de comienzo del horario de verano.
- La configuración del horario estándar especifica el mes, semana, día y hora de comienzo del horario estándar.
- La diferencia con respecto a la zona horaria se aplica siempre al valor de hora del sistema. La diferencia con respecto al horario de verano se aplica únicamente si el horario de verano está en vigor.

Nota

Horario de invierno/verano y configuración del tiempo de inicio estándar

La propiedad "Hora del día" para el "Inicio del horario de verano" de la configuración de dispositivo de la CPU debe ser la hora local.

Códigos de condición: ENO = 1 significa que no se ha producido ningún error. ENO = 0 significa que ha ocurrido un error de ejecución. El código de condición se indica en la salida RET_VAL.

Tabla 8- 11 Códigos de condición

RET_VAL (W#16#....)	Descripción
0000	La hora local actual está en horario estándar.
0001	Se ha configurado el horario de verano y la hora local actual está en horario de verano.
8080	Hora local no disponible
8081	Valor de año no válido
8082	Valor de mes no válido
8083	Valor de día no válido
8084	Valor de hora no válido
8085	Valor de minuto no válido
8086	Valor de segundo no válido
8087	Valor de nanosegundo no válido
80B0	Ha fallado el reloj de tiempo real.

8.1.3 Instrucción Contador de horas de funcionamiento

Tabla 8- 12 Instrucción RTM

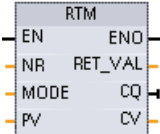
KOP / FUP	SCL	Descripción
	<pre>RTM(NR:=_uint_in_, MODE:=_byte_in_, PV:=_dint_in_, CQ=>_bool_out_, CV=>_dint_out_);</pre>	La instrucción RTM (Contador de horas de funcionamiento) puede inicializar, arrancar, parar y leer las horas de funcionamiento de la CPU.

Tabla 8- 13 Tipos de datos para los parámetros

Parámetro y tipo		Tipo de datos	Descripción
NR	IN	UInt	Número del contador de horas de funcionamiento: (Valores posibles: 0..9)
MODE	IN	Byte	RTM Número de modo de ejecución: <ul style="list-style-type: none"> • 0 = Recuperar datos (el estado se escribe entonces en CQ y el valor actual en CV) • 1 = Arrancar (en el último valor del contador) • 2 = Stop • 4 = Inicializar (al valor especificado en PV) • 5 = Inicializar (al valor especificado en PV) y después arrancar • 6 = Inicializar (al valor especificado en PV) y después parar • 7 = Guardar todos los valores RTM de la CPU en la MC (Memory Card)
PV	IN	DInt	Predeterminar valor de horas para el contador de horas de funcionamiento especificado
RET_VAL	OUT	Int	Resultado de la función / mensaje de error
CQ	OUT	Bool	Estado del contador de horas de servicio (1 = contando)
CV	OUT	DInt	Valor actual de horas de funcionamiento para el contador especificado

La CPU utiliza un máximo de 10 contadores de horas de funcionamiento para conocer las horas de funcionamiento de subsistemas de control críticos. Hay que arrancar cada uno de los contadores individuales con una ejecución RTM para cada temporizador. Todos los contadores de horas de funcionamiento se paran cuando la CPU hace una transición de RUN a STOP. También es posible parar los temporizadores individualmente con una ejecución de RTM, modo 2.

Cuando la CPU hace una transición de STOP a RUN, es necesario reiniciar los temporizadores de horas con una ejecución de RTM para cada temporizador que se arranca. Una vez que el valor de un contador de horas de funcionamiento supera las 2147483647 horas, el conteo se detiene y se transmite el error "Desbordamiento". Hay que ejecutar la instrucción RTM una vez por cada temporizador que se reinicia, o bien modificar el temporizador.

8.1 Fecha y hora

Un fallo de alimentación de la CPU o una desconexión y reconexión provoca un proceso de cierre (power down), que guarda los valores actuales de horas de funcionamiento en una memoria remanente. Cuando la CPU vuelve a arrancar, los valores de horas de funcionamiento guardados se vuelven a cargar en los temporizadores, y los totales anteriores de horas de funcionamiento no se pierden. Los contadores de horas de funcionamiento deben reiniciarse para acumular horas de funcionamiento adicionales.

El programa también puede utilizar la ejecución de RTM en modo 7 para guardar los valores de contador de horas de funcionamiento en una Memory Card. Los estados de todos los temporizadores en el momento en que RTM se ejecuta en modo 7 se guardan en la Memory Card. Los valores memorizados pueden ser incorrectos con el tiempo, ya que los contadores de funcionamiento se arrancan y detienen durante una sesión de funcionamiento del programa. Los valores de la Memory Card deben actualizarse periódicamente para capturar eventos importantes en runtime. La ventaja que se obtiene de guardar los valores RTM en la Memory Card radica en que se puede insertar la Memory Card en una CPU de sustitución, en la que estarán disponibles los valores del programa y los valores RTM memorizados. Si los valores de RTM no se guardaron en la Memory Card, los valores de temporizador se perderán (en una CPU de sustitución).

Nota

Debe evitarse un número excesivo de llamadas del programa para operaciones de escritura en Memory Card


Minimice las operaciones de escritura en Memory Card flash para ampliar la vida útil de las Memory Card.

Tabla 8- 14 Códigos de condición

RET_VAL (W#16#....)	Descripción
0	No hay error
8080	Número del contador de horas de funcionamiento incorrecto
8081	Un valor negativo ha pasado al parámetro PV
8082	Desbordamiento del contador de horas de funcionamiento
8091	El parámetro de entrada MODE contiene un valor no válido.
80B1	El valor no puede guardarse en la MC (MODE=7)

8.1.4 Instrucción SET_TIMEZONE

Tabla 8- 15 Instrucción SET_TIMEZONE

KOP / FUP	SCL	Descripción
	<pre>"SET_TIMEZONE_DB" (REQ:=_bool_in, Timezone:=_struct_in, DONE=>_bool_out_, BUSY=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_);</pre>	Ajusta la zona horaria local y los parámetros de horario de invierno/verano utilizados para convertir la hora del sistema de la CPU en hora local.

¹ En el ejemplo SCL, "SET_TIMEZONE_DB" es el nombre del DB instancia.

Tabla 8- 16 Tipos de datos para los parámetros

Parámetro y tipo	Tipo de datos	Descripción
REQ	IN	Bool
Timezone	IN	TimeTransformationRule
DONE	OUT	Bool
BUSY	OUT	Bool
ERROR	OUT	Bool
STATUS	OUT	Word

Para configurar manualmente los parámetros de zona horaria en la CPU, utilice las propiedades "Hora del día" de la ficha "General" de la configuración del dispositivo.

Utilice la instrucción SET_TIMEZONE para ajustar la configuración de la hora local con programación. Los parámetros de la estructura "TimeTransformationRule" especifican la zona horaria local y el momento de cambio automático entre el horario de invierno y verano.

Tabla 8- 17 "Estructura "TimeTransformationRule"

Parámetro	Tipo de datos	Descripción
Bias	Int	Diferencia horaria entre UTC y hora local [min]
DaylightBias	Int	Diferencia horaria entre horario de invierno y de verano [min]
DaylightStartMonth	USInt	Mes del cambio a horario de verano
DaylightStartWeek	USInt	Semana del cambio a horario de verano: <ul style="list-style-type: none"> • 1 = primera aparición del día de la semana en el mes • ... • 5 = última aparición del día de la semana en el mes

Parámetro	Tipo de datos	Descripción
DaylightStartWeekday	USInt	Día de la semana del cambio a horario de verano: <ul style="list-style-type: none"> • 1 = domingo • ... • 7 = sábado
DaylightStartHour	USInt	Hora del cambio a horario de verano
StandardStartMonth	USInt	Mes de cambio a horario de invierno
StandardStartWeek	USInt	Semana del cambio a horario de invierno: <ul style="list-style-type: none"> • 1 = primera aparición del día de la semana en el mes • ... • 5 = última aparición del día de la semana en el mes
StandardStartWeekday	USInt	Día de la semana del horario de invierno: <ul style="list-style-type: none"> • 1 = domingo • ... • 7 = sábado
StandardStartHour	USInt	Hora del horario de invierno
Time Zone Name	STRING [80]	Nombre de la zona: (GMT +01:00) Ámsterdam, Berlín, Berna, Roma, Estocolmo, Viena

8.2 Cadena y carácter

8.2.1 Sinopsis del tipo de datos String

Tipo de datos STRING

Los datos String se almacenan como encabezado de 2 bytes seguido de 254 bytes de caracteres en código ASCII. Un encabezado String contiene dos longitudes. El primer byte contiene la longitud máxima que se indica entre corchetes cuando se inicializa una cadena o 254 (ajuste predeterminado). El segundo byte del encabezado es la longitud actual, es decir, el número de caracteres válidos de la cadena. La longitud actual debe ser menor o igual a la longitud máxima. El número de bytes almacenados que ocupa el formato String es 2 bytes mayor que la longitud máxima.

Inicialización de los datos String

Los datos de entrada y salida String deben inicializarse como cadenas válidas en la memoria antes de ejecutar cualquier instrucción con cadenas.

Datos String válidos

Una cadena válida tiene una longitud máxima que debe ser mayor que cero pero menor que 255. La longitud actual debe ser menor o igual a la longitud máxima.

Las cadenas no pueden asignarse a áreas de memoria I ni Q.

Encontrará más información en: Formato del tipo de datos String (Página 100).

8.2.2 Instrucción S_MOVE

Tabla 8- 18 Instrucción Desplazar cadena de caracteres

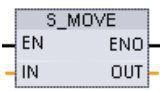
KOP / FUP	SCL	Descripción
	<code>out := in;</code>	Copiar la cadena IN fuente en la posición OUT. La instrucción S_MOVE no afecta a los contenidos de la cadena fuente.

Tabla 8- 19 Tipos de datos para los parámetros

Parámetro	Tipo de datos	Descripción
IN	String	Cadena fuente
OUT	String	Dirección de destino

Si la longitud real de la cadena en la entrada IN excede la longitud máxima de una cadena guardada en la salida OUT, se copia la parte de la cadena IN que cabe en la cadena OUT.

8.2.3 Instrucciones de conversión de cadenas

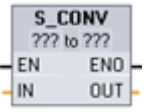
8.2.3.1 Conversión de cadenas en valores y de valores en cadenas

Las siguientes instrucciones permiten convertir cadenas de caracteres numéricos en valores numéricos, y viceversa:

- S_CONV convierte una cadena numérica en un valor numérico, o viceversa.
- STRG_VAL convierte una cadena numérica en un valor numérico con opciones de formato.
- VAL_STRG convierte un valor numérico en una cadena numérica con opciones de formato.

S_CONV (Convertir cadena en valor)

Tabla 8- 20 Instrucciones de conversión de cadenas

KOP / FUP	SCL	Descripción
	<pre>out := <Type>_TO_<Type>(in) ;</pre>	Convierte una cadena de caracteres en el valor correspondiente o viceversa. La instrucción S_CONV no tiene opciones de formato de salida. Gracias a ello, la instrucción S_CONV es más simple pero menos flexible que las instrucciones STRG_VAL y VAL_STRG.

- 1 En KOP y FUP: haga clic en "???" y seleccione el tipo de datos en la lista desplegable.
- 2 En SCL: seleccione S_CONV en las Instrucciones avanzadas y responda las preguntas acerca de los tipos de datos para la conversión. Seguidamente, STEP 7 proporciona la instrucción de conversión adecuada.

Tabla 8- 21 Tipos de datos (cadena en valor)

Parámetro y tipo		Tipo de datos	Descripción
IN	IN	String	Entrada que contiene la cadena de caracteres
OUT	OUT	String, Char, SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal	Valor numérico resultante

La conversión del parámetro IN que contiene la cadena comienza en el primer carácter y continúa hasta el final de la cadena, o bien hasta que se encuentra el primer carácter que no sea "0" a "9", "+", "-" o ".". El resultado se deposita en la ubicación que indica el parámetro OUT. Si el valor numérico resultante no está comprendido en el rango del tipo de datos de OUT, el parámetro OUT se pone a 0 y ENO adopta el estado lógico FALSE (falso). De lo contrario, el parámetro OUT contendrá un resultado válido y ENO adoptará el estado lógico TRUE (verdadero).

Reglas de formato de las cadenas de entrada:

- Si se utiliza un punto decimal en la cadena IN, es preciso utilizar el carácter ".".
- Las comas "," utilizadas como separadores de miles a la izquierda del punto decimal están permitidas, aunque se ignoran.
- Los espacios iniciales se ignoran.

S_CONV (Convertir cadena de caracteres)

Tabla 8- 22 Tipos de datos (valor en cadena)

Parámetro y tipo		Tipo de datos	Descripción
IN	IN	String, Char, SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal	Entrada que contiene el valor numérico
OUT	OUT	String	Cadena de caracteres resultante

Un valor entero, entero sin signo o en coma flotante de la entrada IN se convierte en una cadena de caracteres correspondiente que se deposita en OUT. El parámetro OUT debe referenciar una cadena válida antes de ejecutarse la conversión. Una cadena válida consta de una longitud de cadena máxima en el primer byte, la longitud de cadena actual en el segundo byte y los caracteres de la cadena actuales en los siguientes bytes. La cadena convertida sustituye los caracteres de la cadena OUT comenzando en el primer carácter y ajusta el byte de longitud actual de la cadena OUT. El byte de longitud máxima de la cadena OUT no se modifica.

El número de caracteres que se sustituyen depende del tipo de datos del parámetro IN y del valor numérico. El número de caracteres sustituidos no debe exceder la longitud de la cadena depositada en el parámetro OUT. La longitud máxima (primer byte) de la cadena OUT debe ser mayor o igual al número máximo esperado de caracteres convertidos. La tabla siguiente muestra las longitudes de cadena máximas posibles requeridas para los distintos tipos de datos soportados.

Tabla 8- 23 Longitudes de cadena máximas para cada tipo de datos


Tipo de datos IN	Número máximo de caracteres convertidos en la cadena OUT	Ejemplo	Longitud total de la cadena incluyendo los bytes de longitud máxima y actual
USInt	3	255	5
SInt	4	-128	6
UInt	5	65535	7
Int	6	-32768	8
UDInt	10	4294967295	12
DInt	11	-2147483648	13

Reglas de formato de las cadenas de salida:

- Los valores que se escriben en el parámetro OUT aparecen sin signo "+" inicial.
- Se utiliza la representación en coma fija (no la notación exponencial).
- El carácter de punto "." se utiliza para representar el punto decimal si el parámetro IN tiene el tipo de datos Real.

Instrucción STRG_VAL

Tabla 8- 24 Instrucción de conversión de cadena en valor

KOP / FUP	SCL	Descripción
	<pre>"STRG_VAL" (in:=_string_in, format:=_word_in, p:=uint_in, out=>_variant_out);</pre>	<p>Convierte una cadena de caracteres numéricos en un número entero o en coma flotante correspondiente.</p>

¹ En KOP y FUP: haga clic en "???" y seleccione el tipo de datos en la lista desplegable.

Tabla 8- 25 Tipos de datos para la instrucción STRG_VAL

Parámetro y tipo		Tipo de datos	Descripción
IN	IN	String	Cadena de caracteres ASCII que debe convertirse
FORMAT	IN	Word	Opciones de formato de salida
P	IN	UInt, Byte, USInt	IN: Índice al primer carácter que debe convertirse (primer carácter = 1)
OUT	OUT	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal	Valor numérico convertido

La conversión comienza en el offset de carácter P de la cadena IN y continúa hasta su final, o bien hasta que se encuentra el primer carácter que no sea "+", "-", ".", ";", "e", "E" o "0" a "9". El resultado se deposita en la posición que indica el parámetro OUT.

Los datos String deben inicializarse antes de la ejecución como cadena válida en la memoria.

El parámetro FORMAT de la instrucción STRG_VAL se define a continuación. Los bits no utilizados deben ponerse a cero.

Tabla 8- 26 Formato de la instrucción STRG_VAL

Bit 16								Bit 8	Bit 7							Bit 0	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	f	r

f = Formato de notación 1= Notación exponencial
 0 = Notación en coma fija

r = Formato de punto decimal 1 = "," (coma)
 0 = "." (punto)

Tabla 8- 27 Valores del parámetro FORMAT

FORMAT (W#16#)	Formato de notación	Representación del punto decimal
0000 (estándar)	Coma fija	"."
0001		" , "
0002	Exponencial	"."
0003		" , "
0004 a FFFF	Valores no válidos	

Reglas para la conversión de STRG_VAL:

- Si el carácter de punto "." se utiliza como punto decimal, las comas "," a la izquierda del punto decimal se interpretan como separadores de miles. Las comas están permitidas, aunque se ignoran.
- Si el carácter de coma "," se utiliza como punto decimal, los puntos "." a la izquierda del punto decimal se interpretan como separadores de miles. Estos puntos están permitidos, aunque se ignoran.
- Los espacios iniciales se ignoran.

Instrucción VAL_STRG

Tabla 8- 28 Operación de conversión de valor en cadena

KOP / FUP	SCL	Descripción
	<pre>"VAL_STRG" (in:=_variant_in, size:=_usint_in, prec:=_usint_in, format:=_word_in, p:=uint_in, out=>_string_out);</pre>	<p>Convierte un valor entero, entero sin signo o en coma flotante en la cadena de caracteres correspondiente.</p>

¹ En KOP y FUP: haga clic en "???" y seleccione el tipo de datos en la lista desplegable.

Tabla 8- 29 Tipos de datos para la instrucción VAL_STRG

Parámetro y tipo		Tipo de datos	Descripción
IN	IN	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal	Valor que debe convertirse
SIZE	IN	USInt	Número de caracteres que deben escribirse en la cadena OUT
PREC	IN	USInt	Precisión o tamaño de la parte fraccionaria. Esto no incluye el punto decimal.
FORMAT	IN	Word	Opciones de formato de salida
P	IN	UInt, Byte, USInt	IN: Índice al primer carácter de la cadena OUT que debe convertirse (primer carácter = 1)
OUT	OUT	String	Cadena convertida

El valor que indica el parámetro IN se convierte en una cadena referenciada por el parámetro OUT. El parámetro OUT debe ser una cadena válida antes de ejecutar la conversión.

8.2 Cadena y carácter

La cadena convertida sustituye los caracteres de la cadena OUT, comenzando en el conteo de offset de carácter P hasta el número de caracteres que indica el parámetro SIZE. El número de caracteres de SIZE debe caber en la longitud de la cadena OUT, contando desde la posición del carácter P. Esta instrucción resulta útil para integrar caracteres numéricos en una cadena de texto. Por ejemplo, la cifra "120" puede incorporarse a la cadena "Presión bomba = 120 psi".

El parámetro PREC indica la precisión o el número de dígitos de la parte fraccionaria de la cadena. Si el parámetro IN es un número entero, PREC indica la posición del punto decimal. Por ejemplo, si el valor es 123 y PREC = 1, el resultado es "12,3". La precisión máxima soportada para el tipo de datos Real es 7 dígitos.

Si el parámetro P es mayor que el tamaño actual de la cadena OUT, se agregan espacios hasta la posición P y el resultado se añade al final de la cadena. La conversión finaliza cuando se alcanza la longitud máxima de la cadena OUT.

El parámetro FORMAT de la instrucción VAL_STRG se define a continuación. Los bits no utilizados deben ponerse a cero.

Tabla 8- 30 Formato de la instrucción VAL_STRG

Bit 16								Bit 8	Bit 7								Bit 0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	s	f	r	

s = Carácter de signo	1= usar los signos "+" y "-"
	0 = usar sólo el signo "-"
f = Formato de notación	1= Notación exponencial
	0 = Notación en coma fija
r = Formato de punto decimal	1 = "," (coma)
	0 = "." (punto)

Tabla 8- 31 Valores del parámetro FORMAT

FORMAT (WORD)	Carácter de signo	Formato de notación	Representación del punto decimal
W#16#0000	Sólo "-"	Coma fija	"."
W#16#0001			" "
W#16#0002		Exponencial	"."
W#16#0003			" "
W#16#0004	"+" y "-"	Coma fija	"."
W#16#0005			" "
W#16#0006		Exponencial	"."
W#16#0007			" "
W#16#0008 a W#16#FFFF	Valores no válidos		

Reglas de formato de la cadena del parámetro OUT:

- Si el tamaño de la cadena convertida es menor que el especificado, se insertan espacios en blanco iniciales en el extremo izquierdo de la cadena.
- Si el bit de signo del parámetro FORMAT es FALSE, los valores de los tipos de datos USINT y SINT se escriben en el búfer de salida sin el signo "+" inicial. El "-" signo se utiliza en caso necesario.
<espacios iniciales><dígitos sin ceros iniciales>'.'<dígitos PREC>
- Si el bit de signo es TRUE, los valores de los tipos de datos de entero con o sin signo se escriben en el búfer de salida con un carácter de signo inicial.
<espacios iniciales><signo><dígitos sin ceros iniciales>'.'<dígitos PREC>
- Si el parámetro FORMAT está ajustado a notación exponencial, los valores del tipo de datos Real se escriben en el búfer de salida de la siguiente manera:
<espacios iniciales><signo><dígito> '.' <dígitos PREC>'E' <signo><dígitos sin cero inicial>
- Si el parámetro FORMAT está ajustado a notación en coma fija, los valores del tipo de datos entero, entero sin signo y real se escriben en el búfer de salida de la siguiente manera:
<espacios iniciales><signo><dígitos sin ceros iniciales>'.'<dígitos PREC>
- Los ceros a la izquierda del punto decimal (con excepción del dígito adyacente a éste) se suprimen.
- Los valores a la derecha del punto decimal se redondean para que se correspondan con el número de dígitos a la derecha del punto decimal que indica el parámetro PREC.
- La cadena de salida debe ser como mínimo tres bytes más grande que el número de dígitos a la derecha del punto decimal.
- Los valores se justifican a la derecha en la cadena de salida.

Condiciones que notifica ENO

Si ocurre un error durante la operación de conversión, se devolverán los siguientes resultados:

- ENO se pone a 0.
- OUT se pone a 0, o como se indica en los ejemplos de conversión de una cadena en un valor.
- OUT no se modifica, o como se indica en los ejemplos cuando OUT es una cadena.

Tabla 8- 32 Estado de ENO

ENO	Descripción
1	No hay error
0	Parámetro no permitido o no válido; p. ej. acceso a un DB que no existe
0	Cadena no permitida; la longitud máxima es 0 ó 255
0	Cadena no permitida; la longitud actual excede la longitud máxima
0	El valor numérico convertido es demasiado grande para el tipo de datos de OUT indicado.

ENO	Descripción
0	El tamaño máximo de la cadena del parámetro OUT debe ser lo suficientemente grande para aceptar el número de caracteres que indica el parámetro SIZE, comenzando en el parámetro P de posición de carácter.
0	Valor de P no permitido; P=0 o P es mayor que la longitud actual de la cadena
0	El parámetro SIZE debe ser mayor que el parámetro PREC.

Tabla 8- 33 Ejemplos de S_CONV (convertir cadena en valor)

Cadena IN	Tipo de datos OUT	Valor OUT	ENO
"123"	Int o DInt	123	TRUE
"-00456"	Int o DInt	-456	TRUE
"123.45"	Int o DInt	123	TRUE
"+2345"	Int o DInt	2345	TRUE
"00123AB"	Int o DInt	123	TRUE
"123"	Real	123.0	TRUE
"123.45"	Real	123.45	TRUE
"1.23e-4"	Real	1.23	TRUE
"1.23E-4"	Real	1.23	TRUE
"12,345.67"	Real	12345.67	TRUE
"3.4e39"	Real	3.4	TRUE
"-3.4e39"	Real	-3.4	TRUE
"1.17549e-38"	Real	1.17549	TRUE
"12345"	SInt	0	FALSE
"A123"	N/A	0	FALSE
""	N/A	0	FALSE
"++123"	N/A	0	FALSE
"+-123"	N/A	0	FALSE

Tabla 8- 34 Ejemplos de S_CONV (convertir valor en cadena)

Tipo de datos	Valor IN	Cadena OUT	ENO
UInt	123	"123"	TRUE
UInt	0	"0"	TRUE
UDInt	12345678	"12345678"	TRUE
Real	-INF	"INF"	FALSE
Real	+INF	"INF"	FALSE
Real	NaN	"NaN"	FALSE

Tabla 8- 35 Ejemplos de conversión STRG_VAL

Cadena IN	FORMAT (W#16#....)	Tipo de datos OUT	Valor OUT	ENO
"123"	0000	Int o DInt	123	TRUE
"-00456"	0000	Int o DInt	-456	TRUE
"123.45"	0000	Int o DInt	123	TRUE
"+2345"	0000	Int o DInt	2345	TRUE
"00123AB"	0000	Int o DInt	123	TRUE
"123"	0000	Real	123.0	TRUE
"-00456"	0001	Real	-456.0	TRUE
"+00456"	0001	Real	456.0	TRUE
"123.45"	0000	Real	123.45	TRUE
"123.45"	0001	Real	12345.0	TRUE
"123,45"	0000	Real	12345.0	TRUE
"123,45"	0001	Real	123.45	TRUE
".00123AB"	0001	Real	123.0	TRUE
"1.23e-4"	0000	Real	1.23	TRUE
"1.23E-4"	0000	Real	1.23	TRUE
"1.23E-4"	0002	Real	1.23E-4	TRUE
"12,345.67"	0000	Real	12345.67	TRUE
"12,345.67"	0001	Real	12.345	TRUE
"3.4e39"	0002	Real	+INF	TRUE
"-3.4e39"	0002	Real	-INF	TRUE
"1.1754943e-38" (o menor)	0002	Real	0.0	TRUE
"12345"	N/A	SInt	0	FALSE
"A123"	N/A	N/A	0	FALSE
""	N/A	N/A	0	FALSE
"++123"	N/A	N/A	0	FALSE
"+-123"	N/A	N/A	0	FALSE

Los siguientes ejemplos de conversión VAL_STRG se basan en una cadena OUT que se inicializa de la manera siguiente:

"Current Temp = xxxxxxxxxx C"

, donde el carácter "x" representa caracteres de espacio asignados al valor convertido.

Tabla 8- 36 Ejemplos de conversión VAL_STRG

Tipo de datos	Valor IN	P	SIZE	FORMAT (W#16#....)	PREC	Cadena OUT	ENO
UInt	123	16	10	0000	0	Current Temp = xxxxxxxx123 C	TRUE
UInt	0	16	10	0000	2	Current Temp = xxxxxx0.00 C	TRUE
UDInt	12345678	16	10	0000	3	Current Temp = x12345.678 C	TRUE
UDInt	12345678	16	10	0001	3	Current Temp = x12345,678 C	TRUE

Tipo de datos	Valor IN	P	SIZE	FORMAT (W#16#...)	PREC	Cadena OUT	ENO
Int	123	16	10	0004	0	Current Temp = xxxxxx+123 C	TRUE
Int	-123	16	10	0004	0	Current Temp = xxxxxx-123 C	TRUE
Real	-0.00123	16	10	0004	4	Current Temp = xxx-0.0012 C	TRUE
Real	-0.00123	16	10	0006	4	Current Temp = -1.2300E-3 C	TRUE
Real	-INF	16	10	N/A	4	Current Temp = xxxxxx-INF C	FALSE
Real	+INF	16	10	N/A	4	Current Temp = xxxxxx+INF C	FALSE
Real	NaN	16	10	N/A	4	Current Temp = xxxxxxNaN C	FALSE
UDInt	12345678	16	6	N/A	3	Current Temp = xxxxxxxxxxxx C	FALSE

8.2.3.2 Conversiones de cadena a carácter y de carácter a cadena

Chars_TO_Strg copia una matriz de bytes de caracteres ASCII en una cadena de caracteres.

Strg_TO_Chars copia una cadena de caracteres ASCII en una matriz de bytes de caracteres.

Nota

Sólo los tipos de matriz basados en cero (Array [0..n] of Char) o (Array [0..n] of Byte) están permitidos como parámetro de entrada Chars para la instrucción Chars_TO_Strg o como parámetro IN_OUT Chars para la instrucción Strg_TO_Chars .

Tabla 8- 37 Instrucción Chars_TO_Strg


KOP / FUP	SCL	Descripción
	<pre>Chars_TO_Strg(Chars:=_variant_in_, pChars:=_dint_in_, Cnt:=_uint_in_, Strg=>_string_out_);</pre>	<p>Toda una matriz de caracteres o una parte de ella se copia en una cadena.</p> <p>La cadena de salida debe estar declarada antes de ejecutar Chars_TO_Strg. La cadena se sobrescribe con la operación Chars_TO_Strg.</p> <p>Pueden utilizarse cadenas de todas las longitudes máximas soportadas (1 ... 254).</p> <p>El valor de longitud máxima de una cadena no cambia con la operación Chars_TO_Strg . El proceso de copia de una matriz a una cadena se detiene cuando se ha alcanzado la longitud máxima de la cadena.</p> <p>Un carácter nul '\$00' o 16#00 en la matriz de caracteres funciona como delimitador y finaliza la copia de caracteres en la cadena.</p>

Tabla 8- 38 Tipos de datos para los parámetros (Chars_TO_Strg)

Parámetro y tipo		Tipo de datos	Descripción
Chars	IN	Variante	El parámetro Chars es un puntero hacia una matriz basada en cero [0..n] de caracteres que deben convertirse en una cadena. La matriz puede declararse en un DB o como variables locales en la interfaz del bloque. Ejemplo: "DB1".MyArray apunta a valores de elementos MyArray [0..10] of Char en DB1.
pChars	IN	Dint	Número de elemento del primer carácter en la matriz que debe copiarse. El elemento de matriz [0] es el valor predeterminado.
Cnt	IN	UInt	Recuento de caracteres que se copian: 0 significa todos
Strg	OUT	String	Cadena de destino

Tabla 8- 39 Instrucción Strg_TO_Chars

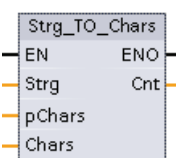
KOP / FUP	SCL	Descripción
	<pre>Strg_TO_Chars (Strg:=_string_in_, pChars:=_dint_in_, Cnt=>_uint_out_, Chars:=_variant_inout_);</pre>	<p>La cadena de entrada completa Strg se copia en una matriz de caracteres en el parámetro IN_OUT Chars.</p> <p>La operación sobrescribe los bytes que empiezan por el número de elemento especificado por el parámetro pChars.</p> <p>Pueden utilizarse cadenas de todas las longitudes máximas soportadas (1 ... 254).</p> <p>No se escribe un delimitador de fin: eso es responsabilidad del usuario. Para establecer un delimitador de fin justo después del último carácter escrito de la matriz, utilice el siguiente número de elemento de la matriz [pChars+Cnt].</p>

Tabla 8- 40 Tipos de datos para los parámetros (Strg_TO_Chars)

Parámetro y tipo		Tipo de datos	Descripción
Strg	IN	String	Cadena fuente
pChars	IN	Dint	Número del elemento para el primer carácter de la cadena escrito en la matriz de destino
Chars	IN_OUT	Variante	El parámetro Chars es un puntero hacia una matriz basada en cero [0..n] de caracteres copiados de una cadena de entrada. La matriz puede declararse en un DB o como variables locales en la interfaz del bloque. Ejemplo: "DB1".MyArray apunta a valores de elementos MyArray [0..10] of Char en DB1.
Cnt	OUT	UInt	Recuento de caracteres copiados

8.2 Cadena y carácter

Tabla 8- 41 Estado de ENO

ENO	Descripción
1	No hay error
0	Chars_TO_Strg: intento de copiar más bytes de caracteres en la cadena de salida de los que permite el byte de longitud máxima en la declaración de cadena
0	Chars_TO_Strg: el valor de carácter cero (16#00) se ha encontrado en la matriz de byte de caracteres de entrada.
0	Strg_TO_Chars: intento de copiar más bytes de caracteres en la matriz de salida de los que permite el límite de número de elementos

8.2.3.3 Conversiones de ASCII a Hex y de Hex a ASCII

Utilice las instrucciones ATH (ASCII a hexadecimal) y HTA (hexadecimal a ASCII) para realizar conversiones entre bytes de caracteres ASCII (0 a 9 y mayúsculas sólo de A a F) y los correspondientes nibbles hexadecimales de 4 bits.

Tabla 8- 42 Instrucción ATH

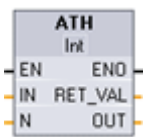
KOP / FUP	SCL	Descripción
	<pre>ret_val := ATH(in:=_variant_in_, n:=_int_in_, out=>_variant_out_);</pre>	Convierte caracteres ASCII en dígitos hexadecimales comprimidos.

Tabla 8- 43 Tipos de datos para la instrucción ATH

Tipo de parámetro	Tipo de datos	Descripción
IN	IN	Variante
N	IN	UInt
RET_VAL	OUT	Word
OUT	OUT	Variante

La conversión comienza en la posición especificada por el parámetro IN y continúa durante N bytes. El resultado se deposita en la posición que indica OUT. Sólo pueden convertirse caracteres ASCII válidos 0 a 9 y mayúsculas de A a F. Cualquier otro carácter se convertirá a cero.

Los caracteres ASCII codificados de 8 bits se convierten en nibbles hexadecimales de 4 bits. Dos caracteres ASCII pueden almacenarse en un único byte.

Los parámetros IN y OUT especifican matrices de bytes y no datos String hexadecimales. Los caracteres ASCII se convierten y depositan en la salida hexadecimal en el mismo orden que se leen. Si existe un número impar de caracteres ASCII, se agregan ceros en el nibble situado más a la derecha del último dígito hexadecimal convertido.

Tabla 8- 44 Ejemplos de conversión ASCII a hexadecimal (ATH)

Bytes de caracteres IN	N	Valor OUT	ENO
'0123'	4	W#16#0123	TRUE
'123AFx1a23'	10	16#123AF01023	FALSE
'a23'	3	W#16#A230	TRUE

Tabla 8- 45 Instrucción HTA


KOP / FUP	SCL	Descripción
	<pre>ret_val := HTA(in:=_variant_in_, n:=_uint_in_, out=>_variant_out_);</pre>	Convierte dígitos hexadecimales comprimidos en los correspondientes bytes de caracteres ASCII.

Tabla 8- 46 Tipos de datos para la instrucción HTA

Parámetro y tipo		Tipo de datos	Descripción
IN	IN	Variante	Puntero a matriz de byte de entrada
N	IN	UInt	Número de bytes que deben convertirse (cada byte de entrada tiene dos nibbles de 4 bits y produce caracteres ASCII 2N)
RET_VAL	OUT	Word	Código de condición de ejecución
OUT	OUT	Variante	Puntero a matriz de byte de caracteres ASCII

La conversión comienza en la posición especificada por el parámetro IN y continúa durante N bytes. Cada nibble de 4 bits convierte a un carácter ASCII único de 8 bits y produce bytes de caracteres 2N ASCII de salida. Todos los bytes 2N de la salida se escriben como caracteres ASCII de 0 a 9 o mayúsculas A a F. El parámetro OUT especifica una matriz de bytes y no una cadena.

Cada nibble del byte hexadecimal se convierte en un carácter en el mismo orden en que se lee (el nibble situado más a la izquierda de un dígito hexadecimal se convierte primero, seguido por el nibble situado más a la derecha del mismo byte).

Tabla 8- 47 Ejemplos de conversión hexadecimal a ASCII (HTA)

Valor IN	N	Bytes de caracteres OUT	ENO (ENO siempre TRUE tras ejecución de HTA)
W#16#0123	2	'0123'	TRUE
DW#16#123AF012	4	'123AF012'	TRUE

Tabla 8- 48 Códigos de condición ATH and HTA

RET_VAL (W#16#....)	Descripción	ENO
0000	No hay error	TRUE
0007	Carácter de entrada ATH no válido: Se ha encontrado un carácter que no era un carácter ASCII 0-9, minúscula a-f o mayúscula A-F	FALSE
8101	Puntero de entrada no permitido o no válido; p. ej. acceso a un DB que no existe.	FALSE
8120	La cadena de entrada tiene un formato no válido, es decir, máx= 0, máx=255, actual>máx o longitud en puntero < máx	FALSE
8182	El búfer de entrada es demasiado pequeño para N	FALSE
8151	Tipo de datos no permitido para búfer de entrada	FALSE
8301	Puntero de salida no permitido o no válido; p. ej. acceso a un DB que no existe.	FALSE
8320	La cadena de salida tiene un formato no válido, es decir, máx= 0, máx=255, actual>máx o longitud en puntero < máx	FALSE
8382	El búfer de salida es demasiado pequeño para N	FALSE
8351	Tipo de datos no permitido para búfer de salida	FALSE

8.2.4 Instrucciones con cadenas

El programa de control puede utilizar las siguientes instrucciones con cadenas y caracteres para crear avisos para displays de operador e históricos del proceso.

8.2.4.1 LEN

Tabla 8- 49 Instrucciones de longitud


KOP / FUP	SCL	Descripción
	<pre>out := LEN(in);</pre>	LEN (longitud) proporciona la longitud actual de la cadena IN en la salida OUT. Una cadena vacía tiene una longitud cero.

Tabla 8- 50 Tipos de datos para los parámetros

Parámetro y tipo	Tipo de datos	Descripción
IN	IN	String
OUT	OUT	Int, DInt, Real, LReal

Tabla 8- 51 Estado de ENO

ENO	Condición	OUT
1	No hay condición de cadena no válida	Longitud de cadena válida
0	La longitud actual de IN excede la longitud máxima de IN	La longitud actual se pone a 0
	La longitud máxima de IN no cabe centro del área de memoria asignado	
	La longitud máxima de IN es 255 (longitud no permitida)	

8.2.4.2 CONCAT

Tabla 8- 52 Instrucción Agrupar cadenas

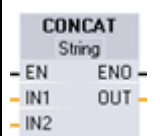
KOP / FUP	SCL	Descripción
	<pre>out := CONCAT(in1, in2);</pre>	CONCAT (agrupar cadenas) agrupa los parámetros de las cadenas IN1 y IN2 para proporcionar una cadena que se deposita en OUT. Tras la agrupación, la cadena IN1 es la parte izquierda y IN2 es la parte derecha de la cadena combinada.

Tabla 8- 53 Tipos de datos para los parámetros

Parámetro y tipo	Tipo de datos	Descripción
IN1	IN	String
IN2	IN	String
OUT	OUT	String
		Cadena combinada (cadena 1 + cadena 2)

Tabla 8- 54 Estado de ENO

ENO	Condición	OUT
1	No se han detectado errores	Caracteres válidos
0	La cadena resultante tras la concatenación excede la longitud máxima de la cadena OUT	Los caracteres de la cadena resultante se copian hasta alcanzarse la longitud máxima de OUT
	La longitud actual de IN1 excede la longitud máxima de IN1, la longitud actual de IN2 excede la longitud máxima de IN2, o la longitud actual de OUT excede la longitud máxima de OUT (cadena no válida)	La longitud actual se pone a 0
	La longitud máxima de IN1, IN2 o OUT no cabe dentro del área de memoria asignado	
	La longitud máxima de IN1 o IN2 es 255, o la longitud máxima de OUT es 0 o 255	

8.2.4.3 LEFT, RIGHT y MID

Tabla 8- 55 Operaciones de subcadenas izquierda, derecha y central

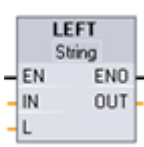
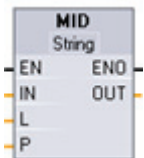
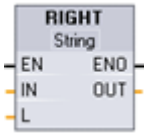
KOP / FUP	SCL	Descripción
	<pre>out := LEFT(in, L);</pre>	<p>LEFT (Leer los caracteres izquierdos de una cadena) crea una subcadena formada por los primeros caracteres L del parámetro de cadena IN.</p> <ul style="list-style-type: none"> • Si L es mayor que la longitud actual de la cadena IN, OUT devuelve la cadena IN completa. • Si la entrada contiene una cadena vacía, OUT devuelve una cadena vacía.
	<pre>out := MID(in, L, p);</pre>	<p>MID (Leer los caracteres centrales de una cadena) provee la parte central de una cadena. La subcadena central tiene una longitud de L caracteres y comienza en la posición de carácter P (inclusive).</p> <p>Si la suma de L y P excede la longitud actual del parámetro de la cadena IN, se devuelve una subcadena que comienza en la posición de carácter P y que continúa hasta el final de la cadena IN.</p>
	<pre>out := RIGHT(in, L);</pre>	<p>RIGHT (Leer los caracteres derechos de una cadena) provee los últimos caracteres L de una cadena.</p> <ul style="list-style-type: none"> • Si L es mayor que la longitud actual de la cadena IN, el parámetro OUT devuelve la cadena IN completa. • Si la entrada contiene una cadena vacía, OUT devuelve una cadena vacía.

Tabla 8- 56 Tipos de datos para los parámetros

Parámetro y tipo		Tipo de datos	Descripción
IN	IN	String	Cadena de entrada
L	IN	Int	Longitud de la subcadena que debe crearse: <ul style="list-style-type: none"> • LEFT utiliza el número de caracteres de la cadena situados más a la izquierda • RIGHT utiliza el número de caracteres de la cadena situados más a la derecha • MID utiliza el número de caracteres comenzando en la posición P de la cadena
P	IN	Int	Sólo MID: Posición del primer carácter de subcadena que debe copiarse P= 1, para la posición de carácter inicial de la cadena IN
OUT	OUT	String	Cadena de salida

Tabla 8- 57 Estado de ENO

ENO	Condición	OUT
1	No se han detectado errores	Caracteres válidos
0	<ul style="list-style-type: none"> L o P es menor o igual que 0 P es mayor que la longitud máxima de IN La longitud actual de IN excede la longitud máxima de IN o la longitud actual de OUT excede la longitud máxima de OUT La longitud máxima de IN o OUT no cabe dentro del área de memoria asignada La longitud máxima de IN o OUT es 0 ó 255 	La longitud actual se pone a 0
	La longitud (L) de la subcadena que debe copiarse excede la longitud máxima de la cadena OUT.	Los caracteres se copian hasta alcanzarse la longitud máxima de OUT
	Sólo MID: L o P es menor o igual que 0	La longitud actual se pone a 0
	Sólo MID: P es mayor que la longitud máxima de IN	
	La longitud actual de IN1 excede la longitud máxima de IN1 o la longitud actual de IN2 excede la longitud máxima de IN2 (cadena no válida)	La longitud actual se pone a 0
	La longitud máxima de IN1, IN2 o OUT no cabe dentro del área de memoria asignado	
La longitud máxima de IN1, IN2 o OUT es 0 ó 255 (longitud no permitida)		

8.2.4.4 DELETE

Tabla 8- 58 Instrucción Borrar caracteres de una cadena


KOP / FUP	SCL	Descripción
	<pre>out := DELETE(in, L, p);</pre>	<p>Borra L caracteres de la cadena IN. El borrado de caracteres comienza en la posición P (inclusive) y la subcadena resultante se deposita en el parámetro OUT.</p> <ul style="list-style-type: none"> Si L es igual a cero, la cadena de entrada se deposita en OUT. Si la suma de L y P es mayor que la longitud de la cadena de entrada, la cadena se borra hasta el final.

Tabla 8- 59 Tipos de datos para los parámetros

Parámetro y tipo		Tipo de datos	Descripción
IN	IN	String	Cadena de entrada
L	IN	Int	Número de caracteres que deben borrarse
P	IN	Int	Posición del primer carácter que debe borrarse: El primer carácter de la cadena IN tiene el número de posición 1
OUT	OUT	String	Cadena de salida

8.2 Cadena y carácter

Tabla 8- 60 Estado de ENO

ENO	Condición	OUT
1	No se han detectado errores	Caracteres válidos
0	P es mayor que la longitud actual de IN	IN se copia en OUT sin borrar caracteres
	La cadena resultante tras borrar los caracteres excede la longitud máxima de la cadena OUT	Los caracteres de la cadena resultante se copian hasta alcanzarse la longitud máxima de OUT
	L es menor que 0, o P es menor o igual que 0	La longitud actual se pone a 0
	La longitud actual de IN excede la longitud máxima de IN o la longitud actual de OUT excede la longitud máxima de OUT	
	La longitud máxima de IN o OUT no cabe dentro del área de memoria asignada	
	La longitud máxima de IN o OUT es 0 ó 255	

8.2.4.5 INSERT

Tabla 8- 61 Instrucción Insertar caracteres en una cadena

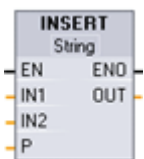
KOP / FUP	SCL	Descripción
	<pre>out := INSERT(in1, in2, p);</pre>	Inserta la cadena IN2 en la cadena IN1. La inserción comienza tras el carácter de la posición P.

Tabla 8- 62 Tipos de datos para los parámetros

Parámetro y tipo	Tipo de datos	Descripción	
IN1	IN	String	Cadena de entrada 1
IN2	IN	String	Cadena de entrada 2
P	IN	Int	Última posición de carácter en la cadena IN1 antes del punto de inserción de la cadena IN2. El primer carácter de la cadena IN1 tiene el número de posición 1.
OUT	OUT	String	Cadena resultante

Tabla 8- 63 Estado de ENO

ENO	Condición	OUT
1	No se han detectado errores	Caracteres válidos
0	P excede la longitud de IN1	IN2 se agrupa con IN1 inmediatamente después del último carácter de IN1
	P es menor que 0	La longitud actual se pone a 0
	La cadena resultante tras la inserción excede la longitud máxima de la cadena OUT	Los caracteres de la cadena resultante se copian hasta alcanzarse la longitud máxima de OUT
	La longitud actual de IN1 excede la longitud máxima de IN1, la longitud actual de IN2 excede la longitud máxima de IN2, o la longitud actual de OUT excede la longitud máxima de OUT (cadena no válida)	La longitud actual se pone a 0
	La longitud máxima de IN1, IN2 o OUT no cabe dentro del área de memoria asignado	
La longitud máxima de IN1 o IN2 es 255, o la longitud máxima de OUT es 0 ó 255		

8.2.4.6 REPLACE

Tabla 8- 64 Instrucción Reemplazar caracteres en una cadena

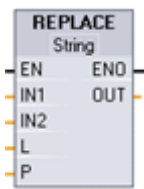
KOP / FUP	SCL	Descripción
	<pre> out := REPLACE(in1:=_string_in_, in2:=_string_in_, L:=_int_in_, p:=_int_in); </pre>	Reemplaza L caracteres en el parámetro de cadena IN1. La sustitución comienza en la posición de carácter P (inclusive) de la cadena IN1, y los caracteres de reemplazo provienen de la cadena IN2.

Tabla 8- 65 Tipos de datos para los parámetros

Parámetro y tipo	Tipo de datos	Descripción	
IN1	IN	String	Cadena de entrada
IN2	IN	String	Cadena de caracteres de reemplazo
L	IN	Int	Número de caracteres que deben reemplazarse
P	IN	Int	Posición del primer carácter que debe reemplazarse
OUT	OUT	String	Cadena resultante

Si el parámetro L es igual a cero, la cadena IN2 se inserta en la posición P de la cadena IN1 sin que se borre ningún carácter de la cadena IN1.

Si P es igual a uno, los primeros caracteres L de la cadena IN1 se reemplazan por caracteres de la cadena IN2.

Tabla 8- 66 Estado de ENO

ENO	Condición	OUT
1	No se han detectado errores	Caracteres válidos
0	P excede la longitud de IN1	IN2 se agrupa con IN1 inmediatamente después del último carácter de IN1
	P cabe en IN1, pero menos de L caracteres permanecen en IN1	IN2 reemplaza los caracteres finales de IN1 comenzando por la posición P
	La cadena resultante tras la sustitución excede la longitud máxima de la cadena OUT	Los caracteres de la cadena resultante se copian hasta alcanzarse la longitud máxima de OUT
	La longitud máxima de IN1 es 0	IN2 caracteres se copian en OUT
	L es menor que 0, o P es menor o igual que 0	La longitud actual se pone a 0
	La longitud actual de IN1 excede la longitud máxima de IN1, la longitud actual de IN2 excede la longitud máxima de IN2, o la longitud actual de OUT excede la longitud máxima de OUT	
	La longitud máxima de IN1, IN2 o OUT no cabe dentro del área de memoria asignado	
	La longitud máxima de IN1 o IN2 es 255, o la longitud máxima de OUT es 0 ó 255	

8.2.4.7 FIND

Tabla 8- 67 Instrucción Buscar caracteres en una cadena

KOP / FUP	SCL	Descripción
	<pre> out := FIND(in1:=_string_in_, in2:=_string_in_); </pre>	Proporciona la posición de carácter de la subcadena especificada en IN2 dentro de la cadena IN1. La búsqueda comienza en el lado izquierdo. La posición de carácter del primer resultado encontrado en la cadena IN2 se devuelve en OUT. Si la cadena IN2 no se encuentra en la cadena IN1, se devuelve cero.

Tabla 8- 68 Tipos de datos para los parámetros

Parámetro y tipo	Tipo de datos	Descripción	
IN1	IN	String	Buscar en esta cadena
IN2	IN	String	Buscar esta cadena
OUT	OUT	Int	Posición de carácter de la cadena IN1 en el primer resultado de búsqueda

Tabla 8- 69 Estado de ENO

ENO	Condición	OUT
1	No se han detectado errores	Posición de carácter válida
0	IN2 es mayor que IN1	La posición de carácter se pone a 0
	La longitud actual de IN1 excede la longitud máxima de IN1 o la longitud actual de IN2 excede la longitud máxima de IN2 (cadena no válida)	
	La longitud máxima de IN1 o IN2 no cabe dentro del área de memoria asignada	
	La longitud máxima de IN1 o IN2 es 255	

8.3 E/S distribuidas (PROFINET, PROFIBUS o AS-i)

8.3.1 Instrucciones de E/S descentralizadas

Las siguientes instrucciones E/S descentralizadas pueden utilizarse con PROFINET, PROFIBUS o AS-i:

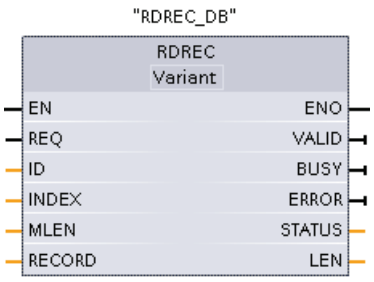
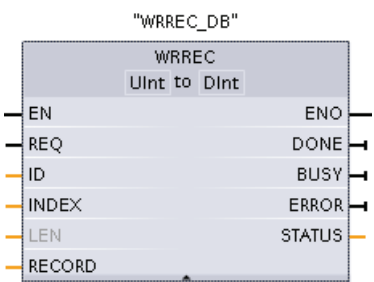
- Instrucción RDREC (Página 286): Se puede leer un registro de datos con el número INDEX desde un módulo o dispositivo.
- Instrucción WRREC (Página 286): Se puede transferir un registro de datos con el número INDEX a un módulo o dispositivo definido por ID.
- Instrucción RALRM (Página 289): Se puede recibir una alarma con toda la información correspondiente de un módulo o dispositivo y suministrar esta información a sus parámetros de salida.
- Instrucción DPRD_DAT (Página 296): Debe leer áreas de datos coherentes mayores de 64 bytes desde un módulo o dispositivo con la instrucción DPRD_DAT.
- Instrucción DPWR_DAT (Página 296): Debe escribir áreas de datos coherentes mayores de 64 bytes desde un módulo o dispositivo con la instrucción DPWR_DAT.

La instrucción DPNRM_DG (Página 298) sólo puede utilizarse con PROFIBUS. Es posible leer los datos de diagnóstico actuales de un esclavo DP en el formato especificado por EN 50 170 volumen 2, PROFIBUS.

8.3.2 RDREC y WRREC

Las instrucciones RDREC (Leer registro) y WRREC (Escribir registro) se pueden utilizar con PROFINET, PROFIBUS y AS-i.

Tabla 8- 70 Instrucciones RDREC y WRREC

KOP / FUP	SCL	Descripción
<p>"RDREC_DB"</p> 	<pre>"RDREC_DB" (req:=_bool_in_, ID:=_word_in_, index:=_dint_in_, mlen:=_uint_in_, valid=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_dword_out_, len=>_uint_out_, record:=_variant_inout_);</pre>	<p>Use la instrucción RDREC para leer un registro con el número INDEX de un componente direccionado por la ID, como un rack central o un componente descentralizado (PROFIBUS DP o PROFINET IO). Asigne el número máximo de bytes que deben leerse en MLEN. La longitud seleccionada del área de destino RECORD debe tener como mínimo la longitud de MLEN bytes.</p>
<p>"WRREC_DB"</p> 	<pre>"WRREC_DB" (req:=_bool_in_, ID:=_word_in_, index:=_dint_in_, len:=_uint_in_, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_dword_out_, record:=_variant_inout_);</pre>	<p>Use la instrucción WRREC para transferir un RECORD con el número de registro INDEX a un esclavo DP o dispositivo PROFINET IO direccionado por la ID, como un módulo del rack central o un componente descentralizado (PROFIBUS DP o PROFINET IO).</p> <p>Asigne la longitud en bytes del registro que debe transmitirse. Así, la longitud seleccionada del área de origen RECORD debe tener como mínimo la longitud de LEN bytes.</p>

- STEP 7 crea el DB automáticamente al insertar la instrucción.
- En los ejemplos SCL, "RDREC_DB" y "WRREC_DB" son los nombres de los DB de instancia.

Tabla 8- 71 Tipos de datos RDREC y WRREC para los parámetros

Parámetro y tipo		Tipo de datos	Descripción
REQ	IN	Bool	REQ = 1: Transferir registro
ID	IN	HW_IO (Word)	<p>Dirección lógica del esclavo DP o componente PROFINET IO (módulo o submódulo):</p> <ul style="list-style-type: none"> • Para un módulo de salida debe activarse el bit 15 (por ejemplo, para dirección 5: ID:= DW#16#8005). • Para un módulo combinado debe especificarse le menor de las dos direcciones. <p>Nota: la ID de dispositivo puede determinarse de uno de los modos siguientes:</p> <ul style="list-style-type: none"> • seleccionando lo siguiente en la "Vista de redes": <ul style="list-style-type: none"> – dispositivo (casilla gris) – "Propiedades" del dispositivo – "ID de hardware" <p>Nota: no todos los dispositivos muestran sus IDs de hardware.</p> • seleccionando lo siguiente en el menú "Árbol del proyecto": <ul style="list-style-type: none"> – variables PLC – tabla de variables predeterminada – tabla de constantes del sistema <p>Se muestran todas las IDs de hardware configuradas para el dispositivo.</p>
INDEX	IN	Byte, Word, USInt, UInt, SInt, Int, DInt	Número de registro
MLEN	IN	Byte, USInt, UInt	Longitud máxima en bytes de la información del registro que debe recuperarse (RDREC)
VALID	OUT	Bool	Se ha recibido un nuevo registro y es válido (RDREC). El bit VALID es TRUE durante un ciclo tras haberse finalizado la última petición sin errores.
DONE	OUT	Bool	El registro se ha transferido (WRREC). El bit DONE es TRUE durante un ciclo tras haberse finalizado la última petición sin errores.
BUSY	OUT	Bool	<ul style="list-style-type: none"> • BUSY = 1: El proceso de lectura (RDREC) o escritura (WRREC) todavía no ha terminado. • BUSY = 0: La transmisión del registro se ha completado.
ERROR	OUT	Bool	ERROR = 1: Se ha producido un error de lectura (RDREC) o escritura (WRREC). El bit ERROR es TRUE durante un ciclo tras haberse finalizado la última petición con un error. El valor del código de error en el parámetro STATUS sólo es válido durante un ciclo en que ERROR = TRUE.
STATUS	OUT	DWord	Estado de bloque o información de error

Parámetro y tipo		Tipo de datos	Descripción
LEN	OUT (RDREC) IN (WRREC)	UInt	<ul style="list-style-type: none"> Longitud de la información de registro recuperada (RDREC) Longitud máxima en bytes del registro que debe transferirse (WRREC)
RECORD	IN_OUT	Variant	<ul style="list-style-type: none"> Área de destino para el registro recuperado (RDREC) Registro (WRREC)

Las instrucciones RDREC y WRREC funcionan de forma asíncrona, de modo que el procesamiento se extiende a lo largo de varias llamadas de la instrucción. Inicie la tarea llamando RDREC o WRREC con REQ = 1.

El estado de la tarea se muestra en el parámetro de salida BUSY y en los dos bytes centrales del parámetro de salida STATUS. La transferencia del registro se ha completado cuando el parámetro de salida BUSY tiene el valor FALSE

TRUE (sólo durante un ciclo) en el parámetro de salida VALID (RDREC) o DONE (WRREC) verifica que el registro se ha transferido correctamente al área de destino RECORD (RDREC) o al dispositivo de destino (WRREC). En caso de RDREC, el parámetro de salida LEN contiene la longitud de los datos recuperados en bytes.

El parámetro de salida ERROR (sólo durante un ciclo cuando ERROR = TRUE) indica si se ha producido un error en una transmisión. En ese caso, el parámetro de salida STATUS (sólo durante un ciclo cuando ERROR = TRUE) contiene la información de error.

Los registros están definidos por el fabricante del dispositivo de hardware. Consulte la documentación del dispositivo del fabricante de hardware para obtener información detallada acerca de un registro.

Nota

Si un esclavo DPV1 se ha configurado mediante un archivo GSD (GSD vers. 3 y superior) y la interfaz DP del maestro DP está ajustada en "compatible con S7", no será posible leer registros de los módulos E/S en el programa de usuario con "RDREC" o escribir en los módulos E/S con "WRREC". En este caso, el maestro DP direcciona el slot equivocado (slot configurado + 3).

Remedio: pone la interfaz del maestro DP a "DPV1".

Nota

Las interfaces de las instrucciones "RDREC" y "WRREC" son idénticas a los FBs "RDREC" y "WRREC" definidos en la norma "PROFIBUS Guideline PROFIBUS Communication and Proxy Function Blocks according to IEC 61131-3".

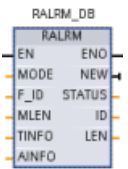
Nota

Si se utiliza "RDREC" o "WRREC" para leer o escribir un registro para PROFINET IO, los valores negativos de los parámetros INDEX, MLEN y LEN se interpretarán como un entero de 16 bits sin signo.

8.3.3 RALRM

La instrucción RALRM (Recibir alarma) puede usarse con PROFINET y PROFIBUS.

Tabla 8- 72 Instrucción RALRM

KOP / FUP	SCL	Descripción
	<pre>"RALRM_DB" (mode:=_int_in_, f_ID:=_word_in_, mlen:=_uint_in_, new=>_bool_out_, status=>_dword_out_, ID=>_word_out_, len=>_uint_out_, tinfo:=_variant_inout_, ainfo:=_variant_inout_);</pre>	<p>Use la instrucción RALRM (leer alarma) para leer la información de una alarma de diagnóstico desde módulos o dispositivos de E/S PROFIBUS o PROFINET.</p> <p>La información de los parámetros de salida contiene la información de inicio del OB al que se ha llamado, así como información sobre el origen de la alarma.</p> <p>Llame a RALRM en un OB de alarma para devolver información sobre los eventos que han producido la alarma. En el S7-1200, sólo se soportan las alarmas de diagnóstico (OB82).</p>

- STEP 7 crea el DB automáticamente al insertar la instrucción.
- En el ejemplo SCL, "RALRM_DB" es el nombre del DB de instancia.

Tabla 8- 73 Tipos de datos para los parámetros

Parámetro y tipo	Tipo de datos	Descripción
MODE	IN	Byte, USInt, SInt, Int Estado operativo
F_ID	IN	HW_IO (Word) La dirección inicial lógica del componente (módulo) del cual deben recibirse alarmas Nota: la ID de dispositivo puede determinarse de uno de los modos siguientes: <ul style="list-style-type: none"> seleccionando lo siguiente en la "Vista de redes": <ul style="list-style-type: none"> dispositivo (casilla gris) "Propiedades" del dispositivo "ID de hardware" Nota: No todos los dispositivos muestran sus IDs de hardware. seleccionando lo siguiente en el menú "Árbol del proyecto": <ul style="list-style-type: none"> variables PLC tabla de variables predeterminada tabla de constantes del sistema Se muestran todas las IDs de hardware configuradas para el dispositivo.
MLEN	IN	Byte, USInt, UInt Longitud máxima en bytes de la información de alarma de datos que debe recibirse. Si MLEN es 0, se permitirá la recepción de tanta información de alarma de datos como esté disponible en el área de destino de AINFO.
NEW	OUT	Bool Se ha recibido una nueva alarma.
STATUS	OUT	DWord Estado de la instrucción RALRM. Consulte "Parámetro STATUS para RDREC, WRREC y RALRM" (Página 292) para obtener más información.

Parámetro y tipo		Tipo de datos	Descripción
ID	OUT	HW_IO (Word)	Identificador de hardware del módulo de E/S que ha provocado la alarma de diagnóstico. Nota: consulte el parámetro F_ID para obtener una explicación de cómo determinar la ID de dispositivo.
LEN	OUT	DWord, UInt, UInt, DInt, Real, LReal	Longitud de la información de alarma de AINFO recibida.
TINFO	IN_OUT	Variant	Información de la tarea: Área de destino para inicio de OB e información de administración. La longitud de TINFO siempre es de 32 bytes.
AINFO	IN_OUT	Variant	Información de alarma: Área de destino para información de encabezado e información de alarma adicional. Para AINFO, indique una longitud de como mínimo el número de bytes de MLEN, si MLEN es mayor que 0. La longitud de AINFO es variable.

Nota

Si se llama "RALRM" en un OB cuyo evento de arranque no es una alarma E/S, la instrucción ofrecerá información reducida en sus salidas en correspondencia.

Asegúrese de utilizar diferentes DBs instancia al llamar "RALRM" en OBs distintos. Si se evalúan datos como resultado de una llamada de "RALRM" fuera del OB de alarma asociado, debería utilizar un DB de instancia separado para cada evento de arranque del OB.

Nota

La interfaz de la instrucción "RALRM" es idéntica al FB "RALRM" definido en la norma "PROFIBUS Guideline PROFIBUS Communication and Proxy Function Blocks according to IEC 61131-3".

Llamar RALRM

La instrucción RALRM se puede llamar en tres modos de operación diferentes (MODE).

Tabla 8- 74 Modos de operación de la instrucción RALRM

MODE	Descripción
0	<ul style="list-style-type: none"> ID contiene la ID de hardware del módulo de E/S que ha disparado la alarma. El parámetro de salida NEW se establece en TRUE. LEN produce una salida de 0. AINFO y TINFO no se actualizan con ninguna información.
1	<ul style="list-style-type: none"> ID contiene la ID de hardware del módulo de E/S que ha disparado la alarma. El parámetro de salida NEW se establece en TRUE. LEN produce una salida con la cantidad de bytes de los datos AINFO que se devuelven. AINFO y TINFO se actualizan con la información de alarma.
2	<p>Si la ID de hardware asignada al parámetro de entrada F_ID ha activado la alarma, entonces:</p> <ul style="list-style-type: none"> ID contiene la ID de hardware del módulo de E/S que ha disparado la alarma. Debe ser igual que el valor de F_ID. El parámetro de salida NEW se establece en TRUE. LEN produce una salida con la cantidad de bytes de los datos AINFO que se devuelven. AINFO y TINFO se actualizan con la información de alarma.

Nota

Si se asigna un área de destino demasiado corta para TINFO o AINFO, RALRM no puede devolver la información completa.

MLEN puede limitar la cantidad de datos de AINFO que se devuelven.

Consulte los parámetros de AINFO y TINFO de la Ayuda on line de STEP 7 para obtener información sobre cómo interpretar los datos de TINFO y AINFO.

8.3.4 Parámetro STATUS para RDREC, WRREC y RALRM

El parámetro de salida STATUS contiene información de error que se interpreta como ARRAY[1...4] OF BYTE con la siguiente estructura:

Tabla 8- 75 Matriz de salida STATUS

Elemento de matriz	Nombre	Descripción
STATUS[1]	Function_Num	<ul style="list-style-type: none"> B#16#00, si no hay errores ID de función de DPV1-PDU: Si se produce un error, B#16#80 se combina lógicamente con 0 (para leer registro: B#16#DE; para escribir registro: B#16#DF). Si no se utiliza ningún elemento de protocolo DPV1, se transferirá B#16#C0 .
STATUS[2]	Error Decode	Ubicación del ID de error
STATUS[3]	Error_Code_1	ID de error
STATUS[4]	Error_Code_2	Extensión de ID de error específica del fabricante

Tabla 8- 76 Valores de STATUS[2]

Error_decode (B#16#...)	Fuente	Descripción
00 a 7F	CPU	Sin errores o alarmas
80	DPV1	Error según CEI 61158-6
81 a 8F	CPU	B#16#8x muestra un error en el "enésimo" parámetro de llamada de la instrucción.
FE, FF	Perfil DP	Error específico de perfil

Tabla 8- 77 Valores de STATUS[3]

Error_decode (B#16#...)	Error_code_1 (B#16#...)	Explicación (DVP1)	Descripción
00	00		Sin errores o alarmas
70	00	Reservado, rechazar	Llamada inicial; sin transferencia de registro activa
	01	Reservado, rechazar	Llamada inicial; transferencia de registro iniciada
	02	Reservado, rechazar	Llamada intermedia; la transferencia de registro ya está activa
80	90	Reservado, permitir	Dirección inicial lógica no válida
	92	Reservado, permitir	Tipo no permitido para puntero Variant
	93	Reservado, permitir	El componente DP direccionado vía ID o F_ID no está configurado.

Error_decode (B#16#....)	Error_code_1 (B#16#....)	Explicación (DVP1)	Descripción
	96		"RALRM (Página 289)" no puede suministrar la información de arranque del OB, la información de administración, de encabezado o de alarma adicional. Para los OBs 4x, 55, 56, 57, 82 y 83 se puede utilizar la instrucción "DPNRM_DG (Página 298)" para leer de forma asíncrona la trama actual del aviso de diagnóstico del esclavo DP relevante (información de dirección procedente de la información de arranque del OB).
	A0	Error de lectura	Acuse negativo al leer del módulo
	A1	Error de escritura	Acuse negativo al escribir del módulo
	A2	Fallo del módulo	Error de protocolo DP en nivel 2 (por ejemplo, fallo de esclavo o problemas de bus)
	A3	Reservado, permitir	<ul style="list-style-type: none"> PROFIBUS DP: Error de protocolo DP con Direct-Data-Link-Mapper o interfaz de usuario/usuario PROFINET IO: Error general CM
	A4	Reservado, permitir	Fallo de comunicación en el bus de comunicación
	A5	Reservado, permitir	-
	A7	Reservado, permitir	Esclavo DP o módulo ocupado (error temporal).
	A8	Conflicto de versiones	Esclavo DP o módulo notifica versiones no compatibles.
	A9	Función no soportada	Función no soportada por esclavo DP o módulo
	AA a AF	Específico de usuario	El esclavo DP o módulo devuelve un error específico de fabricante en su aplicación. Consulte la documentación del fabricante del esclavo DP o módulo.
	B0	Índice no válido	El registro no se conoce en el módulo; número de registro no permitido ≥ 256
	B1	Error de longitud de escritura	<p>La información de longitud en el parámetro RECORD es incorrecta.</p> <ul style="list-style-type: none"> Con "RALRM": error de longitud en AINFO <p>Nota: consulte la Ayuda online de STEP 7 para acceder directamente a información sobre cómo interpretar los búferes "AINFO" devueltos.</p> <ul style="list-style-type: none"> Con "RDREC (Página 286)" y "WRREC (Página 286)": error de longitud en "MLEN"
	B2	Slot no válido	El slot configurado no está ocupado.
	B3	Conflicto de tipos	El tipo de módulo real no coincide con el especificado.
	B4	Área no válida	El esclavo DP o módulo notifica un acceso a un área no válida.
	B5	Conflicto de estado	El esclavo DP o módulo no está listo
	B6	Acceso denegado	El esclavo DP o módulo deniega el acceso.
	B7	Rango no válido	El esclavo DP o módulo notifica un rango no válido de un parámetro o valor.
	B8	Parámetro no válido	El esclavo DP o módulo notifica un parámetro no válido.

Error_decode (B#16#...)	Error_code_1 (B#16#...)	Explicación (DVP1)	Descripción
	B9	Tipo no válido	El esclavo DP o módulo notifica un tipo no válido: <ul style="list-style-type: none"> Con "RDREC (Página 286)": búfer demasiado pequeño (no es posible leer partes de un campo) Con "WRREC (Página 286)": búfer demasiado pequeño (no es posible escribir en partes de un campo)
	BA a BF	Específico de usuario	El esclavo DP o módulo devuelve un error específico de fabricante al acceder. Consulte la documentación del fabricante del esclavo DP o módulo.
	C0	Conflicto de restricción de lectura	<ul style="list-style-type: none"> Con "WRREC (Página 286)": Los datos sólo se pueden escribir cuando la CPU se encuentra en modo STOP. Nota: Ello significa que los datos no pueden escribirse con el programa de usuario. Sólo se pueden escribir datos online con una programadora o un PC. Con "RDREC (Página 286)": el módulo enruta el registro, pero no hay datos o los datos sólo pueden leerse cuando la CPU está en estado STOP. Nota: Si los datos sólo pueden leerse cuando la CPU está en modo STOP, no es posible una evaluación por parte del programa de usuario. En ese caso, sólo pueden leerse datos online con una programadora o un PC.
	C1	Conflicto de restricción de escritura	El módulo aún no ha procesado los datos de la solicitud anterior de escritura en el módulo.
	C2	Recurso ocupado	Actualmente el módulo está procesando el número máximo de tareas posibles para una CPU.
	C3	Recurso no disponible	Los recursos requeridos están ocupados en este momento.
	C4		Error temporal interno. No ha sido posible realizar la tarea. Repetir la tarea. Si se produce este error, compruebe la instalación en busca de fuentes de perturbación eléctrica.
	C5		Esclavo DP o módulo no disponible
	C6		La transferencia del registro se ha cancelado debido a la cancelación por clase de prioridad
	C7		Tarea cancelada debido a re arranque en caliente o en frío del maestro DP.
	C8 a CF		El esclavo DP o módulo devuelve un error de recurso específico de fabricante. Consulte la documentación del fabricante del esclavo DP o módulo.
	Dx	Específico de usuario	Específico de esclavo DP. Véase la descripción del esclavo DP.
81	00 a FF		Error en el parámetro de llamada inicial (con "RALRM (Página 289)": MODE)
	00		Estado operativo no válido

Error_decode (B#16#....)	Error_code_1 (B#16#....)	Explicación (DVP1)	Descripción
82	00 a FF		Error en el segundo parámetro de llamada
88	00 a FF		Error en el octavo parámetro de llamada (con "RALRM (Página 289)": TINFO) Nota: consulte la Ayuda online de STEP 7 para acceder directamente a información sobre cómo interpretar los búferes "TINFO" devueltos.
	01		ID de sintaxis errónea
	23		Estructura de cantidad excedida o área de destino demasiado pequeña
	24		ID de rango errónea
	32		Número de DB/DI fuera del rango de usuario
	3A		El número de DB/DI es CERO para la ID de área DB/DI o el DB/DI especificado no existe.
89	00 a FF		Error en el noveno parámetro de llamada (con "RALRM (Página 289)": AINFO) Nota: consulte la Ayuda online de STEP 7 para acceder directamente a información sobre cómo interpretar los búferes "AINFO" devueltos.
	01		ID de sintaxis errónea
	23		Estructura de cantidad excedida o área de destino demasiado pequeña
	24		ID de rango errónea
	32		Número de DB/DI fuera del rango de usuario
	3A		El número de DB/DI es CERO para la ID de área DB/DI o el DB/DI especificado no existe.
8A	00 a FF		Error en el 10.º parámetro de llamada
8F	00 a FF		Error en el 15.º parámetro de llamada
FE, FF	00 a FF		Error específico de perfil

Elemento de matriz STATUS[4]

Con errores DPV1, el maestro DP transfiere en STATUS[4] a la CPU y a la instrucción. Sin un error de DPV1, este valor se pone a 0 con las siguientes excepciones para RDREC:


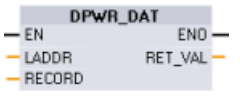
- STATUS[4] contiene la longitud del área de destino de RECORD, si MLEN > la longitud del área de destino de RECORD.
- STATUS[4]=MLEN, si la longitud real del registro < MLEN < la longitud del área de destino de RECORD.
- STATUS[4]=0, si STATUS[4] > 255; debe estar activado

En PROFINET IO, STATUS[4] tiene el valor 0.

8.3.5 DPRD_DAT y DPWR_DAT

Las instrucciones DPRD_DAT (Leer datos coherentes de un esclavo DP normalizado) y DPWR_DAT (Escribir datos coherentes de un esclavo DP normalizado) se pueden utilizar con PROFINET y PROFIBUS.

Tabla 8- 78 Instrucciones DPRD_DAT y DPWR_DAT

KOP / FUP	SCL	Descripción
	<pre>ret_val := DPRD_DAT(laddr:=_word_in_, record=>_variant_out_);</pre>	<p>Utilice la instrucción DPRD_DAT para leer los datos coherentes de un esclavo DP o dispositivo PROFINET IO normalizado. Si no ocurre ningún error durante la transferencia de datos, los datos leídos se introducen en el área de destino especificada por el parámetro RECORD. El área de destino debe tener la misma longitud que la configurada con STEP 7 para el módulo seleccionado. Cuando se llama la instrucción DPRD_DAT, sólo se puede acceder a los datos de un módulo o identificación DP en la dirección inicial configurada.</p>
	<pre>ret_val := DPWR_DAT(laddr:=_word_in_, record:=_variant_in_);</pre>	<p>Utilice la instrucción DPWR_DAT para transferir los datos en RECORD con coherencia al esclavo DO o dispositivo PROFINET IO direccionado. El área de origen debe tener la misma longitud que la configurada con STEP 7 para el módulo seleccionado.</p>

La CPU soporta hasta 64 bytes de datos coherentes. Para áreas de datos coherentes mayores de 64 bytes, deben usarse las instrucciones DPRD_DAT y DPWR_DAT. En caso necesario, estas instrucciones puede utilizarse para áreas de datos de 1 byte o mayores. Si el acceso se deniega, se devuelve el código W#16#8090.

Nota

Si se utiliza las instrucciones DPRD_DAT y DPWR_DAT con datos coherentes, hay que retirar los datos coherentes de la actualización automática de la memoria imagen de proceso. Véase "Conceptos de PLC: Ejecución del programa de usuario" (Página 67) para obtener más información.

Tabla 8- 79 Tipos de datos para los parámetros

Parámetro y tipo		Tipo de datos	Descripción
LADDR	IN	HW_IO (Word)	<ul style="list-style-type: none"> Dirección inicial configurada del área "I" del módulo del cual se van a leer los datos (DPRD_DAT) Dirección inicial configurada del área de salida de la memoria imagen de proceso del módulo en la que se escribirán los datos (DPWR_DAT) <p>Las direcciones deben introducirse en formato hexadecimal (por ejemplo, la dirección de entrada o salida 100 significa: LADDR:=W#16#64).</p>
RECORD	OUT	Variant	Área de destino para los datos de usuario que se han leído (DPRD_DAT) o área de origen de los datos de usuario que deben escribirse (DPWR_DAT). Éste debe tener exactamente la misma longitud que la configurada con STEP 7 para el módulo seleccionado. Sólo se permite el tipo de datos Byte.
RET_VAL	OUT	Int	Si se produce un error mientras la función está activa, el valor de retorno contiene un código de error.

Operaciones DPRD_DAT

El área de destino debe tener la misma longitud que la configuración con STEP 7 para el módulo configurado. Si no se produce ningún error durante la transferencia de datos, los datos leídos se introducen en el área de destino identificada por RECORD.

Si se lee de un esclavo DP estándar con un diseño modular o con varios identificadores, sólo se puede acceder a los datos de un módulo o identificador DP por cada llamada de la instrucción DPRD_DAT especificando la dirección inicial configurada.

Operaciones DPWR_DAT

Los datos de RECORD se transfieren con coherencia al esclavo DP o dispositivo PROFINET IO estándar direccionado. Los datos se transfieren de forma asíncrona, de modo que el proceso de escritura se completa cuando se completa la instrucción.

El área de origen debe tener la misma longitud que la configurada con STEP 7 para el módulo seleccionado.

Si el esclavo DP estándar presenta un diseño modular, sólo se puede acceder a un módulo.

Tabla 8- 80 Códigos de error DPRD_DAT y DPWR_DAT

Código de error	Descripción
0000	No ha ocurrido ningún error
808x	Error de sistema con módulo interfaz DP externo
8090	Se da uno de los siguientes casos: <ul style="list-style-type: none"> No ha configurado un módulo para la dirección base lógica especificada. Ha ignorado la restricción referente a la longitud de los datos coherentes. Ha introducido la dirección inicial en el parámetro LADDR en formato hexadecimal.
8092	Un tipo diferente a Byte se ha especificado en la referencia Any.

8.3 E/S distribuidas (PROFINET, PROFIBUS o AS-i)

Código de error	Descripción
8093	No hay ningún módulo DP o dispositivo PROFINET IO del que se pueden leer (DPRD_DAT) o en el que se pueden escribir (DPWR_DAT) datos coherentes en la dirección lógica especificada en LADDR.
80A0	Error de acceso detectado durante el acceso a los dispositivos I/O (DPRD_DAT).
80A1	Error de acceso detectado durante el acceso a los dispositivos I/O (DPWR_DAT).
80B0	Fallo de esclavo en el módulo de interfaz DP externo
80B1	La longitud del área de destino (DPRD_DAT) o de origen (DPWR_DAT) especificada coincide con la longitud de datos de usuario configurada con STEP 7 Basic.
80B2, 80B3, 80C2, 80Fx	Error de sistema con módulo interfaz DP externo (DPRD_DAT) y (DPWR_DAT)
87xy, 808x	Error de sistema con módulo interfaz DP externo (DPRD_DAT)
85xy	Error de sistema con módulo interfaz DP externo (DPWR_DAT)
80C0	Los datos todavía no han sido leídos por el módulo (DPRD_DAT).
80C1	El módulo aún no ha procesado los datos de la tarea de escritura anterior (DPWR_DAT).
8xyy ¹	Información general del error

Consulte "Instrucciones avanzadas, E/S descentralizadas: Información de error para RDREC, WRREC y RALRM" (Página 292) para más información sobre códigos de error generales.

Nota

Si se accede a esclavos DPV1, la información de error de éstos puede transmitirse del maestro DP a la instrucción.

8.3.6 DPNRM_DG

La instrucción DPNRM_DG (Leer datos de diagnóstico de un esclavo DP) puede utilizarse con PROFIBUS.

Tabla 8- 81 Instrucción DPNRM_DG

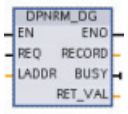
KOP / FUP	SCL	Descripción
	<pre>ret_val := DPNRM_DG(req:= bool_in_, laddr:= word_in_, record=>_variant_out_, busy=> bool_out);</pre>	Utilice la instrucción DPNRM_DG para leer los datos de diagnóstico actuales de un esclavo DP en el formato especificado por EN 50 170 volumen 2, PROFIBUS. Los datos leídos se introducen en el área de destino indicada en RECORD tras una transferencia sin errores.

Tabla 8- 82 Tipos de datos de la instrucción DPNRM_DG para los parámetros

Parámetro y tipo		Tipo de datos	Descripción
REQ	IN	Bool	REQ=1: Solicitud de lectura
LADDR	IN	HW_DPSLAVE	Dirección de diagnóstico configurada para el esclavo DP: tiene que ser la dirección de la estación y no la del dispositivo I/O. Seleccione la estación (y no la imagen del dispositivo) en la vista "Redes" de la "Configuración de dispositivos" para determinar la dirección de diagnóstico. Introduzca las direcciones en formato hexadecimal. Por ejemplo, la dirección de diagnóstico 1022 significa LADDR=W#16#3FE.
RET_VAL	OUT	Int	Si se produce un error mientras la función está activa, el valor de retorno contiene un código de error. Si no se produce ningún error, la longitud real de los datos transferidos se deposita en RET_VAL.
RECORD	OUT	Variant	Área de destino para los datos de diagnóstico leídos. Sólo se permite el tipo de datos Byte. La longitud mínima del registro que debe leerse o del área de destino es 6. La longitud máxima del registro enviado es 240. Los esclavos estándar pueden proporcionar más de 240 bytes de datos de diagnóstico, hasta un máximo de 244 bytes. En ese caso, los primeros 240 bytes se transfieren al área de destino y en los datos se activa el bit de desbordamiento.
BUSY	OUT	Bool	BUSY=1: la tarea de lectura todavía no se ha completado

La tarea de lectura se inicia asignando 1 al parámetro de entrada REQ en la llamada de la instrucción DPNRM_DG. La tarea de lectura se ejecuta de forma asíncrona, lo que significa que requiere varias llamadas de la instrucción DPNRM_DG. El estado de la tarea se indica en los parámetros de salida RET_VAL y BUSY.

Tabla 8- 83 Estructura de datos del diagnóstico de esclavo

Byte	Descripción
0	Estado de estación 1
1	Estado de estación 2
2	Estado de estación 3
3	Número de estación maestra
4	ID del fabricante (byte high)
5	ID del fabricante (byte low)
6 ...	Información de diagnóstico adicional específica de esclavo

Tabla 8- 84 Códigos de error de la instrucción DPNRM_DG

Código de error	Descripción	Restricción
0000	No hay error	-
7000	Primera llamada con REQ=0: Ninguna transferencia de datos activa; BUSY tiene el valor 0.	-
7001	Primera llamada con REQ=1: Ninguna transferencia de datos activa; BUSY tiene el valor 1.	E/S descentralizadas

8.3 E/S distribuidas (PROFINET, PROFIBUS o AS-i)

Código de error	Descripción	Restricción
7002	Llamada intermedia (REQ irrelevante): Transferencia de datos ya activa; BUSY tiene el valor 1.	E/S descentralizadas
8090	Dirección lógica base especificada no válida: No ha dirección base.	-
8092	El tipo especificado en la referencia Any no es Byte.	-
8093	<ul style="list-style-type: none"> Esta instrucción no está permitida para el módulo especificado en LADDR (se permiten módulos DP S7 para S7-1200). LADDR especifica el dispositivo I/O en lugar de especificar la estación. Seleccione la estación (y no la imagen del dispositivo) en la vista "Redes" de la "Configuración de dispositivos" para determinar la dirección de diagnóstico de LADDR. 	-
80A2	<ul style="list-style-type: none"> Error de protocolo DP en nivel 2 (por ejemplo, fallo de esclavo o problemas de bus) Para ET200S los registros no pueden leerse en modo DPV0. 	E/S descentralizadas
80A3	Error de protocolo DP con interfaz de usuario/usuario	E/S descentralizadas
80A4	Problema de comunicación en el bus de comunicación	El error se produce entre la CPU y el modulo de interfaz DP externo.
80B0	<ul style="list-style-type: none"> La instrucción no es posible para el tipo de módulo. El modulo no reconoce el registro. El número de registro 241 no está permitido. 	-
80B1	La longitud especificada en el parámetro RECORD es incorrecta.	Longitud especificada > longitud de registro
80B2	El slot configurado no está ocupado.	-
80B3	El tipo de módulo real no coincide con el requerido	-
80C0	No hay información de diagnóstico.	-
80C1	El módulo aún no ha procesado los datos de la tarea de escritura anterior para el mismo registro.	-
80C2	Actualmente el módulo está procesando el número máximo de tareas posibles para una CPU.	-
80C3	Los recursos requeridos (memoria, etc.) están ocupados en este momento.	-
80C4	Error temporal interno. La tarea no se ha podido procesar. Repetir la tarea. Si este error se produce frecuentemente, compruebe el sistema en busca de fuentes de interferencia eléctrica.	-
80C5	E/S descentralizadas no disponibles	E/S descentralizadas
80C6	La transferencia del registro se ha interrumpido debido a una interrupción por clase de prioridad (rearranque o ejecución de fondo)	E/S descentralizadas
8xyy ¹	Códigos de error generales	

Consulte "Instrucciones avanzadas, E/S descentralizadas: Información de error para RDREC, WRREC y RALRM" (Página 292) para más información sobre códigos de error generales.

8.4 Alarmas

8.4.1 Instrucciones ATTACH y DETACH

Las instrucciones ATTACH y DETACH permiten activar y desactivar subprogramas controlados por eventos de alarma.

Tabla 8- 85 Instrucciones ATTACH y DETACH

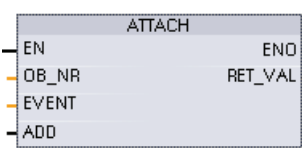
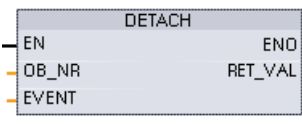
KOP / FUP	SCL	Descripción
	<pre>ret_val := ATTACH(ob_nr:=_int_in_, event:=_event_att_in_, add:=_bool_in_);</pre>	ATTACH habilita la ejecución de un subprograma de OB de alarma para un evento de alarma de proceso.
	<pre>ret_val := DETACH(ob_nr:=_int_in_, event:=_event_att_in_);</pre>	DETACH deshabilita la ejecución de un subprograma de OB de alarma para un evento de alarma de proceso.

Tabla 8- 86 Tipos de datos para los parámetros

Parámetro y tipo		Tipo de datos	Descripción
OB_NR	IN	OB_ATT	Identificador del bloque de organización: Seleccione uno de los OBs de alarma de proceso que fueron creados utilizando la función "Agregar nuevo bloque". Haga doble clic en el campo de parámetro. A continuación, haga clic en el símbolo de ayuda para ver los OBs disponibles.
EVENT	IN	EVENT_ATT	Identificador del evento: Seleccione uno de los eventos de alarma de proceso habilitados en la configuración de dispositivos PLC para las entradas digitales o los contadores rápidos. Haga doble clic en el campo de parámetro. A continuación, haga clic en el símbolo de ayuda para ver los eventos disponibles.
ADD (sólo ATTACH)	IN	Bool	<ul style="list-style-type: none"> ADD = 0 (predeterminado): Este evento reemplaza a todas las asignaciones de eventos anteriores de este OB. ADD = 1: Este evento se agrega a las asignaciones de eventos anteriores de este OB.
RET_VAL	OUT	Int	Código de condición de ejecución

Eventos de alarma de proceso

La CPU soporta los siguientes eventos de alarma de proceso:

- Eventos de flanco ascendente (todas las entradas digitales integradas en la CPU y las entradas digitales de la Signal Board)
 - Un flanco ascendente ocurre cuando la entrada digital cambia de OFF a ON como reacción a una modificación de la señal de un aparato de campo conectado a la entrada.
- Eventos de flanco descendente (todas las entradas digitales integradas en la CPU y las entradas digitales de la Signal Board)
 - Un flanco descendente ocurre cuando la entrada digital cambia de ON a OFF.
- Eventos en los que el valor actual del contador rápido (HSC) = valor de referencia (CV = RV) (HSC 1 a 6)
 - Una alarma CV = RV de un HSC se genera cuando el contaje actual cambia de un valor adyacente al valor que concuerda exactamente con un valor de referencia establecido previamente.
- Eventos de cambio de sentido del HSC (HSC 1 a 6)
 - Un evento de cambio de sentido ocurre cuando se detecta que el sentido de contaje del HSC cambia de ascendente a descendente, o viceversa.
- Eventos de reset externo del HSC (HSC 1 a 6)
 - Algunos modos del HSC permiten asignar una entrada digital como reset externo para poner a cero el valor de contaje del HSC. Un evento de reset externo del HSC ocurre cuando esta entrada cambia de OFF a ON.

Habilitar eventos de alarma de proceso en la configuración de dispositivos

Las alarmas de proceso deben habilitarse durante la configuración de dispositivos. La casilla de habilitación de eventos se debe activar en la configuración de eventos para un canal de entrada digital o un HSC si este evento debe asignarse durante la configuración o en runtime.

Opciones de las casillas de verificación en la configuración de dispositivos PLC:

- Entrada digital
 - Habilitar detección de flancos ascendentes
 - Habilitar detección de flancos descendentes
- Contador rápido (HSC)
 - Habilitar este contador rápido para su uso
 - Generar alarma si el valor del contador es igual al valor de referencia
 - Generar alarma en caso de evento de reset externo
 - Generar alarma en caso de evento de cambio de sentido

Agregar OBs de alarma de proceso nuevos al programa de usuario

De forma predeterminada, ningún OB está asignado a un evento cuando éste se habilita por primera vez. Se indica en la ficha "Alarma de proceso:" Configuración de dispositivo "<no conectado>". Sólo los OB de alarma de proceso pueden asignarse a un evento de alarma de proceso. Todos los OB de alarma de proceso existentes aparecen en la lista desplegable "Alarma de proceso:". Si no se lista ningún OB, es preciso crear un OB de "Alarma de proceso" como se indica a continuación. En la rama "Bloques de programa" del árbol del proyecto:

1. Haga doble clic en "Agregar nuevo bloque", seleccione "Bloque de organización (OB)" y elija "Alarma de proceso".
2. Opcionalmente, puede cambiar el nombre del OB, elegir el lenguaje de programación (KOP o FUP) y seleccionar el número de bloque (cambiar a modo manual y seleccionar un número de bloque diferente del propuesto).
3. Edite el OB y agregue la reacción programada que debe ejecutarse cuando ocurra el evento. Es posible llamar FCs y FBs desde este OB, hasta una profundidad de anidamiento de cuatro niveles.

Parámetro OB_NR

Todos los nombres de OB de alarma de proceso existentes aparecen en la lista desplegable de configuración de dispositivo "Alarma de proceso:" y en la lista desplegable OB_NR del parámetro ATTACH /DETACH.

Parámetro EVENT

Si está habilitado un evento de alarma de proceso, se le asigna un nombre predeterminado y unívoco. Este nombre se puede cambiar editando el campo de edición "Nombre de evento:", pero debe ser un nombre unívoco. Los nombres de eventos se convierten en nombres de variables en la tabla de variables "Constantes" y aparecen en la lista desplegable del parámetro EVENT, en los cuadros de las instrucciones ATTACH y DETACH. El valor de la variable es un número interno utilizado para identificar el evento.

Funcionamiento general

Todo evento de hardware puede asignarse a un OB de alarma de proceso que se pondrá en cola de espera para ser ejecutado cuando ocurra el evento de alarma de proceso. El OB puede asignarse al evento durante la configuración o en runtime.

Es posible asignar o cancelar la asignación de un OB a un evento habilitado al realizar la configuración. Para asignar un OB a un evento durante la configuración hay que usar la lista desplegable "Alarma de proceso:" (haga clic en la flecha hacia abajo de la derecha) y seleccione un OB de la lista desplegable de OBs de alarma de proceso disponibles. Seleccione el nombre de OB deseado en esta lista, o bien elija "<no conectado>" para cancelar la asignación.

8.4 Alarmas

También es posible asignar o deshacer la asignación de un evento de alarma de proceso habilitado en runtime. Utilice las instrucciones ATTACH o DETACH en runtime (varias veces en caso necesario) para asignar o cancelar la asignación de un evento de alarma de proceso al OB respectivo. Si ningún OB está asignado actualmente (debido a que se ha seleccionado "<no conectado>" en la configuración de dispositivos o como resultado de la ejecución de la instrucción DETACH), se ignorará el evento de alarma de proceso habilitado.

Operación DETACH

La instrucción DETACH permite deshacer la asignación de uno o todos los eventos de un OB en particular. Si se especifica un EVENT, se cancelará la asignación sólo de este evento al OB_NR indicado. Todos los demás eventos asignados actualmente a este OB_NR permanecerán asignados. Si no se especifica ningún EVENT, se deshace la asignación de todos los eventos asignados actualmente al OB_NR.

Códigos de condición

Tabla 8- 87 Códigos de condición

RET_VAL (W#16#....)	ENO	Descripción
0000	1	No hay error
0001	1	No hay ninguna asignación que pueda deshacerse (sólo DETACH)
8090	0	El OB no existe
8091	0	Tipo de OB incorrecto
8093	0	El evento no existe

8.4.2 Alarmas cíclicas

8.4.2.1 SET_CINT (Ajustar parámetros de alarma cíclica)

Tabla 8- 88 SET_CINT (Ajustar parámetros de alarma cíclica)

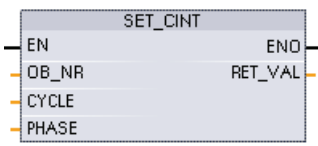
KOP / FUP	SCL	Descripción
	<pre>ret_val := SET_CINT(ob_nr:=_int_in_, cycle:=_udint_in_, phase:=_udint_in_);</pre>	<p>Ajustar el OB de alarma cíclica específico al comienzo de la ejecución cíclica que interrumpe el ciclo del programa.</p>

Tabla 8- 89 Tipos de datos para los parámetros

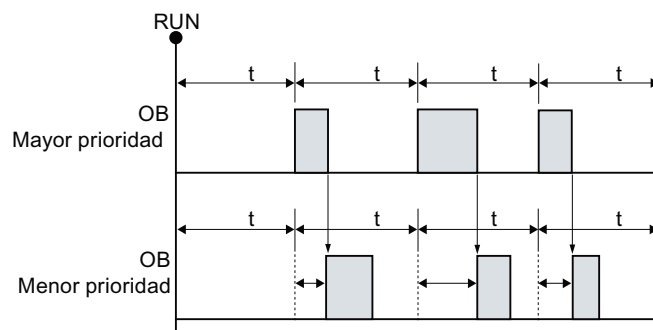
Parámetro y tipo		Tipo de datos	Descripción
OB_NR	IN	OB_CYCLIC	Número de OB (acepta el nombre simbólico)
CYCLE	IN	UDInt	Intervalo de tiempo en microsegundos
PHASE	IN	UDInt	Desfase, en microsegundos
RET_VAL	OUT	Int	Código de condición de ejecución

Ejemplos de parámetros de tiempo:

- Si el tiempo CYCLE = 100 us, el OB de alarma referenciado por OB_NR interrumpirá el programa cíclico cada 100 us. El OB de alarma se ejecuta y a continuación devuelve el control de ejecución al ciclo del programa en el punto de la interrupción.
- Si el tiempo CYCLE = 0, el evento de alarma se desactiva y el OB de alarma no se ejecuta.
- El tiempo PHASE (desfase) es un retardo específico que se produce antes de que comience el intervalo de tiempo CYCLE. Se puede utilizar el desfase para controlar el tiempo de ejecución de OBs de menor prioridad.

Si se llaman OBs de prioridad menor y mayor en el mismo intervalo de tiempo, el OB de menor prioridad sólo se llama una vez que ha finalizado el procesamiento del OB de mayor prioridad. El tiempo de inicio de la ejecución para el OB de menor prioridad puede cambiar en función del tiempo de procesamiento de OBs de mayor prioridad.

Llamada de OB sin desfase



Para reiniciar la ejecución de un OB de menor prioridad en un ciclo de tiempo fijo, el tiempo de desfase debe ser mayor que el tiempo de procesamiento de OBs de mayor prioridad.

Llamada de OB con desfase

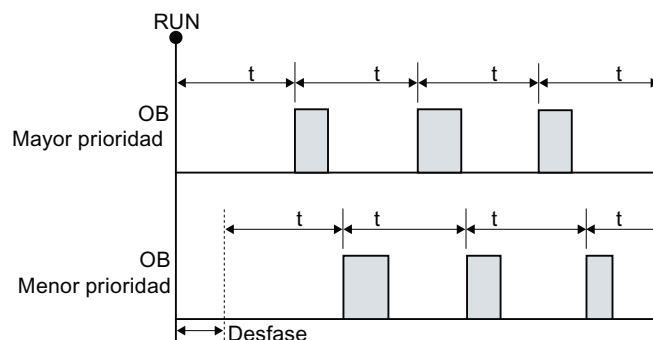


Tabla 8- 90 Códigos de condición

RET_VAL (W#16#....)	Descripción
0000	No hay error
8090	El OB no existe o es del tipo incorrecto
8091	Tiempo de ciclo no válido
8092	Tiempo de desfase no válido
80B2	El OB no tiene ningún evento asignado

8.4.2.2 QRY_CINT (Consultar parámetros de alarma cíclica)

Tabla 8- 91 QRY_CINT (Consultar parámetros de alarma cíclica)

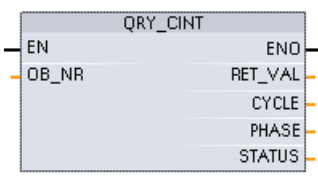
KOP / FUP	SCL	Descripción
	<pre>ret_val := QRY_CINT(ob_nr:=_int_in_, cycle=>_udint_out_, phase=>_udint_out_, status=>_word_out_);</pre>	<p>Recuperar estado de parámetros y ejecución de un OB de alarma cíclica. Los valores devueltos estaban vigentes en el momento en que se ejecutó QRY_CINT.</p>

Tabla 8- 92 Tipos de datos para los parámetros

Parámetro y tipo	Tipo de datos	Descripción
OB_NR	IN	OB_CYCLIC
RET_VAL	OUT	Int
CYCLE	OUT	UDInt
PHASE	OUT	UDInt
STATUS	OUT	Word
		<p>Código de estado de alarma cíclica:</p> <ul style="list-style-type: none"> • Bits 0 a 4, véase la tabla STATUS siguiente • Otros bits, siempre 0

Tabla 8- 93 Parámetro STATUS

Bit	Valor	Descripción
0	0	Durante RUN de la CPU
	1	Durante el arranque
1	0	La alarma está habilitada.
	1	Alarma deshabilitada vía la instrucción DIS_IRT.
2	0	La alarma no está activa o ha transcurrido.
	1	La alarma está activa.
4	0	El OB identificado por OB_NR no existe.
	1	El OB identificado por OB_NR existe.

Bit	Valor	Descripción
Otros bits		Siempre 0

Si se produce un error, RET_VAL muestra el código de error apropiado y el parámetro STATUS es 0.

Tabla 8- 94 Parámetro RET_VAL

RET_VAL (W#16#....)	Descripción
0000	No hay error
8090	El OB no existe o es del tipo incorrecto.
80B2	El OB no tiene ningún evento asignado.

8.4.3 Alarmas de retardo

El procesamiento de las alarmas de retardo se puede iniciar y cancelar con las instrucciones SRT_DINT y CAN_DINT o se puede consultar el estado de la alarma con la instrucción QRY_DINT. Toda alarma de retardo es un evento único que ocurre al cabo del tiempo de retardo indicado. Si el evento de retardo se cancela antes de que transcurra el tiempo de retardo, no ocurrirá la alarma.

Tabla 8- 95 Instrucciones SRT_DINT, CAN_DINT y QRY_DINT

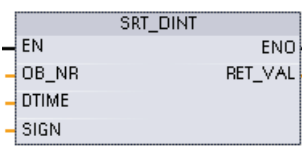
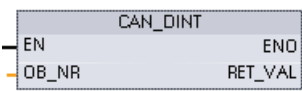
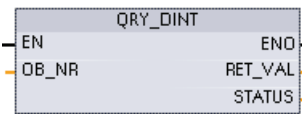
KOP / FUP	SCL	Descripción
	<pre>ret_val := SRT_DINT(ob_nr:=_int_in_, dtime:=_time_in_, sign:=_word_in_);</pre>	SRT_DINT inicia una alarma de retardo que ejecuta un OB una vez transcurrido el tiempo de retardo especificado en el parámetro DTIME.
	<pre>ret_val := CAN_DINT(ob_nr:=_int_in_);</pre>	CAN_DINT cancela una alarma de retardo ya iniciada. El OB de alarma de retardo no se ejecuta en este caso.
	<pre>ret_val := QRY_DINT(ob_nr:=_int_in_, status=>_word_out_);</pre>	QRY_DINT consulta el estado de la alarma de retardo especificada en el parámetro OB_NR.

Tabla 8- 96 Tipos de datos para los parámetros

Parámetro y tipo		Tipo de datos	Descripción
OB_NR	IN	OB_DELAY	Bloque de organización (OB) que debe ejecutarse tras un tiempo de retardo: Seleccione uno de los OBs de alarma de retardo creados utilizando la función "Agregar nuevo bloque" del árbol del proyecto. Haga doble clic en el campo de parámetro. A continuación, haga clic en el símbolo de ayuda para ver los OBs disponibles.
DTIME ¹	IN	Time	Tiempo de retardo (1 hasta 60000 ms)
SIGN ¹	IN	Word	No se utiliza en S7-1200: Se acepta cualquier valor. Debe asignarse un valor para prevenir errores.
RET_VAL	OUT	Int	Código de condición de ejecución
STATUS	OUT	Word	Instrucción QRY_DINT: Estado del OB de alarma de retardo especificado, véase la tabla siguiente

¹ Sólo para SRT_DINT

Operación

La instrucción SRT_DINT especifica un tiempo de retardo, inicia el temporizador de retardo interno y asigna un OB de alarma de retardo al evento de timeout de retardo. Una vez transcurrido el tiempo de retardo especificado, se genera una alarma que dispara la ejecución del OB de alarma de retardo asociado. Una alarma de retardo iniciada puede cancelarse antes de que transcurra el tiempo de retardo especificado, ejecutando para ello la instrucción CAN_DINT. No puede haber más de cuatro eventos de alarma de retardo y de alarma cíclica activos.

Agregar OBs de alarma de retardo al proyecto

Los OBs de alarma de retardo son los únicos que pueden asignarse a las instrucciones SRT_DINT y CAN_DINT. Un proyecto nuevo no contiene aún OBs de alarma de retardo. Es preciso agregar OB de alarma de retardo al proyecto. Para crear un OB de alarma de retardo, proceda del siguiente modo:

1. Haga doble clic en "Agregar nuevo bloque" en la rama "Bloques de programa" del árbol del proyecto, seleccione "Bloque de organización (OB)" y elija "Alarma de retardo".
2. Es posible cambiar el nombre del OB, así como seleccionar el lenguaje de programación y el número de bloque. Cambie al modo de numeración manual si desea asignar un número de bloque diferente del propuesto automáticamente.
3. Edite el OB de alarma de retardo y cree la reacción programada que debe ejecutarse cuando ocurra el evento de timeout de retardo. Es posible llamar FC y FB desde el OB de alarma de retardo, hasta una profundidad de anidamiento de cuatro niveles.
4. Los nuevos nombres de los OB de alarma de retardo estarán disponibles al editar el parámetro OB_NR de las instrucciones SRT_DINT y CAN_DINT.

Parámetro STATUS de QRY_DINT

Tabla 8- 97 Si hay un error (REL_VAL <> 0), entonces STATUS = 0.

Bit	Valor	Descripción
0	0	En RUN
	1	En el arranque
1	0	La alarma está habilitada.
	1	La alarma está deshabilitada.
2	0	La alarma no está activa o ha transcurrido.
	1	La alarma está activa.
4	0	Un OB con un número de OB indicado en OB_NR no existe.
	1	Un OB con un número de OB indicado en OB_NR existe.
Otros bits		Siempre 0

Códigos de condición

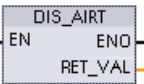
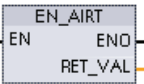
Tabla 8- 98 Códigos de condición para SRT_DINT, CAN_DINT y QRY_DINT

RET_VAL (W#16#...)	Descripción
0000	No ha ocurrido ningún error
8090	Parámetro OB_NR incorrecto
8091	Parámetro DTIME incorrecto
80A0	La alarma de retardo no se ha iniciado.

8.4.4 Alarmas de evento asíncrono

Las instrucciones DIS_AIRT y EN_AIRT sirven para deshabilitar y habilitar el procesamiento de alarmas.

Tabla 8- 99 Instrucciones DIS_AIRT y EN_AIRT

KOP / FUP	SCL	Descripción
	DIS_AIRT () ;	DIS_AIRT retarda el procesamiento de eventos de alarma nuevos. DIS_AIRT se puede ejecutar más de una vez en un OB.
	EN_AIRT () ;	EN_AIRT habilita el procesamiento de eventos de alarma deshabilitados anteriormente con la instrucción DIS_AIRT. Toda ejecución de DIS_AIRT debe ser cancelada por una ejecución de EN_AIRT. Las ejecuciones de EN_AIRT deben ocurrir en un mismo OB, o bien en una FC o FB llamado desde el mismo OB, antes de poder habilitar las alarmas nuevamente para este OB.


 ADVERTENCIA
<p>Si el tiempo de filtro para un canal de entrada digital se reajusta, puede que sea necesario presentar un nuevo valor de entrada de nivel "0" durante un tiempo acumulado de 20 ms para que el filtro esté totalmente operativo ante nuevas entradas. Durante este tiempo, puede que no se detecten o no se cuenten los eventos de pulsación breve "0" cuya duración sea inferior a 20 ms.</p> <p>Este cambio de los tiempos de filtro puede originar un funcionamiento inesperado de los equipos o del proceso, lo que puede causar la muerte o lesiones graves al personal y/o daños a los equipos.</p> <p>Para asegurar que un tiempo de filtro nuevo tenga efecto inmediato, desconecte y vuelva a conectar la CPU.</p>

Tabla 8- 100 Tipos de datos para los parámetros

Parámetro y tipo		Tipo de datos	Descripción
RET_VAL	OUT	Int	Número de retardos = número de ejecuciones de DIS_AIRT en la cola de espera.

El sistema operativo cuenta las ejecuciones de DIS_AIRT. Cada ejecución permanece activa hasta que es cancelada específicamente por una instrucción EN_AIRT o hasta que se termina de procesar el OB actual. Ejemplo: Si se han deshabilitado alarmas cinco veces con cinco ejecuciones de DIS_AIRT, puede cancelarlas con cinco ejecuciones de EN_AIRT antes de que las alarmas se rehabiliten.

Una vez que los eventos de alarma sean habilitadas nuevamente, se procesan las alarmas que ocurrieron cuando estaba activa la instrucción DIS_AIRT, o bien se procesan tan pronto como se haya ejecutado el OB actual.

El parámetro RET_VAL indica el número de veces que se ha inhibido el procesamiento de alarmas. Éste es el número de ejecuciones de DIS_AIRT en la cola de espera. El procesamiento de alarmas sólo se puede habilitar nuevamente cuando el parámetro RET_VAL = 0.

8.5 Diagnóstico (PROFINET o PROFIBUS)

8.5.1 Instrucciones de diagnóstico

Las siguientes instrucciones de diagnóstico se pueden utilizar tanto con PROFINET como con PROFIBUS:

- Instrucción GET_DIAG (Página 316): Permite leer la información de diagnóstico de un dispositivo hardware especificado.
- Instrucción DeviceStates (Página 313): Permite consultar los estados operativos del dispositivo de un dispositivo I/O descentralizado dentro de un subsistema I/O.
- Instrucción ModuleStates (Página 314): Permite consultar los estados operativos de los módulos de un dispositivo I/O descentralizado.
- Instrucción LED (Página 312): Permite leer el estados de los LED de un dispositivo I/O descentralizado.

8.5.2 Eventos de diagnóstico de E/S descentralizadas

Nota

En un sistema PROFIBUS IO, tras una descarga o una desconexión y conexión, la CPU pasará a modo RUN a menos que se establezca la compatibilidad de hardware para permitir módulos sustitutos aceptables (Página 127) y que falten uno o más módulos o que no sea un sustituto aceptable para el módulo configurado.

Como muestra la tabla siguiente, la CPU soporta los diagnósticos que pueden configurarse para los componentes del sistema I/O descentralizado. Cada uno de estos errores genera una entrada en el búfer de diagnóstico.

Tabla 8- 101 Tratamiento de eventos de diagnóstico de PROFINET y PROFIBUS

Tipo de error	¿Hay información de diagnóstico de la estación?	¿Hay una entrada en el búfer de diagnóstico?	Modo de operación de la CPU
Error de diagnóstico	Sí	Sí	Permanece en modo RUN
Fallo del rack o estación	Sí	Sí	Permanece en modo RUN
Error de acceso a E/S ¹	No	Sí	Permanece en modo RUN
Error de acceso a la periferia ²	No	Sí	Permanece en modo RUN
Evento de presencia de módulo	Sí	Sí	Permanece en modo RUN

¹ Causa del ejemplo de error de acceso a E/S: se ha retirado un módulo.

² Causa del ejemplo de error de acceso a la periferia: comunicación acíclica con un submódulo que no está comunicando.

Utilice la instrucción GET_DIAG (Página 316) para obtener información de diagnóstico de cada estación. Esto le permitirá tratar desde el programa los errores encontrados en el dispositivo y, si es necesario, para la CPU a modo STOP. Este método hace necesario especificar el dispositivo de hardware desde el que leer la información de estado.

La instrucción GET_DIAG utiliza la "dirección L" (LADDR) de la estación para obtener el diagnóstico de toda la estación. Esta dirección L se encuentra en la vista de configuración de redes así como seleccionando todo el rack de la estación (todo el área gris), la dirección L se indica en la ficha Propiedades de la estación. La dirección LADDR de cada módulo se encuentra o bien en las propiedades del módulo (en la configuración del dispositivo) o en la tabla de variables predeterminada de la CPU.

8.5.3 Instrucción LED

Tabla 8- 102 Instrucción LED

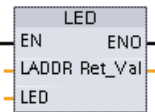
KOP / FUP	SCL	Descripción
	<pre>ret_val := LED(laddr:=_word_in_, LED:=_uint_in_);</pre>	Utilice la instrucción LED para leer el estado de los LEDs en una CPU o interfaz. El estado de LED especificado se devuelve en la salida RET_VAL.

Tabla 8- 103 Tipos de datos para los parámetros

Parámetro y tipo	Tipo de datos	Descripción		
LADDR	IN	HW_IO		
LED	IN	UInt		
		Número de identificación de la CPU o interfaz ¹		
		Número de identificador LED		
		1	RUN/STOP	Color 1 = verde, color 2 = amarillo
		2	Error	Color 1 = rojo
		3	Mantenimiento	Color 1 = amarillo
		4	Redundancia	No aplicable
5	Conexión	Color 1 = verde		
6	Tx/Rx	Color 1 = amarillo		
RET_VAL	OUT	Int		
Estado del LED				

¹ Por ejemplo, es posible seleccionar la CPU (como "PLC_1") o la interfaz PROFINET de la lista desplegable del parámetro.

Tabla 8- 104 Estado de RET_VAL

RET_VAL (W#16#...)	Descripción	
Estado de LED 0 a 9	0	El DB no existe
	1	Off
	2	Color 1 encendido (permanente)

RET_VAL (W#16#...)	Descripción
	3 Color 2 encendido (permanente)
	4 Color 1 parpadea a 2 Hz
	5 Color 2 parpadea a 2 Hz
	6 Color 1 & 2 parpadean de forma alterna a 2 Hz
	7 Color 1 encendido (Tx/Rx)
	8 Color 2 encendido (Tx/Rx)
	9 El estado del LED no está disponible
8091	El dispositivo identificado por LADDR no existe
8092	El dispositivo identificado por LADDR no soporta LEDs
8093	Identificador LED no definido
80Bx	La CPU identificada por LADDR no soporta la instrucción LED

8.5.4 Instrucción DeviceStates

Tabla 8- 105 Instrucción DeviceStates

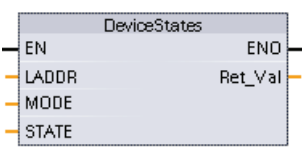
KOP / FUP	SCL	Descripción
	<pre>ret_val := DeviceStates(laddr:=hw_io_in_, mode:=_uint_in_, state:=_variant_inout_);</pre>	<p>DeviceStates determina los estados operativos del dispositivo de E/S de un subsistema de E/S. Tras la ejecución, el parámetro STATE contiene el estado de error de cada dispositivo de E/S en una lista de bits (para LADDR y MODE asignados). Esta información se corresponde con el estado del dispositivo que se indica en la vista de diagnóstico de STEP 7.</p>

Tabla 8- 106 Tipos de datos para los parámetros

Parámetro y tipo	Tipo de datos	Descripción
LADDR IN	HW_IOSYSTEM	Dirección lógica: (Identificador del sistema I/O)
MODE IN	UInt	Tipo de estado: <ul style="list-style-type: none"> • 1: Configuración del dispositivo activa o incompleta. • 2: Dispositivo defectuoso • 3: Dispositivo deshabilitado • 4: El dispositivo existe

Parámetro y tipo		Tipo de datos	Descripción
RET_VAL	OUT	Int	Código de condición de ejecución
STATE ¹	InOut	Variant	<p>Búfer que recibe el estado de error de cada dispositivo: El tipo de datos seleccionado para el parámetro STATE puede ser cualquier tipo de bit (Bool, Byte, Word o DWord) o una matriz del tipo bit.</p> <ul style="list-style-type: none"> • Bit de suma: Bit 0 =1, si uno de los bits de estado de los dispositivos I/O es 1 • Bit de estado: Estado del dispositivo de E/S con número de estación n según el MODE seleccionado. Por ejemplo, MODE = 2 y bit 3 = 1 significa que la estación 3 presenta un defecto.

¹ Para PROFIBUS-DP, la longitud de la información de estado es 128 bits. Para PROFINET IO, la longitud es de 1024 bits.

Tras la ejecución, el parámetro STATE contiene el estado de error de cada dispositivo E/S en forma de lista de bits (para LADDR y MODE asignados).

Tabla 8- 107 Códigos de condición

RET_VAL (W#16#...)	Descripción
0	No hay error
8091	LADDR no existe.
8092	LADDR no direcciona un sistema I/O.
8093	Tipo de datos no válido asignado para el parámetro STATE: Los tipos de datos válidos son (Bool, Byte, Word o Dword) o una matriz de (Bool, Byte, Word o Dword).
80Bx	La instrucción DeviceStates no es soportada por la CPU para este LADDR.
8452	La longitud de todos los datos de estado es demasiado larga para el parámetro STATE asignado. El búfer de STATE contiene un resultado parcial.

8.5.5 Instrucción ModuleStates

Tabla 8- 108 Instrucción ModuleStates

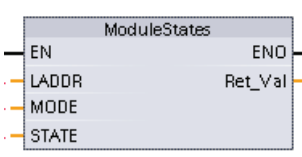
KOP / FUP	SCL	Descripción
	<pre>ret_val := ModuleStates(laddr:=_word_in_, mode:=_uint_in_, state:=_variant_inout);</pre>	<p>ModuleStates determina los estados operativos de los módulos de E/S. Tras la ejecución, el parámetro STATE contiene el estado de error de cada módulo de E/S en una lista de bits (para LADDR y MODE asignados). Esta información se corresponde con el estado del módulo que se indica en la vista de diagnóstico de STEP 7.</p>

Tabla 8- 109 Tipos de datos para los parámetros

Parámetro y tipo		Tipo de datos	Descripción
LADDR	IN	HW_DEVICE	Dirección lógica (identificador de los módulos de E/S)
MODE	IN	UInt	Tipo de estado: <ul style="list-style-type: none"> • 1: Configuración del módulo activa o incompleta. • 2: Módulo defectuoso • 3: Módulo deshabilitado • 4: El módulo existe
RET_VAL	OUT	Int	Estado (código de condición)
STATE ¹	InOut	Variant	Búfer que recibe el estado de error de cada módulo: El tipo de datos utilizado para el parámetro STATE puede ser cualquier tipo de bit (Bool, Byte, Word o DWord) o una matriz del tipo bit. <ul style="list-style-type: none"> • Bit de suma: Bit 0 =1, si uno de los bits de estado del módulo de E/S es 1 • Bit de estado: Estado del módulo de E/S con número de slot <i>n</i> según el MODE seleccionado. Por ejemplo, MODE = 2 y bit 3 = 1 significa que la estación 3 presenta un defecto.

¹ Pueden asignarse 128 bits como máximo. El número de bits necesarios depende del uso del módulo de E/S.

Tabla 8- 110 Códigos de condición

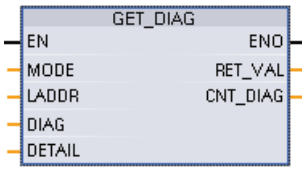
RET_VAL (W#16#...)	Descripción
0	No hay error
8091	El módulo identificado por LADDR no existe.
8092	El módulo identificado por LADDR no direcciona un dispositivo de E/S.
8093	Tipo de datos no válido para el parámetro STATE: Los tipos de datos válidos son (Bool, Byte, Word o Dword) o una matriz de (Bool, Byte, Word o Dword).
80Bx	La instrucción ModuleStates no es soportada por la CPU para este LADDR.
8452	La longitud de todos los datos de estado es demasiado larga para el parámetro STATE asignado. El búfer de STATE contiene un resultado parcial.

8.5.6 Instrucción GET_DIAG

Descripción

Se puede usar la instrucción "GET_DIAG" para leer la información de diagnóstico de un objeto de hardware. El objeto de hardware se selecciona con el parámetro LADDR. Con el parámetro MODE se selecciona la información de diagnóstico que se quiere leer.

Tabla 8- 111 Instrucción GET_DIAG

KOP / FUP	SCL	Descripción
	<pre>ret_val := GET_DIAG(mode:=_uint_in_, laddr:=_word_in_, cnt_diag=>_uint_out_, diag:=_variant_inout_, detail:=_variant_inout_);</pre>	<p>Lee la información de diagnóstico de un dispositivo de hardware especificado.</p>

Parámetros

En la tabla siguiente se muestran los parámetros de la instrucción GET_DIAG:

Tabla 8- 112 Tipos de datos para los parámetros

Parámetro y tipo	Tipo de datos	Descripción
MODE IN	UInt	Utilice el parámetro MODE para seleccionar los datos de diagnóstico que se quieran obtener.
LADDR IN	HW_ANY (Word)	ID de hardware del dispositivo
RET_VAL OUT	Int	Estado de la instrucción
CNT_DIAG OUT	UInt	Número de detalles de diagnóstico obtenidos
DIAG InOut	Variant	Puntero a área de datos para almacenar la información de diagnóstico del modo seleccionado
DETAILS InOut	Variant	Puntero a área de datos para almacenar los detalles de diagnóstico según el modo seleccionado

Parámetro MODE

Según el valor en el parámetro MODE, se obtienen datos diferentes de diagnóstico en los parámetros de salida DIAG, CNT_DIAG y DETAILS:

Tabla 8- 113 Parámetro MODE

MODE	Descripción	DIAG	CNT_DIAG	DETAILS
0	Salida de toda la información de diagnóstico soportada para un módulo como DWord, donde Bit X=1 indica que se soporta el modo X.	Cadena de bits de los modos soportados como DWord, donde Bit X=1 indica que se soporta el modo X.	0	-
1	Salida del estado inherente del objeto de hardware direccionado.	Estado de diagnóstico: Salida según la estructura DIS. Nota: Consulte la información de "Estructura DIS" que aparece a continuación, así como el ejemplo de instrucción GET_DIAG al final del apartado.	0	-
2	Salida del estado de todos los módulos subordinados del objeto de hardware direccionado.	Salida de datos de diagnóstico según la estructura DNN. Nota: Consulte la información de "Estructura DNN" que aparece a continuación, así como el ejemplo de instrucción GET_DIAG al final del apartado.	0	Información de estado de los módulos según la estructura DiagnosticsDetails.

Estructura DIS

Con el parámetro MODE = 1, la información de diagnóstico se obtiene según la estructura DIS. En la tabla siguiente se explican los valores de cada parámetro:

Tabla 8- 114 Estructura de la Diagnostic Information Source (DIS, fuente de información de diagnóstico)

Parámetro	Tipo de datos	Valor	Descripción
MaintenanceState	DWord	Enum	
		0	Mantenimiento no necesario
		1	El módulo o dispositivo está deshabilitado.
		2	-
		3	-
		4	-
		5	Mantenimiento necesario
		6	Mantenimiento solicitado
		7	Error
8	Estado desconocido/error en módulo subordinado		

Parámetro	Tipo de datos	Valor	Descripción
		9	-
		10	Entradas/salidas no disponibles.
Componentstate Detail	DWord	Matriz de bits	Estado de los submódulos del módulo: <ul style="list-style-type: none"> • Bits 0 a 15: mensaje de estado del módulo • Bits 16 a 31: mensaje de estado de la CPU
		0 a 2 (enum)	Información adicional: <ul style="list-style-type: none"> • Bit 0: Sin información adicional • Bit 1: Transferencia no permitida
		3	Bit 3 = 1: Al menos un canal soporta calificadores de diagnóstico.
		4	Bit 4 = 1: Mantenimiento necesario para un canal o un componente como mínimo
		5	Bit 5 = 1: Mantenimiento solicitado para un canal o un componente como mínimo
		6	Bit 6 = 1: Error en un canal o componente como mínimo
		7 a 10	-
		11 a 14	Bit 11 = 1: PNIO: submódulo correcto Bit 12 = 1: PNIO: módulo de repuesto Bit 13 = 1: PNIO: módulo incorrecto Bit 14 = 1: PNIO: módulo desconectado
		15	-
		16 a 31	Información de estado de los módulos generada por la CPU: Bit 16 = 1: Módulo deshabilitado Bit 17 = 1: Modo CiR activo Bit 18 = 1: Entrada no disponible Bit 19 = 1: Salida no disponible Bit 20 = 1: Desbordamiento del búfer de diagnóstico Bit 21 = 1: Diagnóstico no disponible Bits 22 - 31: Reservados (siempre 0)
OwnState	Uint16	Enum	El valor del parámetro OwnState describe el estado de mantenimiento del módulo.
		0	No hay fallos
		1	El módulo o dispositivo está deshabilitado.
		2	Mantenimiento necesario
		3	Mantenimiento solicitado
		4	Error
		5	La CPU no puede acceder al módulo o dispositivo (válido para módulos o dispositivos bajo una CPU).
		6	Entradas/salidas no disponibles.
7	-		
IO State	Uint16	Matriz de bits	Estado de E/S del módulo
		0	Bit 0 = 1: Mantenimiento no necesario
		1	Bit 1 = 1: El módulo o dispositivo está deshabilitado.

Parámetro	Tipo de datos	Valor	Descripción
		2	Bit 2 = 1: Mantenimiento necesario
		3	Bit 3 = 1: Mantenimiento solicitado
		4	Bit 4 = 1: Error
		5	Bit 5 = 1: La CPU no puede acceder al módulo o dispositivo (válido para módulos o dispositivos bajo una CPU).
		6	Entradas/salidas no disponibles.
		7	Calificador; bit 7 = 1, si los bits bit 0, 2 o 3 están a 1
		8 a 15	Reservados (siempre = 0)
		OperatingState	UInt16
0	-		
1	En STOP/actualización de firmware		
2	En STOP/reset de memoria		
3	En STOP/arranque automático		
4	En STOP		
5	Reset de memoria		
6	En ARRANQUE		
7	En RUN		
8	-		
9	En PARADA		
10	-		
11	-		
12	Módulo defectuoso		
13	-		
14	Sin alimentación		
15	CiR		
16	En STOP/sin DIS		
17	In		
18			
19			
20			

Estructura DiagnosticsDetail

Con el parámetro MODE = 2, los detalles de información de diagnóstico se obtienen según la estructura DiagnosticsDetail. En la tabla siguiente se explican los valores de cada parámetro:

Tabla 8- 115 Estructura de DiagnosticsDetail

Parámetro	Tipo de datos	Descripción
ChannelNumber	UInt	Número de canal
Properties	Word	
ALID	UInt	ID de identificación de alarma

Parámetro	Tipo de datos	Descripción
Qualifier	DWord	Calificador de los datos de diagnóstico
ErrorType	UDInt	Tipo de error de canal
ExtErrorType	UDInt	Tipo de error de canal avanzado
AddValue_1	UInt	Valor adicional
AddValue_2	UInt	Valor adicional
AddValue_3	UInt	Valor adicional
AddValue_4	UInt	Valor adicional

Estructura DNN

Con el parámetro MODE = 2, los detalles de información de diagnóstico se obtienen según la estructura DNN. En la tabla siguiente se explican los valores de cada parámetro:

Tabla 8- 116 Estructura del Diagnostic Navigation Node (DNN, nodo de navegación de diagnóstico)

Parámetro	Tipo de datos	Valor	Descripción
SubordinateState	UINT	Enum	Estado del módulo subordinado (consultar parámetro OwnState de la estructura DIS).
SubordinateIOState	WORD	Bitarray	Estado de las entradas y salidas del módulo subordinado (consultar parámetro IO State de la estructura DIS).
DNNmode	WORD	Bitarray	<ul style="list-style-type: none"> • Bit 0 = 0: Diagnóstico habilitado • Bit 0 = 1: Diagnóstico deshabilitado • Bits 1 a 15: Reservado

Parámetro RET_VAL

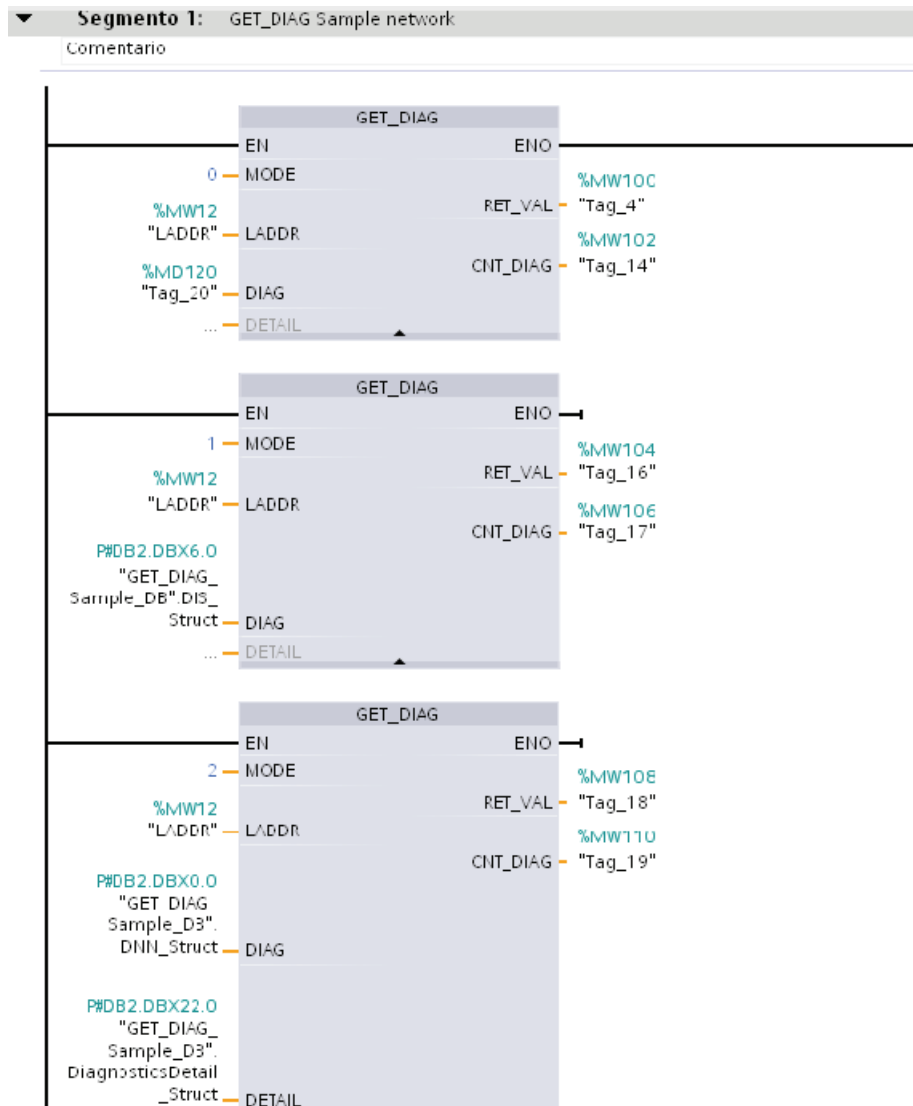
Tabla 8- 117 Códigos de error del parámetro RET_VAL

Código de error (W#16#...)	Descripción
0	No hay error
n	El área de datos en el parámetro DETAILS es demasiado pequeña. No se pueden obtener todos los detalles de los datos de diagnóstico.
8080	Valor no soportado en el parámetro MODE.
8081	El tipo en el parámetro DIAG no está soportado con el modo seleccionado (parámetro MODE).
8082	El tipo en el parámetro DETAILS no está soportado con el modo seleccionado (parámetro MODE).
8090	LADDR no existe.
8091	El canal seleccionado en el parámetro CHANNEL no existe.
80C1	Falta de recursos para ejecución en paralelo

Ejemplo

En el siguiente esquema de lógica de escalera y DB se muestra cómo utilizar los tres modos con las tres estructuras:

- DIS
- DiagnosticsDetail
- DNN



	Nombre	Tipo de datos	Offset	Valor de ...	Rem...	Visible en H...	Comentario
1	Static						
2	DNN_Struct	DNN	0.0		<input type="checkbox"/>	<input checked="" type="checkbox"/>	
3	SubordinateState	UInt	0.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
4	SubordinateIOState	Word	2.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
5	DNNmode	Word	4.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
6	DIS_Struct	DIS	6.0		<input type="checkbox"/>	<input checked="" type="checkbox"/>	
7	MaintenanceState	DWord	0.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
8	ComponentStateDetail	DWord	4.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
9	OwnState	UInt	8.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
10	IOState	Word	10.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
11	OperatingState	UInt	12.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
12	DiagnosticsDetail_Struct	DiagnosticsDetail	22.0		<input type="checkbox"/>	<input checked="" type="checkbox"/>	
13	ChannellNumber	UInt	0.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Channel number
14	Properties	Word	2.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
15	ALID	UInt	4.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Alarm identifier
16	Qualifier	DWord	8.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
17	ErrorType	UDInt	12.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Channel error type
18	ExtErrorType	UDInt	16.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	extended channel error type
19	AddValue_1	UInt	20.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	additional value
20	AddValue_2	UInt	22.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	additional value
21	AddValue_3	UInt	24.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	additional value
22	AddValue_4	UInt	26.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	additional value

- ① DNN
- ② DIS
- ③ DiagnosticsDetail

Nota


En el DB, debe indicar manualmente el tipo de datos para acceder a cada una de las tres estructuras; no hay ninguna lista desplegable para efectuar la selección. Indique los tipos de datos exactamente como se muestra a continuación:

- DNN
- DIS
- DiagnosticsDetail

8.6 Impulso

8.6.1 Instrucción CTRL_PWM

Tabla 8- 118 Instrucción CTRL_PWM (modulación del ancho de impulsos)

KOP / FUP	SCL	Descripción
	<pre>"CTRL_PWM_DB" (PWM:=_word_in_, enable:=_bool_in_, busy=>_bool_out_, status=>_word_out_);</pre>	Ofrece una salida fija de tiempo de ciclo con un ciclo de trabajo variable. La salida PWM se ejecuta continuamente tras haberse iniciado a la frecuencia indicada (tiempo de ciclo). La duración de impulso varía según sea necesario para obtener el control deseado.

- STEP 7 crea el DB automáticamente al insertar la instrucción.
- En el ejemplo SCL, "CTRL_PWM_DB" es el nombre del DB de instancia.

CTRL_HSC

Tabla 8- 119 Tipos de datos para los parámetros

Parámetro y tipo	Tipo de datos	Descripción
PWM	IN	HW_PWM (Word) Identificador PWM: Los nombres de los generadores de impulsos habilitados se convierten en variables en la tabla de variables "Constantes" y están disponibles para ser utilizados como parámetro PWM. (Valor predeterminado: 0)
ENABLE	IN	Bool 1= iniciar generador de impulsos 0 = parar generador de impulsos
BUSY	OUT	Bool Función ocupada (valor predeterminado: 0)
STATUS	OUT	Word Código de condición de ejecución (valor predeterminado: 0)

La instrucción CTRL_PWM almacena la información de parámetros en el DB. El usuario no modifica por separado los parámetros del bloque de datos, sino que la instrucción CTRL_PWM los controla.

Determine el generador de impulsos habilitado que desea utilizar, utilizando su nombre de variable para el parámetro PWM.

Cuando la entrada EN es TRUE, la instrucción PWM_CTRL inicia o detiene el PWM identificado, según el valor de la entrada ENABLE. El valor de la dirección de salida de palabra Q asociada indica la duración de impulso.

Puesto que la CPU procesa la petición cuando se ejecuta la instrucción CTRL_PWM, el parámetro BUSY siempre notifica FALSE. Si se detecta un error, ENO se pone a FALSE y el parámetro STATUS contiene un código de condición.

8.6 Impulso

La duración de impulso se pone al valor inicial ajustado en la configuración de dispositivos cuando la CPU pasa por primera vez al estado operativo RUN. Los valores se escriben en la dirección de palabra Q especificada en la configuración de dispositivos ("Direcciones de salida" / "Dirección inicial:") de la forma requerida para cambiar la duración de impulso. Para escribir la duración de impulso deseada en la palabra Q apropiada se utiliza una instrucción de desplazamiento, conversión, matemática o un cuadro PID. El valor de palabra Q debe estar comprendido en el rango válido (porcentaje, milésimos, diezmilésimos o formato analógico S7).

Nota

Las E/S digitales asignadas a PWM y PTO no se pueden forzar permanentemente

Las E/S digitales utilizadas por los dispositivos con modulación del ancho de impulsos (PWM) y tren de impulsos (PTO) se asignan durante la configuración de dispositivos. Si se asignan direcciones de E/S digitales a estos dispositivos, los valores de las direcciones de E/S asignadas no podrán ser modificados por la función de forzado permanente de la tabla de observación.

Tabla 8- 120 Valor del parámetro STATUS

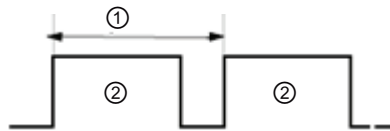
STATUS	Descripción
0	No hay error
80A1	El identificador PWM no direcciona un PWM válido.

Tabla 8- 121 Códigos de condición comunes

Código de condición ¹	Descripción
8022	Área demasiado pequeña para la entrada
8023	Área demasiado pequeña para la salida
8024	Área de entrada no válida
8025	Área de salida no válida
8028	Asignación del bit de entrada no válida
8029	Asignación del bit de salida no válida
8030	El área de salida es un DB de sólo lectura.
803A	El DB no existe.

¹ Cuando se produce uno de estos errores al ejecutar un bloque lógico, la CPU pasa al modo STOP a no ser que se utilicen las instrucciones GetError o GetErrorID dentro de dicho bloque para crear una respuesta programada al error.

8.6.2 Manejo de las salidas de impulso



- ① Tiempo de ciclo
- ② Duración de impulso

El ancho de impulso puede expresarse en centésimas del tiempo de ciclo (0 a 100), milésimas (0 a 1000), diezmilésimas (0 a 10000) o en formato analógico S7. La duración de impulso puede variar entre 0 (sin impulso, siempre off) y escala completa (sin impulso, siempre on).

Puesto que la salida PWM puede variar entre 0 y escala completa, ofrece una salida digital que, en numerosos aspectos, es similar a una salida analógica. Por ejemplo, la salida PWM puede utilizarse para controlar la velocidad de un motor (desde "paro" hasta "a toda velocidad") o la posición de una válvula (desde "cerrada" hasta "totalmente abierta").

Hay dos generadores de impulsos disponibles para controlar las salidas de impulsos rápidos: PWM y PTO (tren de impulsos). Las instrucciones de Motion Control utilizan PTO. Cada generador de impulsos puede asignarse a PWM o PTO, pero no a ambos simultáneamente.

Los dos generadores de impulsos se mapean en las salidas digitales específicas como muestra la tabla siguiente. Es posible utilizar las salidas integradas de la CPU o las salidas opcionales de la Signal Board. La tabla siguiente muestra los números de las salidas (suponiendo que se utiliza la configuración predeterminada de las salidas). Si se ha modificado la numeración de las salidas, sus números serán los que se hayan asignado. Independientemente de ello, PTO1/PWM1 utiliza las dos primeras salidas digitales, en tanto que PTO2/PWM2 utiliza las dos salidas digitales siguientes, bien sea en la CPU o en la Signal Board acoplada. Tenga en cuenta que PWM sólo requiere una salida, mientras que PTO puede utilizar opcionalmente dos salidas por canal. Si una salida no se requiere para una función de impulsos, estará disponible para otros usos.

ATENCIÓN

Los trenes de impulsos no pueden ser utilizados por otras instrucciones del programa de usuario

Si las salidas de la CPU o Signal Board se configuran como generadores de impulsos (para su utilización con la PWM o instrucciones de Motion Control), las direcciones de las salidas correspondientes (Q0.0, Q0.1, Q4.0 y Q4.1) se eliminarán de la memoria Q y no podrán utilizarse para ningún otro fin en el programa de usuario. Si el programa de usuario escribe un valor en una salida utilizada como generador de impulsos, la CPU no escribirá ese valor en la salida física.

Tabla 8- 122 Asignaciones de salidas predeterminadas para los generadores de impulsos

Descripción	Impulso	Sentido
PTO 0		
E/S incorporadas	Q0.0	Q0.1
E/S de la SB	Q4.0	Q4.1
PWM 0		
Salidas incorporadas	Q0.0	-
Salidas de la SB	Q4.0	-
PTO 1		
E/S incorporadas	Q0.2	Q0.3
E/S de la SB	Q4.2	Q4.3
PWM 1		
Salidas incorporadas	Q0.2	-
Salidas de la SB	Q4.2	-
PTO 2		
E/S incorporadas	Q0.4 ¹	Q0.5 ¹
E/S de la SB	Q4.0	Q4.1
PWM 2		
Salidas incorporadas	Q0.4 ¹	-
Salidas de la SB	Q4.1	-
PTO 3		
E/S incorporadas	Q0.6 ²	Q0.7 ²
E/S de la SB	Q4.2	Q4.3
PWM 3		
Salidas incorporadas	Q0.6 ²	-
Salidas de la SB	Q4.3	-

- ¹ La CPU 1211C no tiene las salidas Q0.4, Q0.5, Q0.6 ni Q0.7. Por lo tanto, esas salidas no pueden usarse en la CPU 1211C.
- ² La CPU 1212C no tiene las salidas Q0.6 ni Q0.7. Por lo tanto, esas salidas no pueden usarse en la CPU 1212C.
- ³ Esta tabla es válida para las funciones de PTO/PWM de la CPU 1211C, CPU 1212C, CPU 1214C y CPU 1215C.

8.6.3 Configurar un canal de impulsos para PWM

Para preparar la modulación de ancho de impulsos (PWM), configure primero un canal de impulsos en la configuración de dispositivos, seleccionando la CPU, luego el generador de impulsos (PTO/PWM) y elija PWM1 o PWM2. Habilite el generador de impulsos (casilla de verificación). Si está habilitado un generador de impulsos, se le asigna un nombre predeterminado y unívoco. Este nombre puede cambiarse editando el campo de edición "Nombre:", pero debe ser un nombre unívoco. Los nombres de los generadores de impulsos habilitados se convierten en variables en la tabla de variables "Constantes" y están disponibles para ser utilizados como parámetro PWM de la instrucción CTRL_PWM.

ATENCIÓN

La frecuencia de impulsos máxima de los generadores de salida de impulsos para la salida digital es de 100 KHz (para la CPU), 20 KHz (para una SB estándar) o 200 KHz (para una SB de alta velocidad). Sin embargo, STEP 7 no alerta cuando se configura un eje que, a una velocidad o frecuencia máximas, excede esta limitación de hardware. Ello podría ocasionar problemas en la aplicación. Por tanto, asegúrese de que no se exceda la frecuencia de impulsos máxima del hardware.

Es posible cambiar el nombre del generador de impulsos, agregar un comentario y asignar parámetros como se indica a continuación:

- Generador de impulsos utilizado: PWM o PTO (seleccione PWM)
- Fuente de salida: CPU o SB integrada
- Base de tiempo: milisegundos o microsegundos
- Formato de la duración de impulso:
 - Centésimas (0 a 100)
 - Milésimos (0 a 1000)
 - Diezmilésimos (0 a 10000)
 - Formato analógico S7 (0 a 27648)
- Tiempo de ciclo: Introduzca el valor del tiempo de ciclo. Este valor sólo se puede modificar en la "Configuración de dispositivos".
- Duración de impulso inicial: Introduzca la duración de impulso inicial. El valor de la duración de impulso puede modificarse en runtime.

8.7 Registrar datos

Introduzca la dirección de inicio para configurar las direcciones de salida. Introduzca la dirección de palabra Q en la que desea depositar el valor de la duración de impulso.

ATENCIÓN

Los trenes de impulsos no pueden ser utilizados por otras instrucciones del programa de usuario

Si las salidas de la CPU o Signal Board se configuran como generadores de impulsos (para su utilización con la PWM o instrucciones de Motion Control), las direcciones de las salidas correspondientes (Q0.0, Q0.1, Q4.0 y Q4.1) se eliminarán de la memoria Q y no podrán utilizarse para ningún otro fin en el programa de usuario. Si el programa de usuario escribe un valor en una salida utilizada como generador de impulsos, la CPU no escribirá ese valor en la salida física.

La dirección predeterminada es QW1000 para PWM1 y QW1002 para PWM2. El valor de esta dirección controla el ancho del impulso y se inicializa al valor de la "Duración de impulso inicial:" indicado arriba cada vez que la CPU pasa de STOP a RUN. Este valor de palabra Q puede cambiarse en runtime para modificar la duración de impulso.

8.7 Registrar datos

El programa de control puede usar las instrucciones Data log para almacenar valores de datos de runtime en archivos de registro permanentes. Los archivos de registro se guardan en memorias flash (CPU o Memory Card). Los datos del archivo de registro se guardan en formato CSV (Comma Separated Value) estándar. Los registros se organizan como un archivo de registro circular con un tamaño predeterminado.

Las instrucciones Data log se usan en el programa para crear, abrir o escribir un registro y para cerrar los archivos de registro. El usuario decide qué valores del programa se registran mediante la creación de un búfer de datos que define un registro simple. El búfer de datos se usa como almacenamiento temporal para un nuevo registro. El programa debe desplazar los nuevos valores actuales al búfer durante el tiempo de ejecución. Cuando se actualizan todos los valores de datos actuales, se puede ejecutar la instrucción DataLogWrite para transferir datos del búfer al registro.

Utilice el servidor web integrado en el PLC para administrar los archivos de registro de datos. Descargue los registros recientes o todos los registros, borre registros o archivos de registro con la página web estándar "Registros de datos". Después de transferir un archivo de registro al PC, se pueden analizar los datos con herramientas estándar de hoja de cálculo, como Excel.

8.7.1 Estructura del registro

Los parámetros DATA y HEADER de la instrucción DataLogCreate asignan el tipo de datos y la descripción de encabezado de columna para todos los datos de un registro.

Parámetro DATA para la instrucción DataLogCreate

El parámetro DATA apunta a la memoria utilizada como búfer temporal para un nuevo registro y debe asignarse a una ubicación M o DB.

Es posible asignar un DB completo (derivado de un tipo de datos PLC que se asigna al crear el DB) o parte de un DB (el elemento especificado del DB puede ser cualquier tipo de datos, estructura de tipo de datos, tipo de datos PLC o matriz de datos).

Los tipos de datos de estructura están limitados a un solo nivel de anidamiento. El número total de elementos de datos declarados debe coincidir con el número de columnas especificado en el parámetro de encabezado. El número máximo de elementos de datos que pueden asignarse es de 253 (con sello de tiempo) o de 255 (sin sello de tiempo). Esta restricción mantiene el registro dentro del límite de 256 columnas de una hoja de Excel.

El parámetro DATA puede asignar elementos de datos remanentes o no remanentes en un tipo de DB "estándar" (compatible con S7-300/400) u "optimizado".

Para escribir un registro, primero hay que cargar el registro DATA temporal con nuevos valores de proceso y, a continuación, ejecutar la instrucción DataLogWrite, que guarda nuevos valores de registro en el archivo de registro.

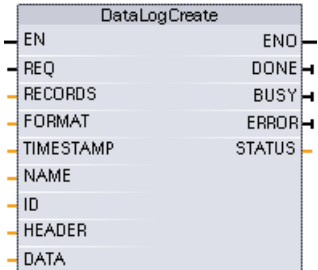
Parámetro HEADER de la instrucción DataLogCreate

El parámetro HEADER apunta a nombres de encabezado de columna para la fila superior de la matriz de datos encriptada en el archivo CSV. Los datos HEADER deben estar ubicados en una memoria DB o M y los caracteres deben cumplir las normas estándar del formato CSV, con comas como carácter de separación entre columnas. El tipo de datos puede ser una secuencia, una matriz de bytes o una matriz de caracteres. Las matrices de caracteres/bytes permiten un tamaño mayor, mientras que las secuencias están limitadas a un máximo de 255 bytes. El parámetro HEADER es opcional. Si HEADER no está asignado, no se crea ninguna fila de encabezado en el archivo de registro.

8.7.2 Instrucciones de programa que controlan registros

8.7.2.1 DataLogCreate

Tabla 8- 123 Instrucción DataLogCreate

KOP / FUP	SCL	Descripción
	<pre>"DataLogCreate_DB" (req:=_bool_in_, records:=_udint_in_, format:=_uint_in_, timestamp:=_uint_in_, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_, name:=_string_inout_, ID:=_dword_inout_, header:=_variant_inout_, data:=_variant_inout_);</pre>	<p>Crea e inicializa un archivo de registro. El archivo se crea en el directorio \DataLogs del PLC, con el nombre indicado en el parámetro NAME, y se abre implícitamente para operaciones de escritura. Las instrucciones Data log se pueden utilizar para que el programa almacene los datos del proceso de tiempo de ejecución en la memoria flash de la CPU.</p> <p>STEP 7 crea automáticamente el DB de instancia asociado al introducir la instrucción.</p>

¹ En el ejemplo SCL, "DataLogCreate_DB" es el nombre del DB de instancia.

Tabla 8- 124 Tipos de datos para los parámetros

Parámetro y tipo	Tipo de datos	Descripción
REQ	IN	Bool Un cambio de señal low a high (flanco ascendente) lanza la operación. (Valor predeterminado: False)
RECORDS	IN	UDint El número máximo de registros que puede contener el registro circular antes de sobrescribir la entrada más antigua: No se incluye el registro de encabezado. Debe haber suficiente memoria de carga disponible en el PLC para poder crear el registro correctamente. (Valor predeterminado: 1)
FORMAT	IN	UInt Formato del registro: <ul style="list-style-type: none"> • 0 - Formato interno (no soportado) • 1 - Valores separados por coma "csv-eng" (valor predeterminado)
TIMESTAMP	IN	UInt Formato del sello horario: no se requieren encabezados de columna para los campos de fecha y hora. El sello de tiempo utiliza la hora del sistema (tiempo universal coordinado, UTC), no la hora local. <ul style="list-style-type: none"> • 0 - Sin sello de tiempo • 1 - Sello de tiempo (valor predeterminado)
NAME	IN	Variant Nombre del registro: El nombre lo especifica el usuario. Esta variante sólo soporta un tipo de datos String y sólo puede ubicarse en una memoria local, DB o M. (Valor predeterminado: '') La referencia del string también se utiliza como nombre del archivo de registro. Los caracteres del nombre deben cumplir las restricciones referentes a los nombres del sistema de archivos de Windows. Los caracteres \ / : * ? " < > y el espacio no están permitidos.

Parámetro y tipo		Tipo de datos	Descripción
ID	In/Out	DWord	Identificador numérico de registro: Este valor generado se guarda para el uso en otras instrucciones de registro. El parámetro ID sólo se utiliza como salida en la instrucción DataLogCreate. (Valor predeterminado: 0) El acceso a este parámetro por nombre simbólico no está permitido.
HEADER	In/Out	Variant	Puntero a nombres de encabezados de columnas de registro para la fila superior de la matriz de datos encriptada en el archivo CSV. (Valor predeterminado: cero). Los datos HEADER deben estar ubicados en una memoria DB o M. Los caracteres deben cumplir las normas del formato CSV estándar, con comas como carácter de separación entre los nombres de las columnas. El tipo de datos puede ser una secuencia, una matriz de bytes o una matriz de caracteres. Las matrices de caracteres/bytes permiten un tamaño mayor, mientras que las secuencias están limitadas a un máximo de 255 bytes. El parámetro HEADER es opcional. Si HEADER no está parametrizado, no se crea ninguna fila de encabezado en el archivo de registro.
DATA	In/Out	Variant	Puntero a la estructura de registro, tipo definido por el usuario (UDT) o matriz. Los datos de registro deben estar ubicados en una memoria DB o M. El parámetro DATA especifica los elementos de datos individuales (columnas) de un registro, así como su tipo. Los tipos de datos de estructura están limitados a un solo nivel de anidamiento. El número de elementos de datos declarados debe coincidir con el número de columnas especificado en el parámetro de encabezado. El número máximo de elementos de datos que pueden asignarse es de 253 (con sello de tiempo) o de 255 (sin sello de tiempo). Esta restricción mantiene el registro dentro del límite de 256 columnas de una hoja de Excel.
DONE	OUT	Bool	El bit DONE es TRUE durante un ciclo tras haberse finalizado la última petición sin errores. (Valor predeterminado: False)
BUSY	OUT	Bool	<ul style="list-style-type: none"> • 0 - Ninguna operación en curso • 1 - Operación en curso
ERROR	OUT	Bool	El bit ERROR es TRUE durante un ciclo tras haberse finalizado la última petición con un error. El valor del código de error en el parámetro STATUS sólo es válido durante un ciclo en que ERROR = TRUE.
STATUS	OUT	Word	Código de condición de ejecución (valor predeterminado: 0)

El archivo de registro se crea con un tamaño fijo predeterminado basado en los parámetros RECORDS y DATA. Los registros se organizan como un archivo de registro circular. Los nuevos registros se agregan al archivo de registro hasta que se almacena el número máximo de registros especificado en el parámetro RECORDS. El siguiente registro que se escriba sobrescribirá el registro más antiguo. Otra tarea de escritura de registro sobrescribirá el siguiente registro más antiguo, etc.

Nota

Para impedir la sobrescritura de registros antiguos se puede utilizar la instrucción DataLogNewFile para crear un registro nuevo basado en el actual, una vez que éste ha guardado el número máximo de registros. Los nuevos registros se guardan en el nuevo archivo de registro. El archivo de registro antiguo y los datos se conservan en la memoria flash.

8.7 Registrar datos

Uso de los recursos de memoria:

- Los registros sólo consumen memoria de carga.
- No existe un límite definido para el número total de registros. El tamaño de todos los registros combinados está limitado por los recursos de memoria de carga disponibles. Sólo pueden abrirse ocho registros de datos al mismo tiempo.
- El número máximo posible para el parámetro RECORDS es el límite correspondiente a un número UDint (4.294.967.295). El límite actual para el parámetro RECORD depende del tamaño de un solo registro, el tamaño de otros registros y los recursos de memoria de carga disponibles. Además, Excel limita el número de filas permitidas en una hoja.

Nota

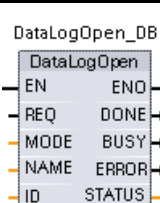
Una operación DataLogCreate se extiende a lo largo de varios ciclos. El tiempo real requerido para la creación de un archivo de registro depende de la estructura y del número de registros. La lógica del programa debe controlar y capturar la transición del bit DONE de DataLogCreate al estado TRUE antes de que el registro pueda utilizarse para otras tareas de registro.

Tabla 8- 125 Valores de ERROR y STATUS

ERROR	STATUS (W#16#...)	Descripción
0	0000	No hay error
0	7000	Llamada sin flanco REQ: BUSY = 0, DONE = 0
0	7001	Primera llamada con flanco de REQ (en curso): BUSY = 1, DONE = 0
0	7002	Llamada Nª (en curso): BUSY = 1, DONE = 0
1	8070	Se está utilizando toda la memoria de instancia interna.
1	807F	Error interno
1	8090	Nombre de archivo no válido
1	8091	El parámetro de nombre no es una referencia de String.
1	8093	El registro ya existe.
1	8097	La longitud de archivo solicitada supera el máximo del sistema de archivos.
1	80B3	La memoria de carga disponible no es suficiente.
1	80B4	La MC (Memory Card) está protegida contra escritura.
1	80C1	Demasiados archivos abiertos: no se permiten más de ocho archivos de registro abiertos.
1	8253	Contaje de registros no válido
1	8353	La selección de formato no es válida
1	8453	La selección de sello no es válida
1	8B24	Asignación no válida del área HEADER: por ejemplo, apuntar a la memoria local
1	8B51	El tipo de datos del parámetro HEADER no es válido
1	8B52	Hay demasiados elementos de datos en el parámetro HEADER
1	8C24	Asignación no válida del área DATA: por ejemplo, apuntar a la memoria local
1	8C51	El tipo de datos del parámetro DATA no es válido
1	8C52	Hay demasiados elementos de datos en el parámetro DATA

8.7.2.2 DataLogOpen

Tabla 8- 126 Instrucción DataLogOpen

KOP / FUP	SCL	Descripción
	<pre>"DataLogOpen_DB" (req:=_bool_in_, mode:=_uint_in_, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_, name:=_string_inout_, ID:=_dword_inout_);</pre>	<p>Abre un archivo de registro ya existente. Es necesario abrir un registro antes de poder escribir registros nuevos en él. Los registros se pueden abrir y cerrar de forma individual. Es posible abrir simultáneamente un máximo de ocho registros de datos.</p> <p>STEP 7 crea automáticamente el DB de instancia asociado al introducir la instrucción.</p>

² En el ejemplo SCL, "DataLogOpen_DB" es el nombre del DB de instancia.

Tabla 8- 127 Tipos de datos para los parámetros

Parámetro y tipo	Tipo de datos	Descripción
REQ	IN	Bool Un cambio de señal low a high (flanco ascendente) lanza la operación. (Valor predeterminado: False)
MODE	IN	UInt Modo de operación: <ul style="list-style-type: none"> 0 - Agregar a los datos existentes (valor predeterminado) 1 - Borrar todos los registros existentes
NAME	IN	Variant Nombre de un registro existente: esta variante sólo soporta un tipo de datos String y sólo puede ubicarse en una memoria local, DB o M. (Valor predeterminado: '')
ID	In/Out	DWord Identificador numérico de un registro. (Valor predeterminado: 0) Nota: el acceso a este parámetro por nombre simbólico no está permitido.
DONE	OUT	Bool El bit DONE es TRUE durante un ciclo tras haberse finalizado la última petición sin errores. (Valor predeterminado: False)
BUSY	OUT	Bool <ul style="list-style-type: none"> 0 - Ninguna operación en curso 1 - Operación en curso
ERROR	OUT	Bool El bit ERROR es TRUE durante un ciclo tras haberse finalizado la última petición con un error. El valor del código de error en el parámetro STATUS sólo es válido durante un único ciclo en que ERROR = TRUE.
STATUS	OUT	Word Código de condición de ejecución (valor predeterminado: 0)

Se puede proporcionar el parámetro NAME o un parámetro ID (ID en forma de entrada) de un registro ya existente. Si se suministran ambos parámetros y una ID válida se corresponde con el registro NAME, se usa la ID y NAME se ignora.

8.7 Registrar datos

NAME debe ser el nombre de un registro creado por la instrucción DataLogCreate. Si sólo se proporciona NAME y NAME especifica un registro válido, se devolverá la ID correspondiente (parámetro ID como salida).

Nota

Uso general de los archivos de registro

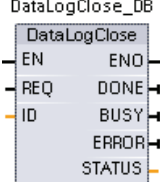
- Los archivos de registro se abren automáticamente después de las operaciones DataLogCreate y DataLogNewFile.
- Los archivos de registro se cierran automáticamente después de una transición de RUN a STOP del PLC o de una desconexión y conexión del PLC.
- Debe haber un archivo de registro abierto antes de que sea posible otra operación DataLogWrite.
- Puede haber un máximo de ocho archivos de registro abiertos a la vez. Si bien pueden existir más de ocho archivos de registro, algunos de ellos deben estar cerrados para que no haya más de ocho abiertos.

Tabla 8- 128 Valores de ERROR y STATUS

ERROR	STATUS (W#16#)	Descripción
0	0000	No hay error
0	0002	Advertencia: el archivo de registro ya está abierto en este programa de aplicación
0	7000	Llamada sin flanco REQ: BUSY = 0, DONE = 0
0	7001	Primera llamada con flanco de REQ (en curso): BUSY = 1, DONE = 0
0	7002	Llamada N° (en curso): BUSY = 1, DONE = 0
1	8070	Se está utilizando toda la memoria de instancia interna.
1	8090	La definición del registro no es coherente con el archivo de registro existente.
1	8091	El parámetro de nombre no es una referencia de String.
1	8092	El registro no existe.
1	80C0	El archivo de registro está bloqueado.
1	80C1	Demasiados archivos abiertos: no se permiten más de ocho archivos de registro abiertos.

8.7.2.3 DataLogClose

Tabla 8- 129 Instrucción DataLogClose

KOP / FUP	SCL	Descripción
DataLogClose_DB 	<pre>"DataLogClose_DB" (req:=_bool_in_, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_, ID:=_dword_inout_);</pre>	<p>Cierra un archivo de registro abierto. Las operaciones DataLogWrite sobre un registro cerrado producen un error. No se permite realizar operaciones de escritura en este registro hasta que no se lleve a cabo otra operación DataLogOpen.</p> <p>La transición al modo STOP cierra todos los archivos de registro abiertos.</p> <p>STEP 7 crea automáticamente el DB de instancia asociado al introducir la instrucción.</p>

² En el ejemplo SCL, "DataLogClose_DB" es el nombre del DB de instancia.

Tabla 8- 130 Tipos de datos para los parámetros

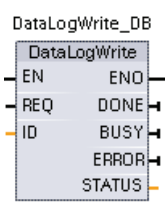
Parámetro y tipo		Tipo de datos	Descripción
REQ	IN	Bool	Un cambio de señal low a high (flanco ascendente) lanza la operación. (Valor predeterminado: False)
ID	In/Out	DWord	Identificador numérico de un registro. Sólo se utiliza como entrada para la instrucción DataLogClose. (Valor predeterminado: 0) Nota: el acceso a este parámetro por nombre simbólico no está permitido.
DONE	OUT	Bool	El bit DONE es TRUE durante un ciclo tras haberse finalizado la última petición sin errores.
BUSY	OUT	Bool	<ul style="list-style-type: none"> 0 - Ninguna operación en curso 1- Operación en curso
ERROR	OUT	Bool	El bit ERROR es TRUE durante un ciclo tras haberse finalizado la última petición con un error. El valor del código de error en el parámetro STATUS sólo es válido durante un único ciclo en que ERROR = TRUE.
STATUS	OUT	Word	Código de condición de ejecución (valor predeterminado: 0)

Tabla 8- 131 Valores de ERROR y STATUS

ERROR	STATUS (W#16#)	Descripción
0	0000	No hay error
0	0001	El registro no se ha abierto
0	7000	Llamada sin flanco REQ: BUSY = 0, DONE = 0
0	7001	Primera llamada con flanco de REQ (en curso): BUSY = 1, DONE = 0
0	7002	Llamada Nª (en curso): BUSY = 1, DONE = 0
1	8092	El registro no existe.

8.7.2.4 DataLogWrite

Tabla 8- 132 Instrucción DataLogWrite

KOP / FUP	SCL	Descripción
	<pre>"DataLogWrite_DB" (req=_bool_in_, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_, ID:=_dword_inout_);</pre>	<p>Escribe un registro en el registro especificado. Para que se permita una operación DataLogWrite es necesario que el registro de destino ya existente esté abierto.</p> <p>STEP 7 crea automáticamente el DB de instancia asociado al introducir la instrucción.</p>

² En el ejemplo SCL, "DataLogWrite_DB" es el nombre del DB de instancia.

Tabla 8- 133 Tipos de datos para los parámetros

Parámetro y tipo		Tipo de datos	Descripción
REQ	IN	Bool	Un cambio de señal low a high (flanco ascendente) lanza la operación. (Valor predeterminado: False)
ID	In/Out	DWord	Identificador numérico de registro. Sólo se utiliza como entrada para la instrucción DataLogWrite. (Valor predeterminado: 0) Nota: el acceso a este parámetro por nombre simbólico no está permitido.
DONE	OUT	Bool	El bit DONE es TRUE durante un ciclo tras haberse finalizado la última petición sin errores.
BUSY	OUT	Bool	<ul style="list-style-type: none"> 0 - Ninguna operación en curso 1 - Operación en curso
ERROR	OUT	Bool	El bit ERROR es TRUE durante un ciclo tras haberse finalizado la última petición con un error. El valor del código de error en el parámetro STATUS sólo es válido durante un único ciclo en que ERROR = TRUE.
STATUS	OUT	Word	Código de condición de ejecución (valor predeterminado: 0)

La dirección de memoria y la estructura de datos del búfer de registro se configura con el parámetro DATA de una instrucción DataLogCreate. Hay que emplear el programa para cargar el búfer de registro con valores de proceso de runtime actuales y, a continuación, ejecutar la instrucción DataLogWrite para desplazar los datos nuevos de registro del búfer al registro.

El parámetro ID identifica un registro y una configuración de registro. El número ID se genera cuando se crea un registro.

Si hay registros vacíos en el archivo de registro circular, se escribe el siguiente registro vacío disponible. Si todos los registros están llenos, se sobrescribe el registro más antiguo.

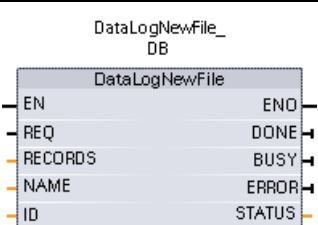
PRECAUCIÓN
Potencial de pérdida de datos de registros durante un fallo de alimentación de la CPU
Si se produce un fallo de alimentación durante una operación DataLogWrite incompleta, se podría perder el registro que se estuviera transfiriendo al registro.

Tabla 8- 134 Valores de ERROR y STATUS

ERROR	STATUS (W#16#)	Descripción
0	0000	No hay error
0	0001	Indica que el registro está lleno: Cada registro se crea con un número máximo específico de registros. Se ha escrito el último registro del número máximo. La siguiente tarea de escritura sobrescribirá el registro más antiguo.
0	7000	Llamada sin flanco REQ: BUSY = 0, DONE = 0
0	7001	Primera llamada con flanco de REQ (en curso): BUSY = 1, DONE = 0
0	7002	Llamada N° (en curso): BUSY = 1, DONE = 0
1	8070	Se está utilizando toda la memoria de instancia interna.
1	8092	El registro no existe.
1	80B0	El archivo de registro no está abierto (sólo para modo de abertura explícita).

8.7.2.5 DataLogNewFile

Tabla 8- 135 Instrucción DataLogNewFile

KOP / FUP	SCL	Descripción
	<pre>"DataLogNewFile_DB" (req:=_bool_in_, records:=_udint_in_, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_, name:=_DataLog_out_, ID:=_dword_inout_);</pre>	<p>Permite al programa crear un archivo de registro nuevo basándose en otro ya existente.</p> <p>STEP 7 crea automáticamente el DB de instancia asociado al introducir la instrucción.</p>

² En el ejemplo SCL, "DataLogNewFile_DB" es el nombre del DB de instancia.

Tabla 8- 136 Tipos de datos para los parámetros

Parámetro y tipo	Tipo de datos	Descripción
REQ	IN	Bool
RECORDS	IN	UDInt
		<p>Un cambio de señal low a high (flanco ascendente) lanza la operación. (Valor predeterminado: False)</p> <p>El número máximo de registros que puede contener el registro circular antes de sobrescribir la entrada más antigua. (Valor predeterminado: 1)</p> <p>No se incluye el registro de encabezado. Debe haber suficiente memoria de carga disponible en la CPU para poder crear el registro correctamente.</p>

8.7 Registrar datos

Parámetro y tipo		Tipo de datos	Descripción
NAME	IN	Variant	Nombre del registro: El nombre lo especifica el usuario. Esta variante sólo soporta un tipo de datos String y sólo puede ubicarse en una memoria local, DB o M. (Valor predeterminado: '') La referencia del string también se utiliza como nombre del archivo de registro. Los caracteres del nombre deben cumplir las restricciones referentes a los nombres del sistema de archivos de Windows. Los caracteres \ / : * ? " < > y el espacio no están permitidos.)
ID	In/Out	DWord	Identificador numérico de registro (valor predeterminado: 0): <ul style="list-style-type: none"> • Durante la ejecución, la entrada ID identifica un registro válido. La nueva configuración de registro se copia desde este registro. • Tras la ejecución, el parámetro ID se convierte en una salida que devuelve la ID del archivo de registro recién creado. Nota: el acceso a este parámetro por nombre simbólico no está permitido.
DONE	OUT	Bool	El bit DONE es TRUE durante un ciclo tras haberse finalizado la última petición sin errores.
BUSY	OUT	Bool	<ul style="list-style-type: none"> • 0 - Ninguna operación en curso • 1 - Operación en curso
ERROR	OUT	Bool	El bit ERROR es TRUE durante un ciclo tras haberse finalizado la última petición con un error. El valor del código de error en el parámetro STATUS sólo es válido durante un único ciclo en que ERROR = TRUE.
STATUS	OUT	Word	Código de condición de ejecución (valor predeterminado: 0)

La instrucción DataLogNewFile se puede ejecutar cuando un registro se llena o se considera que está completo y no se desea perder ningún dato almacenado en él. Es posible crear un archivo de registro vacío basándose en la estructura del archivo de registro lleno. El registro del encabezado se duplica a partir del registro original, junto con las propiedades de este último (búfer de registro DATA, formato de datos y configuración de sello de tiempo). El archivo de registro original se cierra implícitamente y el archivo de registro nuevo se abre también implícitamente.

Disparador del parámetro DataLogWrite: El programa de usuario debe controlar los parámetros ERROR y STATUS de cada operación DataLogWrite. Cuando se escribe el registro final y un registro está lleno, el bit DataLogWrite ERROR se pone a 1 y la palabra DataLogWrite STATUS se pone a 1. Estos valores de ERROR y STATUS son válidos sólo para un ciclo, por lo que la lógica de monitorización debe utilizar ERROR = 1 como una ventana temporal para capturar el valor STATUS y comprobar a continuación STATUS = 1 (el registro está lleno).

Operación DataLogNewFile: Cuando la lógica del programa de usuario recibe la señal de que el registro está lleno, este estado se utiliza para activar una operación DataLogNewFile. DataLogNewFile debe ejecutarse con la ID de un registro ya existente (generalmente lleno) y abierto, pero con un sólo parámetro NAME nuevo. Una vez que la operación DataLogNewFile se ha realizado, se devuelve un nuevo valor de ID de registro (como parámetro de salida), que corresponde al nombre del nuevo registro. El nuevo archivo de registro está implícitamente abierto, y está listo para almacenar nuevos registros. Las operaciones DataLogWrite nuevas directamente relacionadas con el nuevo archivo de registro deben utilizar el valor ID devuelto por la operación DataLogNewFile.

Nota

Una operación DataLogNewFile se extiende a lo largo de varios ciclos. El tiempo real requerido para la creación de un archivo de registro depende de la estructura y del número de registros. La lógica del programa debe controlar y captar la transición del bit DONE de DataLogNewFile al estado TRUE antes de que el registro pueda utilizarse para otras tareas de registro.

Tabla 8- 137 Valores de ERROR y STATUS

ERROR	STATUS (W#16#)	Descripción
0	0000	No hay error
0	7000	Llamada sin flanco REQ: BUSY = 0, DONE = 0
0	7001	Primera llamada con flanco de REQ (en curso): BUSY = 1, DONE = 0
0	7002	Llamada N° (en curso): BUSY = 1, DONE = 0
1	8070	Se está utilizando toda la memoria de instancia interna.
1	8090	Nombre de archivo no válido
1	8091	El parámetro de nombre no es una referencia de String.
1	8092	El registro no existe.
1	8093	El registro ya existe.
1	8097	La longitud de archivo solicitada supera el máximo del sistema de archivos.
1	80B3	La memoria de carga disponible no es suficiente.
1	80B4	La MC está protegida contra escritura.
1	80C1	Demasiados archivos abiertos.

8.7.3 Trabajar con registros de datos

Los archivos de registro se guardan en formato de valores separados por comas (*.csv) en una memoria flash remanente. Los archivos de registro pueden verse utilizando la función de servidor web del PLC o retirando la Memory Card del PLC e insertándola en un lector de tarjetas de PC convencional.

Leer registros con la función de servidor web del PLC

Si el puerto PROFINET del PLC y el equipo están conectados a una red, se puede emplear un navegador de PC (como Microsoft Internet Explorer o Mozilla Firefox) para acceder al servidor web integrado. El PLC puede estar en modo RUN o STOP cuando se utilice el servidor web. Si el PLC está en modo RUN, el programa de control sigue funcionando mientras el servidor web del PLC transfiere los datos de registro a través de la red.

Acceso al servidor web:

1. Habilite el servidor web en la configuración de dispositivos para la CPU de destino (Página 524).
2. Conecte el PC al PLC a través de la red PROFINET (Página 525).
3. Inicie una sesión en el servidor web integrado (Página 527).
4. Descargue los registros recientes o todos los registros, borre registros o archivos de registro con la página web estándar "Data Log" (Página 539).
5. Una vez que se ha descargado al PC una copia de un archivo de registro, se puede abrir el archivo .csv con una aplicación para hojas de cálculo, como Excel.

Ver registros de una Memory Card del PLC

Si la CPU S7-1200 tiene insertada una Memory Card "Programa" del tipo S7-1200, se puede extraer e insertar en una ranura SD (Secure Digital) o MMC (MultiMediaCard) estándar de un PC o una programadora. El PLC se encuentra en modo STOP cuando no tiene insertada la Memory Card, por lo que el programa de control no se ejecutará.

Utilice el explorador de Windows y vaya al directorio \DataLog de la Memory Card. Todos los archivos de registro *.csv se encuentran en ese directorio.

Haga una copia de los archivos de registro y guarde las copias en una unidad local del PC. A continuación se puede utilizar Excel para abrir una copia local de un archivo *.csv, pero no el archivo original, que permanece guardado en la Memory Card.

PRECAUCIÓN

Con ayuda de un lector de tarjetas para PC es posible copiar los archivos de registro de una Memory Card S7-1200, pero no pueden modificarse ni borrarse

La página web de archivos de registro estándar del servidor web es la herramienta recomendada para visualizar, descargar (copiar), borrar (borrar los datos) y eliminar archivos de registro. El servidor web gestiona los archivos de la Memory Card y ayuda a prevenir la modificación accidental o el borrado de datos.

En la navegación directa en el sistema de archivos de la Memory Card con Windows Explorer existe el riesgo de borrar o modificar accidentalmente registros u otros archivos de sistema, lo que podría estropear un archivo o hacer que la Memory Card sea inservible.

Visualización de registros de datos desde un navegador web

Aunque no se utilice la funcionalidad del servidor web, es posible ver registros de datos directamente desde un navegador web como Internet Explorer o Mozilla Firefox. Basta con introducir el texto siguiente en la barra de direcciones del navegador utilizando la dirección IP de la CPU y el nombre real del archivo de registro de datos indicado en STEP 7 en lugar de "MyDataLog":

`http://192.168.0.1/DataLog.html?FileName=MyDataLog.csv`

Las direcciones fijas de los archivos de registro de datos también permiten acceder a ellos a través de herramientas de recopilación de archivos de terceros.

8.7.4 Límites del tamaño de los archivos de registro

Los archivos de registro comparten espacio de la memoria de carga del PLC con el programa, los datos del programa, los datos de configuración, las páginas web definidas por el usuario y los datos del sistema PLC. Un programa de gran tamaño que utilice memoria de carga interna requiere una gran cantidad de memoria de carga y el espacio libre para archivos de registro de datos puede ser insuficiente. En tal caso, se puede utilizar una "Tarjeta de programa" para aumentar el tamaño de la memoria de carga. Las CPU S7-1200 pueden utilizar memoria de carga interna o externa, pero no ambas a la vez.

Véase el capítulo referente a la tarjeta de memoria para conocer los detalles de cómo crear una tarjeta tipo "Programa" (Página 116).

Regla de tamaño máximo de un archivo de registro de datos

El tamaño máximo de un archivo de registro de datos no puede superar el 25% del tamaño de la memoria de carga (interna o externa). Si la aplicación requiere más entradas de registros de datos, utilice la instrucción "DataLogNewFile" para crear un archivo nuevo cuando se hayan completado todos los registros del primer archivo. Consulte la tabla siguiente para conocer los tamaños máximos de un archivo de registro de datos.

Tabla 8- 138 Tamaño de memoria de carga y tamaño máximo de un archivo de registro de datos

Área de datos	CPU 1211C	CPU 1212C	CPU 1214C	CPU 1215C	Almacenamiento de datos
Memoria de carga interna memoria flash	1 MB (250 KB máx. para un archivo de registro de datos)	1 MB (250 KB máx. para un archivo de registro de datos)	4 MB (500 KB máx. para un archivo de registro de datos)	4 MB (500 KB máx. para un archivo de registro de datos)	Programa de usuario y datos de programa, datos de configuración, registros de datos, páginas web definidas por el usuario, y datos del sistema PLC
Memoria de carga externa Tarjetas de memoria flash opcionales como "Tarjeta de programa"	2 MB, 12 MB o 24 MB según el tamaño de la tarjeta SD (500 KB máx. para un archivo de registro con una tarjeta de 2 MB) (6 MB máx. para un archivo de registro de datos con una tarjeta de 24 MB)				

Determinar el tamaño del espacio libre de la memoria de carga

1. Establecer una conexión online entre STEP 7 y el PLC S7-1200 de destino.
2. Descargar el programa en el que se desean agregar operaciones de registros.
3. Crear las posibles páginas web personalizadas que se requieran. (Las páginas web estándar que permiten acceder a registros de datos se guardan en firmware del PLC y no utilizan memoria de carga).
4. Utilizar las herramientas online y de diagnóstico para conocer el tamaño de la memoria de carga y el porcentaje de memoria libre disponible (Página 716).
5. Multiplicar el tamaño de memoria de carga por el porcentaje libre para obtener el tamaño actual de memoria de carga libre.

Regla de tamaño máximo para todos los registros de datos combinados

La cantidad de memoria de carga libre varía durante las operaciones normales, ya que el sistema operativo va utilizando y liberando memoria. Es conveniente limitar el tamaño combinado de todos los archivos de registro a la mitad del espacio libre disponible.

Calcular la memoria necesaria para un único registro

El registro se guarda en forma de bytes de caracteres en el formato de archivo CSV (valores separados por coma). La tabla siguiente muestra el número de bytes necesarios para guardar cada tipo de datos.

Tabla 8- 139 Tamaños de datos de archivos CSV

Tipo de datos	Número de bytes (bytes de datos más byte separador de coma)
Bool	2
Byte	5
Word	7
DWord	12
Char	4
String	<p>Ejemplo 1: MyString[10] El tamaño máximo de la cadena es de 10 caracteres. Caracteres de texto + relleno automático con caracteres vacíos = 10 bytes Comillas de apertura y cierre + caracteres coma = 3 bytes 10 + 3 = 13 bytes en total</p> <p>Ejemplo 2: Mystring2 Si se omiten los corchetes al indicar el tamaño, se asignan de forma predeterminada 254 bytes. Caracteres de texto + relleno automático con caracteres vacíos = 254 bytes Comillas de apertura y cierre + caracteres coma = 3 bytes 254 + 3 = 257 bytes en total</p>
USInt	5
UInt	7

Tipo de datos	Número de bytes (bytes de datos más byte separador de coma)
UDInt	12
SInt	5
Int	7
DInt	12
Real	16
LReal	25
Time	15
DTL	24

El parámetro DATA de DataLogCreate apunta a una estructura que especifica el número de campos de datos y el tipo de datos de cada campo para un registro. La tabla siguiente indica los bytes necesarios en el archivo CSV para cada tipo de datos. Multiplique el número de apariciones del tipo de datos indicado por el número de bytes que requiere. Haga la misma operación para cada tipo de datos del registro y sume el número de bytes para conocer el tamaño total del registro. Agregue un byte para el carácter de final de línea.

Tamaño del registro = suma de los bytes necesarios para todos los campos de datos + 1 (el carácter de final de línea).

Calcular la memoria necesaria para un archivo de registro completo

El parámetro RECORDS de la instrucción DataLogCreate especifica el número máximo de registros de un archivo de registro. Cuando se crea un archivo de registro, se asigna el tamaño de memoria máximo.

Tamaño de archivo de registro = (número de bytes en un registro) x (número de registros).

8.7.5 Programa de ejemplo de registros de datos

Este programa de ejemplo de registro no muestra toda la lógica del programa necesaria para obtener valores de muestreo de un proceso dinámico, pero muestra las operaciones clave de las instrucciones de registro. La estructura y el número de archivos de registro utilizados dependen de las necesidades de control del proceso.

Nota

Uso general de los archivos de registro

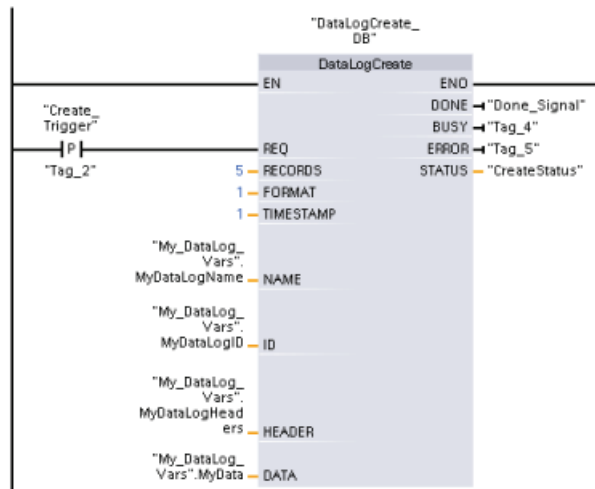
- Los archivos de registro se abren automáticamente después de las operaciones DataLogCreate y DataLogNewFile.
- Los archivos de registro se cierran automáticamente después de una transición de RUN a STOP del PLC o de una desconexión y conexión del PLC.
- Debe haber un archivo de registro abierto antes de que sea posible una operación DataLogWrite.
- Puede haber un máximo de ocho archivos de registro abiertos a la vez. Si bien pueden existir más de ocho archivos de registro, algunos de ellos deben estar cerrados para que no haya más de ocho abiertos.

Ejemplo de programa de registro

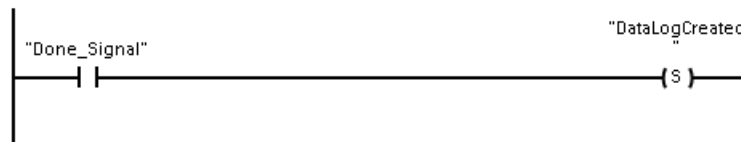
Los nombres del registro, el texto del encabezado y la estructura MyData del ejemplo se crean en un bloque de datos. Las tres variables MyData guardan temporalmente valores de muestreo nuevos. Los valores de muestreo del proceso en las ubicaciones de estos DBs se transfieren a un archivo de registro ejecutando la instrucción DataLogWrite.

My_Datalog_Vars			
	Nombre	Tipo de datos	Valor de arranque
1	Static		
2	MyNEWDatalogName	String	'MyNEWDatalog'
3	MyDatalogName	String	'MyDatalog'
4	MyDatalogID	DWord	0
5	MyDatalogHeaders	String	'Count, Temperature, Pressure'
6	MyData	Struct	
7	MyCount	Int	0
8	MyTemperature	Real	0.0
9	MyPressure	Real	0.0

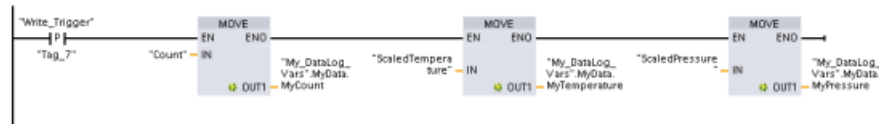
Segmento 1 El flanco ascendente REQ inicia el proceso de creación de registros de datos.



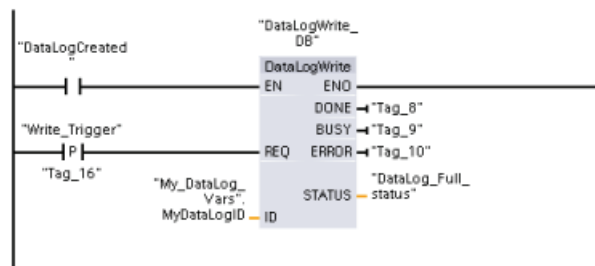
Segmento 2 Capturar la salida DONE de DataLogCreate porque sólo es válida durante un ciclo.



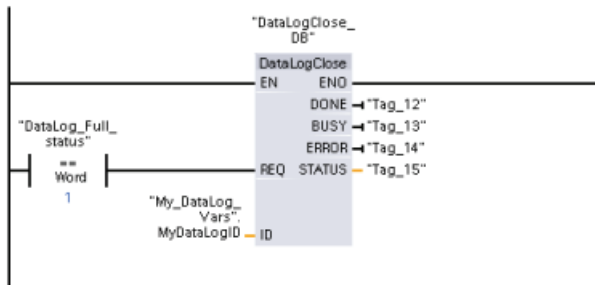
Segmento 3 Una señal de flanco ascendente se dispara cuando deben guardarse datos de proceso nuevos en la estructura MyData.



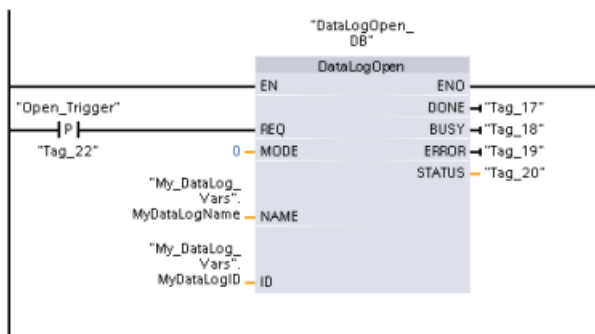
Segmento 4 El estado de la entrada EN se toma como base cuando la operación DataLogCreate ha finalizado. Una operación de creación dura muchos ciclos y debe completarse antes de ejecutar una operación de escritura. La señal de flanco ascendente en la entrada REQ es el evento que dispara una operación de escritura habilitada.



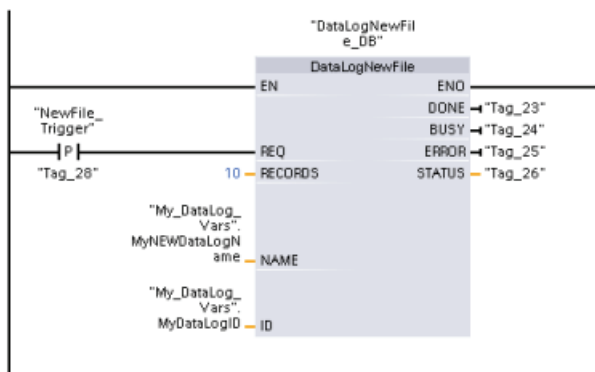
Segmento 5 Cierre el registro una vez se haya escrito el último registro. Después de ejecutar la operación DataLogWrite que escribe el último registro, el estado "lleno" del archivo de registro se indica cuando la salida de ESTADO de DataLogWrite = 1.



Segmento 6 Una entrada REQ DataLogOpen para la señal de flanco ascendente simula la pulsación de un botón por parte del usuario en un HMI que abre un archivo de registro. Si se abre un archivo de registro cuyos registros contienen datos de proceso, la próxima operación DataLogWrite sobrescribirá el registro más antiguo. Quizás prefiera conservar el registro antiguo y crear un registro nuevo, tal como se muestra en el segmento 7.



Segmento 7 El parámetro ID es del tipo IN/OUT. En primer lugar debe proporcionarse el valor ID del registro existente cuya estructura desee copiar. Una vez ha finalizado la operación DataLogNewFile, se escribe un valor ID único para el nuevo registro en la ubicación de referencia de la ID. No se muestra la obtención necesaria del bit DONE = TRUE; consulte los segmentos 1, 2 y 4 para ver un ejemplo de la lógica del bit DONE.



Archivos de registro creados por el programa de ejemplo vistos con el servidor web de la CPU S7-1200



Tabla 8- 140 Ejemplos de archivos .csv descargados vistos con Excel

<p>Dos registros escritos en un archivo de cinco registros como máximo</p>	<table border="1"> <thead> <tr> <th></th> <th>A</th> <th>B</th> <th>C</th> <th>D</th> <th>E</th> <th>F</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Record</td> <td>Date</td> <td>UTC Time</td> <td>Count</td> <td>Temperature</td> <td>Pressure</td> </tr> <tr> <td>2</td> <td>1</td> <td>9/29/2010</td> <td>21:01:46</td> <td>5</td> <td>5.00E+00</td> <td>5.00E+00</td> </tr> <tr> <td>3</td> <td>2</td> <td>9/29/2010</td> <td>21:01:47</td> <td>5</td> <td>5.00E+00</td> <td>5.00E+00</td> </tr> <tr> <td>4</td> <td colspan="6">//END</td> </tr> <tr> <td>5</td> <td colspan="6"></td> </tr> </tbody> </table>		A	B	C	D	E	F	1	Record	Date	UTC Time	Count	Temperature	Pressure	2	1	9/29/2010	21:01:46	5	5.00E+00	5.00E+00	3	2	9/29/2010	21:01:47	5	5.00E+00	5.00E+00	4	//END						5																				
	A	B	C	D	E	F																																																			
1	Record	Date	UTC Time	Count	Temperature	Pressure																																																			
2	1	9/29/2010	21:01:46	5	5.00E+00	5.00E+00																																																			
3	2	9/29/2010	21:01:47	5	5.00E+00	5.00E+00																																																			
4	//END																																																								
5																																																									
<p>Cinco registros en un archivo de registro con un máximo de cinco registros</p>	<table border="1"> <thead> <tr> <th></th> <th>A</th> <th>B</th> <th>C</th> <th>D</th> <th>E</th> <th>F</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Record</td> <td>Date</td> <td>UTC Time</td> <td>Count</td> <td>Temperature</td> <td>Pressure</td> </tr> <tr> <td>2</td> <td>1</td> <td>9/30/2010</td> <td>20:26:56</td> <td>1</td> <td>9.86E+01</td> <td>3.52E+01</td> </tr> <tr> <td>3</td> <td>2</td> <td>9/30/2010</td> <td>20:28:43</td> <td>2</td> <td>1.00E+02</td> <td>3.73E+01</td> </tr> <tr> <td>4</td> <td>3</td> <td>9/30/2010</td> <td>20:29:03</td> <td>3</td> <td>9.99E+01</td> <td>3.68E+01</td> </tr> <tr> <td>5</td> <td>4</td> <td>9/30/2010</td> <td>20:29:21</td> <td>4</td> <td>9.95E+01</td> <td>3.64E+01</td> </tr> <tr> <td>6</td> <td>5</td> <td>9/30/2010</td> <td>20:30:19</td> <td>5</td> <td>9.92E+01</td> <td>3.74E+01</td> </tr> <tr> <td>7</td> <td colspan="6"></td> </tr> </tbody> </table>		A	B	C	D	E	F	1	Record	Date	UTC Time	Count	Temperature	Pressure	2	1	9/30/2010	20:26:56	1	9.86E+01	3.52E+01	3	2	9/30/2010	20:28:43	2	1.00E+02	3.73E+01	4	3	9/30/2010	20:29:03	3	9.99E+01	3.68E+01	5	4	9/30/2010	20:29:21	4	9.95E+01	3.64E+01	6	5	9/30/2010	20:30:19	5	9.92E+01	3.74E+01	7						
	A	B	C	D	E	F																																																			
1	Record	Date	UTC Time	Count	Temperature	Pressure																																																			
2	1	9/30/2010	20:26:56	1	9.86E+01	3.52E+01																																																			
3	2	9/30/2010	20:28:43	2	1.00E+02	3.73E+01																																																			
4	3	9/30/2010	20:29:03	3	9.99E+01	3.68E+01																																																			
5	4	9/30/2010	20:29:21	4	9.95E+01	3.64E+01																																																			
6	5	9/30/2010	20:30:19	5	9.92E+01	3.74E+01																																																			
7																																																									
<p>Después de escribir un registro adicional en el archivo anterior, que ya está lleno, la sexta operación de escritura sobrescribe el registro más antiguo con el sexto registro. Otra operación de escritura sobrescribirá el segundo registro con el séptimo, etc.</p>	<table border="1"> <thead> <tr> <th></th> <th>A</th> <th>B</th> <th>C</th> <th>D</th> <th>E</th> <th>F</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Record</td> <td>Date</td> <td>UTC Time</td> <td>Count</td> <td>Temperature</td> <td>Pressure</td> </tr> <tr> <td>2</td> <td>6</td> <td>9/30/2010</td> <td>20:32:03</td> <td>6</td> <td>9.86E+01</td> <td>3.58E+01</td> </tr> <tr> <td>3</td> <td>2</td> <td>9/30/2010</td> <td>20:28:43</td> <td>2</td> <td>1.00E+02</td> <td>3.73E+01</td> </tr> <tr> <td>4</td> <td>3</td> <td>9/30/2010</td> <td>20:29:03</td> <td>3</td> <td>9.99E+01</td> <td>3.68E+01</td> </tr> <tr> <td>5</td> <td>4</td> <td>9/30/2010</td> <td>20:29:21</td> <td>4</td> <td>9.95E+01</td> <td>3.64E+01</td> </tr> <tr> <td>6</td> <td>5</td> <td>9/30/2010</td> <td>20:30:19</td> <td>5</td> <td>9.92E+01</td> <td>3.74E+01</td> </tr> <tr> <td>7</td> <td colspan="6"></td> </tr> </tbody> </table>		A	B	C	D	E	F	1	Record	Date	UTC Time	Count	Temperature	Pressure	2	6	9/30/2010	20:32:03	6	9.86E+01	3.58E+01	3	2	9/30/2010	20:28:43	2	1.00E+02	3.73E+01	4	3	9/30/2010	20:29:03	3	9.99E+01	3.68E+01	5	4	9/30/2010	20:29:21	4	9.95E+01	3.64E+01	6	5	9/30/2010	20:30:19	5	9.92E+01	3.74E+01	7						
	A	B	C	D	E	F																																																			
1	Record	Date	UTC Time	Count	Temperature	Pressure																																																			
2	6	9/30/2010	20:32:03	6	9.86E+01	3.58E+01																																																			
3	2	9/30/2010	20:28:43	2	1.00E+02	3.73E+01																																																			
4	3	9/30/2010	20:29:03	3	9.99E+01	3.68E+01																																																			
5	4	9/30/2010	20:29:21	4	9.95E+01	3.64E+01																																																			
6	5	9/30/2010	20:30:19	5	9.92E+01	3.74E+01																																																			
7																																																									

8.8 Control de bloques de datos

8.8.1 READ_DBL, WRIT_DBL (Leer de o escribir en un bloque de datos de la memoria de carga)

Tabla 8- 141 Instrucciones READ_DBL y WRIT_DBL

KOP / FUP		Descripción
	<pre> READ_DBL (req:=_bool_in_, srcblk:=_variant_in_, busy=>_bool_out_, dstblk=>_variant_out_); </pre>	<p>Copia valores iniciales del DB o partes de ellos de la memoria de carga a un DB de destino en la memoria de trabajo.</p> <p>El contenido de la memoria de carga no cambia durante el proceso de copia.</p>
	<pre> WRIT_DBL (req:=_bool_in_, srcblk:=_variant_in_, busy=>_bool_out_, dstblk=>_variant_out_); </pre>	<p>Copia valores de DB actuales o partes de ellos de la memoria de trabajo a un DB de destino en la memoria de carga.</p> <p>El contenido de la memoria de trabajo no cambia durante el proceso de copia.</p>

Tabla 8- 142 Tipos de datos para los parámetros

Parámetro y tipo	Tipo de datos	Descripción
REQ IN	BOOL	Una señal high inicia la operación si BUSY = 0.
SRCBLK IN	VARIANT	READ_DBL: Puntero al bloque de datos fuente en la memoria de carga WRIT_DBL: Puntero al bloque de datos fuente en la memoria de trabajo
RET_VAL OUT	INT	Código de condición de ejecución
BUSY OUT	BOOL	BUSY = 1 indica que el proceso de lectura/escritura no se ha completado.
DSTBLK OUT	VARIANT	READ_DBL: Puntero al bloque de datos de destino en la memoria de trabajo WRIT_DBL: Puntero al bloque de datos de destino en la memoria de carga

Típicamente, un DB se almacena tanto en memoria de carga (flash) como en memoria de trabajo (RAM). Los valores de arranque (valores iniciales) siempre se almacenan en memoria de carga, y los valores actuales siempre se almacenan en memoria de trabajo. READ_DBL se puede utilizar para copiar un conjunto de valores de arranque de la memoria de carga a los valores actuales de un DB en memoria de trabajo que se referencia mediante el programa. WRIT_DBL se puede utilizar para actualizar los valores de arranque guardados en la memoria de carga interna o en una Memory Card a partir de los valores actuales de la memoria de trabajo.

Nota**Evite excesivas operaciones de escritura en memoria flash (WRIT_DBL)**

La instrucción WRIT_DBL realiza operaciones de escritura en una memoria flash (memoria de carga interna o Memory Card). WRIT_DBL debe utilizarse para actualizaciones poco frecuentes según cambia un proceso de producción.

Antes de poder utilizar estas instrucciones es necesario que los bloques de datos utilizados por READ_DBL y WRIT_DBL hayan sido creados previamente por STEP 7. Si el DB de origen se crea como tipo "estándar", el DB de destino también debe tener el tipo "estándar". Si el bloque de datos de origen se crea como tipo "optimizado", el bloque de datos de destino también debe tener el tipo "optimizado".

Si los DB son estándar, se puede especificar un nombre de etiqueta o un valor de P#. El valor de P# permite especificar y copiar cualquier número de elementos del tamaño especificado (Byte, Word o DWord). De este modo, se puede copiar todo un DB o parte de él. Si los DB están optimizados, solamente se puede especificar un nombre de variable; no se puede utilizar el operador P#. Si se especifica un nombre de variable para los DB estándar u optimizados (o para otros tipos de memoria de trabajo), entonces se copia lo que referencia este nombre de variable, sea lo que sea. Puede ser un tipo definido por el usuario, una matriz o un elemento básico. El tipo Struct sólo puede ser utilizado por estas instrucciones si el DB es estándar y no optimizado. Se debe utilizar un tipo definido por el usuario (UDT) si es una estructura en memoria optimizada. Sólo un tipo definido por el usuario garantiza que los "tipos de datos" sean exactamente los mismos para las estructuras de origen y destino.

Nota**Uso de estructuras (tipo de datos Struct) en un DB "optimizado"**

Al usar un tipo de datos Struct con DB "optimizados", primero debe crearse un tipo de datos definido por el usuario (UDT) para la Struct. Después, se configuran los DB de origen y de destino en el UDT. El UDT asegura que los tipos de datos dentro de la Struct sean coherentes para ambos DB.

Para DB "estándar", se usa la Struct sin crear un UDT.

READ_DBL y WRIT_DBL se ejecutan de forma asíncrona respecto al ciclo del programa. El procesamiento se extiende en varias llamadas de READ_DBL y WRIT_DBL. La tarea de transferencia de DB se inicia llamando con REQ = 1 y a continuación controlado las salidas BUSY y RET_VAL para determinar cuándo se ha completado la transferencia de datos y si lo ha hecho correctamente.

Para garantizar la coherencia de datos, no modifique el área de destino durante el procesamiento de READ_DBL o el área de origen durante el procesamiento de WRIT_DBL (es decir, mientras el parámetro BUSY sea TRUE).

Restricciones de los parámetros SRCBLK y DSTBLK:

- Un bloque de datos debe haber sido creado antes de poder ser referenciado.
- La longitud de un puntero VARIANT del tipo BOOL debe ser divisible entre 8.
- La longitud de un puntero VARIANT del tipo STRING debe ser el mismo en los punteros fuente y de destino.

Información de de recetas y de configuración de la máquina

Se pueden utilizar las instrucciones READ_DBL y WRIT_DBL para gestionar información de recetas o de configuración de la máquina. En esencia, esto constituye otro método de obtener datos remanentes para los valores que no cambian con frecuencia, aunque se desee limitar el número de escrituras para impedir que la memoria flash se desgaste prematuramente. Ello permite aumentar de forma eficaz la cantidad de memoria remanente más allá de la soportada para los datos normales remanentes tras el apagado, al menos para los valores que no cambian con frecuencia. La información de recetas o la información de configuración de la máquina de la memoria de trabajo se puede guardar en la memoria de carga mediante la instrucción WRIT_DBL, y dicha información se puede recuperar de la memoria de carga a la memoria de trabajo mediante la instrucción READ_DBL.

Tabla 8- 143 Códigos de condición

RET_VAL (W#16#...)	Descripción
0000	No hay error
0081	Advertencia: Que el área de origen sea menor que la de destino. Los datos fuente se copian al completo con los bytes adicionales en el área de destino sin modificación.
7000	Llamada con REQ = 0: BUSY = 0
7001	Primera llamada con REQ = 1 (en curso): BUSY = 1
7002	Llamada N° (en curso): BUSY = 1
8051	Error de tipo de bloque de datos
8081	El área de origen es mayor que la de destino. El área de destino está llena por completo y los bytes restantes del origen se ignoran.
8251	Error de tipo de bloque de datos de origen
82B1	El bloque de datos de origen falta
82C0	El DB de origen está siendo editado por otra instrucción o función de comunicación.
8551	Error de tipo de bloque de datos de destino
85B1	El bloque de datos de destino falta
85C0	El DB de destino está siendo editado por otra instrucción o función de comunicación.
80C3	Actualmente hay más de 50 instrucciones READ_DBL o 50 WRIT_DBL en cola de ejecución.

8.9 Códigos de error comunes para las instrucciones "Avanzadas"

Tabla 8- 144 Códigos de condición comunes para las instrucciones avanzadas

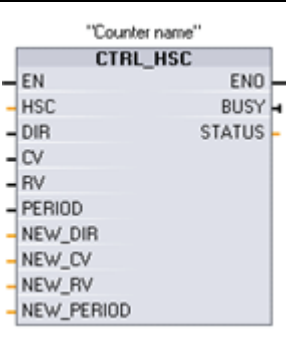
Código de condición (W#16#....) ¹	Descripción
8022	Área demasiado pequeña para la entrada
8023	Área demasiado pequeña para la salida
8024	Área de entrada no válida
8025	Área de salida no válida
8028	Asignación del bit de entrada no válida
8029	Asignación del bit de salida no válida
8030	El área de salida es un DB de sólo lectura.
803A	El DB no existe.

¹ Cuando se produce uno de estos errores al ejecutar un bloque lógico, la CPU pasa al estado operativo STOP a no ser que se utilicen las instrucciones GetError o GetErrorID dentro de dicho bloque para crear una respuesta programada al error.

Instrucciones tecnológicas

9.1 Contador rápido

Tabla 9- 1 Instrucción CTRL_HSC

KOP / FUP	SCL	Descripción
	<pre>"CTRL_HSC_0_DB" (hsc:=_hw_hsc_in_, dir:=_bool_in_, cv:=_bool_in_, rv:=_bool_in_, period:=_bool_in_, new_dir:=_int_in_, new_cv:=_int_in_, new_rv:=_dint_in_, new_period:=_int_in_, busy:=_bool_out_, status:=_word_out_);</pre>	<p>Toda instrucción CTRL_HSC utiliza una estructura almacenada en un DB para conservar los datos. El DB se asigna cuando la instrucción CTRL_HSC se coloca en el editor.</p>

- STEP 7 crea el DB automáticamente al insertar la instrucción.
- En el ejemplo SCL, "CTRL_HSC_0_DB" es el nombre del DB de instancia.

Tabla 9- 2 Tipos de datos para los parámetros

Parámetro y tipo	Tipo de datos	Descripción	
HSC	IN	HW_HSC	Identificador del HSC
DIR ^{1, 2}	IN	Bool	1 = solicitar nuevo sentido de conteo
CV ¹	IN	Bool	1 = solicitar activación del nuevo valor del contador
RV ¹	IN	Bool	1 = solicitar activación del nuevo valor de referencia
PERIOD ¹	IN	Bool	1 = solicitar activación del nuevo periodo (sólo para el modo de medición de frecuencia)
NEW_DIR	IN	Int	Nuevo sentido: 1= hacia delante, -1= hacia atrás
NEW_CV	IN	DInt	Nuevo valor del contador
NEW_RV	IN	DInt	Nuevo valor de referencia
NEW_PERIOD	IN	Int	Nuevo periodo en segundos: 0.01, 0.1 ó 1 (sólo para el modo de medición de frecuencia)
BUSY ³	OUT	Bool	La función está ocupada
STATUS	OUT	Word	Código de condición de ejecución

- Si no se solicita la actualización de un parámetro, se ignoran los valores de entrada correspondientes.
- El parámetro DIR es válido sólo si el sentido de conteo se ajusta a "Programa de usuario (control interno de sentido)". La forma de utilizar este parámetro se determina en la configuración de dispositivos HSC.
- En un HSC en la CPU o en la Signal Board, el parámetro BUSY siempre está puesto a 0.

9.1 Contador rápido

Configure los parámetros de cada HSC en la configuración de dispositivos de la CPU: Modo de contaje, conexiones de E/S, asignación de alarmas y funcionamiento como contador rápido o dispositivo para medir la frecuencia de pulsos.

Algunos de los parámetros del HSC se pueden modificar mediante el programa de usuario para ofrecer un control de programa del proceso de contaje:

- Ajustar el sentido de contaje al valor NEW_DIR
- Ajustar el valor de contaje actual al valor NEW_CV
- Ajustar el valor de referencia al valor NEW_RV
- Ajustar el valor Periodo (para el modo de medición de frecuencia) al valor NEW_PERIOD

Si las siguientes marcas booleanas están puestas a 1 cuando se ejecuta la instrucción CTRL_HSC, el valor NEW_xxx correspondiente se carga en el contador. Las peticiones múltiples (varias marcas se activan simultáneamente) se procesan en una sola ejecución de la instrucción CTRL_HSC.

- DIR = 1 es una petición para cargar un valor NEW_DIR, 0 = ningún cambio
- CV = 1 es una petición para cargar un valor NEW_CV, 0 = ningún cambio
- RV = 1 es una petición para cargar un valor NEW_RV, 0 = ningún cambio
- PERIOD = 1 es una petición para cargar un valor NEW_PERIOD, 0 = ningún cambio

Generalmente, la instrucción CTRL_HSC se inserta en un OB de alarma de proceso que se ejecuta cuando se dispara el evento de alarma de proceso del contador. Por ejemplo, si un evento CV=RV dispara la alarma del contador, un OB de alarma de proceso ejecuta la instrucción CTRL_HSC y puede cambiar el valor de referencia cargando un valor NEW_RV.

El valor de contaje actual no está disponible en los parámetros de CTRL_HSC. La dirección de la memoria imagen de proceso que almacena el valor de contaje actual se asigna al configurar el hardware del contador rápido. Se puede emplear la lógica del programa para leer directamente el valor de contaje. El valor devuelto al programa será un contaje correcto para el instante en el que se leyó el contador. El contador continuará contando eventos rápidos. Por tanto, el valor de contaje real puede cambiar antes de que el programa finalice un proceso utilizando un valor de contaje antiguo.

Códigos de condición: En caso de error, ENO se pone a 0 y la salida STATUS devuelve un código de condición.

Tabla 9- 3 Valores STATUS (W#16#)

STATUS	Descripción
0	No hay error
80A1	El identificador HSC no direcciona un HSC
80B1	Valor de NEW_DIR no permitido
80B2	Valor de NEW_CV no permitido
80B3	Valor de NEW_RV no permitido
80B4	Valor de NEW_PERIOD no permitido
80C0	Acceso múltiple al contador rápido
80D0	Contador rápido (HSC) no habilitado en la configuración hardware de la CPU

9.1.1 Funcionamiento del contador rápido

El contador rápido (HSC) realiza el conteo de eventos que se producen con mayor rapidez que la frecuencia de ejecución del OB. Si los eventos que se deben contar se producen con la frecuencia de ejecución del OB, utilice las instrucciones de conteo CTU, CTD o CTUD. Si el evento se produce con mayor rapidez que la frecuencia de ejecución del OB, utilice el HSC. La instrucción CTRL_HSC permite al programa de usuario cambiar algunos de los parámetros del HSC.

Ejemplo: Se puede emplear el HSC como entrada para un encoder rotativo. El encoder rotativo ofrece un número determinado de valores de conteo por revolución, así como un impulso de reset que ocurre una vez por revolución. El o los relojes y el impulso de reset del encoder suministran las entradas para el contador rápido.

El primero de los valores predeterminados se carga en el HSC y las salidas se activan durante el periodo en el que el conteo actual es menor que el valor predeterminado. El HSC pone a disposición una alarma cuando el conteo actual es igual al valor predeterminado, al ocurrir un reset y también al producirse un cambio de sentido.

Cuando el conteo actual es igual al valor predeterminado y se presenta un evento de alarma, se carga un nuevo valor predeterminado y se activa el siguiente estado para las salidas. Cuando ocurre el evento de alarma de reset, se activan el primer valor predeterminado y los primeros estados de las salidas y se repite el ciclo.

Puesto que las alarmas ocurren con una frecuencia mucho menor que la frecuencia de conteo del HSC, es posible implementar un control preciso de las operaciones rápidas con un impacto relativamente bajo en el ciclo de la CPU. El método de asociar alarmas permite cargar cada valor predeterminado nuevo en un subprograma por separado, lo que simplifica el control del estado. (Alternativamente, todos los eventos de alarma se pueden ejecutar en un solo subprograma.)

Tabla 9- 4 Frecuencia máxima (KHz)

HSC		Fase simple	Cuadratura AB y dos fases
HSC1	CPU	100 KHz	80 KHz
	SB rápidas	200 KHz	160 KHz
	SB	30 KHz	20 KHz
HSC2	CPU	100 KHz	80 KHz
	SB rápidas	200 KHz	160 KHz
	SB	30 KHz	20 KHz
HSC3	CPU	100 KHz	80 KHz
HSC4	CPU	30 KHz	20 KHz
HSC5	CPU	30 KHz	20 KHz
	SB rápidas	200 KHz	160 KHz
	SB	30 KHz	20 KHz
HSC6	CPU	30 KHz	20 KHz
	SB rápidas	200 KHz	160 KHz
	SB	30 KHz	20 KHz

Seleccionar las funciones del HSC

Todos los HSCs funcionan de la misma manera en el mismo modo de operación del contador. Hay cuatro tipos básicos de HSCs, a saber:

- Contador de fase simple con control interno del sentido de contaje
- Contador de fase simple con control externo del sentido de contaje
- Contador de dos fases con 2 entradas de reloj
- Contador A/B

Todo tipo de HSC puede utilizarse con o sin entrada de reset. Cuando se activa la entrada de reset (con algunas restricciones, v. la tabla siguiente), el valor actual se borra y se mantiene borrado hasta que se desactive la entrada de reset.

- Función de frecuencia: Algunos modos del HSC (Modo de contaje) permiten configurarlo de manera que notifique la frecuencia en vez del contaje actual de impulsos. Hay tres periodos de medición de frecuencia disponibles: 0,01, 0,1 ó 1,0 segundos.

El periodo de medición de frecuencia determina cada cuánto calcula y notifica el HSC un nuevo valor de frecuencia. La frecuencia notificada es un valor promedio determinado por el número total de contajes en el último periodo de medición. Si la frecuencia cambia rápidamente, el valor notificado será el valor medio entre la frecuencia más alta y más baja registrada durante el periodo de medición. La frecuencia se indica siempre en hertzios (impulsos por segundo), independientemente del ajuste del periodo de medición de frecuencia.

- Modos y entradas de contador: La tabla siguiente muestra las entradas utilizadas para las funciones de reloj, control de sentido y reset asociadas al HSC.

Una misma entrada no se puede utilizar para dos funciones diferentes. Sin embargo, cualquier entrada que no se esté utilizando en el modo actual del HSC se puede usar para otro fin. Por ejemplo, si el HSC1 está en un modo que utiliza entradas integradas, pero que no usa el reset externo (I0.3), la entrada I0.3 puede utilizarse para alarmas de flanco o para el HSC2.

Tabla 9- 5 Modos de contaje del HSC

Tipo	Entrada 1	Entrada 2	Entrada 3	Función
Contador de fase simple con control interno del sentido de contaje	Reloj	(Opcional: sentido)	-	Contaje o frecuencia
			Desactivar	Contaje
Contador de fase simple con control externo del sentido de contaje	Reloj	Sentido	-	Contaje o frecuencia
			Desactivar	Contaje
Contador de dos fases con 2 entradas de reloj	Reloj adelante	Reloj atrás	-	Contaje o frecuencia
			Desactivar	Contaje
Contador en cuadratura (fases A/B)	Fase A	Fase B	-	Contaje o frecuencia
			Reset ¹	Contaje

¹ Para un encoder: fase Z, inicio

Direcciones de entrada del HSC

Nota

Las E/S digitales utilizadas por los contadores rápidos se asignan durante la configuración de dispositivos. Si se asignan direcciones de E/S digitales a dichos dispositivos, los valores de las direcciones de E/S asignadas no podrán modificarse utilizando la función de forzado permanente de una tabla de observación.

Al configurar la CPU, se puede habilitar y configurar cada HSC. La CPU asigna automáticamente las direcciones de entrada de cada HSC según su configuración. (Algunos HSCs permiten seleccionar el uso de las entradas integradas de la CPU o de las entradas de una SB.)

ATENCIÓN

Como se indica en las tablas siguientes, las asignaciones predeterminadas de las señales opcionales para los diferentes HSCs se solapan. Por ejemplo, la inicialización externa opcional del HSC 1 usa la misma entrada que una de las entradas del HSC 2.

Asegúrese siempre de haber configurado los HSCs de forma que **no** haya dos HSCs usando una entrada.

La tabla siguiente muestra las asignaciones de entradas de HSC tanto para las E/S integradas de la CPU 1211C como para una SB. (Si la SB sólo tiene 2 entradas, sólo están disponibles las entradas 4.0 y 4.1).

- Para fase simple: C es la entrada de reloj, [d] es la entrada de sentido opcional y [R] es una entrada de inicialización externa opcional. (La inicialización sólo está disponible para el modo de "contaje").
- Para dos fases: CU es la entrada de reloj adelante, CD es la entrada de reloj atrás, y [R] es una entrada de inicialización externa opcional. (La inicialización sólo está disponible para el modo de "contaje").
- Para cuadratura de fases AB: A es la entrada de reloj A, B es la entrada de reloj B, y [R] es una entrada de inicialización externa opcional. (La inicialización sólo está disponible para el modo de "contaje").

Tabla 9- 6 Asignaciones de entradas de HSC para la CPU 1211C

HSC		Entrada integrada de CPU (0.x)						Entrada de SB (predeterminado 4.x) ³			
		0	1	2	3	4	5	0	1	2	3
HSC 1 ¹	1 fase	C	[d]		[R]			C	[d]		[R]
	2 fases	CU	CD		[R]			CU	CD		[R]
	Fase AB	A	B		[R]			A	B		[R]
HSC 2 ¹	1 fase		[R]	C	[d]				[R]	C	[d]
	2 fases		[R]	CU	CD				[R]	CU	CD
	Fase AB		[R]	A	B				[R]	A	B
HSC 3	1 fase					C	[d]				

9.1 Contador rápido

HSC		Entrada integrada de CPU (0.x)						Entrada de SB (predeterminado 4.x) ³			
		0	1	2	3	4	5	0	1	2	3
	2 fases					CU	CD				
	Fase AB					A	B				
HSC 5 ²	1 fase							C	[d]		[R]
	2 fases							CU	CD		[R]
	Fase AB							A	B		[R]
HSC 6 ²	1 fase								[R]	C	[d]
	2 fases								[R]	CU	CD
	Fase AB								[R]	A	B

- ¹ HSC 1 y HSC 2 se pueden configurar tanto para las entradas integradas como para una SB.
- ² HSC 5 y HSC 6 sólo están disponibles con una SB. HSC 6 sólo está disponible con una SB de cuatro entradas.
- ³ Una SB con sólo dos entradas únicamente ofrece las entradas 4.0 y 4.1.

La tabla siguiente muestra las asignaciones de entradas de HSC tanto para las E/S integradas de la CPU 1212C como para una SB. (Si la SB sólo tiene 2 entradas, sólo están disponibles las entradas 4.0 y 4.1).

- Para fase simple: C es la entrada de reloj, [d] es la entrada de sentido opcional y [R] es una entrada de inicialización externa opcional. (La inicialización sólo está disponible para el modo de "contaje").
- Para dos fases: CU es la entrada de reloj adelante, CD es la entrada de reloj atrás, y [R] es una entrada de inicialización externa opcional. (La inicialización sólo está disponible para el modo de "contaje").
- Para cuadratura de fases AB: A es la entrada de reloj A, B es la entrada de reloj B, y [R] es una entrada de inicialización externa opcional. (La inicialización sólo está disponible para el modo de "contaje").

Tabla 9-7 Asignaciones de entradas de HSC para la CPU 1212C

HSC		Entrada integrada de CPU (0.x)								Entrada de SB (4.x) ³			
		0	1	2	3	4	5	6	7	0	1	2	3
HSC 1 ¹	1 fase	C	[d]		[R]					C	[d]		[R]
	2 fases	CU	CD		[R]					CU	CD		[R]
	Fase AB	A	B		[R]					A	B		[R]
HSC 2 ¹	1 fase		[R]	C	[d]						[R]	C	[d]
	2 fases		[R]	CU	CD						[R]	CU	CD
	Fase AB		[R]	A	B						[R]	A	B
HSC 3	1 fase					C	[d]		[R]				
	2 fases					CU	CD		[R]				
	Fase AB					A	B		[R]				
HSC 4	1 fase						[R]	C	[d]				
	2 fases						[R]	CU	CD				
	Fase AB						[R]	A	B				

HSC		Entrada integrada de CPU (0.x)							Entrada de SB (4.x) ³				
		0	1	2	3	4	5	6	7	0	1	2	3
HSC 5 ²	1 fase									C	[d]		[R]
	2 fases									CU	CD		[R]
	Fase AB									A	B		[R]
HSC 6 ²	1 fase										[R]	C	[d]
	2 fases										[R]	CU	CD
	Fase AB										[R]	A	B

¹ HSC 1 y HSC 2 se pueden configurar tanto para las entradas integradas como para una SB.

² HSC 5 y HSC 6 sólo están disponibles con una SB. HSC 6 sólo está disponible con una SB de cuatro entradas.

³ Una SB con sólo dos entradas únicamente ofrece las entradas 4.0 y 4.1.

Las dos tablas siguientes muestran las asignaciones de entradas de HSC tanto para las E/S integradas de la CPU 1214C como para una SB opcional, en caso de haberla.

- Para fase simple: C es la entrada de reloj, [d] es la entrada de sentido opcional y [R] es una entrada de inicialización externa opcional. (La inicialización sólo está disponible para el modo de "contaje").
- Para dos fases: CU es la entrada de reloj adelante, CD es la entrada de reloj atrás, y [R] es una entrada de inicialización externa opcional. (La inicialización sólo está disponible para el modo de "contaje").
- Para cuadratura de fases AB: A es la entrada de reloj A, B es la entrada de reloj B, y [R] es una entrada de inicialización externa opcional. (La inicialización sólo está disponible para el modo de "contaje").

Tabla 9- 8 Asignaciones de entradas HSC para la CPU 1214C y la CPU 1215C (sólo entradas integradas)

HSC		Entrada digital 0 (predeterminado: 0.x)							Entrada digital 1 (predeterminado: 1.x)						
		0	1	2	3	4	5	6	7	0	1	2	3	4	5
HSC 1 ¹	1 fase	C	[d]		[R]										
	2 fases	CU	CD		[R]										
	Fase AB	A	B		[R]										
HSC 2 ¹	1 fase		[R]	C	[d]										
	2 fases		[R]	CU	CD										
	Fase AB		[R]	A	B										
HSC 3	1 fase					C	[d]		[R]						
	2 fases					CU	CD		[R]						
	Fase AB					A	B		[R]						
HSC 4	1 fase						[R]	C	[d]						
	2 fases						[R]	CU	CD						
	Fase AB						[R]	A	B						
HSC 5 ¹	1 fase									C	[d]	[R]			
	2 fases									CU	CD	[R]			
	Fase AB									A	B	[R]			

9.1 Contador rápido

HSC		Entrada digital 0 (predeterminado: 0.x)							Entrada digital 1 (predeterminado: 1.x)						
		0	1	2	3	4	5	6	7	0	1	2	3	4	5
HSC 6 ¹	1 fase												C	[d]	[R]
	2 fases												CU	CD	[R]
	Fase AB												A	B	[R]

¹ HSC 1, HSC 2, HSC 5 y HSC 6 se pueden configurar tanto para las entradas integradas como para una SB.

Tabla 9- 9 Asignaciones de entradas de HSC para SBs

HSC ¹		Entradas de SB (predeterminado: 4.x) ²			
		0	1	2	3
HSC 1	1 fase	C	[d]		[R]
	2 fases	CU	CD		[R]
	Fase AB	A	B		[R]
HSC 2	1 fase		[R]	C	[d]
	2 fases		[R]	CU	CD
	Fase AB		[R]	A	B
HSC 5	1 fase	C	[d]		[R]
	2 fases	CU	CD		[R]
	Fase AB	A	B		[R]
HSC 6	1 fase		[R]	C	[d]
	2 fases		[R]	CU	CD
	Fase AB		[R]	A	B

¹ Para CPU 1214C: HSC 1, HSC 2, HSC 5 y HSC 6 se pueden configurar tanto para las entradas integradas como para una SB.

² Una SB con sólo dos entradas únicamente ofrece las entradas 4.0 y 4.1.

Acceso al valor actual del HSC

Nota

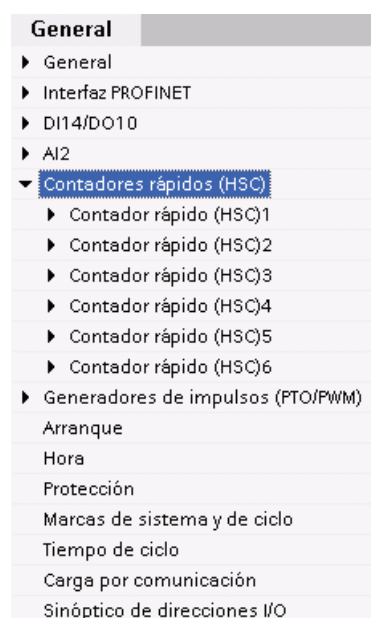
Al activar un generador de impulsos para emplearlo como PTO, se asigna un HSC correspondiente a dicha PTO. HSC1 se asigna al PTO1, mientras que HSC2 se asigna al PTO2. El HSC asignado pertenece por completo al canal del PTO y se desactiva la salida habitual del HSC. El valor del HSC sólo se usa para funciones internas. No se puede controlar el valor actual (por ejemplo, en ID1000) cuando se producen impulsos.

La CPU almacena el valor actual de cada HSC en una dirección de entrada (I). La tabla siguiente muestra las direcciones predeterminadas asignadas al valor actual de cada HSC. La dirección I del valor actual se puede cambiar modificando las propiedades de la CPU en la "Configuración de dispositivos".

Tabla 9- 10 Valor actual de HSC

HSC	Tipo de datos	Dirección predeterminada
HSC1	DInt	ID1000
HSC2	DInt	ID1004
HSC3	DInt	ID1008
HSC4	DInt	ID1012
HSC5	DInt	ID1016
HSC6	DInt	ID1020

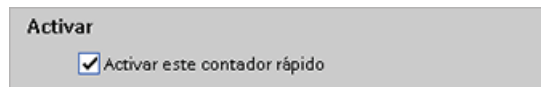
9.1.2 Configuración del HSC



La CPU permite configurar hasta 6 contadores rápidos. Las "Propiedades" de la CPU se editan para configurar los parámetros de cada HSC.

Utilice la instrucción CTRL_HSC en el programa de usuario para controlar el funcionamiento del HSC.

Active el HSC específico seleccionando la opción "Habilitar" de dicho HSC.



Nota

Al habilitar el contador rápido y seleccionar entradas para el mismo, los filtros de entrada para esas entradas se ajustan a 800 ns. Cada entrada tiene una sola configuración de filtro, aplicable a todos los usos: entradas de proceso, interrupciones, captura de impulsos y entradas de HSC.

⚠️ ADVERTENCIA

Si el tiempo de filtro para un canal de entrada digital se reajusta, puede que sea necesario presentar un nuevo valor de entrada de nivel "0" durante un tiempo acumulado de 20 ms para que el filtro esté totalmente operativo ante nuevas entradas. Durante este tiempo, puede que no se detecten o no se cuenten los eventos de pulsación breve "0" cuya duración sea inferior a 20 ms.

Este cambio de los tiempos de filtro puede originar un funcionamiento inesperado de los equipos o del proceso, lo que puede causar la muerte o lesiones graves al personal y/o daños a los equipos.

Para asegurar que un tiempo de filtro nuevo tenga efecto inmediato, desconecte y vuelva a conectar la CPU.

Tras habilitar el HSC se deben configurar los demás parámetros, tales como la función del contador, los valores iniciales, las opciones de reset y los eventos de alarma.

The screenshot shows a configuration window for HSC with the following settings:

- Modo: Contaje
- Fase servicio: Monofásica
- Origen señal: Entrada de CPU integrada
- Sentido de conteo dado por: Programa de usuario (control i)
- Sentido de conteo inicial: Incrementar contador
- Período de medición de frecuencia: +/- sec

Para obtener información sobre la configuración del HSC, consulte el apartado Configuración de la CPU (Página 127).

9.2 Control PID

STEP 7 ofrece las siguientes instrucciones PID para la CPU S7-1200:

- La instrucción PID_Compact se utiliza para controlar procesos técnicos con variables continuas de entrada y salida.
- La instrucción PID_3Step se utiliza para controlar dispositivos accionados por motor, como válvulas que requieren señales discretas para las acciones de apertura y cierre.

Nota

Los cambios realizados en la configuración PID y las descargas en modo RUN no son efectivos hasta que la CPU no pasa del modo STOP a RUN.

Ambas instrucciones PID (PID_3Step y PID_Compact) pueden calcular las acciones P, I y D durante el arranque (si se han configurado para "optimización inicial"). También es posible configurar la instrucción para la "optimización fina" con el fin de optimizar los parámetros. No es necesario especificar los parámetros manualmente.

Nota

Ejecute la instrucción PID en intervalos regulares del tiempo de muestreo (preferentemente en un OB cíclico).

Puesto que el lazo PID necesita cierto tiempo para responder a los cambios del valor de control, no debe calcularse el valor de salida en cada ciclo. No ejecute la instrucción PID en el OB de ciclo del programa principal (p. ej. OB 1).

El tiempo de muestreo del algoritmo PID representa el intervalo entre dos cálculos del valor de salida (valor de control). El valor de salida se calcula durante el autoajuste y se redondea a un múltiplo del tiempo de ciclo. Las demás funciones de la instrucción PID se ejecutan en cada llamada.

Algoritmo PID

El regulador PID (Proporcional/Integral/Derivativo) mide el intervalo de tiempo entre dos llamadas y evalúa el resultado para controlar el tiempo de muestreo. En cada cambio de modo y en el primer arranque se genera un valor medio del tiempo de muestreo. Dicho valor se utiliza como referencia para la función de vigilancia y para realizar cálculos. La vigilancia incluye el tiempo de medición actual entre dos llamadas y el valor medio del tiempo de muestreo definido del regulador.

El valor de salida del regulador PID está formado por tres acciones:

- P (proporcional): cuando se calcula con la acción "P", el valor de salida es proporcional a la diferencia entre la consigna y el valor de proceso (valor de entrada).
- I (integral): cuando se calcula con la acción "I", el valor de salida aumenta en proporción a la duración de la diferencia entre la consigna y el valor de proceso (valor de entrada) para corregir la diferencia al final.
- D (derivativo): cuando se calcula con la acción "D", el valor de salida aumenta como una función de la tasa de incremento de cambio de la diferencia entre la consigna y el valor de proceso (valor de entrada). El valor de salida se corrige a la consigna lo más rápido posible.

El regulador PID utiliza la siguiente fórmula para calcular el valor de salida de la instrucción PID_Compact.

$$y = K_p \left[(b \cdot w - x) + \frac{1}{T_i \cdot s} (w - x) + \frac{T_d \cdot s}{a \cdot T_d \cdot s + 1} (c \cdot w - x) \right]$$

y	Valor de salida	x	Valor de proceso
w	Consigna	s	Operador laplaciano
K _p	Ganancia proporcional (acción P)	a	Coefficiente de retardo derivativo (acción D)
T _i	Tiempo de acción integral (acción I)	b	Ponderación de acción proporcional (acción P)
T _D	Tiempo de acción derivativa (acción D)	c	Ponderación de la acción derivativa (acción D)

El regulador PID utiliza la siguiente fórmula para calcular el valor de salida de la instrucción PID_3Step.

$$\Delta y = K_p \cdot s \cdot \left[(b \cdot w - x) + \frac{1}{T_i \cdot s} (w - x) + \frac{T_d \cdot s}{a \cdot T_d \cdot s + 1} (c \cdot w - x) \right]$$

y	Valor de salida	x	Valor de proceso
w	Consigna	s	Operador laplaciano
K _p	Ganancia proporcional (acción P)	a	Coefficiente de retardo derivativo (acción D)
T _i	Tiempo de acción integral (acción I)	b	Ponderación de acción proporcional (acción P)
T _D	Tiempo de acción derivativa (acción D)	c	Ponderación de acción derivativa (acción D)

9.2.1 Insertar la instrucción PID y un objeto tecnológico

STEP 7 ofrece dos instrucciones de control PID:

- La instrucción PID_Compact y su objeto tecnológico ofrecen un regulador PID universal con optimización. El objeto tecnológico contiene todos los ajustes para el lazo de regulación.
- La instrucción PID_3Step y su objeto tecnológico ofrecen un regulador PID con ajustes específicos para válvulas accionadas por motor. El objeto tecnológico contiene todos los ajustes para el lazo de regulación. El regulador PID_3Step dispone de dos salidas booleanas adicionales.

Después de crear el objeto tecnológico, es necesario configurar los parámetros (Página 381). También deben ajustarse los parámetros de optimización ("optimización inicial" durante el arranque u "optimización fina" manual) para poner el regulador PID en servicio (Página 383).

Tabla 9- 11 Insertar la instrucción PID y el objeto tecnológico

Cuando se inserta una instrucción PID en el programa de usuario, STEP 7 crea automáticamente un objeto tecnológico y un DB de instancia para dicha instrucción. El DB de instancia contiene todos los parámetros que se utilizan para la instrucción PID. Cada instrucción PID debe tener su propio DB de instancia unívoco para funcionar correctamente.

Después de insertar la instrucción PID y crear el objeto tecnológico y el DB de instancia, se configuran los parámetros del objeto tecnológico (Página 381).

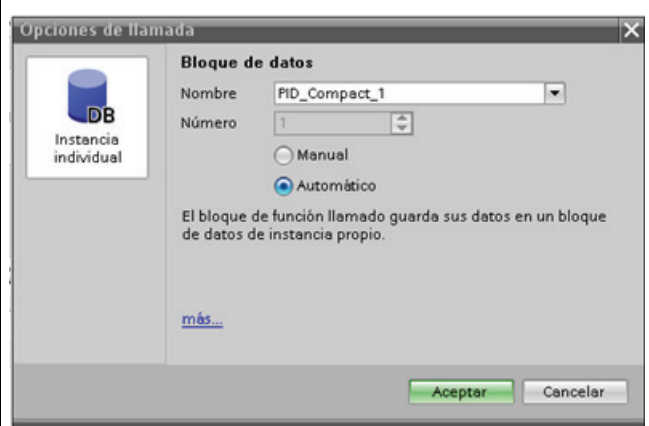
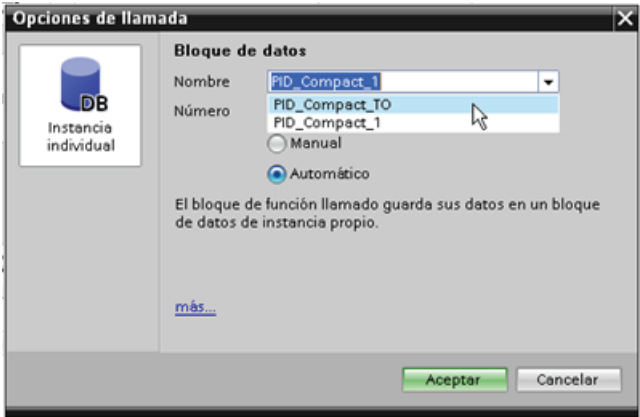
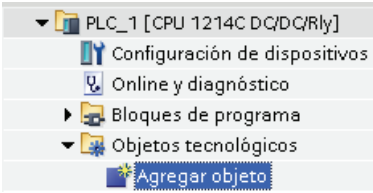
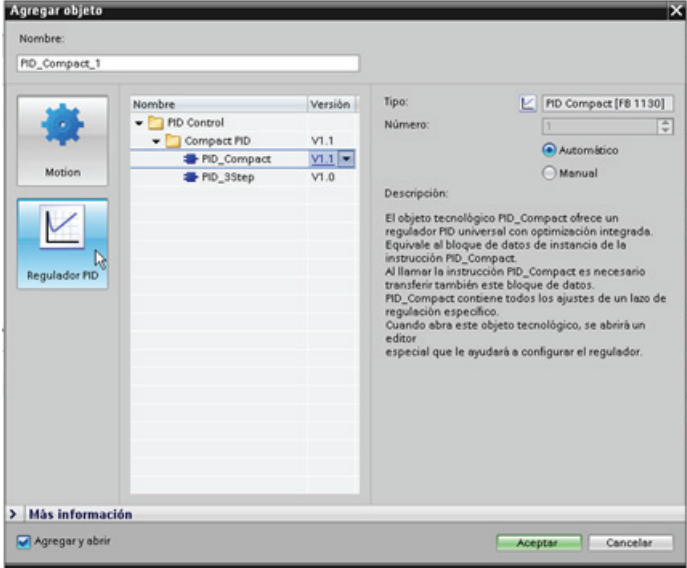


Tabla 9- 12 (Opcional) Crear un objeto tecnológico desde el árbol del proyecto

<p>También es posible crear objetos tecnológicos para el proyecto antes de insertar la instrucción PID. Si se crea el objeto tecnológico antes de insertar una instrucción PID en el programa de usuario, puede seleccionarse dicho objeto tecnológico al insertar la instrucción PID.</p>	
<p>Para crear un objeto tecnológico, haga doble clic en el icono "Agregar objeto" del árbol del proyecto.</p>	
<p>Haga clic en el icono "Regulación" y seleccione el objeto tecnológico para el tipo de regulador PID (PID_Compact o PID_3Step). Es posible crear un nombre opcional para el objeto tecnológico. Haga clic en "Aceptar" para crear el objeto tecnológico.</p>	

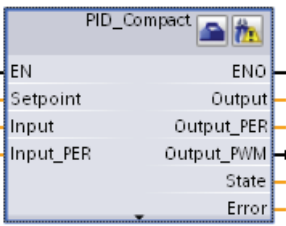
9.2.2 Instrucción PID_Compact

El regulador PID utiliza la siguiente fórmula para calcular el valor de salida de la instrucción PID_Compact.

$$y = K_p \left[(b \cdot w - x) + \frac{1}{T_i \cdot s} (w - x) + \frac{T_d \cdot s}{a \cdot T_d \cdot s + 1} (c \cdot w - x) \right]$$

y	Valor de salida	x	Valor de proceso
w	Consigna	s	Operador laplaciano
K _p	Ganancia proporcional (acción P)	a	Coefficiente de retardo derivativo (acción D)
T _i	Tiempo de acción integral (acción I)	b	Ponderación de acción proporcional (acción P)
T _d	Tiempo de acción derivativa (acción D)	c	Ponderación de acción derivativa (acción D)

Tabla 9- 13 Instrucción PID_Compact

KOP / FUP	SCL	Descripción
	<pre>"PID_Compact_1" (Setpoint:= _real_in_, Input:= _real_in_, Input_PER:= _word_in_, ManualEnable:= _bool_in_, ManualValue:= _real_in_, Reset:= _bool_in_, ScaledInput=> _real_out_, Output=> _real_out_, Output_PER=> _word_out_, Output_PWM=> _bool_out_, SetpointLimit_H=> _bool_out_, SetpointLimit_L=> _bool_out_, InputWarning_H=> _bool_out_, InputWarning_L=> _bool_out_, State=> _int_out_, Error=> _dword_out_);</pre>	<p>PID_Compact ofrece un regulador PID con autoajuste para modo automático y modo manual. PID_Compact es un regulador PIDT1 con anti-windup y ponderación de las acciones P y D.</p>

- STEP 7 crea automáticamente el objeto tecnológico y el DB de instancia al insertar la instrucción. El DB de instancia contiene los parámetros del objeto tecnológico.
- En el ejemplo SCL, "PID_Compact_1" es el nombre del DB de instancia.

Tabla 9- 14 Tipos de datos para los parámetros

Parámetro y tipo	Tipo de datos	Descripción
Setpoint	IN	Real Consigna del regulador PID en modo automático. Valor predeterminado: 0.0
Input	IN	Real Valor de proceso. Valor predeterminado: 0.0 También debe ponerse sPid_Cmpt.b_Input_PER_On = FALSE.

Parámetro y tipo		Tipo de datos	Descripción
Input_PER	IN	Word	Valor de proceso analógico (opcional). Valor predeterminado: W#16#0 También debe ponerse sPid_Cmpt.b_Input_PER_On = TRUE.
ManualEnable	IN	Bool	Activa o desactiva el modo de operación manual. Valor predeterminado: FALSE: <ul style="list-style-type: none"> • PID_Compact V1.0 y V1.2: Cuando la CPU pasa a RUN, si ManualEnable = TRUE, PID_Compact inicia en modo manual. No es necesario que haya una transición FALSE a TRUE para poner el PID_Compact en modo manual. • PID_Compact V1.1: Cuando la CPU pasa a RUN y ManualEnable = TRUE, el PID Compact inicia en el último estado. Debe haber una transición de TRUE a FALSE y a TRUE para poner el PID_Compact en modo manual.
ManualValue	IN	Real	Valor de proceso para operación manual. Valor predeterminado: 0.0
Reset	IN	Bool	El parámetro Reset reinicia el regulador. Valor predeterminado: FALSE En el apartado "Respuesta a reset", más adelante, hallará los gráficos de respuesta a reset del PID_Compact V1.1 y V1.0.
ScaledInput	OUT	Real	Valor de proceso escalado. Valor predeterminado: 0.0
Output ¹	OUT	Real	Valor de salida. Valor predeterminado: 0.0
Output_PER ¹	OUT	Word	Valor de salida analógico. Valor predeterminado: W#16#0
Output_PWM ¹	OUT	Bool	Valor de salida para la modulación del ancho de impulso. Valor predeterminado: FALSE
SetpointLimit_H	OUT	Bool	Límite superior de consigna. Valor predeterminado: FALSE Si SetpointLimit_H = TRUE, se ha alcanzado el límite superior absoluto de la consigna. Valor predeterminado: FALSE
SetpointLimit_L	OUT	Bool	Límite inferior de consigna. Valor predeterminado: FALSE Si SetpointLimit_L = TRUE, se ha alcanzado el límite inferior absoluto de la consigna. Valor predeterminado: FALSE
InputWarning_H	OUT	Bool	Si InputWarning_H = TRUE, el valor de proceso ha alcanzado o excedido el límite superior de advertencia. Valor predeterminado: FALSE
InputWarning_L	OUT	Bool	Si InputWarning_L = TRUE, el valor de proceso ha alcanzado el límite inferior de advertencia. Valor predeterminado: FALSE
State	OUT	Int	Estado operativo actual del regulador PID. Valor predeterminado: 0 Utilice sRet.i_Mode para cambiar de estado. <ul style="list-style-type: none"> • State = 0: Inactivo • State = 1: Optimización inicial • State = 2: Optimización fina manual • State = 3: Modo automático • State = 4: Modo manual
ErrorBits	OUT	DWord	La tabla de parámetros (Página 372) de la instrucción ErrorBits de PID_Compact define los mensajes de error. Valor predeterminado: DW#16#0000 (sin errores)

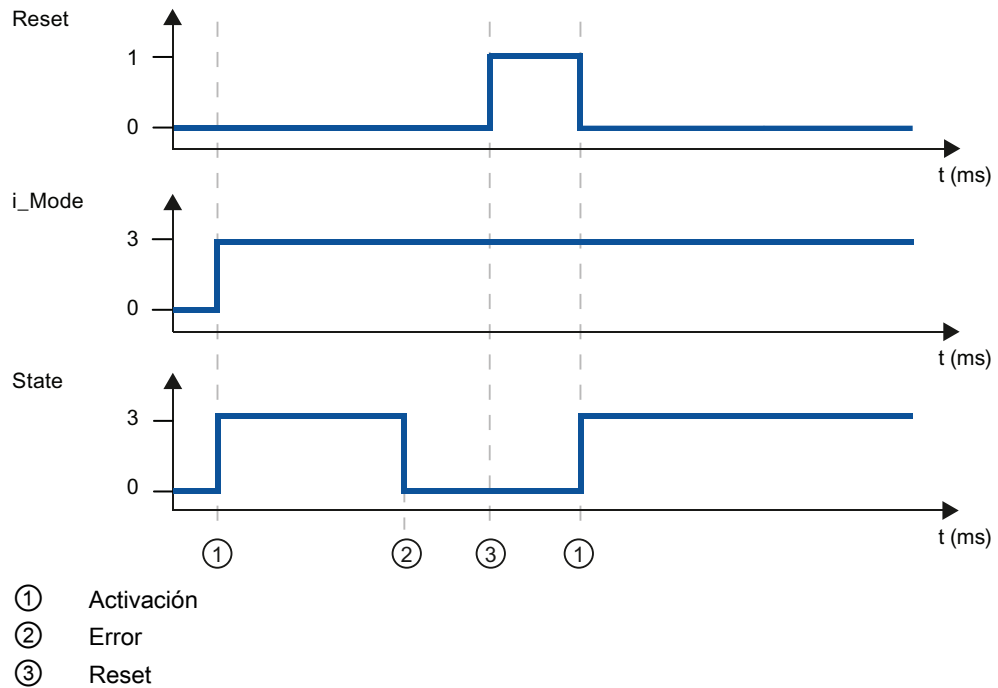
¹ Las salidas de los parámetros Output, Output_PER y Output_PWM pueden utilizarse paralelamente.

Respuesta a Reset

La respuesta a Reset = TRUE depende de la versión de la instrucción de PID_Compact.

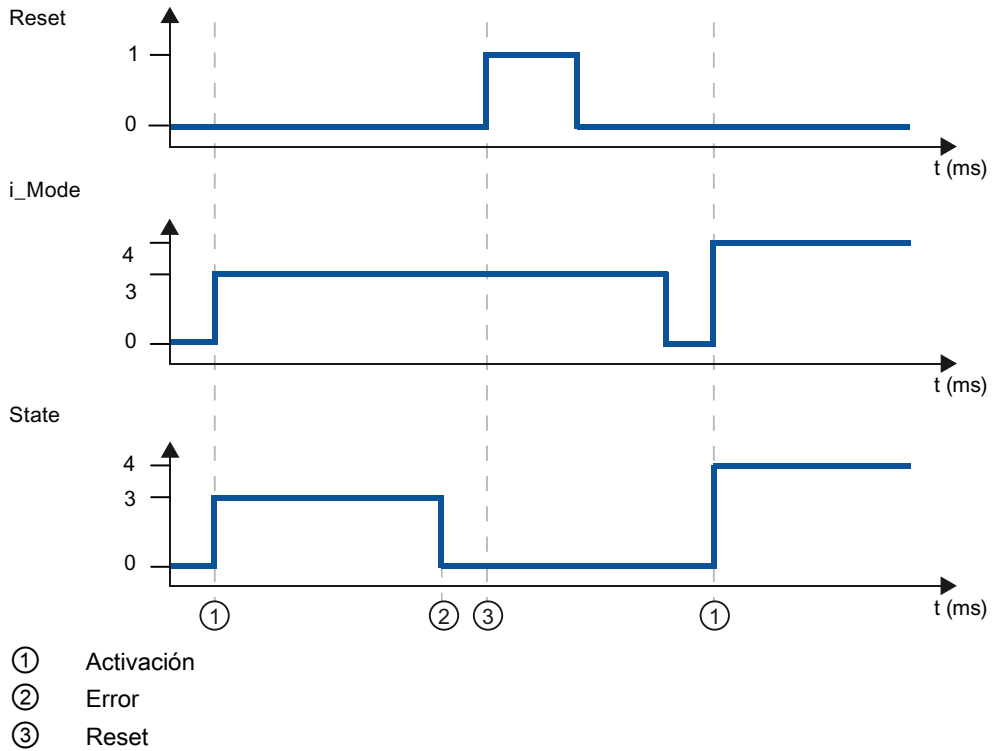
Respuesta a Reset, PID_Compact V1.1

Un flanco ascendente en Reset resetea los errores y advertencias, y borra la acción integral. Un flanco descendente en Reset activa un cambio al modo de operación activo más reciente.



Respuesta a Reset, PID_Compact V1.0

Un flanco ascendente en Reset resetea los errores y advertencias, y borra la acción integral. El regulador no se vuelve a activar hasta el siguiente flanco en i_Mode.



Operación del regulador PID_Compact

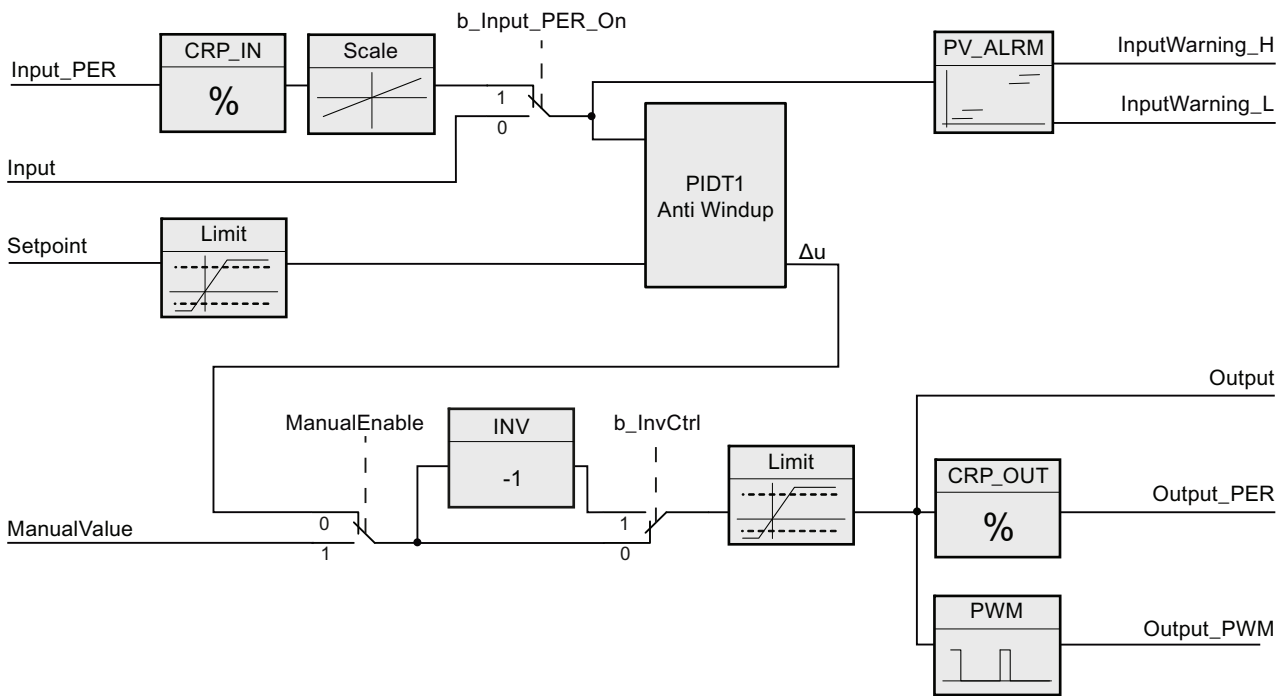


Figura 9-1 Operación del regulador PID_Compact

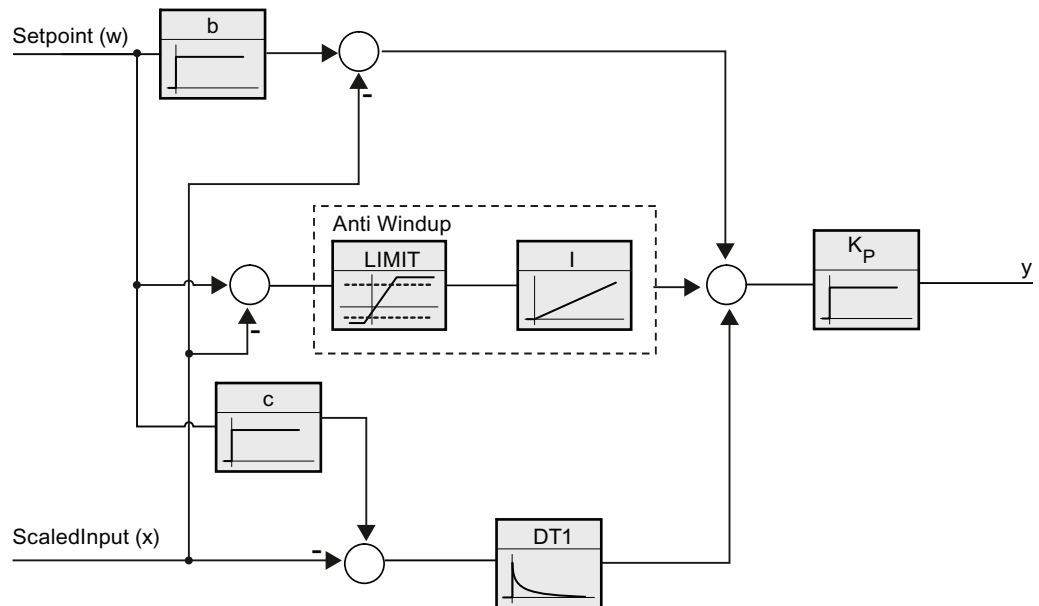


Figura 9-2 Operación del regulador PID_Compact como regulador PIDT1 con anti-windup

9.2.3 Parámetros de la instrucción ErrorBit de PID_Compact

Si hay varios errores pendientes, los valores de los códigos de error se muestran mediante suma binaria. La indicación del código de error 0003, por ejemplo, indica que también están pendientes los errores 0001 y 0002.

Tabla 9- 15 Parámetros de la instrucción ErrorBit de PID_Compact

ErrorBit (DW#16#...)	Descripción
0000	No hay error
0001	El parámetro "Input" se encuentra fuera de los límites de valores de proceso. Input > sPid_Cmpt.r_Pv_HImor Input < sPid_Cmpt.r_Pv_LIm No se puede volver a iniciar el actuador mientras no se elimine el error.
0002	Valor no válido en el parámetro "Input_PER". Compruebe si hay un error pendiente en la entrada analógica.
0004	Error durante la optimización fina: no es posible mantener la oscilación del valor de proceso.
0008	Error al iniciar la pre-optimización. El valor de proceso está demasiado próximo a la consigna. Inicie la optimización fina.
0010	La consigna se ha cambiado durante la optimización del regulador.
0020	No se puede preoptimizar en modo automático o durante la optimización fina.
0040	Error en la optimización fina: la consigna está demasiado próxima a los límites de consigna.
0080	Configuración incorrecta de los límites del valor de salida. Compruebe si los límites del valor de salida se han configurado correctamente y haga concordar el sentido en el que opera la regulación.
0100	Un error durante la optimización del regulador ha provocado parámetros no válidos.
0200	Valor no válido en el parámetro "Input": El formato numérico del valor no es válido.
0400	Fallo en el cálculo del valor de salida. Compruebe los parámetros PID.
0800	Error de tiempo de muestreo: No se ha llamado a PID_Compact durante el tiempo de muestreo del OB de alarma cíclica.
1000	Valor no válido en el parámetro "Setpoint": El formato numérico del valor no es válido.

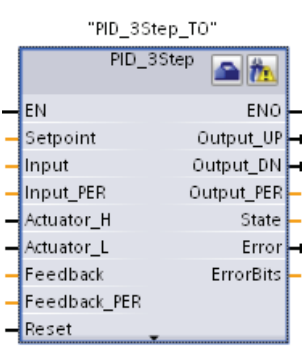
9.2.4 Instrucción PID_3STEP

El regulador PID utiliza la siguiente fórmula para calcular el valor de salida de la instrucción PID_3Step.

$$\Delta y = K_p \cdot s \cdot \left[(b \cdot w - x) + \frac{1}{T_i \cdot s} (w - x) + \frac{T_d \cdot s}{a \cdot T_d \cdot s + 1} (c \cdot w - x) \right]$$

y	Valor de salida	x	Valor de proceso
w	Consigna	s	Operador laplaciano
K _p	Ganancia proporcional (acción P)	a	Coefficiente de retardo derivativo (acción D)
T _i	Tiempo de acción integral (acción I)	b	Ponderación de acción proporcional (acción P)
T _d	Tiempo de acción derivativa (acción D)	c	Ponderación de acción derivativa (acción D)

Tabla 9- 16 Instrucción PID_3Step

KOP / FUP	SCL	Descripción
	<pre>"PID_3Step_1" (SetpoInt:=_real_in_, Input:=_real_in_, ManualValue:=_real_in_, Feedback:=_real_in_, InputPer:=_word_in_, FeedbackPer:=_word_in_, ManualEnable:=_bool_in_, ManualUP:=_bool_in_, ManualDN:=_bool_in_, ActuatorH:=_bool_in_, ActuatorL:=_bool_in_, Reset:=_bool_in_, ScaledInput=>_real_out_, ScaledFeedback=>_real_out_, ErrorBits=>_dword_out_, OutputPer=>_word_out_, State=>_int_out_, OutputUP=>_bool_out_, OutputDN=>_bool_out_, SetpoIntLimitH=>_bool_out_, SetpoIntLimitL=>_bool_out_, InputWarningH=>_bool_out_, InputWarningL=>_bool_out_, Error=>_bool_out_);</pre>	<p>PID_3Step configura un regulador PID con capacidades de autoajuste que se ha optimizado para válvulas accionadas por motor y actuadores. Proporciona dos salidas booleanas.</p> <p>PID_3Step es un regulador PIDT1 con anti-windup y ponderación de las acciones P y D.</p>

- STEP 7 crea automáticamente el objeto tecnológico y el DB de instancia al insertar la instrucción. El DB de instancia contiene los parámetros del objeto tecnológico.
- En el ejemplo SCL, "PID_3Step_1" es el nombre del DB de instancia.

Tabla 9- 17 Tipos de datos para los parámetros

Parámetro y tipo		Tipo de datos	Descripción
Setpoint	IN	Real	Consigna del regulador PID en modo automático. Valor predeterminado: 0.0
Input	IN	Real	Valor de proceso. Valor predeterminado: 0.0 También debe ponerse Config.InputPEROn = FALSE.
Input_PER	IN	Word	Valor de proceso analógico (opcional). Valor predeterminado: W#16#0 También debe ponerse Config.InputPEROn = TRUE.
ManualEnable	IN	Bool	Activa o desactiva el modo de operación manual. Valor predeterminado: FALSE <ul style="list-style-type: none"> En el flanco del cambio de FALSE a TRUE, el regulador PID conmuta a modo manual, State = 4 y Retain.Mode permanecen invariables. En el flanco del cambio de TRUE a FALSE, el regulador PID conmuta al último estado operativo activo y State = Retain.Mode.
ManualUP	IN	Bool	En modo manual, cada flanco ascendente abre la válvula un 5% del rango de posicionamiento total o por el tiempo de posicionamiento mínimo del motor. ManualUP solo se evalúa si no se utiliza Output_PER y si no hay realimentación de posición. Valor predeterminado: FALSE <ul style="list-style-type: none"> Si Output_PER es FALSE, la entrada manual activa Output_UP durante el tiempo correspondiente a un movimiento del 5% del dispositivo. Si Config.ActuatorEndStopOn es TRUE, Output_UP no se activa si Actuator_H es TRUE.
ManualDN	IN	Bool	En modo manual, cada flanco ascendente cierra la válvula un 5% del rango de posicionamiento total o por el tiempo de posicionamiento mínimo del motor. ManualDN sólo se evalúa si no se utiliza Output_PER y si no hay realimentación de posición. Valor predeterminado: FALSE <ul style="list-style-type: none"> Si Output_PER es FALSE, la entrada manual activa Output_DN durante el tiempo correspondiente a un movimiento del 5% del dispositivo. Si Config.ActuatorEndStopOn es TRUE, Output_DN no se activa si Actuator_L es TRUE.
ManualValue	IN	Real	Valor de proceso para operación manual. Valor predeterminado: 0.0 En modo manual, el usuario especifica la posición absoluta de la válvula. ManualValue sólo se evalúa si se utiliza OutputPer o si hay realimentación de posición. Valor predeterminado: 0.0
Feedback	IN	Real	Realimentación de posición de la válvula. Valor predeterminado: 0.0 Para utilizar Feedback, ponga Config.FeedbackPerOn = FALSE.

Parámetro y tipo		Tipo de datos	Descripción
Feedback_PER	IN	Word	Realimentación analógica de la posición de la válvula. Valor predeterminado: W#16#0 Para utilizar Feedback_PER, ponga Config.FeedbackPerOn = TRUE. Feedback_PER se escala mediante los parámetros siguientes: <ul style="list-style-type: none"> • Config.FeedbackScaling.LowerPointIn • Config.FeedbackScaling.UpperPointIn • Config.FeedbackScaling.LowerPointOut • Config.FeedbackScaling.UpperPointOut
Actuator_H	IN	Bool	Si Actuator_H = TRUE, la válvula está en la posición límite superior y ya no se mueve en esa dirección. Valor predeterminado: FALSE
Actuator_L	IN	Bool	Si Actuator_L = TRUE, la válvula está en la posición límite inferior y ya no se mueve en esa dirección. Valor predeterminado: FALSE
Reset	IN	Bool	Reinicia el regulador PID. Valor predeterminado: FALSE Si hay un flanco FALSE a TRUE: <ul style="list-style-type: none"> • Modo de operación "Inactivo" • Valor de entrada = 0 • Se reinician los valores intermedios del regulador. (Los parámetros PID se conservan.) Si hay un flanco TRUE a FALSE, cambio al modo activo más reciente.
ScaledInput	OUT	Real	Valor de proceso escalado
ScaledFeedback	OUT	Real	Posición de válvula escalada
Output_PER	OUT	Word	Valor de salida analógico. Si Config.OutputPerOn = TRUE, se usa el parámetro Output_PER.
Output_UP	OUT	Bool	Valor de salida digital para abrir la válvula. Valor predeterminado: FALSE Si Config.OutputPerOn = FALSE, se usa el parámetro Output_UP.
Output_DN	OUT	Bool	Valor de salida digital para cerrar la válvula. Valor predeterminado: FALSE Si Config.OutputPerOn = FALSE, se usa el parámetro Output_DN.
SetpointLimitH	OUT	Bool	Límite superior de consigna. Valor predeterminado: FALSE Si SetpointLimitH = TRUE, se ha alcanzado el límite superior absoluto de la consigna. En la CPU, la consigna está limitada por el límite superior absoluto configurado del valor real.
SetpointLimitL	OUT	Bool	Límite inferior de consigna. Valor predeterminado: FALSE Si SetpointLimitL = TRUE, se ha alcanzado el límite inferior absoluto de la consigna. En la CPU la consigna está limitada por el límite inferior absoluto configurado del valor real.
InputWarningH	OUT	Bool	Si InputWarningH = TRUE, el valor de entrada ha alcanzado o excedido el límite superior de advertencia. Valor predeterminado: FALSE
InputWarningL	OUT	Bool	Si InputWarningL = TRUE, el valor de entrada ha alcanzado o excedido el límite inferior de advertencia. Valor predeterminado: FALSE

Parámetro y tipo	Tipo de datos	Descripción	
State	OUT	Int	Estado operativo actual del regulador PID. Valor predeterminado: 0 Utilice Retain.Mode para cambiar de estado operativo: <ul style="list-style-type: none"> • State = 0: Inactivo • State = 1: Optimización inicial • State = 2: Optimización fina manual • State = 3: Modo automático • State = 4: Modo manual • State = 5: Aproximación al valor de salida sustitutivo • State = 6: Medición del tiempo de transición • State = 7: Aproximación al valor de salida sustitutivo con monitorización de errores • State = 8: Monitorización de errores
Error	OUT	Bool	Si Error = TRUE, hay como mínimo un mensaje de error pendiente. Valor predeterminado: FALSE
ErrorBits	OUT	DWord	La tabla de parámetros (Página 379) de la instrucción ErrorBits de PID_3STEP define los mensajes de error. Valor predeterminado: DW#16#0000 (sin errores)

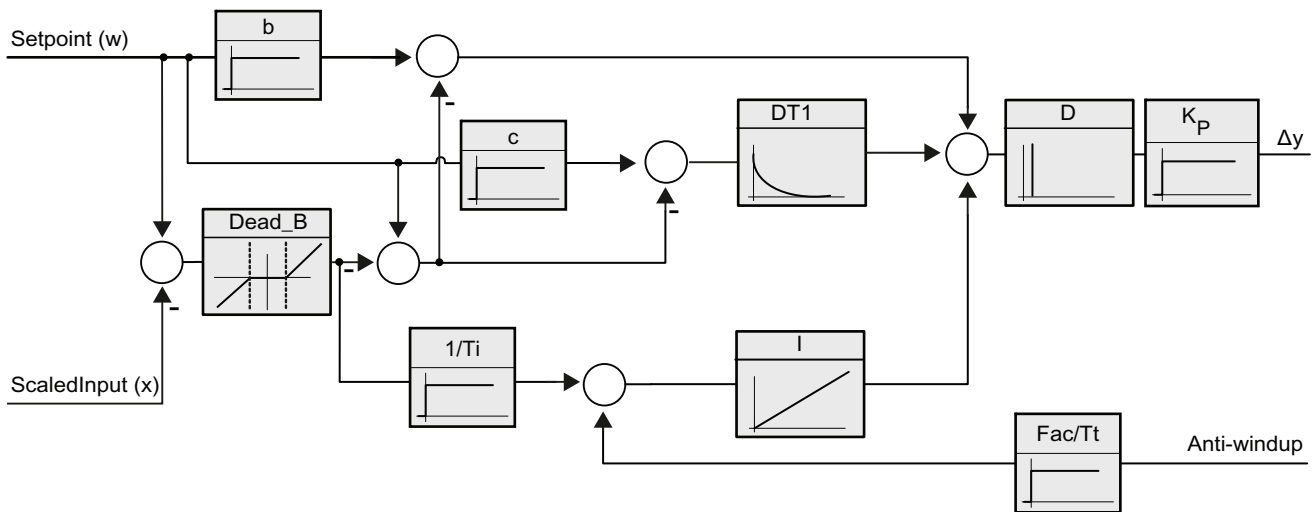


Figura 9-3 Operación del regulador PID_3Step como regulador PIDT1 con anti-windup

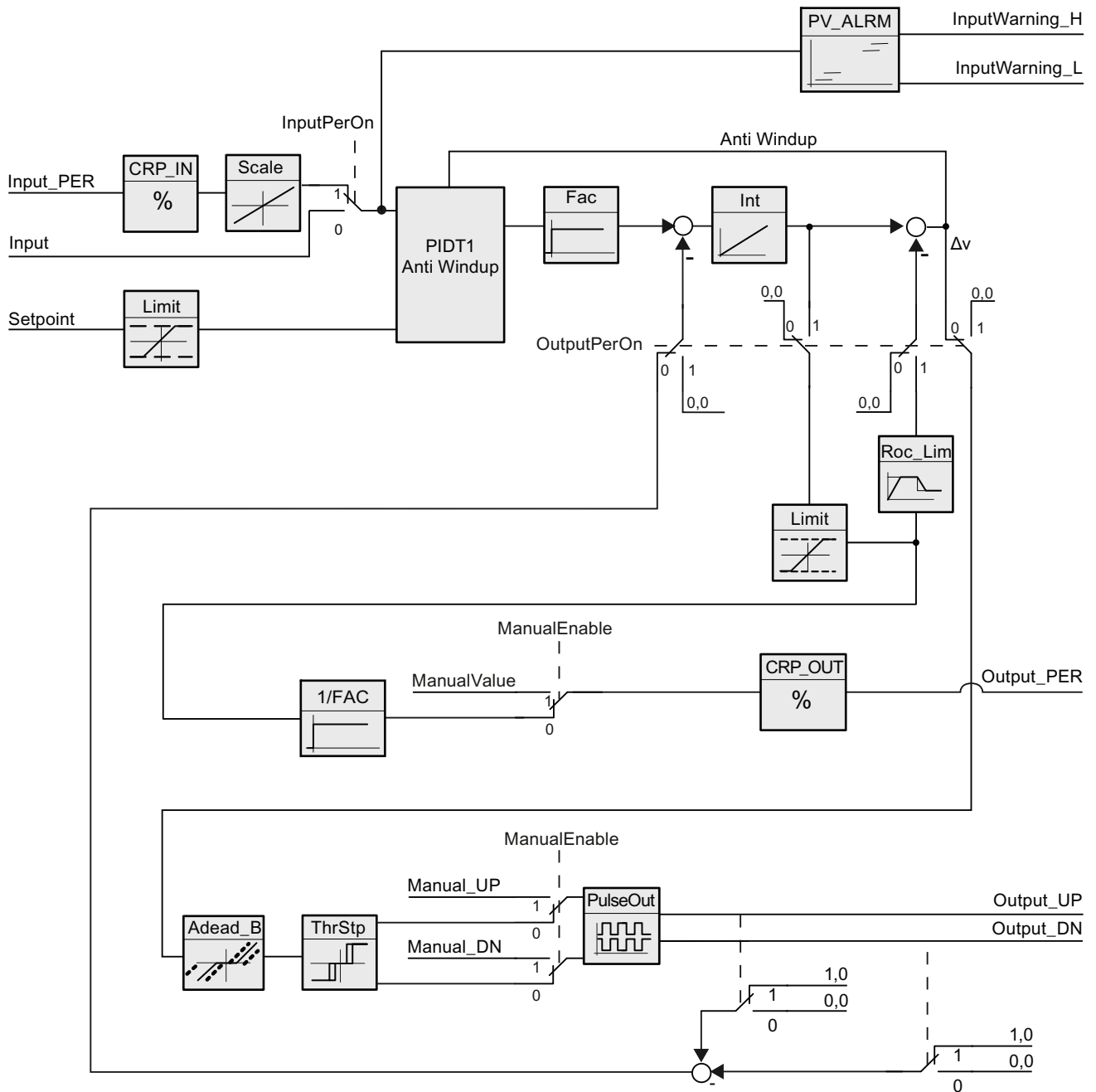


Figura 9-4 Operación del regulador PID_3Step sin realimentación de posición

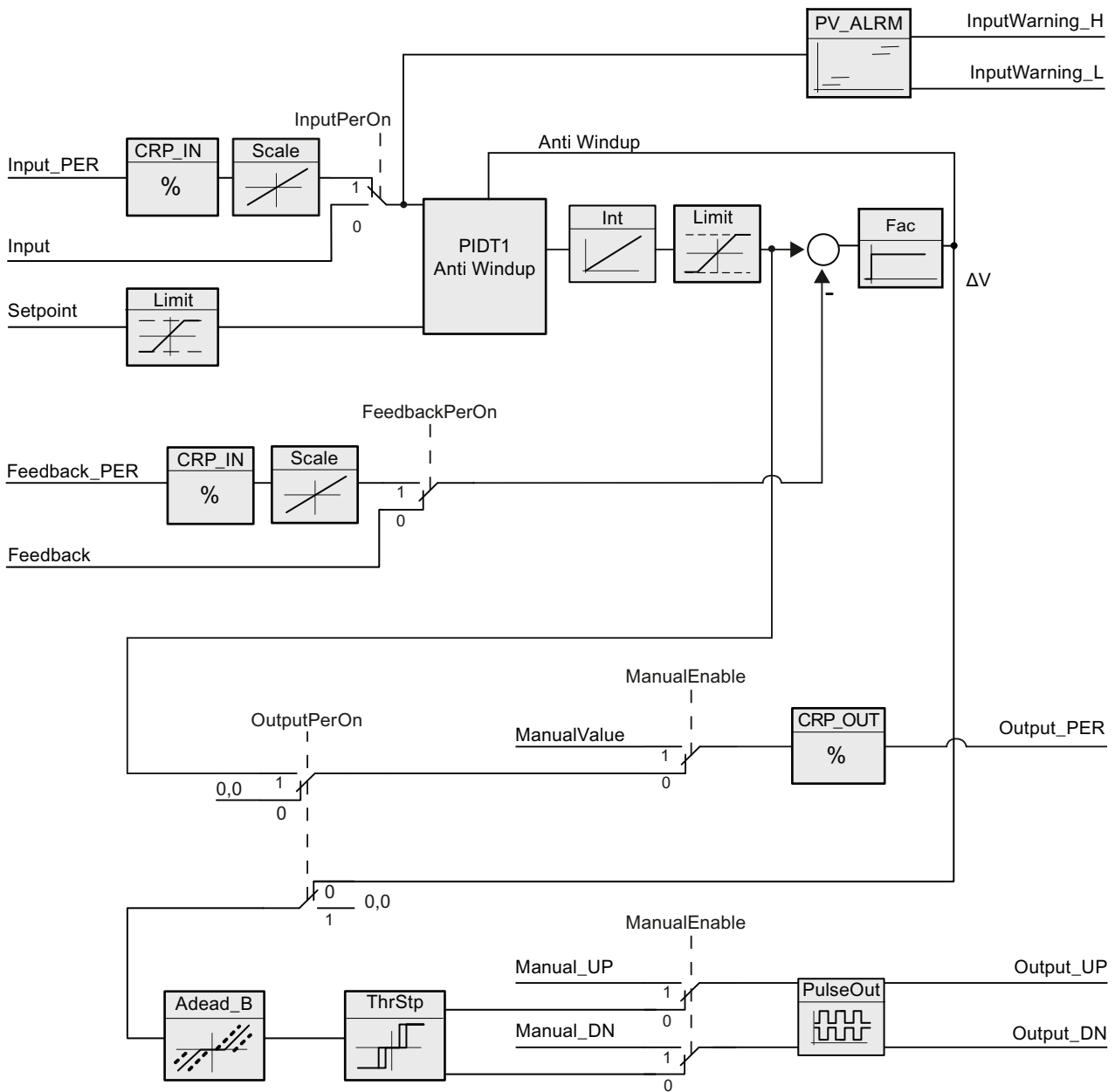


Figura 9-5 Operación del regulador PID_3Step con realimentación de posición habilitada

9.2.5 Parámetros de la instrucción ErrorBit de PID_3STEP

Si hay varios errores pendientes, los valores de los códigos de error se muestran mediante suma binaria. La indicación del código de error 0003, por ejemplo, indica que también están pendientes los errores 0001 y 0002.

Tabla 9- 18 Parámetros de la instrucción ErrorBit de PID_3STEP

ErrorBit (DW#16#...)	Descripción
0000	No hay error
0001	El parámetro "Input" se encuentra fuera de los límites de valores de proceso: Input > Config.InputUpperLimit o Input < Config.InputLowerLimit Si ActivateRecoverMode = TRUE y ErrorBehaviour = 1, el actuador se mueve hasta el valor de salida sustitutivo. Si ActivateRecoverMode = TRUE y ErrorBehaviour = 0, el actuador se detiene en su posición actual. Si ActivateRecoverMode = FALSE, el actuador se detiene en su posición actual. PID_3STEP V1.1: Se puede mover el actuador en modo manual. PID_3STEP V1.0: En este estado no es posible el modo manual. No se puede volver a iniciar el actuador mientras no se elimine el error.
0002	Valor no válido en el parámetro "Input_PER". Compruebe si hay un error pendiente en la entrada analógica. Si el modo automático estaba activo antes de que ocurriera el error, ActivateRecoverMode = TRUE y el error ya no está pendiente. PID_3STEP vuelve a modo automático.
0004	Error durante la optimización fina: no es posible mantener la oscilación del valor de proceso.
0008	Error al iniciar la pre-optimización. El valor de proceso está demasiado próximo a la consigna. Inicie la optimización fina.
0010	La consigna no se puede cambiar durante la optimización fina.
0020	No se puede preoptimizar en modo automático o durante la optimización fina.
0040	Error en la optimización fina: la consigna está demasiado próxima a los límites de consigna.
0080	Error durante la preoptimización. Configuración incorrecta de los límites del valor de salida. Compruebe si los límites del valor de salida se han configurado correctamente y haga concordar el sentido en el que opera la regulación.
0100	Un error durante la optimización fina ha provocado parámetros no válidos.
0200	Valor no válido en el parámetro "Input": El formato numérico del valor no es válido. Si el modo automático estaba activo antes de que ocurriera el error, ActivateRecoverMode = TRUE y el error ya no está pendiente. PID_3STEP vuelve a modo automático.
0400	Fallo en el cálculo del valor de salida. Compruebe los parámetros PID.

ErrorBit (DW#16#...)	Descripción
0800	<p>Error de tiempo de muestreo: No se ha llamado a PID_3STEP durante el tiempo de muestreo del OB de alarma cíclica.</p> <p>Si el modo automático estaba activo antes de que ocurriera el error, ActivateRecoverMode = TRUE y el error ya no está pendiente. PID_3STEP vuelve a modo automático.</p>
1000	<p>Valor no válido en el parámetro "Setpoint": El formato numérico del valor no es válido.</p> <p>Si el modo automático estaba activo antes de que ocurriera el error, ActivateRecoverMode = TRUE y el error ya no está pendiente. PID_3STEP vuelve a modo automático.</p>
2000	<p>Valor no válido en el parámetro Feedback_PER.</p> <p>Compruebe si hay un error pendiente en la entrada analógica.</p> <p>No se puede mover el actuador al valor de salida sustitutivo, y no se mueve de la posición actual. En este estado no es posible el modo manual. Se ha de deshabilitar la realimentación de posición (Config. FeedbackOn = FALSE) para sacar al actuador de este estado.</p> <p>Si el modo automático estaba activo antes de que ocurriera el error, ActivateRecoverMode = TRUE y el error ya no está pendiente. PID_3STEP vuelve a modo automático.</p>
4000	<p>Valor no válido en el parámetro Feedback. El formato numérico del valor no es válido.</p> <p>No se puede mover el actuador al valor de salida sustitutivo, y no se mueve de la posición actual. En este estado no es posible el modo manual. Se ha de deshabilitar la realimentación de posición (Config. FeedbackOn = FALSE) para sacar al actuador de este estado.</p> <p>Si el modo automático estaba activo antes de que ocurriera el error, ActivateRecoverMode = TRUE y el error ya no está pendiente. PID_3STEP vuelve a modo automático.</p>
8000	<p>Error en la realimentación de posición digital. Actuator_H = TRUE y Actuator_L = TRUE.</p> <p>No se puede mover el actuador al valor de salida sustitutivo, y no se mueve de la posición actual. En este estado no es posible el modo manual.</p> <p>Se ha de deshabilitar "Endstop signals actuador" (Config. FeedbackOn = FALSE) para sacar al actuador de este estado.</p> <p>Si el modo automático estaba activo antes de que ocurriera el error, ActivateRecoverMode = TRUE y el error ya no está pendiente. PID_3STEP vuelve a modo automático.</p>

9.2.6 Configurar el regulador PID

Los parámetros del objeto tecnológico determinan el funcionamiento del regulador PID. Utilice el icono para abrir el editor de configuración.

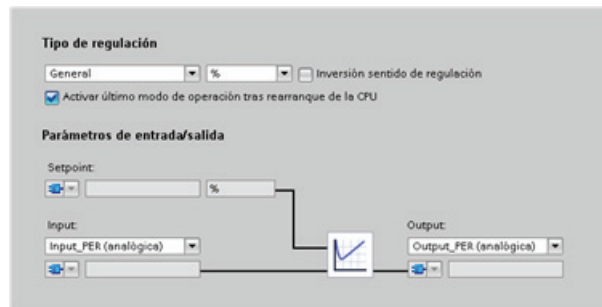


Figura 9-6 Editor de configuración para PID_Compact (ajustes básicos)

Tabla 9- 19 Ajustes de configuración de muestreo para la instrucción PID_Compact

Configuración	Descripción	
Basic	Tipo de regulador	Selecciona las unidades de ingeniería.
	Invertir la lógica de control	Permite seleccionar un lazo PID de acción inversa. <ul style="list-style-type: none"> Si no está seleccionado, el lazo PID está en modo de acción directa y la salida del lazo PID se incrementa si el valor de entrada < la consigna. Si está seleccionado, la salida del lazo PID se incrementa si el valor de entrada > la consigna.
	Habilitar el último estado después de reiniciar la CPU	Reinicia el lazo PID después de resetearlo o si se ha excedido un límite de entrada y se ha vuelto al rango válido.
	Entrada	Selecciona el parámetro Input o Input_PER (analógico) para el valor de proceso. Input_PER puede proceder directamente de un módulo de entrada analógico.
	Salida	Selecciona el parámetro Output o Output_PER (analógico) para el valor de salida. Output_PER puede ir directamente a un módulo de salida analógico.
Valor de proceso	Escala tanto el rango como los límites del valor de proceso. Si el valor de proceso rebasa por defecto el límite inferior o por exceso el límite superior, el lazo PID pasa al estado inactivo y pone el valor de salida a 0. Para utilizar Input_PER hay que escalar el valor de proceso analógico (valor de entrada).	

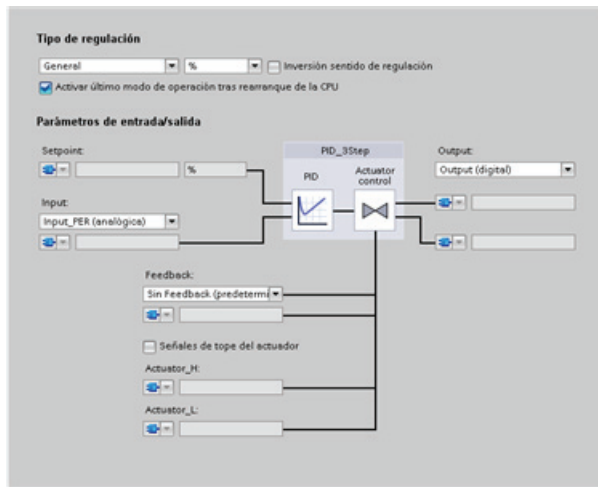


Figura 9-7 Editor de configuración para PID_3Step (ajustes básicos)

Tabla 9- 20 Ajustes de configuración de muestreo para la instrucción PID_3Step

Configuración		Descripción
Basic	Tipo de regulador	Selecciona las unidades de ingeniería.
	Invertir la lógica de control	Permite seleccionar un lazo PID de acción inversa. <ul style="list-style-type: none"> • Si no está seleccionado, el lazo PID está en modo de acción directa y la salida del lazo PID se incrementa si el valor de entrada < la consigna. • Si está seleccionado, la salida del lazo PID se incrementa si el valor de entrada > la consigna.
	Habilitar el último estado después de reiniciar la CPU	Reinicia el lazo PID después de resetearlo o si se ha excedido un límite de entrada y se ha vuelto al rango válido.
	Entrada	Selecciona el parámetro Input o Input_PER (analógico) para el valor de proceso. Input_PER puede proceder directamente de un módulo de entrada analógico.
	Salida	Selecciona si deben utilizarse las salidas digitales (Output_UP y Output_DN) o la salida analógica (Output_PER) para el valor de salida.
	Realimentación	Selecciona el tipo de estado de dispositivo devuelto al lazo PID: <ul style="list-style-type: none"> • Sin realimentación (predeterminado) • Realimentación • Feedback_PER
Valor de proceso	<p>Escala tanto el rango como los límites del valor de proceso. Si el valor de proceso rebasa por defecto el límite inferior o por exceso el límite superior, el lazo PID pasa al estado inactivo y pone el valor de salida a 0.</p> <p>Para utilizar Input_PER hay que escalar el valor de proceso analógico (valor de entrada).</p>	
Actuador	Tiempo de transición del motor	Establece el tiempo entre la abertura y el cierre de la válvula. (Encontrará este valor en la hoja de datos o en la placa frontal de la válvula.)
	Tiempo de actividad mínimo	Establece el tiempo de movimiento mínimo de la válvula. (Encontrará este valor en la hoja de datos o en la placa frontal de la válvula.)
	Tiempo de reposo mínimo	Establece el tiempo de pausa mínimo de la válvula. (Encontrará este valor en la hoja de datos o en la placa frontal de la válvula.)

Configuración		Descripción
	Comportamiento de error	Define el comportamiento de la válvula cuando se detecta un error o cuando se resetea el lazo PID. Si se elige utilizar una posición de sustitución, introduzca la "posición de seguridad". Para una realimentación o salida analógica, seleccione un valor entre el límite superior o inferior de la salida. Para salidas digitales, puede elegir sólo entre 0% (off) y 100% (on).
	Escarlar realimentación de posición ¹	<ul style="list-style-type: none"> "Límite superior de parada" y "Límite inferior de parada" definen la posición positiva máxima (completamente abierta) y la posición negativa máxima (completamente cerrada). El "Límite superior de parada" debe ser mayor que el "Límite inferior de parada". "Límite superior del valor de proceso" y "Límite inferior del valor de proceso" definen las posiciones superior e inferior de la válvula durante el ajuste y el modo automático. "FeedbackPER" ("Low" y "High") define la realimentación analógica de la posición de la válvula. "FeedbackPER High" debe ser mayor que "FeedbackPER Low".

¹ "Escarlar realimentación de posición" sólo es editable si se ha activado "Realimentación" en los ajustes "básicos".

9.2.7 Puesta en servicio del regulador PID

Utilice el editor de puesta en servicio para configurar el regulador PID de modo que se autoajuste al arrancar y durante el funcionamiento. Para abrir el editor de puesta en servicio, haga clic en el icono de la instrucción o del árbol del proyecto.



Tabla 9- 21 Ejemplo de pantalla de ajuste (PID_3Step)

- **Medición:** para visualizar la consigna, el valor de proceso (valor de entrada) y el valor de salida en una curva de tiempo real, introduzca el tiempo de muestreo y haga clic en el botón "Iniciar".
- **Modo de optimización:** para optimizar el lazo PID, seleccione "Optimización inicial" u "Optimización fina" (manual) y haga clic en el botón "Iniciar". El regulador PID pasa por diferentes fases para calcular la respuesta del sistema y los tiempos de actualización. Los parámetros de optimización adecuados se calculan a partir de estos valores.

Una vez finalizado el proceso de optimización, los parámetros nuevos se guardan haciendo clic en el botón "Cargar parámetros PID" de la sección "Parámetros PID" del editor de puesta en servicio.

Si se produce un error durante la optimización, el valor de salida del PID pasa a 0. En este caso, el modo PID se pone a "Inactivo". El estado indica el error.

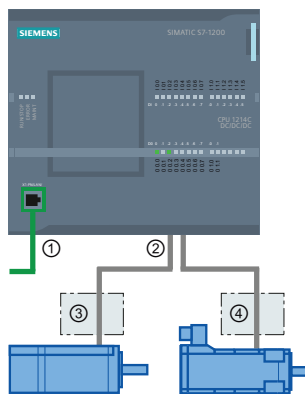
9.3 Motion control

La CPU ofrece funciones de control de movimiento para el uso de motores paso a paso y servomotores con interfaz por impulso. Las funciones de control de movimiento controlan y monitorizan los accionamientos.

- El objeto tecnológico "Eje" configura los datos mecánicos del accionamiento, así como su interfaz, sus parámetros dinámicos y otras propiedades.
- Las salidas de dirección e impulso de la CPU deben configurarse para controlar el accionamiento.
- El programa de usuario utiliza las instrucciones de Motion Control para controlar el eje e iniciar las tareas de desplazamiento.
- La interfaz de PROFINET se utiliza para establecer la conexión online entre la CPU y la programadora. Además de las funciones online de la CPU, hay funciones de puesta en marcha y diagnóstico adicionales para el control de movimiento.

Nota

Los cambios realizados en la configuración de control de movimiento y las descargas en modo RUN no son efectivos hasta que la CPU no pasa del estado operativo STOP al estado operativo RUN.



- ① PROFINET
- ② Salidas de dirección e impulso
- ③ Etapa de potencia para motor paso a paso
- ④ Etapa de potencia para servomotor

Las variantes DC/DC/DC de la CPU S7-1200 cuentan con salidas incorporadas para ejercer un control directo de los accionamientos. Las variantes de relé de la CPU necesitan una Signal Board con salidas DC para el control de los accionamientos.

Una Signal Board (SB) amplía las E/S incorporadas para incluir algunas E/S adicionales. Puede emplearse una SB con dos salidas digitales como generadores de impulsos y sentido para controlar un motor. Puede emplearse una SB con cuatro salidas digitales como generadores de impulso y sentido para controlar dos motores. No se pueden usar salidas de relé integradas como generadores de impulsos para controlar motores.

Nota

Los trenes de impulsos no pueden ser utilizados por otras instrucciones del programa de usuario

Si las salidas de la CPU o Signal Board se configuran como generadores de impulsos (para su utilización con PWM o instrucciones de Motion Control), las direcciones de salida correspondientes (de Q0.0 a Q0.3, de Q4.0 a Q4.3) se eliminan de la memoria Q y no pueden utilizarse para ningún otro fin en el programa de usuario. Si el programa de usuario escribe un valor en una salida utilizada como generador de impulsos, la CPU no escribirá ese valor en la salida física.

Tabla 9- 22 Número máximo de unidades controlables

Tipo de CPU		Ninguna SB instalada	Con una SB (2 salidas DC)	Con una SB (4 salidas DC)
CPU 1211C	DC/DC/DC	2	2	2
	AC/DC/relé	0	1	2
	DC/DC/relé	0	1	2
CPU 1212C	DC/DC/DC	2	2	2
	AC/DC/relé	0	1	2
	DC/DC/relé	0	1	2
CPU 1214C	DC/DC/DC	2	2	2
	AC/DC/relé	0	1	2
	DC/DC/relé	0	1	2
CPU 1215C	DC/DC/DC	4	4	4
	AC/DC/relé	0	1	2
	DC/DC/relé	0	1	2

Tabla 9- 23 Frecuencias límite de salidas de impulsos

Salida de impulsos	Frecuencia
Integrada	2 PTO: $2 \text{ Hz} \leq f \leq 100 \text{ KHz}$
	2 PTO: $2 \text{ Hz} \leq f \leq 20 \text{ KHz}$
SB estándar	$2 \text{ Hz} \leq f \leq 20 \text{ KHz}$
SB de alta velocidad (200 KHz)	Instrucciones MC V2: $2 \text{ Hz} \leq f \leq 200 \text{ KHz}$
	Instrucciones MC V1: $2 \text{ Hz} \leq f \leq 100 \text{ KHz}$ ¹

¹ Las instrucciones MC V1 soportan una frecuencia máxima de 100 KHz.

ATENCIÓN

La frecuencia de impulsos máxima de los generadores de salida de impulsos es de 100 KHz para las salidas digitales de la CPU, 20 KHz para las salidas digitales de la SB estándar y 200 KHz para las salidas digitales de las SB de alta velocidad (o 100 KHz para instrucciones MC V1).

Configurar un generador de impulsos

1. Agregue un objeto tecnológico:
 - En el árbol de proyectos, expanda el nodo "Objetos tecnológicos" y seleccione "Agregar objeto".
 - Seleccione el icono "Eje" (cambie el nombre si fuera necesario) y haga clic en "Aceptar" para abrir el editor de configuración para el objeto de eje.
 - Visualice las propiedades de selección de PTO para el control del eje en la sección de parámetros básicos y elija el impulso que desee. Tenga en cuenta las dos salidas Q asignadas para los impulsos y el sentido.

Nota

Si el PTO no se ha configurado anteriormente en las propiedades de la CPU, se configura para utilizar una de las salidas incorporadas.

Si se utiliza una salida de Signal Board, seleccione el botón "Configuración de dispositivos" para ir a las propiedades de la CPU. Bajo "Parametrización", en "Opciones de impulsos", configure la fuente de salida para una salida de Signal Board. "Pulse_1" y "Pulse_3" son las únicas salidas de impulsos disponibles en Signal Board.

- Configure el resto de los parámetros básicos y avanzados.

2. Programe la aplicación: Introduzca la instrucción MC_Power en un bloque lógico.
 - Para la entrada de eje, seleccione el objeto tecnológico Eje que ha creado y configurado.
 - Si se establece la entrada Enable a TRUE, pueden utilizarse las demás instrucciones de movimiento.
 - Si se establece la entrada Enable a FALSE, se cancelan las demás instrucciones de movimiento.

Nota

Incluya sólo una instrucción MC_Power por eje.

3. Inserte las demás instrucciones de movimiento para obtener el desplazamiento necesario.

Nota

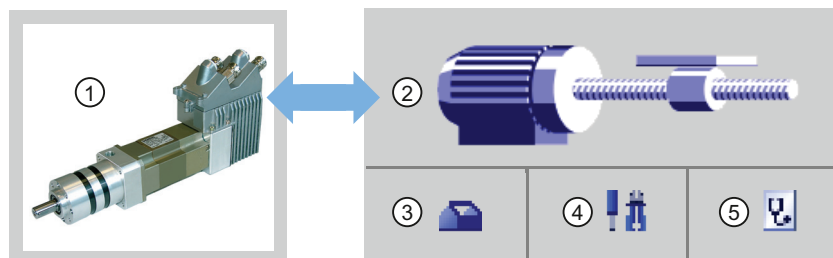
Configurar un generador de impulsos para salidas de Signal Board: Seleccione las propiedades "Generadores de impulsos (PTO/PWM)" de una CPU (en la configuración de dispositivos) y habilite un generador de impulsos. Hay dos generadores de impulsos disponibles para cada CPU S7-1200 V1.0, V2.0 y V2.1; para la CPU S7-1200 V2.2. y V3.0 hay cuatro generadores de impulsos disponibles. En esta misma área de configuración, en "Opciones de impulsos", seleccione que el generador de impulsos se use como: "PTO".

Nota

La CPU calcula las tareas de movimiento en "fragmentos" o segmentos de 10 ms. Una vez se ha ejecutado un fragmento, el próximo ya está esperando en la cola para ser ejecutado. Si se interrumpe la tarea de movimiento de un eje (ejecutando otra tarea de movimiento para dicho eje), la nueva tarea de movimiento no puede ejecutarse durante un máximo de 20 ms (el resto de la fracción actual más la fracción en cola).

9.3.1**Configurar el eje**

STEP 7 ofrece las herramientas de configuración, puesta en servicio y diagnóstico del objeto tecnológico "Eje".



- | | | | |
|---|--------------------|---|--------------------|
| ① | Accionamiento | ④ | Puesta en servicio |
| ② | Objeto tecnológico | ⑤ | Diagnóstico |
| ③ | Configuración | | |

Nota

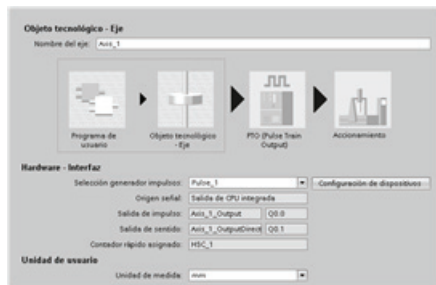
El PTO necesita la funcionalidad interna de un contador rápido (HSC). Esto significa que el contador rápido correspondiente no puede utilizarse en ningún otro lugar.

La asignación entre PTO y HSC es fija. Cuando se activa PTO1, se conecta a HSC1. Cuando se activa PTO2, se conecta a HSC2. Esto sólo es aplicable a las CPU S7-1200 V1.0, V2.0, V2.1 y V2.2. Las CPU S7-1200 V3.0 no tienen esta limitación.

No se puede controlar el valor actual (por ejemplo, en ID 1000) cuando se producen impulsos.

Tabla 9- 24 Herramientas de STEP 7 para el control de movimiento

Herramienta	Descripción
Configuración	<p>Configura las propiedades siguientes del objeto tecnológico "Eje":</p> <ul style="list-style-type: none"> • Selección del PTO que se va a utilizar y configuración de la interfaz del accionamiento • Propiedades de los mecanismos y de la velocidad de transmisión del accionamiento (o máquina o sistema) • Propiedades de los límites de posición, las animaciones y la referenciación <p>Guarde la configuración en el bloque de datos del objeto tecnológico.</p>
Puesta en servicio	<p>Prueba la función del eje sin tener que crear un programa de usuario. Cuando se inicia la herramienta, se muestra el panel de control. Los comandos siguientes están disponibles en el panel de control:</p> <ul style="list-style-type: none"> • Habilitación y deshabilitación del eje • Desplazamiento del eje en modo Jog • Posicionamiento del eje en términos absolutos y relativos • Referenciación del eje • Confirmación de errores <p>Para los comandos de desplazamiento es posible especificar la velocidad y la aceleración o deceleración. El panel de control también muestra el estado del eje actual.</p>
Diagnóstico	<p>Controla el estado actual y la información de error del eje y del accionamiento.</p>



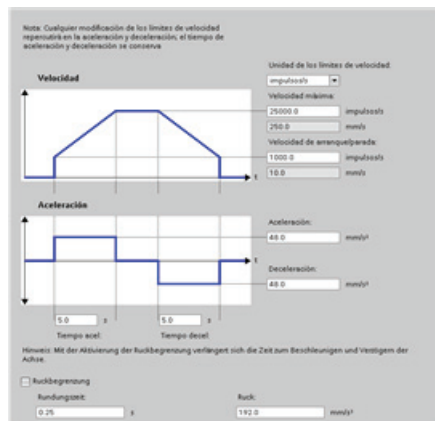
Después de crear el objeto tecnológico para el eje, se configura el eje definiendo los parámetros básicos, como el PTO y la configuración de la interfaz del accionamiento. También se configuran las demás propiedades del eje, como los límites de posición, las animaciones y la referenciación.

ATENCIÓN

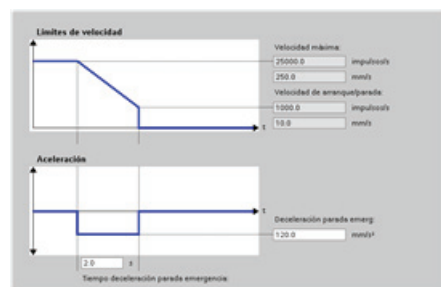
Puede que haya que adaptar los valores de los parámetros de entrada de las instrucciones de control de movimiento a la nueva unidad del programa de usuario.



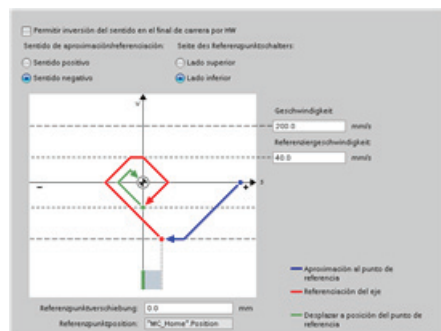
Configure las propiedades de las señales y los mecanismos del accionamiento, así como la vigilancia de posición (finales de carrera por hardware y software).



Se configuran las animaciones del movimiento y el comportamiento del comando de parada de emergencia.



Además, se configura el comportamiento de referenciación (pasiva y activa).



Utilice el panel de mando de "Puesta en servicio" para probar la funcionalidad independientemente del programa de usuario.



Haga clic en el icono "Inicio" para poner el eje en servicio.

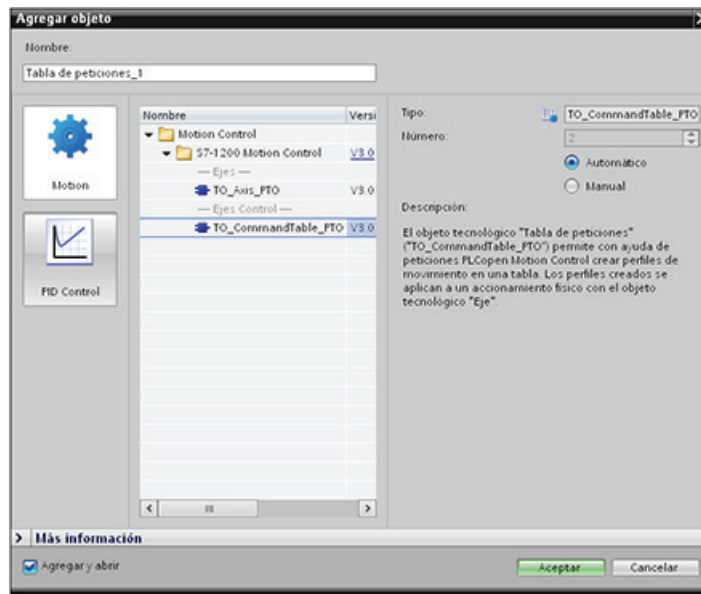
El panel de control muestra el estado actual del eje. No sólo es posible habilitar y deshabilitar el eje, sino también probar el posicionamiento del mismo (tanto en términos absolutos como relativos) y especificar la velocidad, aceleración y deceleración. También pueden probarse las tareas de referenciación y Jog. El panel de mando permite además acusar errores.

9.3.2 Configurar TO_CommandTable_PTO

Puede configurar una instrucción CommandTable mediante los objetos tecnológicos.

Agregar un objeto tecnológico

1. En el árbol de proyectos, expanda el nodo "Objetos tecnológicos" y seleccione "Agregar objeto".
2. Seleccione el icono "CommandTable" (cambie el nombre si es necesario) y haga clic en "Aceptar" para abrir el editor de configuración para el objeto CommandTable.



Planificar los pasos para su aplicación

La secuencia de movimientos deseada se puede crear en la ventana de configuración "Tabla de comandos" y comprobar el resultado con la vista gráfica del diagrama de tendencias.

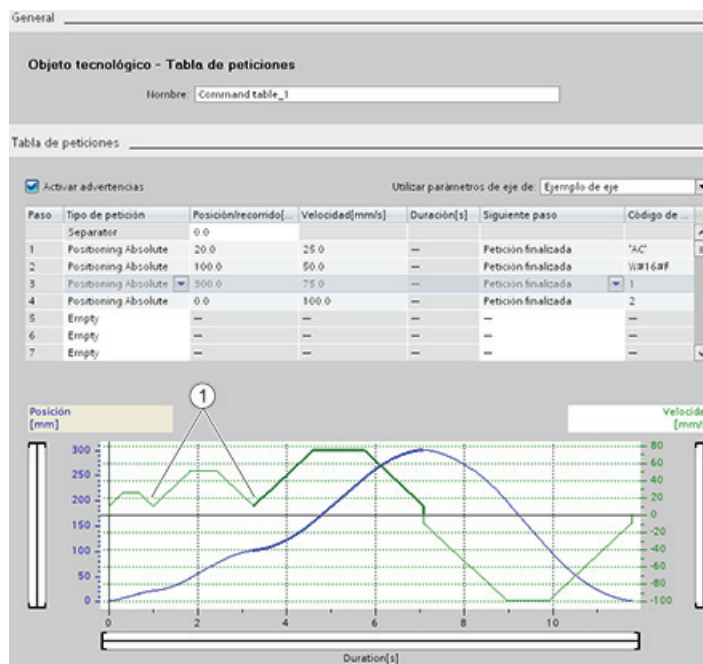
Se pueden seleccionar los tipos de comandos que se van a utilizar para procesar la tabla de comandos. Se pueden introducir hasta 32 pasos. Los comandos se procesan en secuencia, lo que permite generar con facilidad un complejo perfil de movimiento.

Tabla 9- 25 Tipos de comandos de MC_CommandTable

Tipo de comando	Descripción
Empty	El comando vacío sirve como comodín para los comandos que se añadan. La entrada vacía se ignora cuando se procesa la tabla de comandos.
Halt	Detiene el eje. Nota: El comando sólo tiene lugar tras un comando "Velocity setpoint".
Positioning Relative	Coloca el eje basándose en la distancia. El comando mueve el eje según la distancia y la velocidad dadas.
Positioning Absolute	Coloca el eje basándose en la posición. El comando mueve el eje a la posición indicada, a la velocidad especificada.

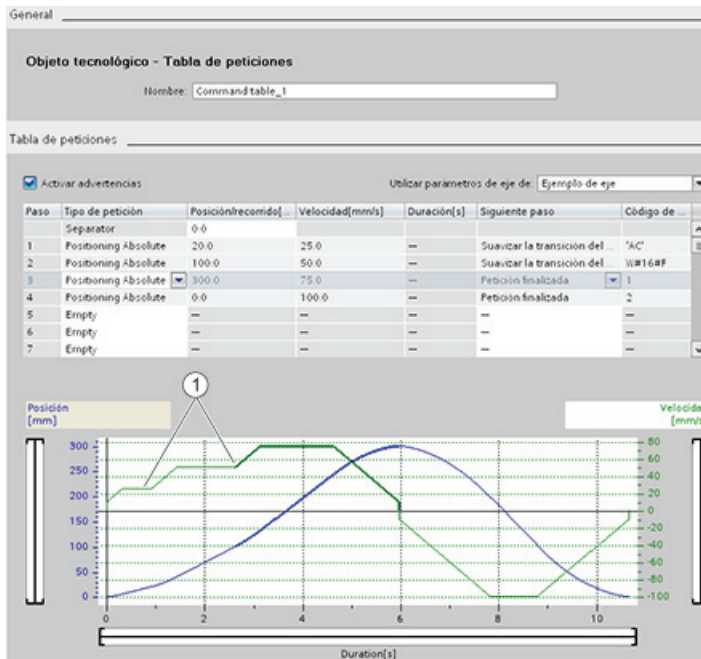
Tipo de comando	Descripción
Velocity setpoint	Mueve el eje a la velocidad dada.
Wait	Espera hasta que finaliza el período determinado. El comando "Wait" no detiene un movimiento de desplazamiento activo.
Separator	Añade una línea de tipo "Separator" encima de la línea seleccionada. La línea separadora permite definir más de un perfil en una misma tabla de comandos.

En la siguiente figura, se utiliza "Command complete" como transición al siguiente paso. Este tipo de transición permite que el dispositivo decelere a la velocidad de arranque/parada y luego vuelva a acelerar al inicio del siguiente paso.



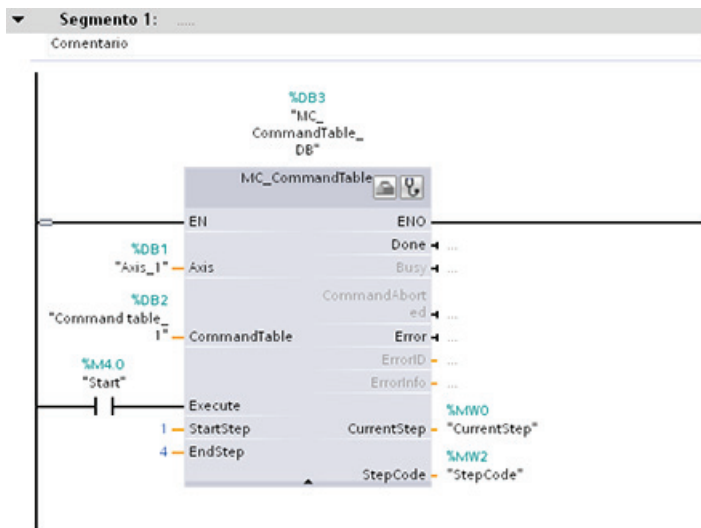
① El eje decelera a la velocidad de arranque/parada entre pasos.

En la siguiente figura, se utiliza "Blending motion" como transición al siguiente paso. Este tipo de transición permite que el dispositivo mantenga su velocidad en el inicio del siguiente paso, por lo que la transición del dispositivo de un paso al siguiente es fluida. El uso de esta transición puede reducir el tiempo total necesario para que un perfil se ejecute por completo. Sin ella, este ejemplo tardaría siete segundos en ejecutarse. Con ella, el tiempo de ejecución se reduce en un segundo, siendo el total de seis segundos.



① El eje sigue moviéndose y acelera o decelera a la velocidad del siguiente paso, ahorrando tiempo y desgaste mecánico.

El funcionamiento de CommandTable se controla mediante una instrucción MC_CommandTable, como se muestra a continuación:



9.3.3 Instrucciones de Motion Control

Nota

La CPU calcula las tareas de movimiento en "fragmentos" o segmentos de 10 ms. Una vez se ha ejecutado un fragmento, el próximo ya está esperando en la cola para ser ejecutado. Si se interrumpe la tarea de movimiento de un eje (ejecutando otra tarea de movimiento para dicho eje), la nueva tarea de movimiento no puede ejecutarse durante un máximo de 20 ms (el resto de la fracción actual más la fracción en cola).

9.3.3.1 Instrucción MC_Power

ATENCIÓN

Si el eje se desconecta debido a un error, se habilitará de nuevo automáticamente una vez que el error haya sido eliminado y acusado. Para ello es necesario que el parámetro de entrada Enable haya conservado el valor TRUE durante el proceso.

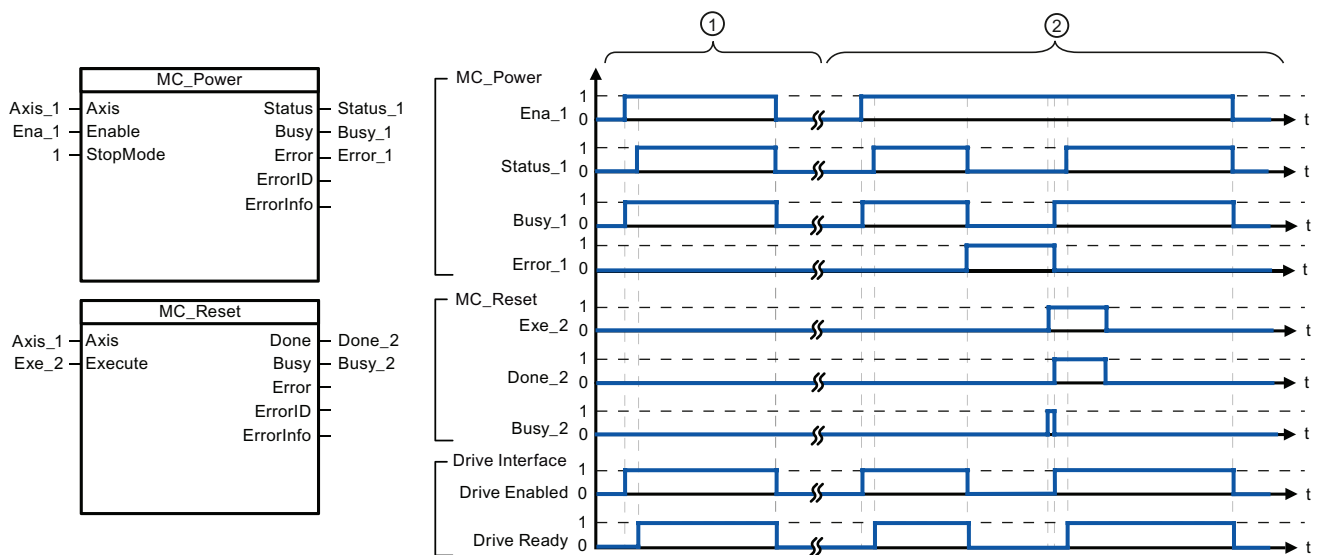
Tabla 9- 26 Instrucción MC_Power

KOP / FUP	SCL	Descripción
	<pre>"MC_Power_DB" (Axis:=_multi_fb_in_, Enable:=_bool_in_, StopMode:=_int_in_, Status=>_bool_out_, Busy=>_bool_out_, Error=>_bool_out_, ErrorID=>_word_out_, ErrorInfo=>_word_out_);</pre>	<p>La instrucción MC_Power de Motion Control habilita o deshabilita un eje. Antes de poder habilitar o deshabilitar el eje, garantice las siguientes condiciones:</p> <ul style="list-style-type: none"> • El objeto tecnológico está configurado correctamente. • No hay ningún error pendiente que impida la habilitación. <p>La ejecución de MC_Power no puede cancelarse por una tarea de Motion Control. La deshabilitación del eje (parámetro de entrada Enable = FALSE) cancela todas las tareas de Motion Control para el objeto tecnológico asociado.</p>

- 1 STEP 7 crea el DB automáticamente al insertar la instrucción.
- 2 En el ejemplo SCL, "MC_Power_DB" es el nombre del DB de instancia.

Tabla 9- 27 Parámetros de la instrucción MC_Power

Parámetro y tipo		Tipo de datos	Descripción
Axis	IN	TO_Axis_1	Objeto tecnológico Eje
Enable	IN	Bool	<ul style="list-style-type: none"> FALSE (predeterminado): Todas las tareas activas se cancelan en función del "StopMode" parametrizado y el eje se detiene. TRUE: Motion Control intenta habilitar el eje.
StopMode	IN	Int	<ul style="list-style-type: none"> 0: Parada de emergencia: Si hay pendiente una solicitud de deshabilitación del eje, el eje se frena con la deceleración de emergencia configurada. El eje se deshabilita una vez que se detiene. 1: Parada inmediata: Si hay pendiente una solicitud de deshabilitación del eje, el eje se deshabilita sin deceleración. La transmisión de impulsos se detiene inmediatamente. 2: Parada de emergencia con control de tirones: Si hay pendiente una solicitud de deshabilitación del eje, el eje se frena con la deceleración de parada de emergencia configurada. Si el control de tirones está activado, los tirones configurados se tienen en cuenta. El eje se deshabilita una vez que se detiene.
Status	OUT	Bool	<p>Se habilita Status del eje:</p> <ul style="list-style-type: none"> FALSE: El eje está deshabilitado: <ul style="list-style-type: none"> El eje no ejecuta tareas de Motion Control y no acepta ninguna tarea nueva (excepción: tarea MC_Reset). El eje no está referenciado. Tras la deshabilitación, el estado no cambia a FALSE hasta que el eje se detiene por completo. TRUE: El eje está habilitado: <ul style="list-style-type: none"> El eje está listo para ejecutar tareas de Motion Control. Tras la habilitación del eje, el estado no cambia a TRUE hasta que esté presente la señal "Accionamiento listo". Si no se ha configurado la interfaz de accionamiento "Accionamiento listo" en la configuración del eje, el estado cambia inmediatamente a TRUE.
Busy	OUT	Bool	<p>FALSE: MC_Power no está activo. TRUE: MC_Power está activo.</p>
Error	OUT	Bool	<p>FALSE: Sin error TRUE: Se ha producido un error en la instrucción de Motion Control "MC_Power" o en el objeto tecnológico asociado. La causa del error se indica en los parámetros "ErrorID" y "ErrorInfo".</p>
ErrorID	OUT	Word	ID de error del parámetro "Error"
ErrorInfo	OUT	Word	ID de info de error del parámetro "ErrorID"



- ① Se activa y, a continuación, se desactiva un eje. Una vez que el accionamiento ha indicado a la CPU que está listo, la activación correcta puede leerse a través de "Status_1".
- ② Tras la activación de un eje, se ha producido un error que ha hecho que el eje se desactive. El error se elimina y se acusa con "MC_Reset". El eje se vuelve a activar.

Para habilitar un eje con interfaz de accionamiento configurada, proceda del siguiente modo:

1. Compruebe que se dan las condiciones arriba indicadas.
2. Inicialice el parámetro de entrada "StopMode" con el valor deseado. Ponga el parámetro de entrada "Enable" a TRUE.

La salida de habilitación para "Accionamiento habilitado" cambia a TRUE para permitir la alimentación eléctrica del accionamiento. La CPU espera la señal de "Accionamiento listo" del accionamiento.

Una vez que la señal "Accionamiento listo" está disponible en la entrada de disponibilidad configurada de la CPU, el eje se habilita. El parámetro de salida "Status" y la variable de objeto tecnológico <Nombre de eje>.StatusBits.Enable tienen el valor TRUE.

Para habilitar un eje sin interfaz de accionamiento configurada, proceda del siguiente modo:

1. Compruebe que se dan las condiciones arriba indicadas.
2. Inicialice el parámetro de entrada "StopMode" con el valor deseado. Ponga el parámetro de entrada "Enable" a TRUE. El eje está habilitado. Parámetro de salida "Status" y variable de objeto tecnológico <Nombre de eje>.StatusBits.Enable tienen el valor TRUE.

Para deshabilitar el eje, proceda del siguiente modo:


1. Haga que el eje se detenga por completo.

Se puede saber cuándo el eje está detenido por completo en la variable de objeto tecnológico <Nombre de eje>.StatusBits.StandStill.

2. Ponga el parámetro de entrada "Enable" a TRUE una vez que el eje se haya parado por completo.
3. Si los parámetros de salida "Busy" y Status y la variable de objeto tecnológico <Nombre de eje>.StatusBits.Enable tienen el valor FALSE, la deshabilitación del eje ha finalizado.

9.3.3.2 Instrucción MC_Reset

Tabla 9- 28 Instrucción MC_Reset

KOP / FUP	SCL	Descripción
	<pre>"MC_Reset_DB" (Axis:=_multi_fb_in_, Execute:=_bool_in_, Restart:=_bool_in_, Done=>_bool_out_, Busy=>_bool_out_, Error=>_bool_out_, ErrorID=>_word_out_, ErrorInfo=>_word_out_);</pre>	<p>Utilice la instrucción MC_Reset para acusar "Error operativo con parada de eje" y "Error de configuración". Los errores que requieren acuse pueden encontrarse en la "Lista de ErrorIDs y ErrorInfos" en "Solución".</p> <p>Antes de utilizar la instrucción MC_Reset es necesario haber eliminado la causa de cualquier error de configuración presente sujeto a acuse (por ejemplo, cambiando un valor de aceleración no válido en el objeto tecnológico "Eje" a un valor válido).</p> <p>En V3.0 y posteriores, el comando Restart permite descargar la configuración del eje a la memoria de trabajo en el modo de operación RUN.</p>

- STEP 7 crea el DB automáticamente al insertar la instrucción.
- En el ejemplo SCL, "MC_Reset_DB" es el nombre del DB de instancia.

La tarea MC_Reset no puede ser interrumpida por otra tarea de Motion Control. Las tareas MC_Reset nuevas no interrumpen ninguna otra tarea de Motion Control activa.

Tabla 9- 29 Parámetros de la instrucción MC_Reset

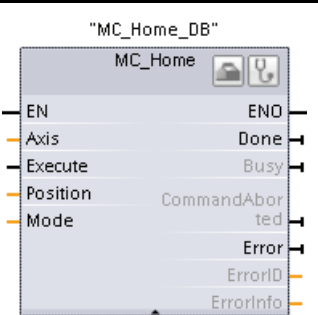
Parámetro y tipo	Tipo de datos	Descripción
Axis	IN	TO_Axis_1 Objeto tecnológico Eje
Execute	IN	Bool Inicio de la tarea con flanco positivo
Restart	IN	Bool TRUE = Descargar la configuración del eje desde la memoria de carga a la memoria de trabajo. El comando sólo se puede ejecutar con el eje deshabilitado. FALSE = Acusa el recibo de los errores pendientes
Done	OUT	Bool TRUE = El error ha sido acusado.
Busy	OUT	Bool TRUE = La tarea está siendo ejecutada.
Error	OUT	Bool TRUE = Se ha producido un error durante la ejecución de la tarea. La causa del error se indica en los parámetros "ErrorID" y "ErrorInfo".
ErrorID	OUTP	Word ID de error del parámetro "Error"
ErrorInfo	OUT	Word ID de info de error del parámetro "ErrorID"

Para acusar un error con MC_Reset, proceda del siguiente modo:

- Compruebe que se dan las condiciones arriba indicadas.
- Inicie el acuse del error con un flanco ascendente en el parámetro de entrada Execute.
- El error ha sido acusado cuando Done es TRUE y la variable <Nombre de eje>.StatusBits.Error del objeto tecnológico es FALSE.

9.3.3.3 Instrucción MC_Home

Tabla 9- 30 Instrucción MC_Home

KOP / FUP	SCL	Descripción
	<pre>"MC_Home_DB" (Axis:=_multi_fb_in_, Execute:=_bool_in_, Position:=_real_in_, Mode:=_int_in_, Done=>_bool_out_, Busy=>_bool_out_, CommandAborted=>_bool_out_, Error=>_bool_out_, ErrorID=>_word_out_, ErrorInfo=>_word_out_);</pre>	<p>Utilice la instrucción MC_Home para cuadrar las coordenadas del eje con la posición física real del accionamiento. Se requiere una referenciación para posicionar el eje de forma absoluta:</p> <p>Para utilizar la instrucción MC_Home primero es necesario haber habilitado el eje.</p>

- STEP 7 crea el DB automáticamente al insertar la instrucción.
- En el ejemplo SCL, "MC_Home_DB" es el nombre del DB de instancia.

Están disponibles los siguientes tipos de referenciación:

- Referenciación directa absoluta (Mode = 0): La posición actual del eje se ajusta al valor del parámetro "Position".
- Referenciación directa relativa (Mode = 1): La posición actual del eje se desplaza según el valor del parámetro "Position".
- Referenciación pasiva (Mode = 2): Durante la referenciación pasiva, la instrucción MC_Home no realiza ningún movimiento de referenciación. El movimiento necesario para este paso debe ser implementado por el usuario mediante otras instrucciones de Motion Control. El eje está referenciado cuando se detecta el interruptor de punto de referencia.
- Referenciación activa (Mode = 3): El procedimiento de referenciación se ejecuta automáticamente.

Tabla 9- 31 Parámetros de la instrucción MC_Home

Parámetro y tipo	Tipo de datos	Descripción
Axis	IN	TO_Axis_PTO Objeto tecnológico Eje
Execute	IN	Bool Inicio de la tarea con flanco positivo
Position	IN	Real <ul style="list-style-type: none"> Mode = 0, 2 y 3 (posición absoluta del eje tras haber completado la operación de referenciación) Mode = 1 (Valor de corrección de la posición actual del eje) Valores límite: $-1.0e^{12} \leq \text{Position} \leq 1.0e^{12}$

Parámetro y tipo		Tipo de datos	Descripción
Mode	IN	Int	<p>Modo de referenciación</p> <ul style="list-style-type: none"> 0: Referenciación directa absoluta La nueva posición del eje es el valor de posición del parámetro "Position". 1: Referenciación directa relativa La nueva posición del eje es la posición actual del eje + el valor de posición del parámetro "Position". 2: Referenciación pasiva Referenciación acorde con la configuración del eje. Tras la referenciación, el valor del parámetro "Position" se ajusta como la nueva posición del eje. 3: Referenciación activa Aproximación al punto de referencia conforme a la configuración del eje. Tras la referenciación, el valor del parámetro "Position" se ajusta como la nueva posición del eje.
Done	OUT	Bool	TRUE = Tarea completada
Busy	OUT	Bool	TRUE = La tarea está siendo ejecutada.
CommandAborted	OUT	Bool	TRUE = La tarea ha sido interrumpida por otra durante la ejecución.
Error	OUT	Bool	TRUE = Se ha producido un error durante la ejecución de la tarea. La causa del error se indica en los parámetros "ErrorID" y "ErrorInfo".
ErrorID	OUT	Word	ID de error del parámetro "Error"
ErrorInfo	OUT	Word	ID de info de error del parámetro "ErrorID"

Nota

La referenciación del eje se pierde en las siguientes condiciones

- Deshabilitación del eje por la instrucción MC_Power
- Conmutación entre control automático y manual
- Una vez iniciada una referenciación activa (tras haber completado correctamente la operación de referenciación, la referenciación del eje vuelve a estar disponible).
- Tras desconectar y volver a conectar la alimentación de la CPU.
- Tras rearranque de la CPU (RUN a STOP o STOP a RUN)

Para referenciar el eje, proceda del siguiente modo:

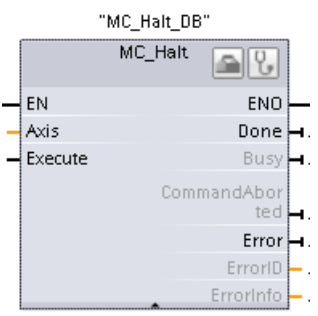
1. Compruebe que se dan las condiciones arriba indicadas.
2. Inicializar los parámetros de entrada necesarios con valores e iniciar la operación de referenciación con un flanco ascendente en el parámetro de entrada "Execute".
3. Si el parámetro de salida "Done" y la variable de objeto tecnológico <Nombre de eje>.StatusBits.HomingDone tienen el valor TRUE, la referenciación ha finalizado.

Tabla 9- 32 Respuesta de corrección

Modo	Descripción	
0 ó 1	La tarea MC_Home no puede ser interrumpida por otra tarea de Motion Control. La tarea MC_Home nueva no interrumpe ninguna otra tarea de Motion Control activa. Las tareas de movimiento relacionadas con la posición se reanudan tras la referenciación de acuerdo con la posición de referenciación (valor depositado en el parámetro de entrada Position).	
2	La tarea MC_Home puede ser interrumpida por las siguientes tareas de Motion Control: Tarea MC_Home, Mode = 2, 3: La tarea MC_Home nueva interrumpe las siguientes tareas de Motion Control activas. Tarea MC_Home, Mode = 2: Las tareas de movimiento relacionadas con la posición se reanudan tras la referenciación de acuerdo con la posición de referenciación (valor depositado en el parámetro de entrada Position).	
3	La tarea MC_Home puede ser interrumpida por las siguientes tareas de Motion Control: <ul style="list-style-type: none"> • MC_Home Mode = 3 • MC_Halt • MC_MoveAbsolute • MC_MoveRelative • MC_MoveVelocity • MC_MoveJog 	La tarea MC_Home nueva interrumpe las siguientes tareas de Motion Control activas: <ul style="list-style-type: none"> • Modo MC_Home = 2, 3 • MC_Halt • MC_MoveAbsolute • MC_MoveRelative • MC_MoveVelocity • MC_MoveJog

9.3.3.4 Instrucción MC_Halt

Tabla 9- 33 Instrucción MC_Halt

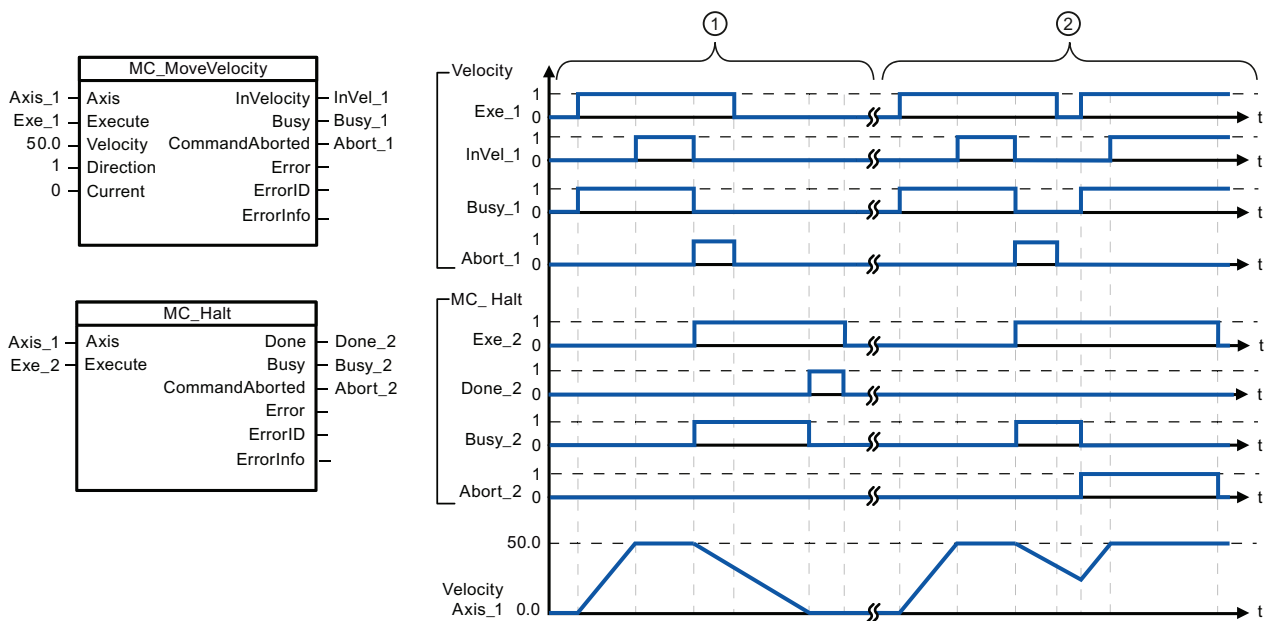
KOP / FUP	SCL	Descripción
	<pre>"MC_Halt_DB" (Axis:= multi_fb_in_, Execute:=_bool_in_, Done=>_bool_out_, Busy=>_bool_out_, CommandAborted=>_bool_out_, Error=>_bool_out_, ErrorID=>_word_out_, ErrorInfo=>_word_out_);</pre>	<p>Utilice la instrucción MC_Halt para parar todo el movimiento y detener el eje por completo. La posición de eje totalmente detenido no está definida.</p> <p>Para utilizar la instrucción MC_Halt primero es necesario haber habilitado el eje.</p>

- 1 STEP 7 crea el DB automáticamente al insertar la instrucción.
- 2 En el ejemplo SCL, "MC_Halt_DB" es el nombre del DB de instancia.

Tabla 9- 34 Parámetros de la instrucción MC_Halt

Parámetro y tipo	Tipo de datos	Descripción
Axis	IN	TO_Axis_1
Execute	IN	Bool
		Objeto tecnológico Eje
		Inicio de la tarea con flanco positivo

Parámetro y tipo	Tipo de datos	Descripción	
Done	OUT	Bool	TRUE = Alcanzada la velocidad cero
Busy	OUT	Bool	TRUE = La tarea está siendo ejecutada.
CommandAborted	OUT	Bool	TRUE = La tarea ha sido interrumpida por otra durante la ejecución.
Error	OUT	Bool	TRUE = Se ha producido un error durante la ejecución de la tarea. La causa del error se indica en los parámetros "ErrorID" y "ErrorInfo".
ErrorID	OUT	Word	ID de error del parámetro "Error"
ErrorInfo	OUT	Word	ID de info de error del parámetro "ErrorID"



Los siguientes valores se han configurado en la ventana "Animaciones > General": Aceleración = 10.0 y deceleración = 5.0

- ① Una tarea MC_Halt frena el eje hasta que éste se para por completo. El estado de "eje parado" se asigna vía "Done_2".
- ② Mientras una tarea MC_Halt esté frenando el eje, la tarea en cuestión quedará interrumpida por cualquier otra tarea de movimiento. La interrupción se indica a través de "Abort_2".

Respuesta de corrección

La tarea MC_Halt puede ser interrumpida por las siguientes tareas de Motion Control.

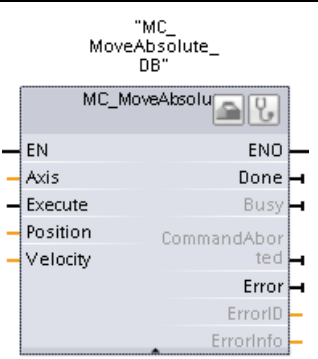
- MC_Home Mode = 3
- MC_Halt
- MC_MoveAbsolute
- MC_MoveRelative
- MC_MoveVelocity
- MC_MoveJog

La tarea MC_Halt nueva interrumpe las siguientes tareas de Motion Control activas:

- MC_Home Mode = 3
- MC_Halt
- MC_MoveAbsolute
- MC_MoveRelative
- MC_MoveVelocity
- MC_MoveJog

9.3.3.5 Instrucción MC_MoveAbsolute

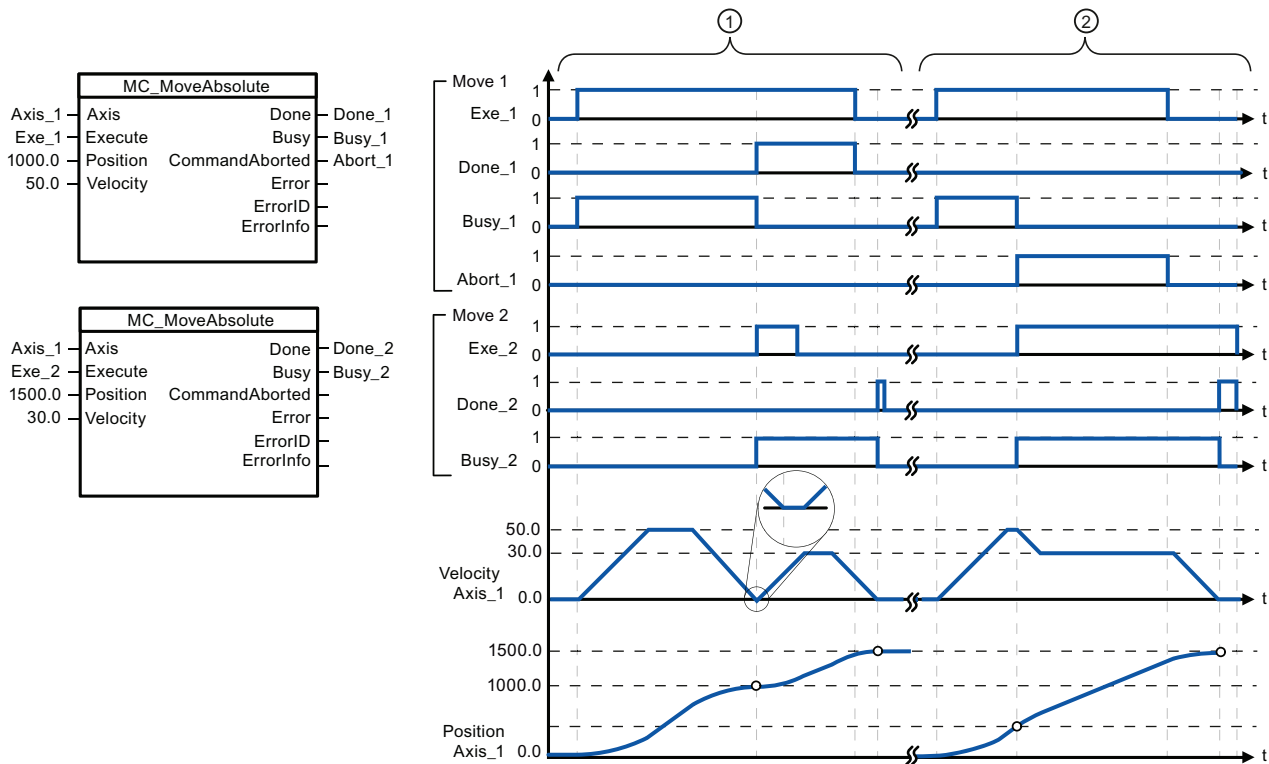
Tabla 9- 35 Instrucción MC_MoveAbsolute

KOP / FUP	SCL	Descripción
	<pre>"MC_MoveAbsolute_DB" (Axis:=_multi_fb_in_, Execute:=_bool_in_, Position:=_real_in_, Velocity:=_real_in_, Done=>_bool_out_, Busy=>_bool_out_, CommandAborted=>_bool_out_, Error=>_bool_out_, ErrorID=>_word_out_, ErrorInfo=>_word_out_);</pre>	<p>Utilice la instrucción MC_MoveAbsolute para iniciar un movimiento de posicionamiento del eje a una posición absoluta.</p> <p>Para utilizar la instrucción MC_MoveAbsolute primero es necesario haber habilitado y referenciado el eje.</p>

- 1 STEP 7 crea el DB automáticamente al insertar la instrucción.
- 2 En el ejemplo SCL, "MC_MoveAbsolute_DB" es el nombre del DB de instancia.

Tabla 9- 36 Parámetros de la instrucción MC_MoveAbsolute

Parámetro y tipo	Tipo de datos	Descripción
Axis	IN	TO_Axis_1 Objeto tecnológico Eje
Execute	IN	Bool Inicio de la tarea con flanco positivo (valor predeterminado: False)
Position	IN	Real Posición de destino absoluta (valor predeterminado: 0.0) Valores límite: $-1.0e^{12} \leq \text{Position} \leq 1.0e^{12}$
Velocity	IN	Real Velocidad del eje (valor predeterminado: 10.0) Esta velocidad no siempre se alcanza, debido a la aceleración y deceleración configurada y a la posición de destino a la que hay que aproximarse. Valores límite: $\text{velocidad inicio/parada} \leq \text{Velocity} \leq \text{velocidad máxima}$
Done	OUT	Bool TRUE = Posición de destino absoluta alcanzada
Busy	OUT	Bool TRUE = La tarea está siendo ejecutada.
CommandAborted	OUT	Bool TRUE = La tarea ha sido interrumpida por otra durante la ejecución.
Error	OUT	Bool TRUE = Se ha producido un error durante la ejecución de la tarea. La causa del error se indica en los parámetros "ErrorID" y "ErrorInfo".
ErrorID	OUT	Word ID de error del parámetro "Error" (valor predeterminado: 0000)
ErrorInfo	OUT	Word ID de información de error del parámetro "ErrorID" (valor predeterminado: 0000)



Los siguientes valores se han configurado en la ventana "Animaciones > General": Aceleración = 10.0 y deceleración = 10.0

- ① Un eje se desplaza a la posición absoluta 1000,0 con una tarea MC_MoveAbsolute. Cuando el eje alcanza la posición de destino, se indica a través de "Done_1". Cuando "Done_1" = TRUE, se inicia otra tarea MC_MoveAbsolute, con la posición de destino 1500.0. Debido a los tiempos de respuesta (por ejemplo, tiempo de ciclo del programa de usuario, etc.), el eje se para brevemente (véase el detalle ampliado). Cuando el eje alcanza la nueva posición de destino, esto se indica a través de "Done_2".
- ② Una tarea MC_MoveAbsolute activa queda interrumpida por otra tarea MC_MoveAbsolute. La interrupción se indica a través de "Abort_1". A continuación, el eje se desplaza a la nueva velocidad hasta la nueva posición de destino 1500.0. Cuando se alcanza la nueva posición de destino, esto se indica a través de "Done_2".

Respuesta de corrección

La tarea MC_MoveAbsolute puede ser interrumpida por las siguientes tareas de Motion Control.

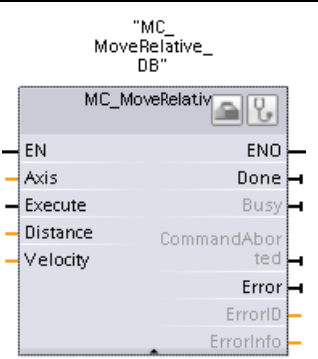
- MC_Home Mode = 3
- MC_Halt
- MC_MoveAbsolute
- MC_MoveRelative
- MC_MoveVelocity
- MC_MoveJog

La tarea MC_MoveAbsolute nueva interrumpe las siguientes tareas de Motion Control activas:

- MC_Home Mode = 3
- MC_Halt
- MC_MoveAbsolute
- MC_MoveRelative
- MC_MoveVelocity
- MC_MoveJog

9.3.3.6 Instrucción MC_MoveRelative

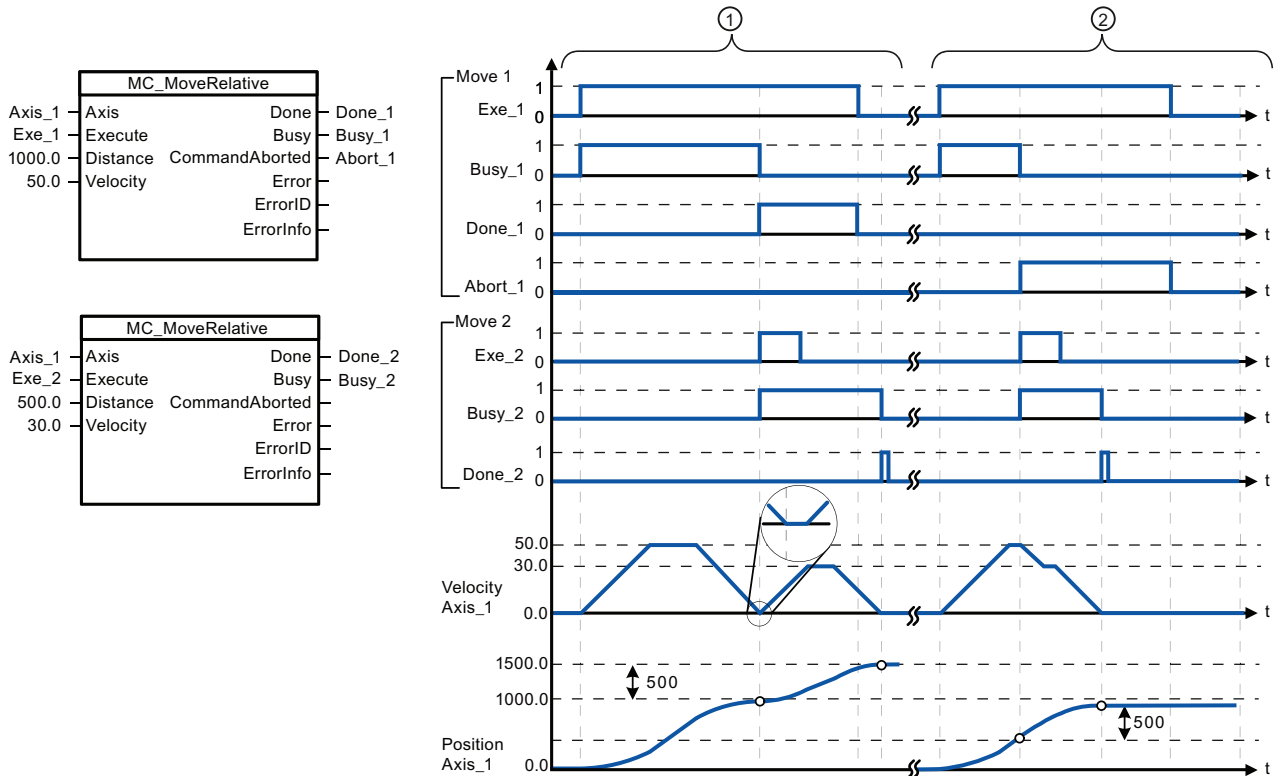
Tabla 9- 37 Instrucción MC_MoveRelative

KOP / FUP	SCL	Descripción
	<pre>"MC_MoveRelative_DB" (Axis:=_multi_fb_in_, Execute:=_bool_in_, Distance:=_real_in_, Velocity:=_real_in_, Done=>_bool_out_, Busy=>_bool_out_, CommandAborted=>_bool_out_, Error=>_bool_out_, ErrorID=>_word_out_, ErrorInfo=>_word_out_);</pre>	<p>Utilice la instrucción MC_MoveRelative para iniciar un movimiento de posicionamiento relativo a la posición inicial.</p> <p>Para utilizar la instrucción MC_MoveRelative primero es necesario haber habilitado el eje.</p>

- 1 STEP 7 crea el DB automáticamente al insertar la instrucción.
- 2 En el ejemplo SCL, "MC_MoveRelative_DB" es el nombre del DB de instancia.

Tabla 9- 38 Parámetros de la instrucción MC_MoveRelative

Parámetro y tipo	Tipo de datos	Descripción
Axis	IN	TO_Axis_1 Objeto tecnológico Eje
Execute	IN	Bool Inicio de la tarea con flanco positivo (valor predeterminado: False)
Distance	IN	Real Distancia de desplazamiento para la operación de posicionamiento: 0.0 Valores límite: $-1.0e^{12} \leq \text{Distance} \leq 1.0e^{12}$
Velocity	IN	Real Velocidad del eje (valor predeterminado: 10.0) Esta velocidad no siempre se alcanza debido a la aceleración y deceleración configurada y a la distancia que debe recorrerse. Valores límite: Velocidad de inicio/parada $\leq \text{Velocity} \leq$ velocidad máxima
Done	OUT	Bool TRUE = Posición de destino alcanzada
Busy	OUT	Bool TRUE = La tarea está siendo ejecutada.
CommandAborted	OUT	Bool TRUE = La tarea ha sido interrumpida por otra durante la ejecución.
Error	OUT	Bool TRUE = Se ha producido un error durante la ejecución de la tarea. La causa del error se indica en los parámetros "ErrorID" y "ErrorInfo".
ErrorID	OUT	Word ID de error del parámetro "Error" (valor predeterminado: 0000)
ErrorInfo	OUT	Word ID de información de error del parámetro "ErrorID" (valor predeterminado: 0000)



Los siguientes valores se han configurado en la ventana "Animaciones > General": Aceleración = 10.0 y deceleración = 10.0

- ① El eje se mueve con la tarea MC_MoveRelative durante la distancia ("Distance") 1000.0. Cuando el eje alcanza la posición de destino, se indica a través de "Done_1". Cuando "Done_1" = TRUE, se inicia otra tarea MC_MoveRelative, con una distancia de desplazamiento 500.0. Debido a los tiempos de respuesta (por ejemplo, tiempo de ciclo del programa de usuario), el eje se para brevemente (véase el detalle ampliado). Cuando el eje alcanza la nueva posición de destino, esto se indica a través de "Done_2".
- ② Una tarea MC_MoveRelative activa queda interrumpida por otra tarea MC_MoveRelative. La interrupción se indica a través de "Abort_1". A continuación, el eje se desplaza a la nueva velocidad con la nueva distancia ("Distance") 500.0. Cuando se alcanza la nueva posición de destino, esto se indica a través de "Done_2".

Respuesta de corrección

La tarea MC_MoveRelative puede ser interrumpida por las siguientes tareas de Motion Control.

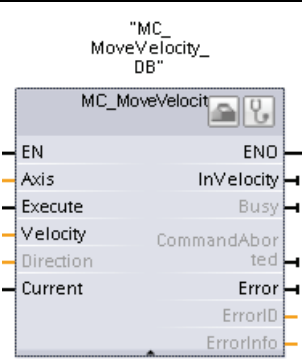
- MC_Home Mode = 3
- MC_Halt
- MC_MoveAbsolute
- MC_MoveRelative
- MC_MoveVelocity
- MC_MoveJog

La tarea MC_MoveRelative nueva interrumpe las siguientes tareas de Motion Control activas:

- MC_Home Mode = 3
- MC_Halt
- MC_MoveAbsolute
- MC_MoveRelative
- MC_MoveVelocity
- MC_MoveJog

9.3.3.7 Instrucción MC_MoveVelocity

Tabla 9- 39 Instrucción MC_MoveVelocity

KOP / FUP	SCL	Descripción
	<pre>"MC_MoveVelocity_DB" (Axis:=_multi_fb_in_, Execute:=_bool_in_, Velocity:=_real_in_, Direction:=_int_in_, Current:=_bool_in_, InVelocity=>_bool_out_, Busy=>_bool_out_, CommandAborted=>_bool_out_, Error=>_bool_out_, ErrorID=>_word_out_, ErrorInfo=>_word_out_);</pre>	<p>Utilice la instrucción MC_MoveVelocity para mover el eje constantemente a la velocidad especificada.</p> <p>Para utilizar la instrucción MC_MoveVelocity primero es necesario haber habilitado el eje.</p>

- 1 STEP 7 crea el DB automáticamente al insertar la instrucción.
- 2 En el ejemplo SCL, "MC_MoveVelocity_DB" es el nombre del DB de instancia.

Tabla 9- 40 Parámetros de la instrucción MC_MoveVelocity

Parámetro y tipo	Tipo de datos	Descripción
Axis	IN	TO_Axis_1
Execute	IN	Bool
Velocity	IN	Real
Direction	IN	Int
Current	IN	Bool

Objeto tecnológico Eje

Inicio de la tarea con flanco positivo (valor predeterminado: False)

Especificación de velocidad para movimiento del eje (valor predeterminado: 10.0)
Valores límite: Velocidad de inicio/parada \leq |Velocity| \leq velocidad máxima
(Se permite Velocity = 0,0)

Especificación de dirección:

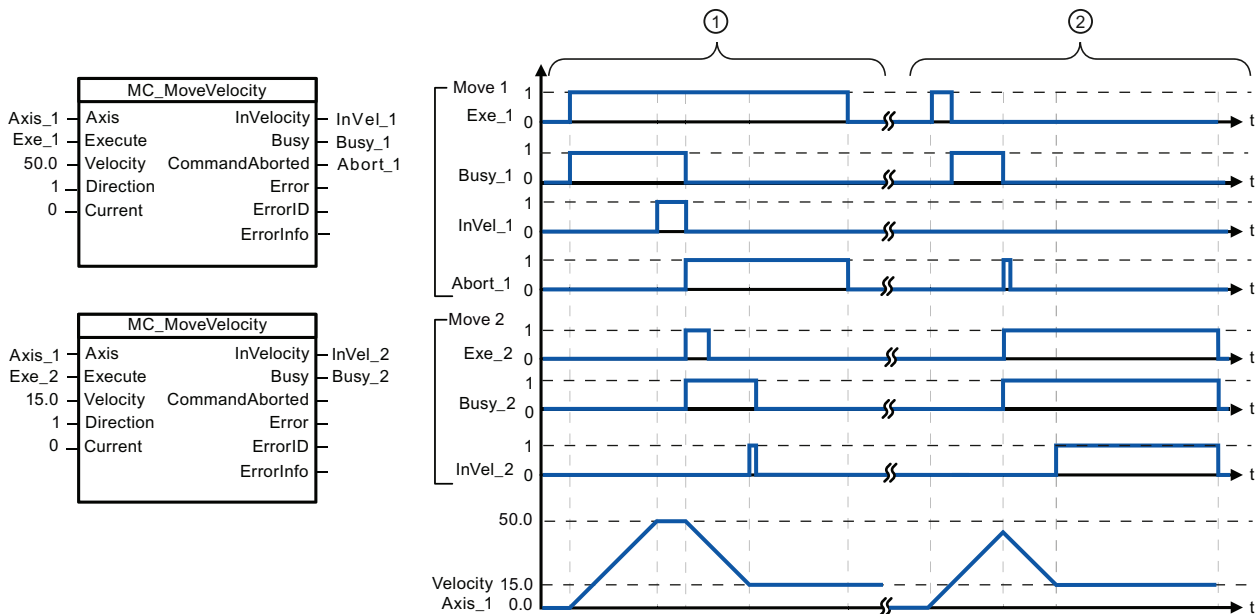
- 0: la dirección de rotación corresponde a la señal del valor depositado en el parámetro "Velocity" (valor predeterminado)
- 1: dirección positiva de rotación (la señal del valor del parámetro "Velocity" se ignora).
- 2: dirección negativa de rotación (la señal del valor del parámetro "Velocity" se ignora).

Mantener velocidad actual:

- FALSE: "Mantener velocidad actual" está desactivado. Se utilizan los valores de los parámetros "Velocity" y "Direction". (Valor predeterminado)
- TRUE: "Mantener velocidad actual" está activado. Los valores de los parámetros "Velocity" y "Direction" no se tienen en cuenta.

Cuando el eje retoma el movimiento a la velocidad actual, el parámetro "InVelocity" vuelve a adoptar el valor TRUE.

Parámetro y tipo		Tipo de datos	Descripción
InVelocity	OUT	Bool	TRUE: <ul style="list-style-type: none"> Si "Current" = FALSE: Se ha alcanzado la velocidad especificada en el parámetro "Velocity". Si "Current" = TRUE: El eje se desplaza a la velocidad actual en el tiempo inicial.
Busy	OUT	Bool	TRUE = La tarea está siendo ejecutada.
CommandAborted	OUT	Bool	TRUE = La tarea ha sido interrumpida por otra durante la ejecución.
Error	OUT	Bool	TRUE = Se ha producido un error durante la ejecución de la tarea. La causa del error se indica en los parámetros "ErrorID" y "ErrorInfo".
ErrorID	OUT	Word	ID de error del parámetro "Error" (valor predeterminado: 0000)
ErrorInfo	OUT	Word	ID de información de error del parámetro "ErrorID" (valor predeterminado: 0000)



Los siguientes valores se han configurado en la ventana "Animaciones > General": Aceleración = 10.0 y deceleración = 10.0

- ① Una tarea MC_MoveVelocity activa señala con "InVel_1" que se ha alcanzado la velocidad de destino. Entonces se interrumpe por otra tarea MC_MoveVelocity. La interrupción se indica a través de "Abort_1". Cuando se alcanza la nueva velocidad de destino 15.0, esto se indica a través de "InVel_2". El eje sigue desplazándose a la nueva velocidad constante.
- ② Una tarea MC_MoveVelocity activa queda interrumpida por otra tarea MC_MoveVelocity antes de alcanzar la velocidad de destino. La interrupción se indica a través de "Abort_1". Cuando se alcanza la nueva velocidad de destino 15.0, esto se indica a través de "InVel_2". El eje sigue desplazándose a la nueva velocidad constante.

Respuesta de corrección

La tarea MC_MoveVelocity puede ser interrumpida por las siguientes tareas de Motion Control:

- MC_Home Mode = 3
- MC_Halt
- MC_MoveAbsolute
- MC_MoveRelative
- MC_MoveVelocity
- MC_MoveJog

La tarea MC_MoveVelocity nueva interrumpe las siguientes tareas de Motion Control activas:

- MC_Home Mode = 3
- MC_Halt
- MC_MoveAbsolute
- MC_MoveRelative
- MC_MoveVelocity
- MC_MoveJog

Nota

Comportamiento con velocidad ajustada cero (Velocity = 0.0)

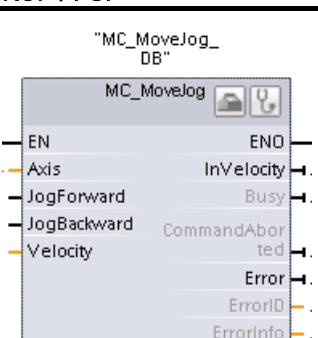
Una tarea MC_MoveVelocity con "Velocity" = 0.0 (como una tarea MC_Halt) interrumpe cualquier tarea activa de Motion Control con la deceleración configurada. Cuando el eje se para, el parámetro de salida "InVelocity" indica TRUE como mínimo durante un ciclo.

"Busy" indica el valor TRUE durante la operación de deceleración y cambia a FALSE junto con "InVelocity". Si se activa el parámetro "Execute" = TRUE, "InVelocity" y "Busy" están enclavados.

Cuando la tarea MC_MoveVelocity ha comenzado, el bit de estado "SpeedCommand" se activa en el objeto tecnológico en cuestión. El bit de estado "ConstantVelocity" se activa cuando el eje se detiene por completo. Ambos bits se adaptan a la nueva situación cuando comienza una nueva tarea de Motion Control.

9.3.3.8 Instrucción MC_MoveJog

Tabla 9- 41 Instrucción MC_MoveJog

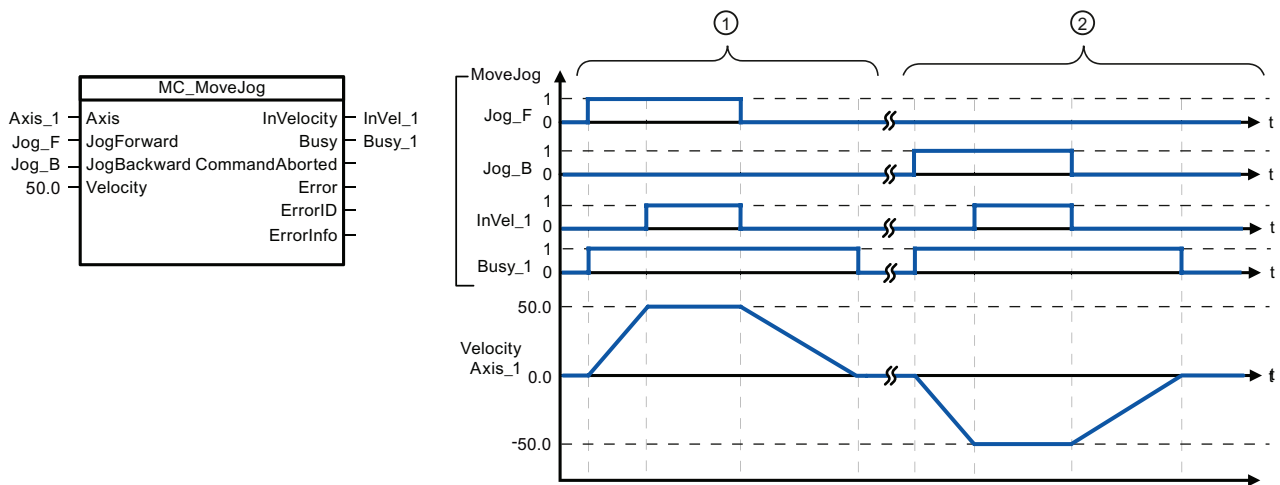
KOP / FUP	SCL	Descripción
	<pre>"MC_MoveJog_DB" (Axis:= _multi_fb_in_, JogForward:= _bool_in_, JogBackward:= _bool_in_, Velocity:= _real_in_, InVelocity=> _bool_out_, Busy=> _bool_out_, CommandAborted=> _bool_out_, Error=> _bool_out_, ErrorID=> _word_out_, ErrorInfo=> _word_out_);</pre>	<p>Utilice la instrucción MC_MoveJog para mover el eje constantemente a la velocidad específica en modo paso a paso. Esta instrucción se suele utilizar con fines de prueba y mantenimiento.</p> <p>Para utilizar la instrucción MC_MoveJog primero es necesario haber habilitado el eje.</p>

- 1 STEP 7 crea el DB automáticamente al insertar la instrucción.
- 2 En el ejemplo SCL, "MC_MoveJog_DB" es el nombre del DB de instancia.

Tabla 9- 42 Parámetros de la instrucción MC_MoveJog

Parámetro y tipo		Tipo de datos	Descripción
Axis	IN	TO_Axis_1	Objeto tecnológico Eje
JogForward ¹	IN	Bool	Mientras el parámetro sea TRUE, el eje se mueve en dirección positiva a la velocidad especificada en el parámetro "Velocity". La señal del valor del parámetro "Velocity" se ignora. (Valor predeterminado: False)
JogBackward ¹	IN	Bool	Mientras el parámetro sea TRUE, el eje se mueve en dirección negativa a la velocidad especificada en el parámetro "Velocity". La señal del valor del parámetro "Velocity" se ignora. (Valor predeterminado: False)
Velocity	IN	Real	Predeterminar velocidad para modo paso a paso (valor predeterminado: 10.0) Valores límite: Velocidad de inicio/parada ≤ Velocity ≤ velocidad máxima
InVelocity	OUT	Bool	TRUE = Se ha alcanzado la velocidad especificada en el parámetro "Velocity".
Busy	OUT	Bool	TRUE = La tarea está siendo ejecutada.
CommandAborted	OUT	Bool	TRUE = La tarea ha sido interrumpida por otra durante la ejecución.
Error	OUT	Bool	TRUE = Se ha producido un error durante la ejecución de la tarea. La causa del error se indica en los parámetros "ErrorID" y "ErrorInfo".
ErrorID	OUT	Word	ID de error del parámetro "Error" (valor predeterminado: 0000)
ErrorInfo	OUT	Word	ID de información de error del parámetro "ErrorID" (valor predeterminado: 0000)

¹ Si ambos parámetros JogForward y JogBackward son TRUE al mismo tiempo, el eje se detiene con la deceleración configurada. Un error se indica en los parámetros "Error", "ErrorID" y "ErrorInfo".



Los siguientes valores se han configurado en la ventana "Animaciones > General": Aceleración = 10.0 y deceleración = 5.0

- ① El eje se mueve en dirección positiva en modo paso a paso con "Jog_F". Cuando se alcanza la velocidad de destino 50.0, esto se indica a través de "InVelo_1". El eje se frena hasta parar de nuevo tras la inicialización de Jog_F.
- ② El eje se mueve en dirección negativa en modo paso a paso con "Jog_B". Cuando se alcanza la velocidad de destino 50.0, esto se indica a través de "InVelo_1". El eje se frena hasta parar de nuevo tras la inicialización de Jog_B.

Respuesta de corrección

La tarea MC_MoveJog puede ser interrumpida por las siguientes tareas de Motion Control.

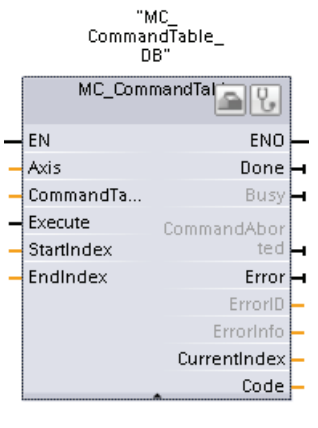
- MC_Home Mode = 3
- MC_Halt
- MC_MoveAbsolute
- MC_MoveRelative
- MC_MoveVelocity
- MC_MoveJog

La tarea MC_MoveJog nueva interrumpe las siguientes tareas de Motion Control activas:

- MC_Home Mode = 3
- MC_Halt
- MC_MoveAbsolute
- MC_MoveRelative
- MC_MoveVelocity
- MC_MoveJog

9.3.3.9 Instrucción MC_CommandTable

Tabla 9- 43 Instrucción MC_CommandTable

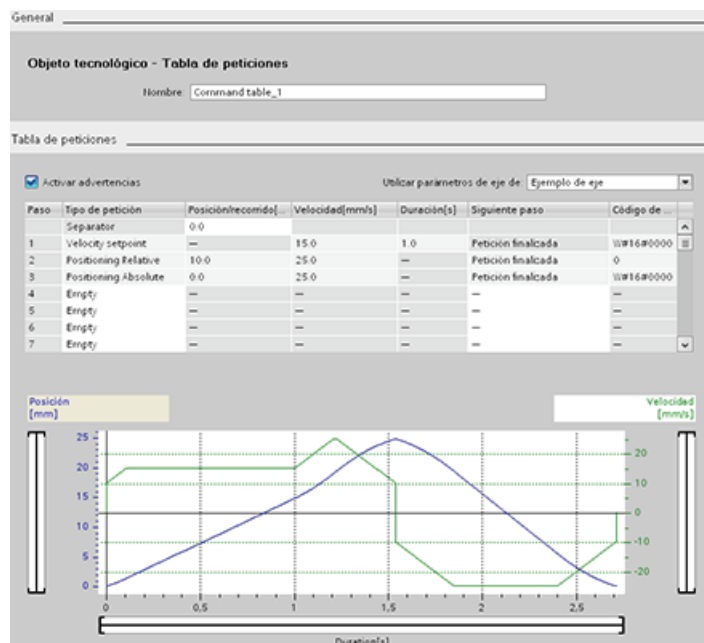
KOP / FUP	SCL	Descripción
	<pre>"MC_CommandTable_DB" (Axis:=_multi_fb_in_, CommandTable:=_multi_fb_in_, Execute:=_bool_in_, StartIndex:=_uint_in_, EndIndex:=_uint_in_, Done=>_bool_out_, Busy=>_bool_out_, CommandAborted=>_bool_out_, Error=>_bool_out_, ErrorID=>_word_out_, ErrorInfo=>_word_out_, CurrentIndex=>_uint_out_, Code=>_word_out_);</pre>	<p>Ejecuta una serie de movimientos individuales para un eje controlado por motor que se combinan en una secuencia de movimientos.</p> <p>Los movimientos individuales están configurados en la tabla de comandos de un objeto tecnológico para la salida del tren de impulsos (TO_CommandTable_PTO).</p>

- STEP 7 crea el DB automáticamente al insertar la instrucción.
- En el ejemplo SCL, "MC_CommandTable_DB" es el nombre del DB de instancia.

Tabla 9- 44 Parámetros de la instrucción MC_CommandTable

Parámetro y tipo	Tipo de datos	Valor inicial	Descripción	
Axis	IN	TO_Axis_1	-	Objeto tecnológico Eje
Table	IN	TO_CommandTable_1	-	Tabla de comandos de objeto tecnológico
Execute	IN	Bool	FALSE	Iniciar tarea con flanco ascendente
StartIndex	IN	Int	1	Iniciar el procesamiento de la tabla de comandos con este paso Límites: $1 \leq \text{StartIndex} \leq \text{EndIndex}$
EndIndex	IN	Int	32	Finalizar el procesamiento de la tabla de comandos con este paso Límites: $\text{StartIndex} \leq \text{EndIndex} \leq 32$
Done	OUT	Bool	FALSE	Procesamiento de MC_CommandTable completado correctamente
Busy	OUT	Bool	FALSE	Operación en marcha.
CommandAborted	OUT	Bool	FALSE	La tarea ha sido interrumpida por otra tarea durante el procesamiento.
Error	OUT	Bool	FALSE	Se ha producido un error durante el procesamiento. La causa se indica en los parámetros ErrorID y ErrorInfo.
ErrorID	OUT	Word	16#0000	Identificador del error
ErrorInfo	OUT	Word	16#0000	Información de error
Step	OUT	Int	0	Paso actualmente en curso
Code	OUT	Word	16#0000	Identificador definido por usuario del paso actualmente en curso

La secuencia de movimientos deseada se puede crear en la ventana de configuración "Tabla de comandos" y comprobar el resultado con la vista gráfica del diagrama de tendencias.



Se pueden seleccionar los tipos de comandos que se van a utilizar para procesar la tabla de comandos. Se pueden introducir hasta 32 tareas. Los comandos se procesan en secuencia.

Tabla 9- 45 Tipos de comandos de MC_CommandTable

Tipo de comando	Descripción
Empty	El comando vacío sirve como comodín para los comandos que se añadan. La entrada vacía se ignora cuando se procesa la tabla de comandos.
Halt	Detiene el eje. Nota: El comando sólo tiene lugar tras un comando "Velocity setpoint".
Positioning Relative	Coloca el eje basándose en la distancia. El comando mueve el eje según la distancia y la velocidad dadas.
Positioning Absolute	Coloca el eje basándose en la posición. El comando mueve el eje a la posición indicada, a la velocidad especificada.
Velocity setpoint	Mueve el eje a la velocidad dada.
Wait	Espera hasta que finaliza el período determinado. El comando "Wait" no detiene un movimiento de desplazamiento activo.
Separator	Añade una línea de tipo "Separator" encima de la línea seleccionada. La línea separadora permite definir más de un perfil en una misma tabla de comandos.

Requisitos para la ejecución de MC_CommandTable:

- El objeto tecnológico TO_Axis_PTO V2.0 debe estar configurado correctamente.
- El objeto tecnológico TO_CommandTable_PTO debe estar configurado correctamente.
- El eje debe estar habilitado.

Respuesta de corrección

La tarea MC_CommandTable puede ser interrumpida por las siguientes tareas de Motion Control.

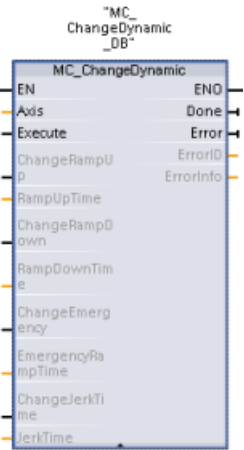
- MC_Home Mode = 3
- MC_Halt
- MC_MoveAbsolute
- MC_MoveRelative
- MC_MoveVelocity
- MC_MoveJog
- MC_CommandTable

La tarea MC_CommandTable nueva interrumpe las siguientes tareas de Motion Control activas:

- MC_Home Mode = 3
- MC_Halt
- MC_MoveAbsolute
- MC_MoveRelative
- MC_MoveVelocity
- MC_MoveJog
- MC_CommandTable
- La tarea actual de Motion Control con el lanzamiento del primer comando "Positioning Relative", "Positioning Absolute", "Velocity setpoint" o "Halt".

9.3.3.10 MC_ChangeDynamic

Tabla 9- 46 Instrucción MC_ChangeDynamic

KOP / FUP	SCL	Descripción
	<pre>"MC_ChangeDynamic_DB" (Execute:=_bool_in_, ChangeRampUp:=_bool_in_, RampUpTime:=_real_in_, ChangeRampDown:=_bool_in_, RampDownTime:=_real_in_, ChangeEmergency:=_bool_in_, EmergencyRampTime:=_real_in_, ChangeJerkTime:=_bool_in_, JerkTime:=_real_in_, Done=>_bool_out_, Error=>_bool_out_, ErrorID=>_word_out_, ErrorInfo=>_word_out_);</pre>	<p>Modifica los ajustes dinámicos de un eje de control de movimiento:</p> <ul style="list-style-type: none"> • Cambia el valor del tiempo de aceleración • Cambia el valor del tiempo de deceleración • Cambia el valor del tiempo de deceleración de parada de emergencia • Cambiar el valor del tiempo de suavizado (tirones)

- 1 STEP 7 crea el DB automáticamente al insertar la instrucción.
- 2 En el ejemplo SCL, "MC_ChangeDynamic_DB" es el nombre del DB de instancia.

Tabla 9- 47 Parámetros de la instrucción MC_ChangeDynamic

Parámetro y tipo		Tipo de datos	Descripción
Axis	IN	TO_Axis_1	Objeto tecnológico Eje
Execute	IN	Bool	Inicio del comando con flanco ascendente. Valor predeterminado: FALSE
ChangeRampUp	IN	Bool	TRUE = Cambia el tiempo de aceleración según el parámetro de entrada "RampUpTime". Valor predeterminado: FALSE
RampUpTime	IN	Real	Tiempo (en segundos) para acelerar desde parada a la velocidad máxima configurada sin límite de tirones. Valor predeterminado: 5.00 El cambio afectará a la variable <Nombre de eje>. Config.DynamicDefaults.Acceleration. La eficacia del cambio se muestra en la descripción de esta variable.
ChangeRampDown	IN	Bool	TRUE = Cambia el tiempo de deceleración según el parámetro de entrada "RampDownTime". Valor predeterminado: FALSE
RampDownTime	IN	Real	Tiempo (en segundos) para decelerar el eje desde la velocidad máxima configurada hasta la parada sin limitador de tirones. Valor predeterminado: 5.00 El cambio afectará a la variable <Nombre de eje>. Config.DynamicDefaults.Deceleration. La eficacia del cambio se muestra en la descripción de esta variable.
ChangeEmergency	IN	Bool	TRUE = Cambia el tiempo de deceleración de parada de emergencia según el parámetro de entrada "EmergencyRampTime". Valor predeterminado: FALSE
EmergencyRampTime	IN	Real	Tiempo (en segundos) para decelerar el eje desde la velocidad máxima configurada hasta la parada sin limitador de tirones y en modo de parada de emergencia. Valor predeterminado: 2.00 El cambio afectará a la variable <Nombre de eje>. Config.DynamicDefaults.EmergencyDeceleration. La eficacia del cambio se muestra en la descripción de esta variable.
ChangeJerkTime	IN	Bool	TRUE = Modificar el tiempo de suavizado en función del parámetro de entrada "JerkTime". Valor predeterminado: FALSE
JerkTime	IN	Real	Tiempo de suavizado (en segundos) usado para las rampas de aceleración y de deceleración del eje. Valor predeterminado: 0.25 El cambio afectará a la variable <Nombre de eje>. Config.DynamicDefaults.Jerk. La eficacia del cambio se muestra en la descripción de esa variable.
Done	OUT	Bool	TRUE = Los valores cambiados se han escrito en el bloque de datos tecnológico. La descripción de las variables indicará cuándo el cambio será efectivo. Valor predeterminado: FALSE
Error	OUT	Bool	TRUE = Se ha producido un error al ejecutar el comando. La causa del error se indica en los parámetros "ErrorID" y "ErrorInfo". Valor predeterminado: FALSE
ErrorID	OUT	Word	Identificador del error. Valor predeterminado: 16#0000
ErrorInfo	IN	Word	Información de error. Valor predeterminado: 16#0000

Requisitos para la ejecución de MC_ChangeDynamic:

- El objeto tecnológico TO_Axis_PTO V2.0 debe estar configurado correctamente.
- El eje debe estar habilitado.

Respuesta de corrección

Un comando MC_ChangeDynamic no puede ser interrumpido por ningún otro comando de control de movimiento.

Un nuevo comando MC_ChangeDynamic no interrumpe ninguna tarea de control de movimiento activa.

Nota

Los parámetros de entrada "RampUpTime", "RampDownTime", "EmergencyRampTime" y "RoundingOffTime" pueden especificarse con valores que pongan fuera de los límites admisibles los parámetros resultantes del eje "aceleración", "retardo", "retardo de parada de emergencia" y "tirones".

Asegúrese de mantener los parámetros MC_ChangeDynamic dentro de los límites de la configuración dinámica para el objeto tecnológico Eje.

9.3.4 Operación de Motion Control para S7-1200

9.3.4.1 Salidas de la CPU utilizadas para control de movimiento

La CPU proporciona cuatro generadores de salida de impulsos. Cada generador de salida de impulsos ofrece una salida de impulsos y una salida de sentido para controlar un motor paso a paso, o bien un servomotor con interfaz de impulsos. La salida de impulsos proporciona al accionamiento los impulsos necesarios para el movimiento del motor. La salida de sentido controla el sentido de desplazamiento del accionamiento.

Las salidas de impulso y sentido están asignadas entre sí de forma permanente. Las salidas integradas de la CPU y las salidas de una Signal Board pueden utilizarse como salidas de impulso y sentido. Al configurar el dispositivo se elige entre las salidas integradas de la CPU y las salidas de la Signal Board en la ficha "Propiedades" de Generadores de impulsos (PTO/PWM). Sólo PTO (Pulse Train Output) se aplica al control de movimiento.

La salida de PTO genera una salida de onda cuadrada a frecuencia variable. La generación de impulsos se controla mediante información de configuración y ejecución suministrada por la configuración hardware o SFC/SFB.

Según la elección del usuario mientras la CPU está en modo RUN, bien los valores almacenados en la imagen de proceso, bien las salidas de generador de impulso, accionan las salidas digitales. En modo STOP, el generador de PTO no controla las salidas.

Tabla 9- 48 Asignaciones de direcciones de las salidas de impulso y sentido

Utilización de salidas para el control de movimiento		
	Impulso	Sentido
PTO 0		
E/S incorporadas	Q0.0	Q0.1
E/S de la SB	Q4.0	Q4.1
PTO 1		
E/S incorporadas	Q0.2	Q0.3
E/S de la SB	Q4.2	Q4.3
PTO 2		
E/S incorporadas	Q0.4 ¹	Q0.5 ¹
E/S de la SB	Q4.0	Q4.1
PTO 3		
E/S incorporadas	Q0.6 ²	Q0.7 ²
E/S de la SB	Q4.2	Q4.3

¹ La CPU 1211C no tiene las salidas Q0.4, Q0.5, Q0.6 ni Q0.7. Por lo tanto, esas salidas no pueden usarse en la CPU 1211C.

² La CPU 1212C no tiene las salidas Q0.6 ni Q0.7. Por lo tanto, esas salidas no pueden usarse en la CPU 1212C.

³ Esta tabla es válida para las funciones de PTO de la CPU 1211C, CPU 1212C, CPU 1214C y CPU 1215C.

Interfaz del accionamiento

Para el control de movimiento es posible configurar opcionalmente una interfaz del accionamiento para "Accionamiento habilitado" y "Accionamiento listo". Cuando se utiliza la interfaz del accionamiento, la salida digital para habilitar el accionamiento y la entrada digital para "accionamiento listo" pueden seleccionarse libremente.

Nota

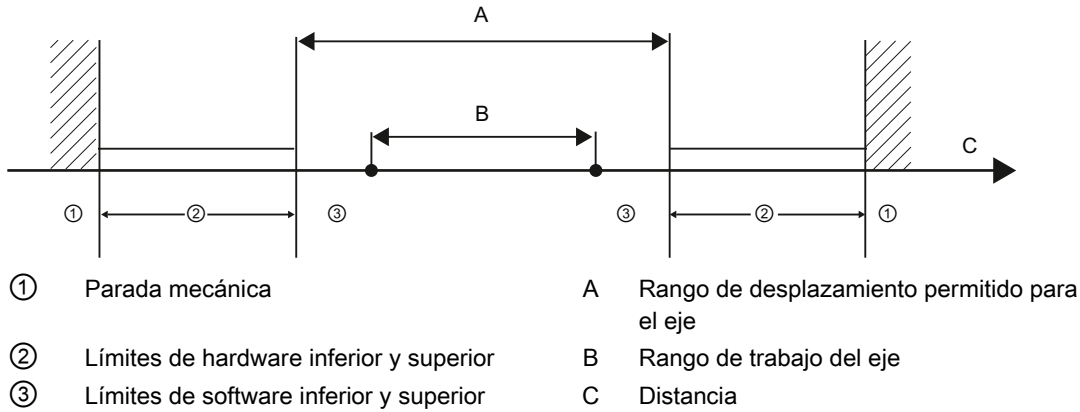
El firmware tomará el control mediante las salidas correspondientes de impulso y sentido si se ha seleccionado la PTO (Pulse Train Output) y se ha asignado a un eje.

En el momento de hacerse cargo de la función de control, la conexión entre la memoria imagen de proceso y la salida de periferia también se desconectará. Mientras que el usuario tiene la posibilidad de escribir en la memoria imagen de proceso de las salidas de impulsos y sentido por medio del programa de usuario o la tabla de observación, ésta no se transfiere nunca a la salida de periferia. Por consiguiente, tampoco es posible vigilar la salida de periferia por medio del programa de usuario o la tabla de observación. La información leída simplemente refleja el valor de la memoria imagen de proceso y no concuerda con el estado actual de la salida de periferia en ningún sentido.

En todas las demás salidas de la CPU que no son utilizadas permanentemente por el firmware de la CPU, el estado de la salida de periferia puede controlarse o vigilarse mediante la memoria imagen de proceso, como siempre.

9.3.4.2 Finales de carrera por hardware y software para Motion Control

Utilice los finales de carrera por hardware y software para limitar el "rango de desplazamiento permitido" y el "rango de trabajo" del eje.



Los finales de carrera por hardware y software deben haberse activado en la configuración o en el programa de usuario antes de ser utilizados. Los finales de carrera por software sólo están activos tras referenciar el eje.

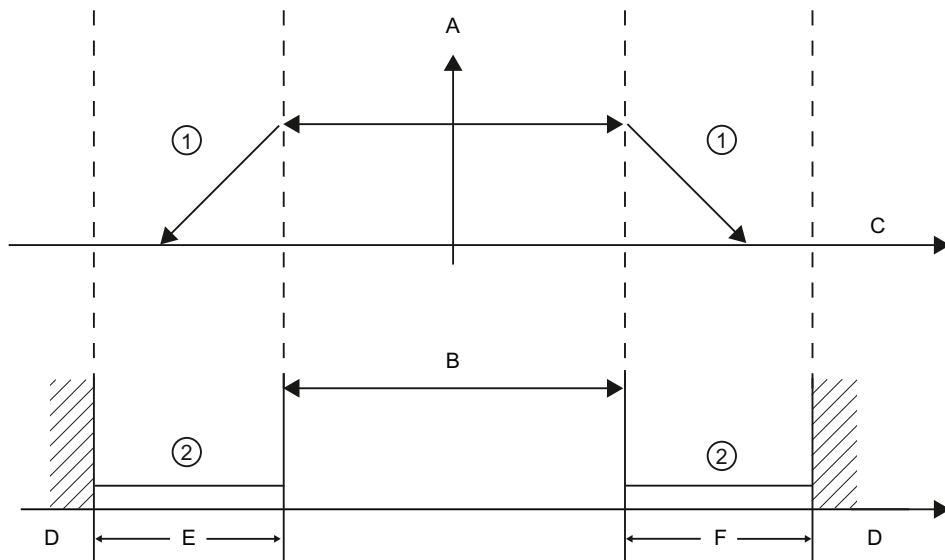
Finales de carrera por hardware

Los finales de carrera por hardware determinan el rango de desplazamiento máximo del eje. Los finales de carrera por hardware son elementos conmutadores físicos que deben estar conectados a entradas aptas para salida de la CPU. Utilice sólo finales de carrera por hardware que permanezcan activados permanentemente tras la aproximación. Este estado de conmutación sólo puede ser revocado una vez que se regrese al rango de desplazamiento válido.

Tabla 9- 49 Entradas disponibles para generadores de impulsos

Descripción	RPS	LIM-	LIM+
PTO 0			
E/S incorporadas		10.0 - 11.5	
E/S de la SB		14.0 - 14.3	
PTO 1			
E/S incorporadas		10.0 - 11.5	
E/S de la SB		14.0 - 14.3	
PTO 2			
E/S incorporadas		10.0 - 11.5	
E/S de la SB		14.0 - 14.3	
PTO 3			
E/S incorporadas		10.0 - 11.5	
E/S de la SB		14.0 - 14.3	

Una vez que se ha completado la aproximación a los finales de carrera por hardware, el eje frena hasta detenerse por completo con la deceleración de emergencia configurada. La deceleración específica de emergencia debe ser suficiente para parar el eje de forma fiable antes de la parada mecánica. El siguiente diagrama representa el comportamiento del eje tras la aproximación a los finales de carrera por hardware.



- ① El eje frena hasta detenerse con la deceleración de emergencia configurada.
- ② Rango en el cual los finales de carrera por hardware señalizan el estado "approached" (aproximación completada).
- A [Velocidad]
- B Rango de desplazamiento permitido
- C Distancia
- D Parada mecánica
- E Final de carrera por hardware inferior
- F Final de carrera por hardware superior

ADVERTENCIA

Si el tiempo de filtro para un canal de entrada digital se reajusta, puede que sea necesario presentar un nuevo valor de entrada de nivel "0" durante un tiempo acumulado de 20 ms para que el filtro esté totalmente operativo ante nuevas entradas. Durante este tiempo, puede que no se detecten o no se cuenten los eventos de pulsación breve "0" cuya duración sea inferior a 20 ms.

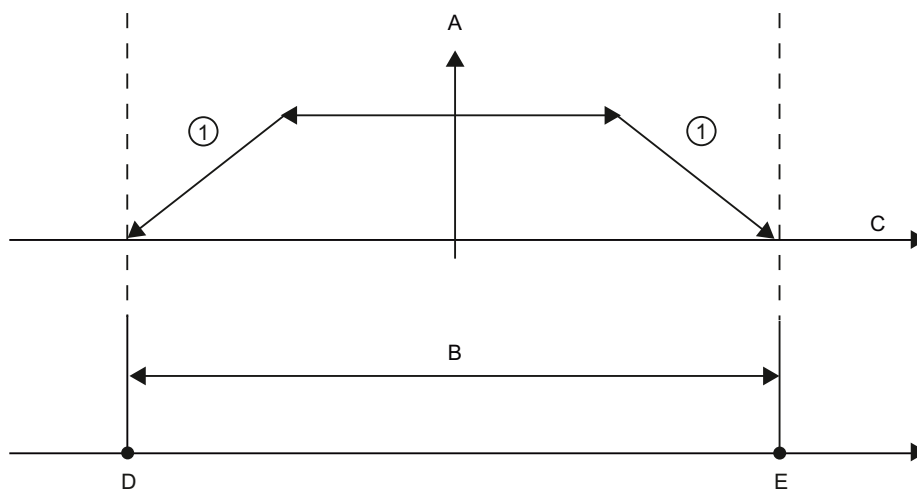
Este cambio de los tiempos de filtro puede originar un funcionamiento inesperado de los equipos o del proceso, lo que puede causar la muerte o lesiones graves al personal y/o daños a los equipos.

Para asegurar que un tiempo de filtro nuevo tenga efecto inmediato, desconecte y vuelva a conectar la CPU.

Finales de carrera por software

Los finales de carrera por software limitan el "rango de trabajo" del eje. Deben estar dentro de los finales de carrera por hardware relativos al rango de desplazamiento. Dado que las posiciones de los finales de carrera por software pueden ajustarse con flexibilidad, el rango operativo del eje puede delimitarse individualmente en función del perfil de desplazamiento. Al contrario que los finales de carrera por hardware, los de software se implementan exclusivamente mediante software, y no requieren elementos conmutadores propios.

Si los finales de carrera por software se activan, cualquier movimiento activo se detiene en la posición del final de carrera por software. El eje frena con la deceleración configurada. El siguiente diagrama representa el comportamiento del eje hasta alcanzar los finales de carrera por software.



- ① El eje frena hasta detenerse con la deceleración configurada.
- A [Velocidad]
- B Rango de trabajo
- C Distancia
- D Final de carrera por software inferior
- E Final de carrera por software superior

Utilice finales de carrera por hardware si una posición final mecánica está situada tras los finales de carrera por software y existe riesgo de daño mecánico.

Información adicional

El programa de usuario puede omitir los finales de carrera por hardware y software habilitando o deshabilitando ambas funciones. La selección se realiza en el DB de eje.

- Para habilitar o deshabilitar la función de final de carrera por hardware, acceda a la variable "Active" (Bool) de la ruta de DB "<nombre de eje>/Config/PositionLimits_HW". El estado de la variable "Activo" habilita o deshabilita el uso de finales de posición por hardware.
- Para habilitar o deshabilitar la función de final de carrera por software, acceda a la variable "Activo" (Bool) de la ruta de DB "<nombre de eje>/Config/PositionLimits_SW". El estado de la variable "Activo" habilita o deshabilita los finales de carrera por software.

También es posible modificar los límites de posición por software con el programa de usuario (por ejemplo para aumentar la flexibilidad para la configuración de la máquina o reducir el tiempo de cambio). El programa de usuario puede escribir nuevos valores en las variables " MinPosition " y " MaxPosition " (unidades de ingeniería en formato Real) en el DB "<Nombre de eje>/Config/PositionLimits_SW".

9.3.4.3 Referenciación

Referenciar significa hacer concordar las coordenadas del eje con la posición física real del accionamiento. (Si en este momento el accionamiento se encuentra en la posición x, el eje se ajustará para que se encuentre en la posición x.) Para ejes controlados por posición, las entradas e indicaciones referentes a la posición se refieren exactamente a esas coordenadas del eje.

Nota

La concordancia entre las coordenadas del eje y la situación real es determinante. Este paso es necesario para garantizar que la posición de destino absoluta del eje también se alcance con precisión con el accionamiento.

La instrucción MC_Home inicia la referenciación del eje.

Hay 4 funciones de referenciación diferentes. Las dos primeras funciones permiten al usuario ajustar la posición actual del eje y las dos segundas posicionar el eje respecto a un sensor de referenciación.

- Modo 0 - Referenciación directa absoluta: Al ejecutarlo, este modo le dice al eje exactamente dónde está. Asigna a la variable de posición interna el valor de la entrada Position de la instrucción de referenciación. Se usa para la calibración y ajuste de máquinas.

La posición del eje se ajusta independientemente del interruptor de punto de referencia. Los movimientos activos no se interrumpen. El valor del parámetro de la entrada Position de la instrucción MC_Home se fija inmediatamente como punto de referencia del eje. Para asignar el punto de referencia a una posición mecánica precisa, el eje debe estar parado en la posición en el momento de la referenciación.

- Modo 1 - Referenciación directa relativa: Al ejecutarlo, este método utiliza la variable de posición interna y le añade el valor de la entrada Position de la instrucción de referenciación. Este método se suele utilizar para offsets de máquinas.

La posición del eje se ajusta independientemente del interruptor de punto de referencia. Los movimientos activos no se interrumpen. La siguiente afirmación es válida para la posición del eje tras la referenciación: Nueva posición del eje = posición actual del eje + valor del parámetro Position de la instrucción MC_Home.

- Modo 2 - Referenciación pasiva: Cuando el eje se mueve y pasa el interruptor de punto de referencia, la posición actual se fija como posición de referencia. Esta función ayuda a reducir el desgaste normal de la máquina y el juego de los cojinetes y prevenir la necesidad de compensar el desgaste manualmente. Igual que antes, la entrada Position de la instrucción de referenciación se añade a la posición indicada por el interruptor del punto de referencia, facilitando así el offset de la posición de referencia.

Durante la referenciación pasiva, la instrucción MC_Home no realiza ningún movimiento de referenciación. El movimiento necesario para este paso debe ser implementado por el usuario mediante otras instrucciones de Motion Control. El eje está referenciado según la configuración cuando se detecta el interruptor de punto de referencia. Los movimientos de desplazamiento activos no se interrumpen con el inicio de la referenciación pasiva.

- Modo 3 - Referenciación activa: Este modo es el método más preciso para la referenciación del eje. La dirección y velocidad iniciales del movimiento se configuran en los parámetros de configuración avanzada del objeto tecnológico para la referenciación. Depende de la configuración de la máquina. También existe la posibilidad de determinar si el flanco inicial o descendente en la señal del interruptor de punto de referencia es la posición de referencia. Virtualmente todos los sensores tienen un rango activo y si se ha utilizado la posición de Siempre conectado como señal de referencia, existe la posibilidad de errar en la posición de referencia, ya que el rango activo de señal de conexión podría cubrir cierta distancia. Al aplicar el flanco inicial o descendente de la señal se consigue una posición de referencia mucho más precisa. Igual que ocurre con todos los métodos restantes, el valor de la entrada Position de la instrucciones de posicionamiento se añade a la posición referenciada por hardware.

En el modo de referenciación activa, la instrucción MC_Home realiza la aproximación necesaria al punto de referencia. El eje está referenciado según la configuración cuando se detecta el interruptor de punto de referencia. Los movimientos activos se interrumpen.

Los modos 0 y 1 no requieren un movimiento del eje. Suelen utilizarse para tareas de configuración y calibración. En los modos 2 y 3 es necesario que el eje se mueva y pase por un sensor configurado en el objeto tecnológico "Eje" como interruptor de punto de referencia. El punto de referencia que puede situarse en el área de trabajo del eje o fuera del área normal de trabajo, pero dentro del rango de movimiento.

Configuración de parámetros de referenciación

Configure los parámetros para la referenciación activa y pasiva en la ventana de configuración "Homing". El método de referenciación se ajusta utilizando el parámetro de entrada "Mode" de la instrucción de Motion Control. En este caso, Mode = 2 significa referenciación pasiva y Mode = 3 referenciación activa.

ATENCIÓN

Utilice una de las siguientes medidas para garantizar que la máquina no se desplace a una posición final mecánica en caso de invertir la dirección:

- Mantenga una velocidad de aproximación reducida
- Aumente la aceleración/deceleración configurada
- Aumente la distancia entre el final de carrera por hardware y la parada mecánica

Tabla 9- 50 Parámetros de configuración para referenciar el eje

Parámetro	Descripción
Entrada interruptor de punto de referencia (Referenciación activa y pasiva)	<p>Seleccione la entrada digital para el interruptor de punto de referencia de la lista desplegable. La entrada debe soportar alarmas. Las entradas integradas de la CPU y las entradas de una Signal Board opcional pueden seleccionarse como entradas para el interruptor de punto de referencia.</p> <p>El tiempo de filtro predeterminado para las salidas digitales es 6,4 ms. Cuando las entradas digitales se utilizan como interruptor de punto de referencia, pueden producirse desaceleraciones no deseadas y, por lo tanto, imprecisiones. Dependiendo de la velocidad reducida y la extensión del interruptor de punto de referencia, es posible que dicho punto no se detecte. El tiempo de filtrado puede configurarse en "Filtros de entrada" en la configuración de dispositivos de las entradas digitales.</p> <p>El tiempo de filtrado especificado debe ser menor que la duración de la señal de entrada en el interruptor del punto de referencia.</p>
Inversión automática tras alcanzar los finales de carrera por hardware (Sólo referenciación activa)	<p>Active la casilla de verificación para utilizar el final de carrera por hardware como leva de inversión para la aproximación al punto de referencia. Los finales de carrera por hardware deben estar configurados y activados para la inversión direccional.</p> <p>Cuando se alcanza el final de carrera por hardware durante la referenciación activa, el eje frena a la deceleración configurada (no con la deceleración de emergencia) e invierte la dirección. El interruptor de punto de referencia se explora entonces en dirección inversa.</p> <p>Si la inversión direccional no está activada y el eje llega al final de carrera por hardware durante la referenciación activa, la aproximación al punto de referencia se interrumpe con un error y el eje se frena con la deceleración de emergencia.</p>
Dirección de aproximación (Referenciación activa y pasiva)	<p>Al seleccionar la dirección se determina la "dirección de aproximación" utilizada durante la referenciación activa para buscar el interruptor del punto de referencia, y también la dirección de referenciación. La dirección de referenciación especifica la dirección de desplazamiento que el eje utiliza para aproximarse al lado configurado del interruptor de punto de referencia para llevar a cabo la operación de referenciación.</p>
Interruptor de punto de referencia (Referenciación activa y pasiva)	<ul style="list-style-type: none"> • Referenciación activa: Seleccione si desea que el eje se referencie en el lado izquierdo o derecho del interruptor de punto de referencia. Dependiendo de la posición de inicio del eje y de la configuración de los parámetros de referenciación, la secuencia de aproximación al punto de referencia puede divergir del diagrama mostrado en la ventana de configuración. • Referenciación pasiva: Con la referenciación pasiva, los movimientos de desplazamiento con fines de referenciación deben ser implementados por el usuario mediante comandos de movimiento. El lado del interruptor del punto de referenciación en que se produce la referenciación depende de los siguientes factores: <ul style="list-style-type: none"> – Configuración de la dirección de aproximación – Configuración del interruptor de punto de referencia – Dirección actual de desplazamiento durante la referenciación pasiva
Velocidad de aproximación (Sólo referenciación activa)	<p>Especifique la velocidad a la que debe localizarse el interruptor de punto de referencia durante la aproximación al punto de referencia.</p> <p>Valores límite (independientemente de la unidad seleccionada por el usuario): Velocidad inicio/parada \leq velocidad de aproximación \leq velocidad máxima</p>

Parámetro	Descripción
Velocidad reducida (Sólo referenciación activa)	Especifique la velocidad a la cual el eje se aproxima al interruptor de punto de referencia en la referenciación. Valores límite (independientemente de la unidad seleccionada por el usuario): Velocidad inicio/parada ≤ velocidad reducida ≤ velocidad máxima
Offset de posición de referencia (Sólo referenciación activa)	Si la posición de referencia deseada varía de la posición del interruptor de punto de referencia, en este campo también se puede especificar el offset de la posición de referencia. Si el valor no es 0, el eje ejecuta las siguientes acciones tras la referenciación en el interruptor de punto de referencia: 1. Mueve el eje a velocidad reducida según el valor del offset de la posición de referencia. 2. Una vez alcanzada la posición del offset, la posición del eje se fija como posición de referencia absoluta. La posición de referencia absoluta se especifica con el parámetro "Position" de la instrucción de Motion Control "MC_Home". Valores límite (independientemente de la unidad seleccionada por el usuario): -1.0e12 ≤ offset de posición de referencia ≤ 1.0e12

Tabla 9- 51 Factores que afectan a la referenciación

Factores influyentes:			Resultado:
Configuración Dirección de aproximación	Configuración Interruptor de punto de referencia	Dirección de desplazamiento actual	Referenciación activa Interruptor de punto de referencia
positivo	"Lado izquierdo (negativa)"	Dirección positiva	Izquierda
		Dirección negativa	Derecha
positivo	"Lado derecho (positiva)"	Dirección positiva	Derecha
		Dirección negativa	Izquierda
negativo	"Lado izquierdo (negativa)"	Dirección positiva	Derecha
		Dirección negativa	Izquierda
negativo	"Lado derecho (positiva)"	Dirección positiva	Izquierda
		Dirección negativa	Derecha

Secuencia de referenciación activa

La referenciación activa se inicia con la instrucción de Motion Control "MC_Home" (parámetro de entrada Mode = 3). En este caso, el parámetro de entrada "Position" especifica las coordenadas del punto de referencia absoluto. Como alternativa, la referenciación activa se puede iniciar desde el panel de mando con fines de prueba.

El siguiente diagrama muestra un ejemplo de curva característica para una aproximación activa al punto de referencia con los siguientes parámetros de configuración:

- "Dirección de aproximación" = "Dirección de aproximación positiva"
- "Interruptor de punto de referencia" = "Lado derecho (positiva)"
- Valor de "Offset de posición de referencia" > 0

Tabla 9- 52 Características de velocidad de la referenciación MC

Funcionamiento		Notas	
		A	Velocidad de aproximación
		B	Velocidad reducida
		C	Coordenada de posición de referencia
		D	Offset de posición de referencia
①	Fase de búsqueda (segmento azul de la curva): Cuando la referenciación activa comienza, el eje acelera a la velocidad de aproximación configurada y busca el interruptor de punto de referencia a esa velocidad.		
②	Aproximación a punto de referencia (sección roja de la curva): Cuando se detecta el interruptor de punto de referencia, el eje del ejemplo frena e invierte la dirección para referenciarse en el lado configurado del interruptor de punto de referencia con la velocidad reducida configurada.		
③	Desplazamiento a la posición del punto de referencia (segmento verde de la curva): Tras la referenciación en el interruptor de punto de referencia, el eje se desplaza a las coordenadas de punto de referencia configuradas a la velocidad reducida especificada. Cuando alcanza las "coordenadas del punto de referencia", el eje se detiene en el valor de posición especificado en el parámetro de entrada Position de la instrucción MC_Home.		

Nota

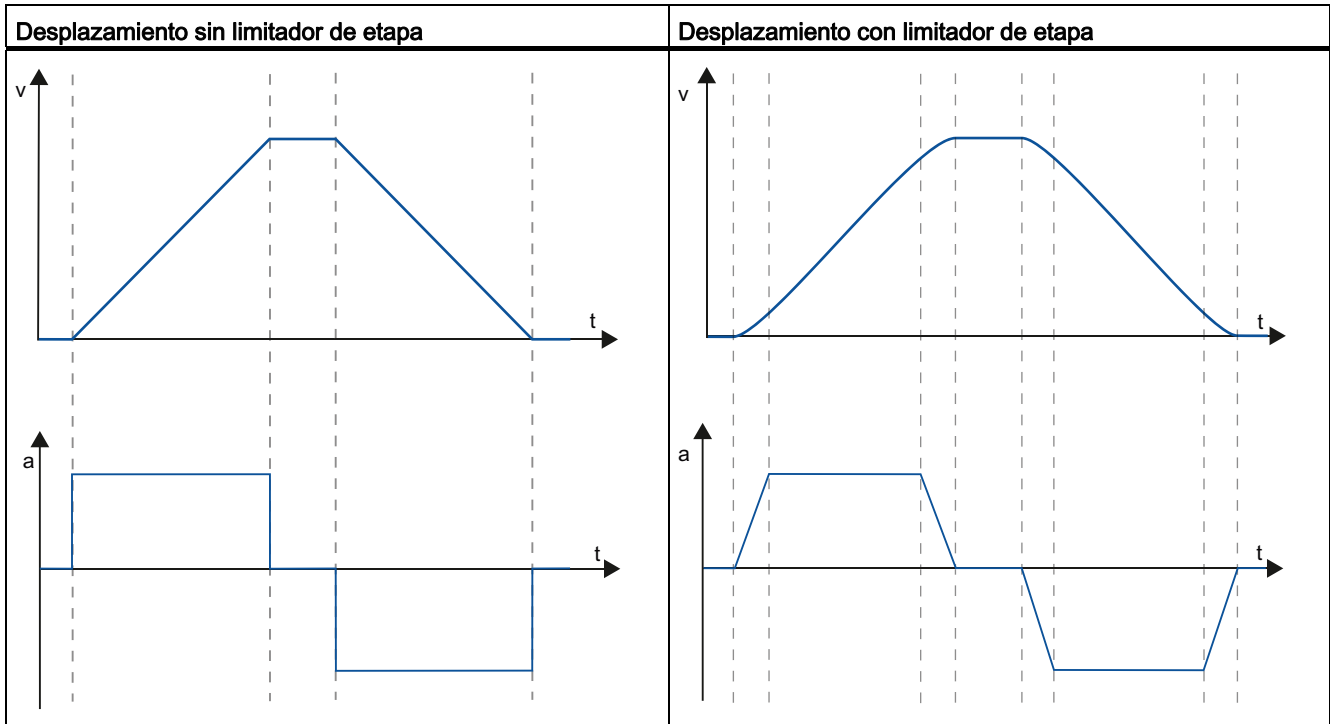
Si la búsqueda de referenciación no funciona de la forma esperada, compruebe las entradas asignadas a los finales de carrera de hardware o al punto de referencia. Estas entradas pueden tener las alarmas de flanco deshabilitadas en la configuración de dispositivo.

Revise los datos de configuración para el objeto tecnológico Eje para ver qué entradas (si hay alguna) están asignadas para "HW Low Limit Switch Input", "HW High Limit Switch Input" y "Input reference point switch". A continuación, abra la configuración de dispositivo para la CPU y examine cada una de las entradas asignadas. Compruebe que ambas opciones "Activar detección del flanco ascendente" y "Activar detección del flanco descendente" están seleccionadas. Si no es así, borre las entradas especificadas en la configuración del eje y vuelva a seleccionarlas.

9.3.4.4 Limitación de tirones

La limitación de tirones permite reducir los esfuerzos de los aparatos durante la rampa de aceleración y deceleración. El valor de aceleración y deceleración no cambia repentinamente cuando el limitador de etapa está activo; se adapta en una fase de transición. La figura inferior muestra las curvas de velocidad y aceleración sin y con limitación de tirones.

Tabla 9- 53 Limitación de tirones



La limitación de etapa consigue un perfil de velocidad "suavizado" del movimiento del eje. De este modo se garantiza el arranque y frenado suaves de una cinta transportadora, por ejemplo.

9.3.5 Puesta en servicio

Función de diagnóstico "Bits de estado y error"

La función de diagnóstico "Bits de estado y error" se utiliza para vigilar los principales avisos de estado y mensajes de error del eje. La visualización de la función de diagnóstico está disponible en modo online con el modo "Control manual" y con "Control automático" cuando el eje está activo.

Tabla 9- 54 Estado del eje

Estado	Descripción
Habilitado	El eje está habilitado y listo para ser controlado mediante las tareas de Motion Control. (Variable del objeto tecnológico: <Nombre del eje>.StatusBits.Enable)
Referenciado	El eje está referenciado y es capaz de ejecutar tareas de posicionamiento absoluto de la instrucción de Motion Control "MC_MoveAbsolute". El eje no tiene que estar referenciado para la referenciación relativa. Situaciones especiales: <ul style="list-style-type: none"> • Durante la referenciación activa, el estado es FALSE. • Si un eje referenciado está sometido a referenciación pasiva, el estado se pone a TRUE durante la referenciación pasiva. (Variable del objeto tecnológico: <Nombre del eje>.StatusBits.HomingDone)
Error	Se ha producido un error en el objeto tecnológico "Eje". Encontrará más información sobre el error con control automático en los parámetros ErrorID y ErrorInfo de las instrucciones de Motion Control. En modo manual, el campo "Último error" del panel de mando muestra información detallada acerca de la causa del error. (Variable del objeto tecnológico: <Nombre del eje>.StatusBits.Error)
Panel de mando activo	El modo "Control manual" se ha activado en el panel de mando. El panel de mando tiene prioridad de control sobre el objeto tecnológico "Eje". El eje no puede controlarse desde el programa de usuario. (Variable del objeto tecnológico: <Nombre del eje>.StatusBits.ControlPanelActive)

Tabla 9- 55 Estado del accionamiento

Estado	Descripción
Accionamiento listo	El accionamiento está listo para funcionar. (Variable del objeto tecnológico: <Nombre del eje>.StatusBits.DriveReady)
Error	El accionamiento ha notificado un error después de que fallara su señal de disponibilidad. (Variable del objeto tecnológico: <Nombre del eje>.ErrorBits.DriveFault)

Tabla 9- 56 Estado del movimiento del eje

Estado	Descripción
Parada	El eje está parado. (Variable del objeto tecnológico: <Nombre del eje>.StatusBits.StandStill)
Aceleración	El eje acelera. (Variable del objeto tecnológico: <Nombre del eje>.StatusBits.Acceleration)
Velocidad constante	El eje se desplaza a una velocidad constante. (Variable del objeto tecnológico: <Nombre del eje>.StatusBits.ConstantVelocity)
Deceleración	El eje desacelera. (Variable del objeto tecnológico: <Nombre del eje>.StatusBits.Deceleration)

Tabla 9- 57 Estado del modo de movimiento

Estado	Descripción
Posicionamiento	El eje ejecuta una tarea de posicionamiento de la instrucción de Motion Control "MC_MoveAbsolute" o "MC_MoveRelative" o del panel de mando. (Variable del objeto tecnológico: <Nombre del eje>.StatusBits.PositioningCommand)
Comando de velocidad	El eje ejecuta una tarea a la velocidad establecida de la instrucción de Motion Control "MC_MoveVelocity" o "MC_MoveJog" o del panel de mando. (Variable del objeto tecnológico: <Nombre del eje>.StatusBits.SpeedCommand)
Referenciación	El eje ejecuta una tarea de referenciación de la instrucción de Motion Control "MC_Home" o del panel de mando. (Variable del objeto tecnológico: <Nombre del eje>.StatusBits.Homing)

Tabla 9- 58 Bits de error

Error	Descripción
Final de carrera software mín. alcanzado	Se ha alcanzado el final de carrera por software inferior. (Variable del objeto tecnológico: <Nombre del eje>.ErrorBits.SwLimitMinReached)
Final de carrera software mín. rebasado	Se ha rebasado el final de carrera por software inferior. (Variable del objeto tecnológico: <Nombre del eje>.ErrorBits.SwLimitMinExceeded)
Final de carrera software máx. alcanzado	Se ha alcanzado el final de carrera por software superior. (Variable del objeto tecnológico: <Nombre del eje>.ErrorBits.SwLimitMaxReached)
Final de carrera software máx. rebasado	Se ha rebasado el final de carrera por software superior. (Variable del objeto tecnológico: <Nombre del eje>.ErrorBits.SwLimitMaxExceeded)
Final de carrera hardware negativo	Se ha completado la aproximación al final de carrera por hardware inferior. (Variable del objeto tecnológico: <Nombre del eje>.ErrorBits.HwLimitMin)
Final de carrera hardware positivo	Se ha finalizado la aproximación al final de carrera por hardware superior. (Variable del objeto tecnológico: <Nombre del eje>.ErrorBits.HwLimitMax)
PTO y HSC ya se utilizan	Un segundo eje utiliza la misma PTO y HSC y está habilitado con "MC_Power". (Variable del objeto tecnológico: <Nombre del eje>.ErrorBits.HwUsed)

Error	Descripción
Error de configuración	El objeto tecnológico "Eje" se ha configurado erróneamente o los datos de configuración editables se han modificado incorrectamente durante el runtime del programa de usuario. (Variable del objeto tecnológico: <Nombre del eje>.ErrorBits.ConfigFault)
Error general	Se ha producido un error interno. (Variable del objeto tecnológico: <Nombre del eje>.ErrorBits.SystemFault)

Función de diagnóstico "Estado de movimiento"

La función de diagnóstico "Estado de movimiento" se utiliza para vigilar el estado de movimiento del eje. La visualización de la función de diagnóstico está disponible en modo online con el modo "Control manual" y con "Control automático" cuando el eje está activo.

Tabla 9- 59 Estado de movimiento

Estado	Descripción
Posición de destino	El campo "Posición de destino" indica la posición de destino actual de una tarea de posicionamiento activa de la instrucción de Motion Control "MC_MoveAbsolute" o "MC_MoveRelative" o del panel de mando. El valor de la "Posición de destino" sólo es válida mientras se ejecuta una tarea de posicionamiento. (Variable del objeto tecnológico: <Nombre del eje>.MotionStatus.TargetPosition)
Posición actual	El campo "Posición actual" indica la posición actual del eje. Si el eje no está referenciado, el valor indica la posición en relación a la posición de habilitación del eje. (Variable del objeto tecnológico: <Nombre del eje>.MotionStatus.Position)
Velocidad actual	El campo "Velocidad actual" indica la velocidad actual del eje. (Variable del objeto tecnológico: <Nombre del eje>.MotionStatus.Velocity)

Tabla 9- 60 Límites dinámicos

Límite dinámico	Descripción
Velocidad	El campo "Velocidad" indica la velocidad máxima configurada para el eje. (Variable del objeto tecnológico: <Nombre del eje>.Config.DynamicLimits.MaxVelocity)
Aceleración	El campo "Aceleración" indica la aceleración configurada actualmente para el eje. (Variable del objeto tecnológico: <Nombre del eje>.Config.DynamicDefaults.Acceleration)
Deceleración	El campo "Deceleración" indica la deceleración configurada actualmente para el eje. (Variable del objeto tecnológico: <Nombre del eje>.Config.DynamicDefaults.Deceleration)

9.3.6 Vigilar comandos activos

9.3.6.1 Vigilar instrucciones MC con un parámetro de salida "Done"

Las instrucciones Motion Control con el parámetro de salida "Done" se inician mediante el parámetro de entrada "Execute" y tienen una finalización definida (p. ej. con la instrucción Motion Control "MC_Home": referenciación correcta). La tarea se ha completado y el eje está parado.

- El parámetro de salida "Done" indica el valor TRUE si la tarea se ha completado correctamente.
- Los parámetros de salida "Busy", "CommandAborted" y "Error" indican que la tarea todavía se está procesando, se ha cancelado o hay un error pendiente, respectivamente. La instrucción Motion Control "MC_Reset" no puede cancelarse, por lo que no tiene el parámetro de salida "CommandAborted".
 - Durante el procesamiento de la tarea Motion Control, el parámetro de salida "Busy" indica el valor TRUE. Si la tarea se ha completado, cancelado o detenido a causa de un error, el parámetro de salida "Busy" cambia a FALSE. El cambio se produce independientemente de la señal en el parámetro de entrada "Execute".
 - Los parámetros de salida "Done", "CommandAborted" y "Error" indican el valor TRUE durante un ciclo como mínimo. Los avisos de estado se congelan mientras el parámetro de entrada "Execute" está puesto a TRUE.

Las tareas de las siguientes instrucciones Motion Control tienen una finalización definida:

- MC_Reset
- MC_Home
- MC_Halt
- MC_MoveAbsolute
- MC_MoveRelative

El comportamiento del bit de estado se muestra más abajo en algunas situaciones de ejemplo.

- El primer ejemplo muestra el comportamiento del eje en una tarea finalizada. Si la tarea Motion Control se ha ejecutado por completo en el momento de su finalización, se indica con el valor TRUE en el parámetro de salida "Done". El estado de señal del parámetro de entrada "Execute" influye en el tiempo de visualización del parámetro de salida "Done".
- El segundo ejemplo muestra el comportamiento del eje en una tarea cancelada. Si la tarea Motion Control se cancela durante la ejecución, se indica con el valor TRUE en el parámetro de salida "CommandAborted". El estado de señal del parámetro de entrada "Execute" influye en el tiempo de visualización del parámetro de salida "CommandAborted".
- El tercer ejemplo muestra el comportamiento del eje cuando se produce un error. Si se produce un error durante la ejecución de la tarea Motion Control, se indica con el valor TRUE en el parámetro de salida "Error". El estado de señal del parámetro de entrada "Execute" influye en el tiempo de visualización del parámetro de salida "Error".

Tabla 9- 61 Ejemplo 1: ejecución completa de la tarea

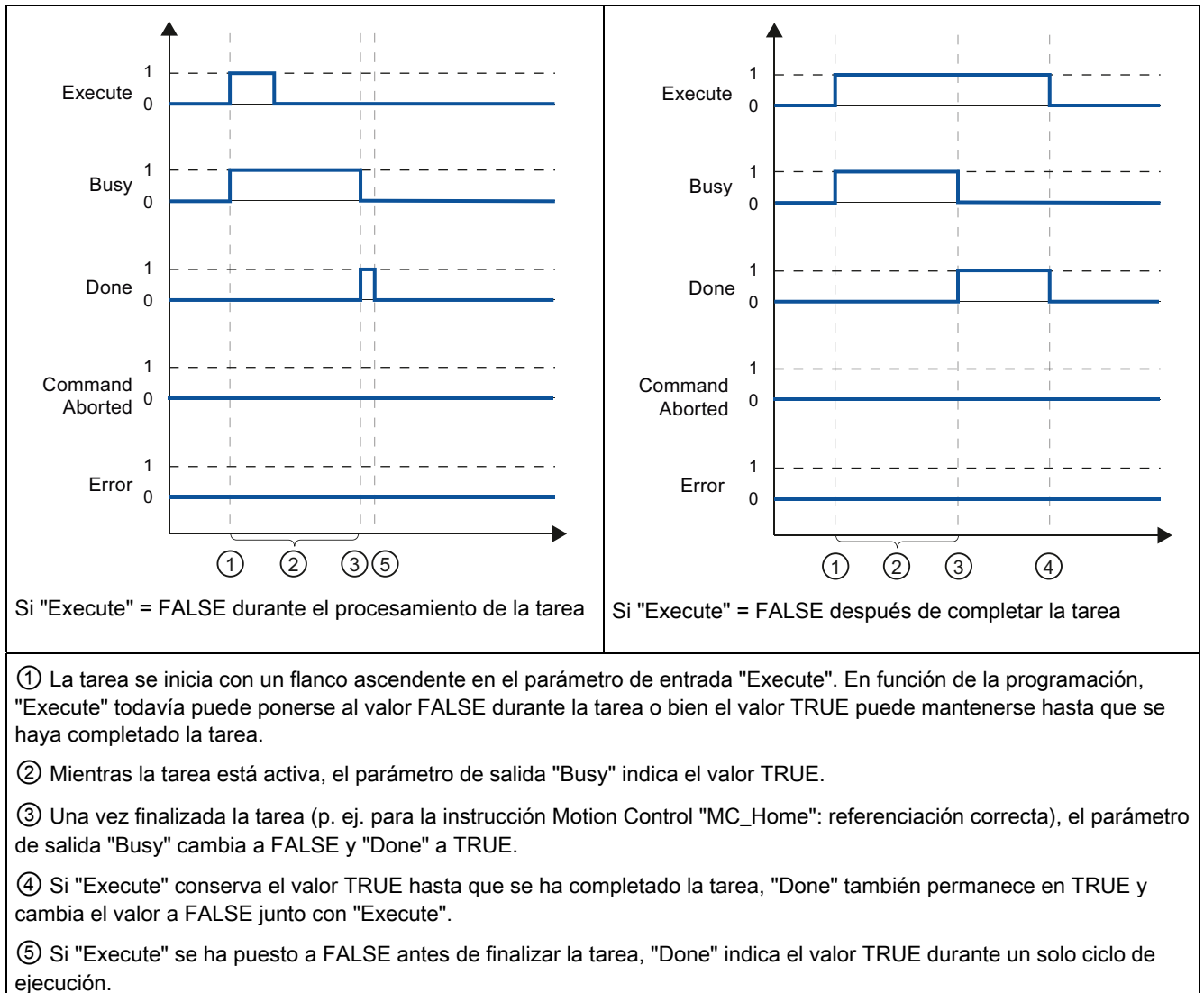
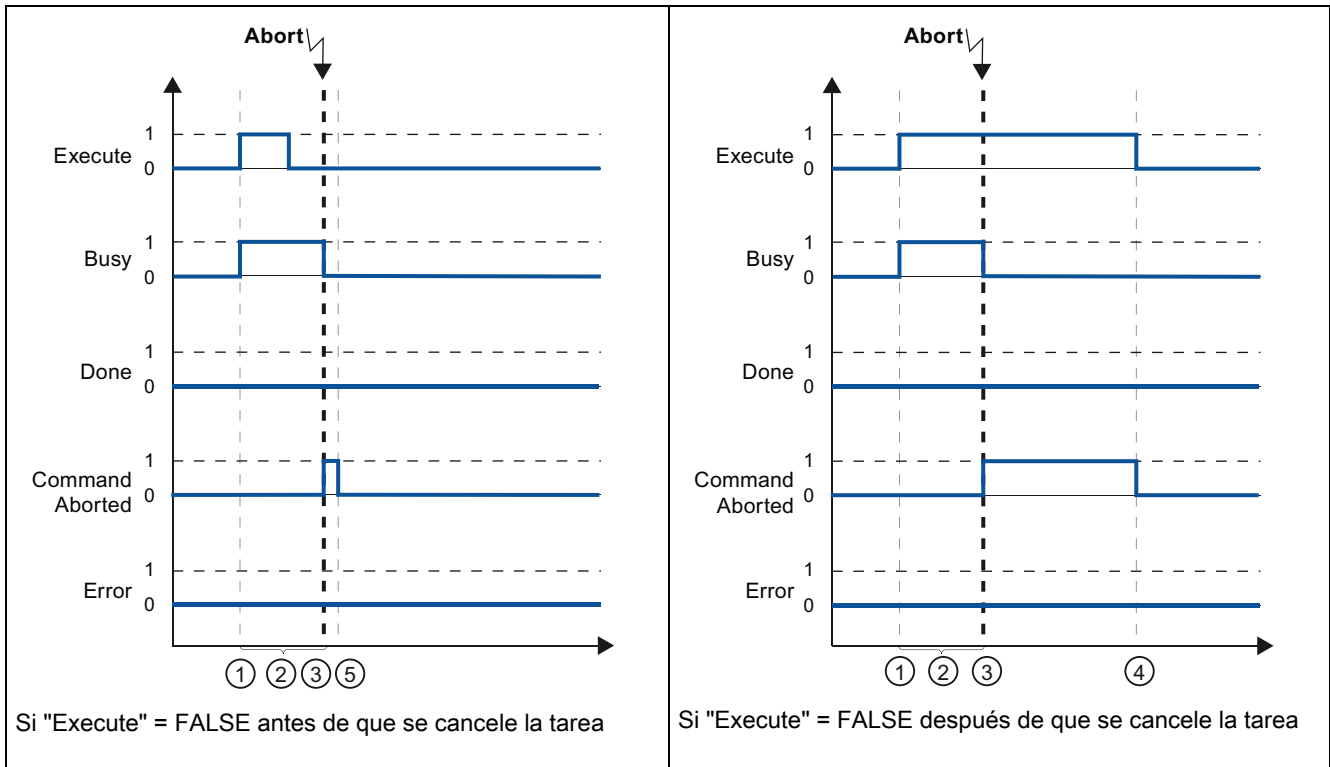
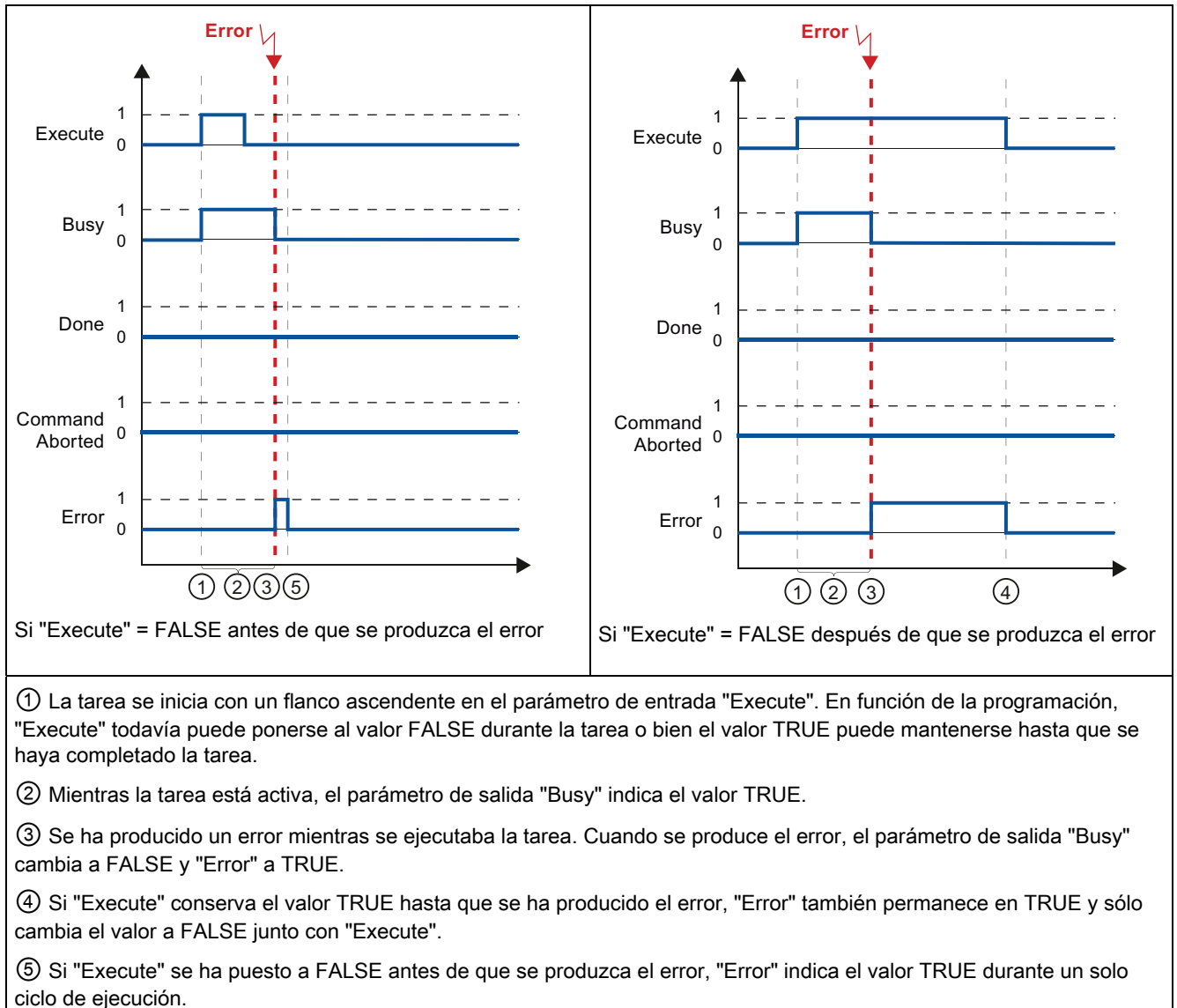


Tabla 9- 62 Ejemplo 2: cancelación de la tarea



- ① La tarea se inicia con un flanco ascendente en el parámetro de entrada "Execute". En función de la programación, "Execute" todavía puede ponerse al valor FALSE durante la tarea o bien el valor TRUE puede mantenerse hasta que se haya completado la tarea.
- ② Mientras la tarea está activa, el parámetro de salida "Busy" indica el valor TRUE.
- ③ Mientras se ejecuta la tarea, ésta es cancelada por otra tarea Motion Control. Si se cancela la tarea, el parámetro de salida "Busy" cambia a FALSE y "CommandAborted" a TRUE.
- ④ Si "Execute" conserva el valor TRUE hasta que se ha cancelado la tarea, "CommandAborted" también permanece en TRUE y cambia el valor a FALSE junto con "Execute".
- ⑤ Si "Execute" se ha puesto a FALSE antes de cancelar la tarea, "CommandAborted" indica el valor TRUE durante un solo ciclo de ejecución.

Tabla 9- 63 Ejemplo 3: error al ejecutar la tarea



9.3.6.2 Vigilar la instrucción MC_Velocity

Las tareas de la instrucción Motion Control "MC_MoveVelocity" se ejecutan de manera constante a la velocidad especificada.

- Las tareas de la instrucción Motion Control "MC_MoveVelocity" no tienen un final definido. El objetivo de la tarea se cumple cuando se alcanza la velocidad parametrizada por primera vez y el eje se desplaza a una velocidad constante. Cuando se ha alcanzado la velocidad parametrizada, se indica con el valor TRUE en el parámetro de salida "InVelocity".
- La tarea se completa cuando se ha alcanzado la velocidad parametrizada y el parámetro de entrada "Execute" se ha puesto al valor FALSE. Sin embargo, el movimiento del eje no se completa una vez finalizada la tarea. Por ejemplo, el movimiento del eje puede detenerse con la tarea Motion Control "MC_Halt".
- Los parámetros de salida "Busy", "CommandAborted" y "Error" indican que la tarea todavía se está procesando, se ha cancelado o hay un error pendiente, respectivamente.
 - Durante la ejecución de la tarea Motion Control, el parámetro de salida "Busy" indica el valor TRUE. Si la tarea se ha completado, cancelado o detenido a causa de un error, el parámetro de salida "Busy" cambia a FALSE. El cambio se produce independientemente de la señal en el parámetro de entrada "Execute".
 - Los parámetros de salida "InVelocity", "CommandAborted" y "Error" indican el valor TRUE durante un ciclo como mínimo, siempre que se cumplan las condiciones. Los avisos de estado están enclavados mientras el parámetro de entrada "Execute" está puesto a TRUE.

El comportamiento del bit de estado se muestra más abajo en algunas situaciones de ejemplo.

- El primer ejemplo muestra el comportamiento cuando el eje alcanza la velocidad parametrizada. Si la tarea Motion Control se ha ejecutado en el momento en que se alcanza la velocidad parametrizada, se indica con el valor TRUE en el parámetro de salida "InVelocity". El estado de señal del parámetro de entrada "Execute" influye en la duración de visualización del parámetro de salida "InVelocity".
- El segundo ejemplo muestra el comportamiento si la tarea se cancela antes de alcanzar la velocidad parametrizada. Si la tarea Motion Control se cancela antes de alcanzar la velocidad parametrizada, se indica con el valor TRUE en el parámetro de salida "CommandAborted". El estado de señal del parámetro de entrada "Execute" influye en el tiempo de visualización del parámetro de salida "CommandAborted".
- El tercer ejemplo muestra el comportamiento del eje cuando se produce un error antes de alcanzar la velocidad parametrizada. Si se produce un error durante la ejecución de la tarea Motion Control antes de alcanzar la velocidad parametrizada, se indica con el valor TRUE en el parámetro de salida "Error". El estado de señal del parámetro de entrada "Execute" influye en la duración de visualización del parámetro de salida "Error".

Tabla 9- 64 Ejemplo 1: si se ha alcanzado la velocidad parametrizada

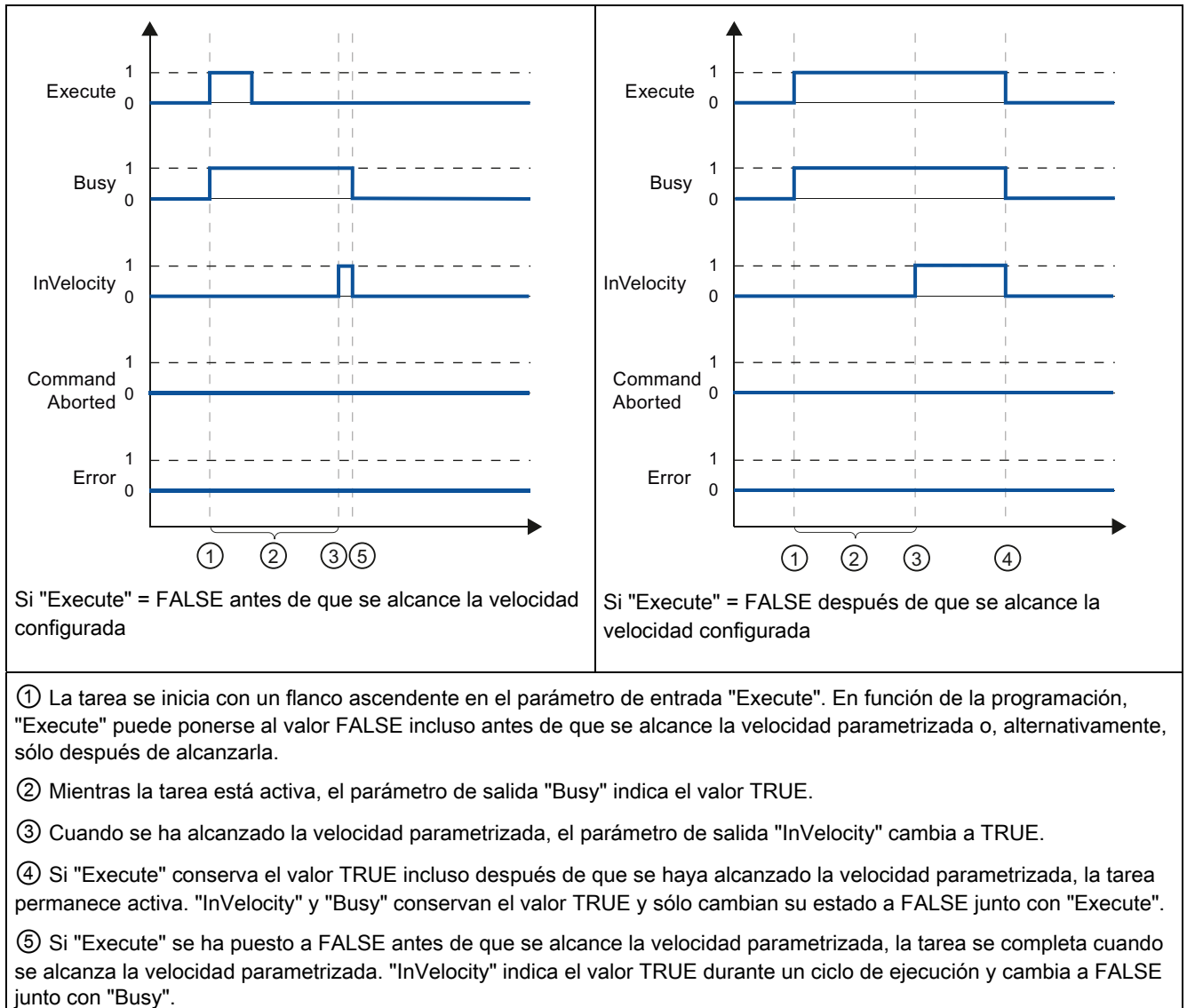
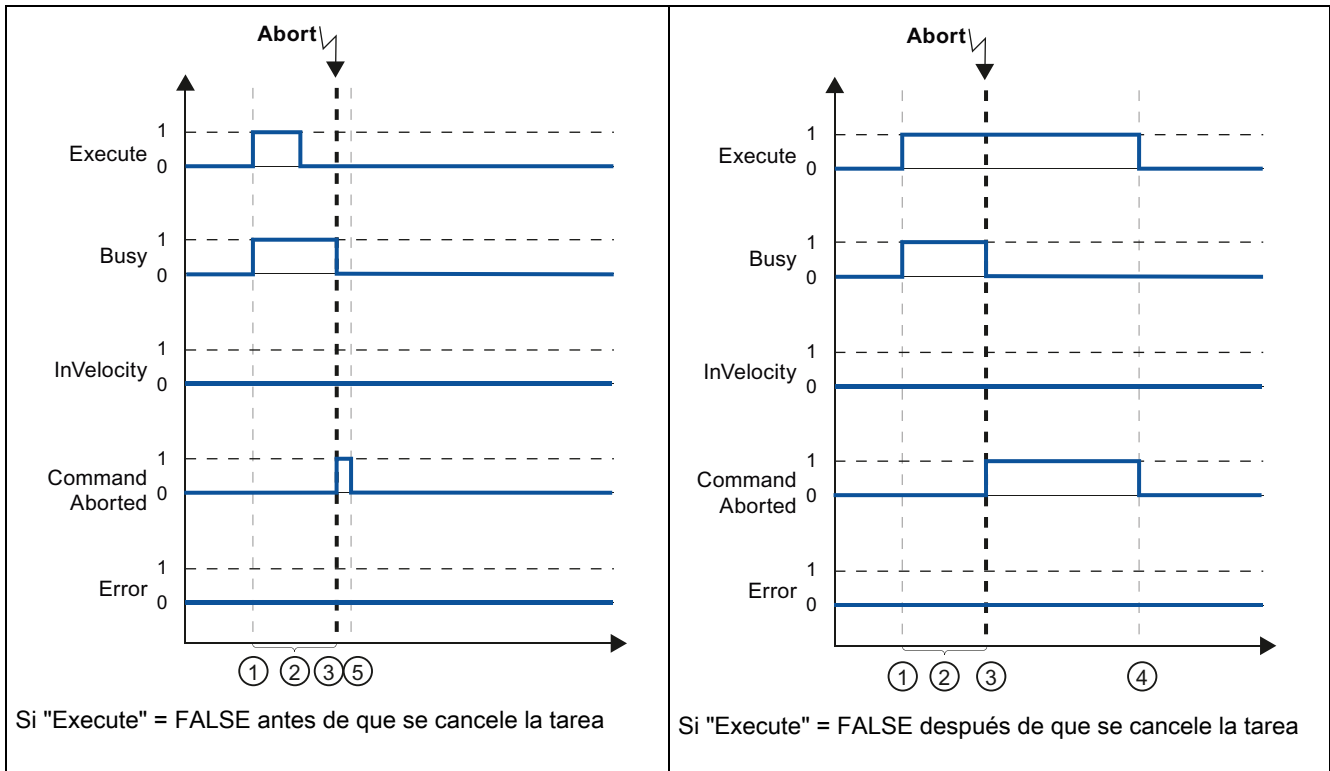


Tabla 9- 65 Ejemplo 2: si la tarea se cancela antes de alcanzar la velocidad parametrizada



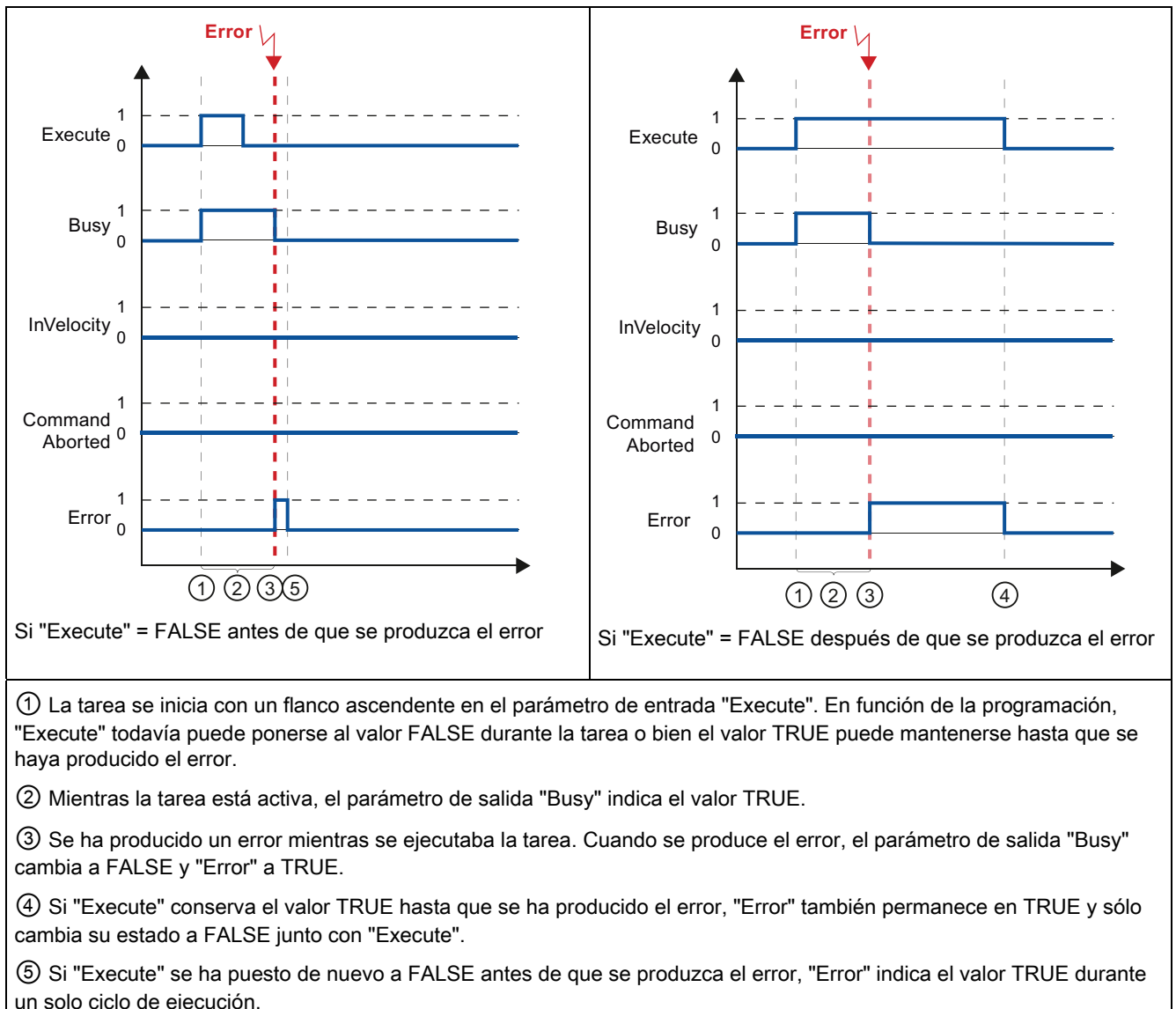
- ① La tarea se inicia con un flanco ascendente en el parámetro de entrada "Execute". En función de la programación, "Execute" todavía puede ponerse al valor FALSE durante la tarea o bien el valor TRUE puede mantenerse hasta que se haya cancelado la tarea.
- ② Mientras la tarea está activa, el parámetro de salida "Busy" indica el valor TRUE.
- ③ Mientras se ejecuta la tarea, ésta es cancelada por otra tarea Motion Control. Si se cancela la tarea, el parámetro de salida "Busy" cambia a FALSE y "CommandAborted" a TRUE.
- ④ Si "Execute" conserva el valor TRUE hasta que se ha cancelado la tarea, "CommandAborted" también permanece en TRUE y cambia su estado a FALSE junto con "Execute".
- ⑤ Si "Execute" se ha puesto de nuevo a FALSE antes de cancelar la tarea, "CommandAborted" indica el valor TRUE durante un solo ciclo de ejecución.

Nota

En las condiciones siguientes no se indica la cancelación en el parámetro de salida "CommandAborted":

- Se ha alcanzado la velocidad parametrizada, el parámetro de entrada "Execute" tiene el valor FALSE y se ha iniciado una nueva tarea Motion Control.
- Cuando se ha alcanzado la velocidad parametrizada y el parámetro de entrada "Execute" tiene el valor FALSE, la tarea se ha completado. Por consiguiente, el inicio de una tarea nueva no se señala como cancelación.

Tabla 9- 66 Ejemplo 3: si se produce un error antes de alcanzar la velocidad parametrizada



- ① La tarea se inicia con un flanco ascendente en el parámetro de entrada "Execute". En función de la programación, "Execute" todavía puede ponerse al valor FALSE durante la tarea o bien el valor TRUE puede mantenerse hasta que se haya producido el error.
- ② Mientras la tarea está activa, el parámetro de salida "Busy" indica el valor TRUE.
- ③ Se ha producido un error mientras se ejecutaba la tarea. Cuando se produce el error, el parámetro de salida "Busy" cambia a FALSE y "Error" a TRUE.
- ④ Si "Execute" conserva el valor TRUE hasta que se ha producido el error, "Error" también permanece en TRUE y sólo cambia su estado a FALSE junto con "Execute".
- ⑤ Si "Execute" se ha puesto de nuevo a FALSE antes de que se produzca el error, "Error" indica el valor TRUE durante un solo ciclo de ejecución.

Nota

En las condiciones siguientes no se indica un error en el parámetro de salida "Error":

- Se ha alcanzado la velocidad parametrizada, el parámetro de entrada "Execute" tiene el valor FALSE y se ha producido un error en el eje (se ha completado la aproximación al final de carrera por software, por ejemplo).
- Cuando se ha alcanzado la velocidad parametrizada y el parámetro de entrada "Execute" tiene el valor FALSE, la tarea se ha completado. Una vez completada la tarea, el error de eje sólo se indica en la instrucción Motion Control "MC_Power".

9.3.6.3 Vigilar la instrucción MC_MoveJog

Las tareas de la instrucción Motion Control "MC_MoveJog" implementan una operación Jog.

- Las tareas Motion Control "MC_MoveJog" no tienen un final definido. El objetivo de la tarea se cumple cuando se alcanza la velocidad parametrizada por primera vez y el eje se desplaza a una velocidad constante. Cuando se ha alcanzado la velocidad parametrizada, se indica con el valor TRUE en el parámetro de salida "InVelocity".
- La petición está completa cuando el parámetro de entrada "JogForward" o "JogBackward" se han puesto al valor FALSE y el eje se ha parado.
- Los parámetros de salida "Busy", "CommandAborted" y "Error" indican que la tarea todavía se está procesando, se ha cancelado o hay un error pendiente, respectivamente.
 - Durante el procesamiento de la tarea Motion Control, el parámetro de salida "Busy" indica el valor TRUE. Si la tarea se ha completado, cancelado o detenido a causa de un error, el parámetro de salida "Busy" cambia a FALSE.
 - El parámetro de salida "InVelocity" indica el estado TRUE mientras el eje se mueve a la velocidad parametrizada. Los parámetros de salida "CommandAborted" y "Error" indican el estado durante un ciclo como mínimo. Los avisos de estado están enclavados mientras el parámetro de entrada "JogForward" o "JogBackward" está puesto a TRUE.

El comportamiento del bit de estado se muestra más abajo en algunas situaciones de ejemplo.

- El primer ejemplo muestra el comportamiento del eje si se ha alcanzado y se mantiene la velocidad parametrizada. Si la tarea Motion Control se ha ejecutado en el momento en que se alcanza la velocidad parametrizada, se indica con el valor TRUE en el parámetro de salida "InVelocity".
- El segundo ejemplo muestra el comportamiento del eje si la tarea se cancela. Si la tarea Motion Control se cancela durante la ejecución, se indica con el valor TRUE en el parámetro de salida "CommandAborted". El comportamiento es independiente de si se ha alcanzado la velocidad parametrizada o no.
- El tercer ejemplo muestra el comportamiento del eje cuando se produce un error. Si se produce un error durante la ejecución de la tarea Motion Control, se indica con el valor TRUE en el parámetro de salida "Error". El comportamiento es independiente de si se ha alcanzado la velocidad parametrizada o no.

Tabla 9- 67 Ejemplo 1: si se ha alcanzado y se mantiene la velocidad parametrizada

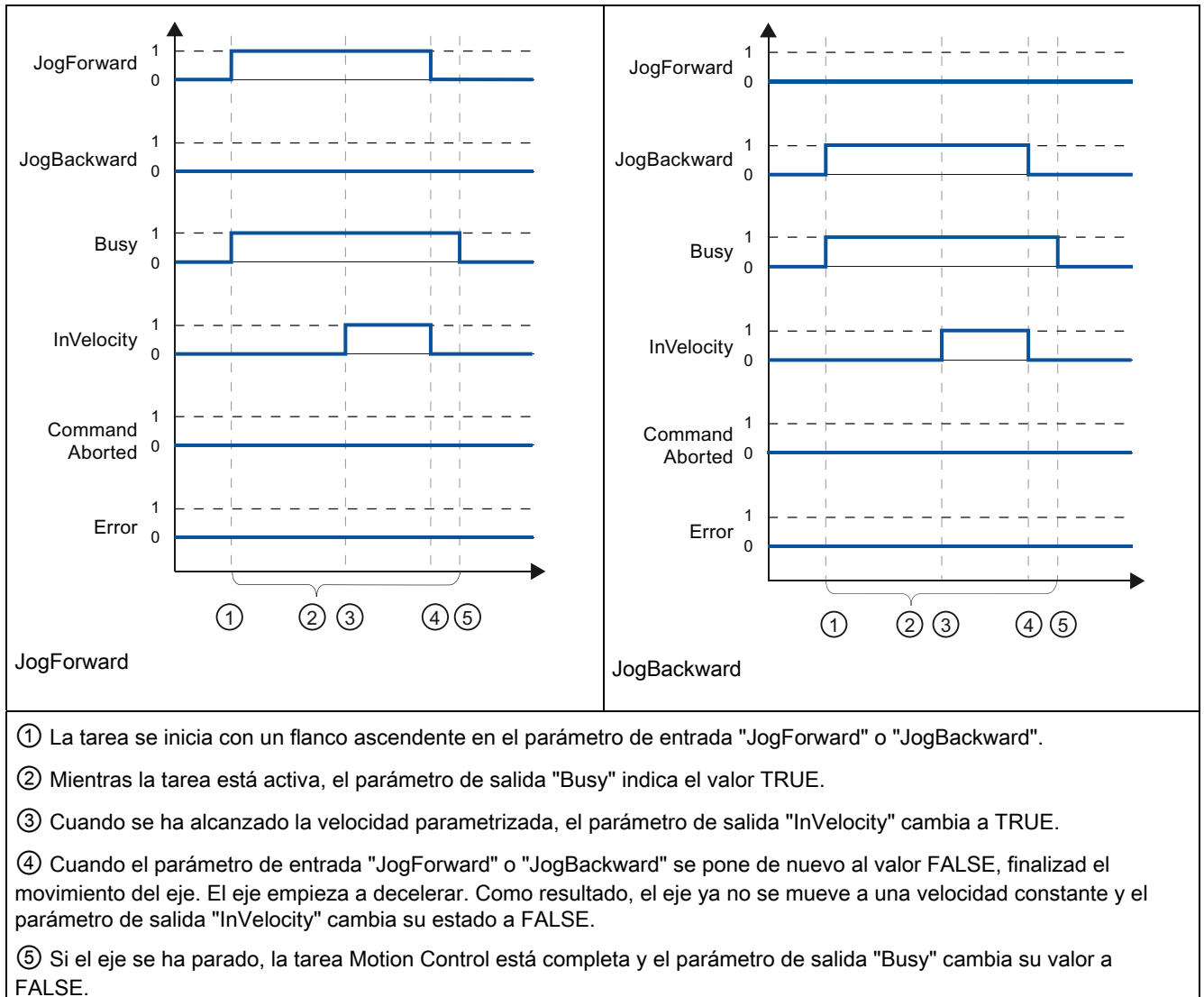
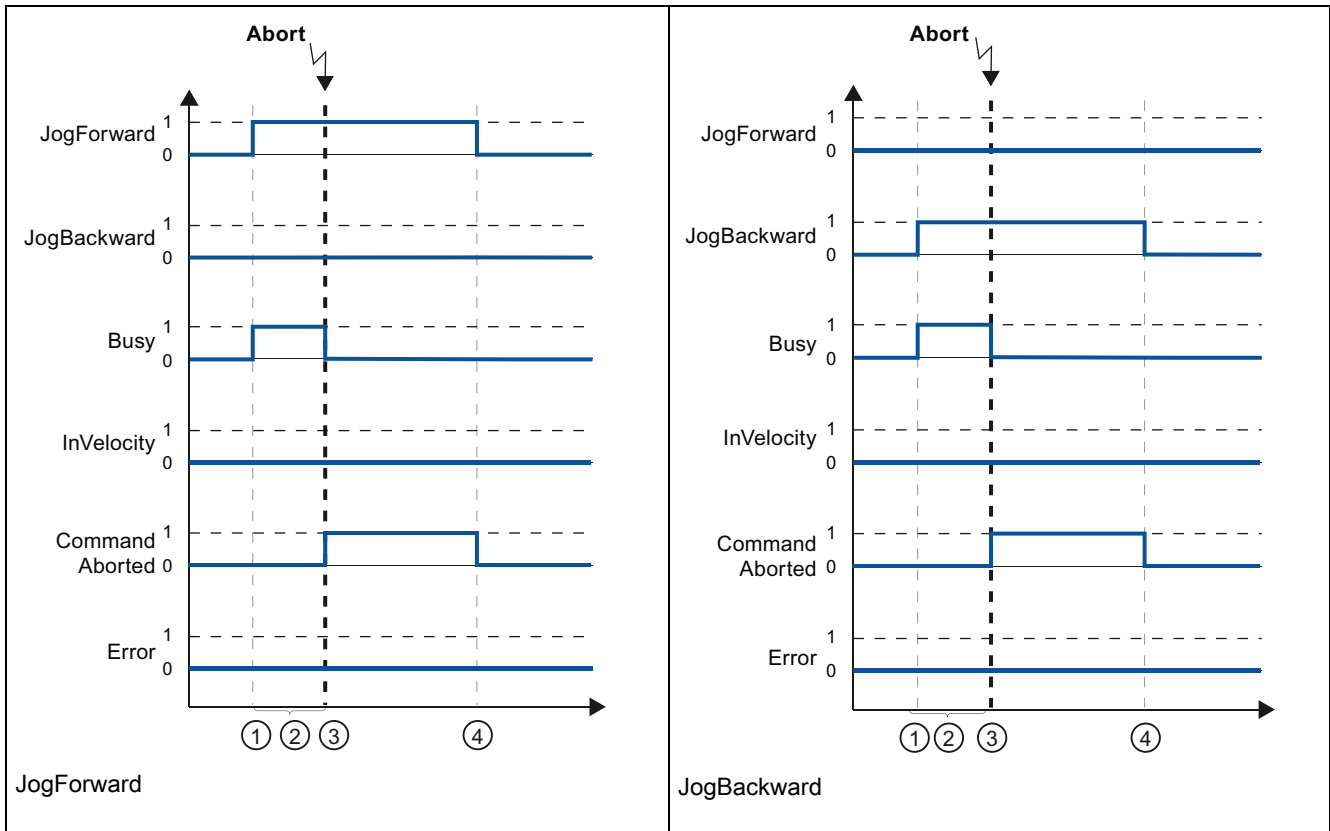


Tabla 9- 68 Ejemplo 2: si la tarea se cancela durante la ejecución



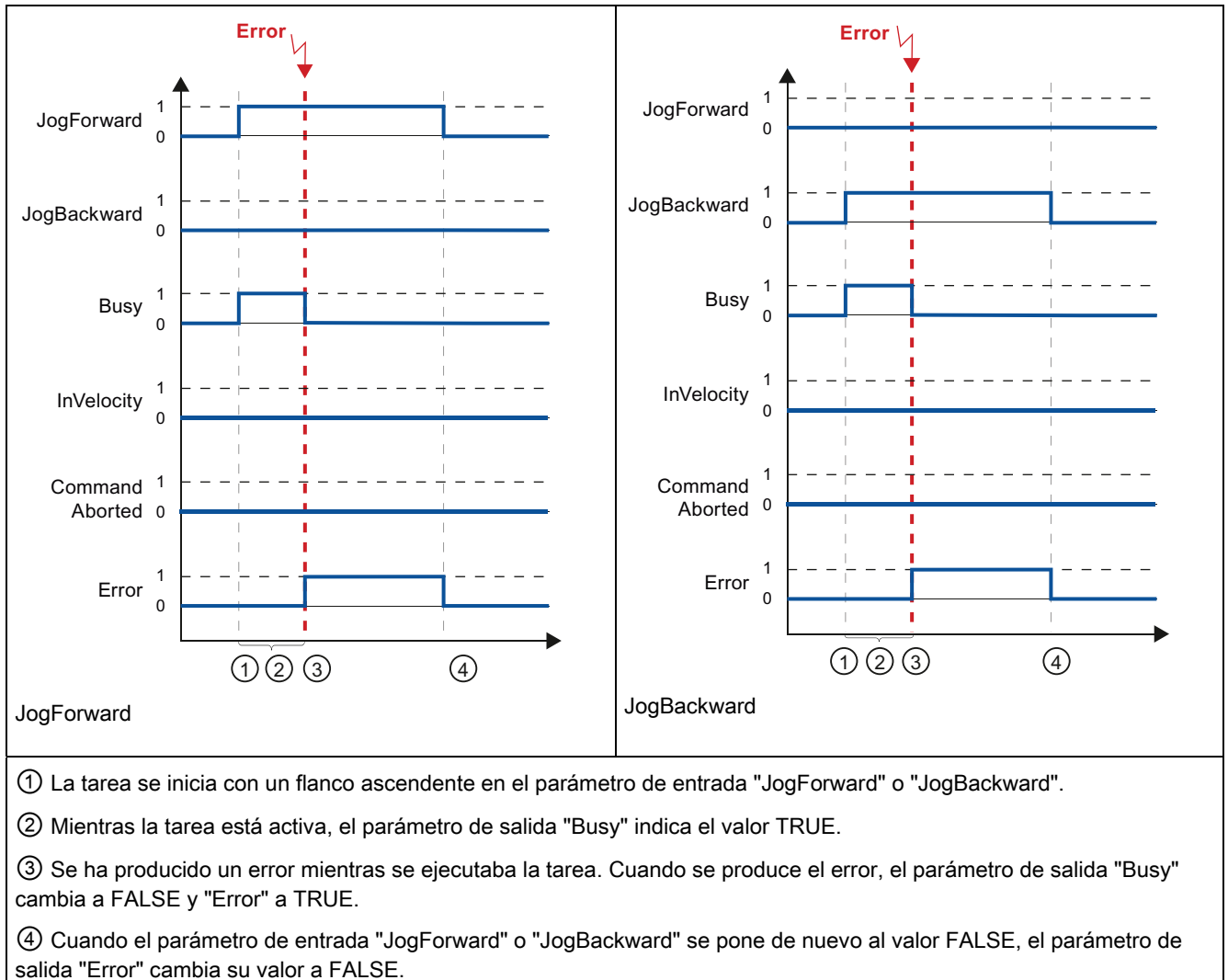
- ① La tarea se inicia con un flanco ascendente en el parámetro de entrada "JogForward" o "JogBackward".
- ② Mientras la tarea está activa, el parámetro de salida "Busy" indica el valor TRUE.
- ③ Mientras se ejecuta la tarea, ésta es cancelada por otra tarea Motion Control. Si se cancela la tarea, el parámetro de salida "Busy" cambia a FALSE y "CommandAborted" a TRUE.
- ④ Cuando el parámetro de entrada "JogForward" o "JogBackward" se pone de nuevo al valor FALSE, el parámetro de salida "CommandAborted" cambia su valor a FALSE.

Nota

La cancelación de la tarea se indica en el parámetro de salida "CommandAborted" durante un solo ciclo de ejecución, siempre que se cumplan todas las condiciones siguientes:

Los parámetros de entrada "JogForward" y "JogBackward" tienen el valor FALSE (pero el eje sigue desacelerando) y se ha iniciado una nueva tarea Motion Control.

Tabla 9- 69 Ejemplo 3: si se ha producido un error al ejecutar la tarea

**Nota**

La aparición del error se indica en el parámetro de salida "Error" durante un solo ciclo de ejecución, siempre que se cumplan todas las condiciones siguientes:

Los parámetros de entrada "JogForward" y "JogBackward" tienen el valor FALSE (pero el eje sigue desacelerando) y se ha producido un nuevo error (se ha completado la aproximación al final de carrera por software, por ejemplo).

Comunicación

S7-1200 ofrece varios tipos de comunicación entre CPU, y programadoras, HMI y otras CPU.

PROFINET

PROFINET se usa para intercambiar datos a través del programa de usuario con otros interlocutores vía Ethernet:

- La CPU ofrece este soporte para PROFINET y PROFIBUS:
 - En V3.0, PROFINET soporta un máximo de 16 dispositivos IO con un máximo de 256 submódulos. PROFIBUS permite 3 maestros PROFIBUS DP independientes, con 32 dispositivos IO y un máximo de 512 submódulos por dispositivo IO.
 - En V2.2, PROFINET soporta un máximo de 8 dispositivos IO con un máximo de 128 submódulos (si se han configurado ocho o menos submódulos o esclavos PROFIBUS). PROFIBUS soporta un máximo de 16 dispositivos PROFIBUS IO en un maestro, con un máximo de 256 submódulos por dispositivo IO.
- Comunicación S7
- Protocolo User Datagram Protocol (UDP)
- ISO on TCP (RFC 1006)
- Transport Control Protocol (TCP)

Controlador IO PROFINET RT

Como controlador IO que utiliza PROFINET RT, la CPU ofrece este soporte en la red PN local o a través de un acoplador PN/PN (link). Para obtener más información, consulte PROFIBUS and PROFINET International, PI (www.profinet.com):

- En V3.0, el S7-1200 se comunica con un máximo de 16 dispositivos PN.
- En V2.2, S7-1200 se comunica con un máximo de 8 dispositivos PN.

PROFIBUS

PROFIBUS se usa para intercambiar datos a través del programa de usuario con otros interlocutores a través de la red PROFIBUS:

- Con CM 1242-5 la CPU funciona como esclavo PROFIBUS DP.
- Con CM 1243-5 la CPU funciona como maestro PROFIBUS DP de clase 1.
- En V3.0 los esclavos PROFIBUS DP, maestros PROFIBUS DP y ASi (los tres módulos de comunicación del lado izquierdo) y PROFINET están separados.

10.1 Número de conexiones soportadas de comunicación asíncrona

- En V2.2 la CPU ofrece este soporte para PROFINET y PROFIBUS:
 - Un total de 16 dispositivos y 256 submódulos, con un máximo de 16 dispositivos esclavos y 256 submódulos PROFIBUS DP (si no hay dispositivos o submódulos PROFINET IO configurados).

Nota

En V2.2 los 16 dispositivos PROFINET y PROFIBUS en total comprenden:

- Los módulos esclavos PROFIBUS DP conectados por el maestro PROFIBUS DP (CM 1243-5)
- Cualquier módulo esclavo PROFIBUS DP (CM 1242-5) conectado a la CPU
- Cualquier dispositivo PROFINET conectado a la CPU a través del puerto PROFINET

Por ejemplo, una configuración con tres CM PROFIBUS (un CM 1243-5 maestro y dos CM 1242-5 esclavos) reduce el número máximo de módulos esclavos a los que puede acceder el maestro PROFIBUS DP (CM 1243-5) a 14.

- AS-i: el CM 1243-2 maestro AS-i del S7-1200 permite conectar una red AS-i a una CPU S7-1200.
- Comunicación S7 de CPU a CPU

Comunicación Teleservice

En TeleService vía GPRS, una estación de ingeniería en la que está instalado STEP 7 se comunica mediante la red GSM e Internet con una estación SIMATIC S7-1200 con un CP 1242-7. La conexión se ejecuta mediante un servidor de telecontrol que actúa de intermediario y que está conectado a Internet.

10.1 Número de conexiones soportadas de comunicación asíncrona

La CPU soporta el siguiente número máximo de conexiones asíncronas simultáneas para PROFINET y PROFIBUS:

- 8 conexiones para Open User Communications (activas o pasiva): TSEND_C, TRCV_C, TCON, TDISCON, TSEND y TRCV.
- 3 conexiones S7 de CPU a CPU para datos GET/PUT de servidor
- 8 conexiones S7 de CPU a CPU para datos GET/PUT de cliente

Nota

Las CPUs S7-1200, S7-300 y S7-400 usan las instrucciones GET y PUT para la comunicación S7 de CPU a CPU. Las CPUs S7-200 usan instrucciones ETHx_XFER para la comunicación S7 de CPU a CPU.

- Conexiones HMI: La CPU proporciona conexiones HMI dedicadas que admiten un máximo de 3 dispositivos HMI. Se pueden tener hasta 2 SIMATIC Comfort Panel. El número total de HMI depende del tipo de paneles HMI indicados en la configuración. Así, por ejemplo, pueden conectarse a la CPU hasta tres SIMATIC Basic Panels o bien hasta dos SIMATIC Comfort Panel con un Basic Panel.
- Conexiones PG: La CPU proporciona conexiones que admiten 1 programadora (PG).
- Conexiones Webserver (HTTP): La CPU proporciona conexiones para Webserver.

10.2 PROFINET

10.2.1 Conexión local/interlocutor

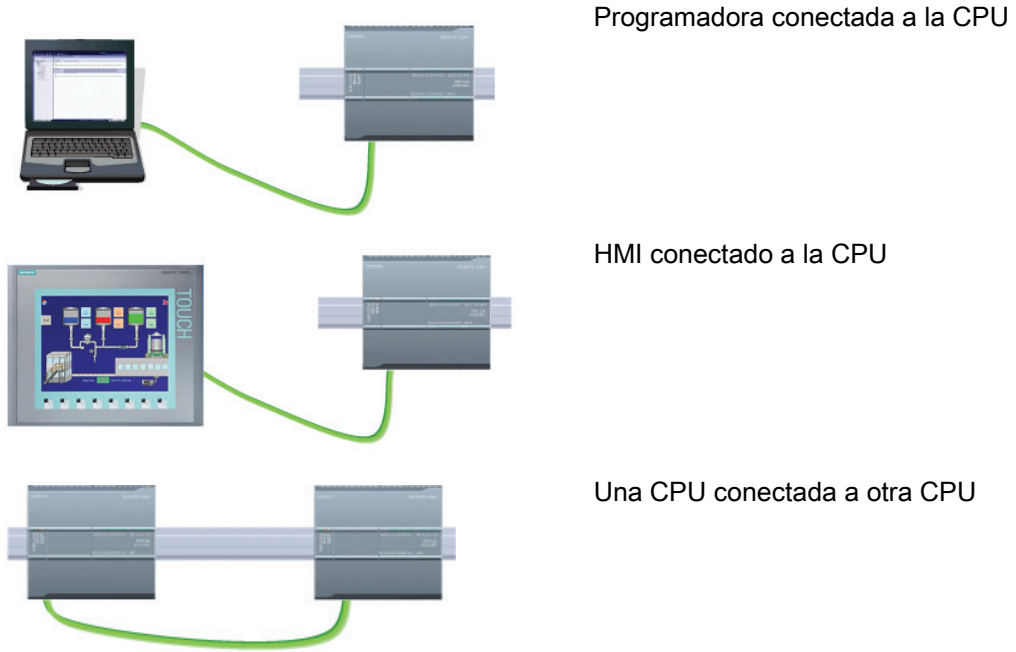
Una conexión local/interlocutor (remoto) define una asignación lógica de dos interlocutores para establecer servicios de comunicación. Una conexión define lo siguiente:

- Interlocutores involucrados (uno activo, otro pasivo)
- Tipo de conexión (por ejemplo una conexión de PLC, HMI o dispositivo)
- Ruta de conexión

Los interlocutores ejecutan las instrucciones necesarias para configurar y establecer la conexión. El punto final activo y el punto final pasivo de la comunicación se especifican mediante parámetros. Una vez configurada y establecida la conexión, la CPU la mantiene y la vigila automáticamente. Consulte la sección "Configurar la vía de conexión local/interlocutor" (Página 132) para obtener información sobre la configuración de parámetros para la conexión.

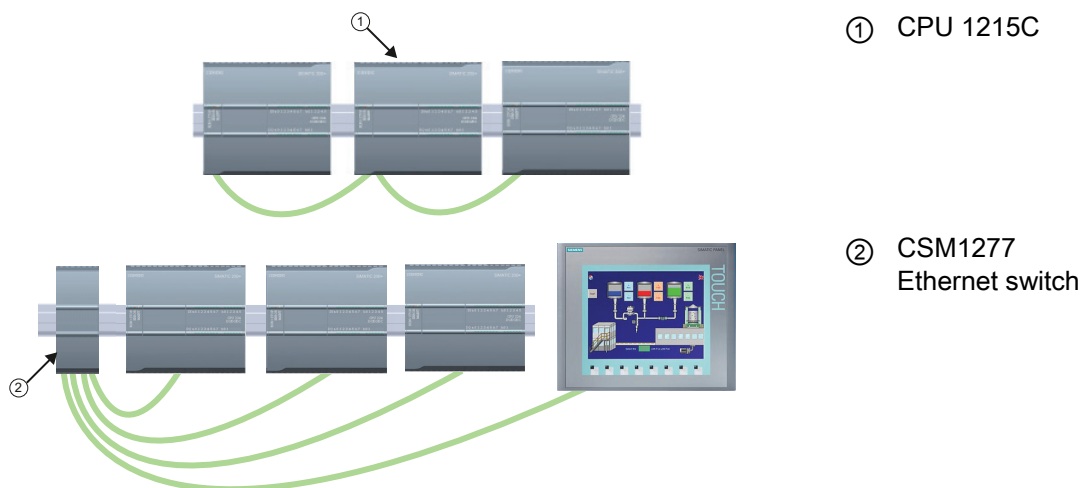
Si la conexión finaliza (p. ej. debido a una interrupción de la línea), el interlocutor activo intenta restablecer la conexión configurada. No es necesario volver a ejecutar la instrucción.

La CPU se puede comunicar con otras CPUs, con programadoras, con dispositivos HMI y con dispositivos no Siemens que utilicen protocolos de comunicación TCP estándar.



Conmutación Ethernet

El puerto PROFINET de las CPU 1211C, 1212C y 1214C no contiene ningún switch Ethernet. Una conexión directa entre una programadora o HMI y una CPU no requiere un conmutador Ethernet. Sin embargo, una red con más de dos CPUs o dispositivos HMI sí requiere un conmutador Ethernet.



La CPU 1215C sí que tiene un switch Ethernet de 2 puertos integrado. Se puede tener una red con una CPU 1215C y otras dos CPU S7-1200. También puede utilizarse el switch Ethernet de 4 puertos CSM1277 de montaje en rack para conectar varias CPU y dispositivos HMI.

10.2.2 Open User Communication

10.2.2.1 ID de conexión para las instrucciones PROFINET

Cuando se insertan las instrucciones PROFINET TSEND_C, TRCV_C o TCON en el programa de usuario, STEP 7 crea un DB de instancia para configurar el canal de comunicaciones (o conexión) entre los dispositivos. Utilice las "Propiedades" de la instrucción para configurar los parámetros de la conexión. Entre los parámetros se encuentra la ID de conexión para dicha conexión.

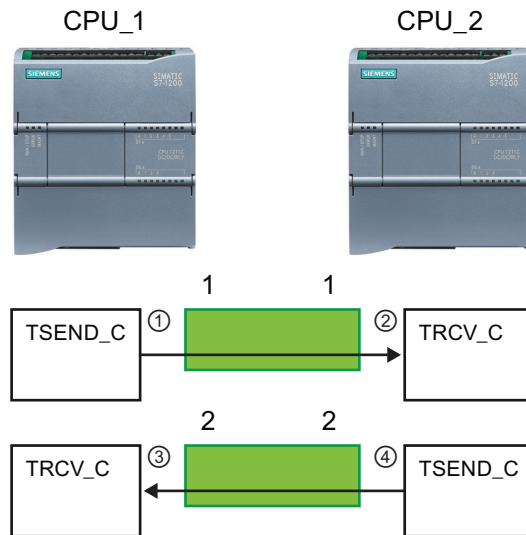
- La ID de conexión debe ser unívoca para la CPU. Cada conexión que se cree debe tener un DB y una ID de conexión distintos.
- Tanto la CPU local como la CPU interlocutora pueden utilizar el mismo número de ID de conexión para la misma conexión, aunque los números de ID de conexión no tienen que coincidir necesariamente. El número de ID de conexión sólo es relevante para las instrucciones PROFINET dentro del programa de usuario de la CPU individual.
- Puede utilizarse cualquier número para la ID de conexión de la CPU. Sin embargo, si se configuran las ID de conexión secuencialmente empezando por "1", es más fácil saber el número de conexiones que se están utilizando en una CPU concreta.

Nota

Cada instrucción TSEND_C, TRCV_C o TCON del programa de usuario crea una conexión nueva. Es importante utilizar la ID de conexión correcta para cada conexión.

El ejemplo siguiente muestra la comunicación entre dos CPUs que utilizan 2 conexiones separadas para transmitir y recibir datos.

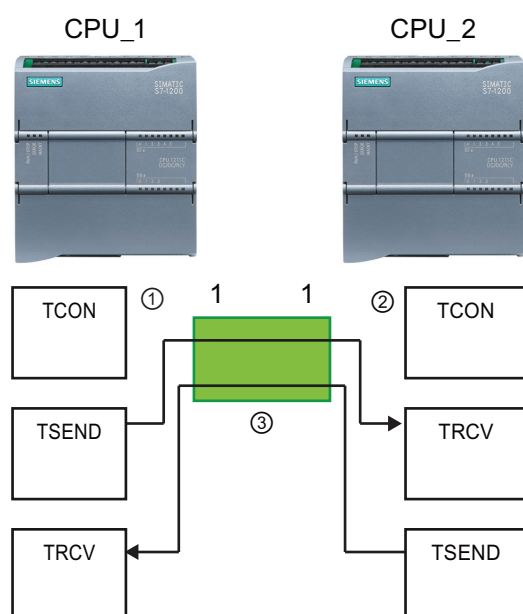
- La instrucción TSEND_C de la CPU_1 enlaza con TRCV_V de la CPU_2 a través de la primera conexión ("ID de conexión 1" tanto en la CPU_1 como en la CPU_2).
- La instrucción TRCV_C de la CPU_1 enlaza con TSEND_C de la CPU_2 a través de la segunda conexión ("ID de conexión 2" tanto en la CPU_1 como en la CPU_2).



- ① TSEND_C en la CPU_1 crea una conexión y asigna una ID de conexión a dicha conexión (ID de conexión 1 para la CPU_1).
- ② TRCV_C en la CPU_2 crea la conexión para la CPU_2 y asigna la ID de conexión (ID de conexión 1 para la CPU_2).
- ③ TRCV_C en la CPU_1 crea una segunda conexión para CPU_1 y asigna una ID de conexión distinta para dicha conexión (ID de conexión 2 para la CPU_1).
- ④ TSEND_C en la CPU_2 crea una segunda conexión y asigna una ID de conexión distinta para dicha conexión (ID de conexión 2 para la CPU_2).

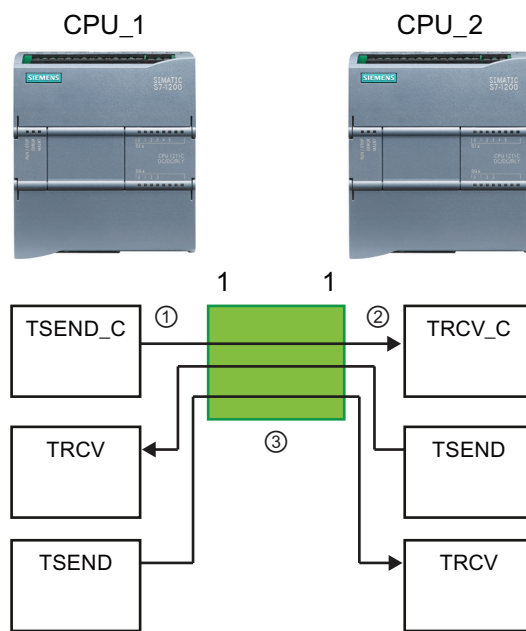
El ejemplo siguiente muestra la comunicación entre dos CPUs que utilizan 1 conexión tanto para transmitir como para recibir datos.

- Cada CPU utiliza una instrucción TCON para configurar la conexión entre las dos CPUs.
- La instrucción TSEND de la CPU_1 enlaza con la instrucción TRCV de la CPU_2 a través de la ID de conexión ("ID de conexión 1") configurada por la instrucción TCON de la CPU_1. La instrucción TRCV de la CPU_2 enlaza con la instrucción TSEND de la CPU_1 a través de la ID de conexión ("ID de conexión 1") configurada por la instrucción TCON de la CPU_2.
- La instrucción TSEND de la CPU_2 enlaza con la instrucción TRCV de la CPU_1 a través de la ID de conexión ("ID de conexión 1") configurada por la instrucción TCON de la CPU_2. La instrucción TRCV de la CPU_1 enlaza con la instrucción TSEND de la CPU_2 a través de la ID de conexión ("ID de conexión 1") configurada por la instrucción TCON de la CPU_1.



- ① TCON en la CPU_1 crea una conexión y asigna una ID de conexión a dicha conexión en la CPU_1 (ID=1).
- ② TCON en la CPU_2 crea una conexión y asigna una ID de conexión a dicha conexión en la CPU_2 (ID=1).
- ③ TSEND y TRCV en la CPU_1 utilizan la ID de conexión creada por TCON en la CPU_1 (ID=1). TSEND y TRCV en la CPU_2 utilizan la ID de conexión creada por TCON en la CPU_2 (ID=1).

Tal como se muestra en el ejemplo siguiente, también es posible utilizar instrucciones TSEND y TRCV individuales para comunicarse a través de una conexión creada por una instrucción TSEND_C o TRCV_C. Las instrucciones TSEND y TRCV no crean por sí solas una conexión nueva, por lo que deben utilizar el DB y la ID de conexión creados por una instrucción TSEND_C, TRCV_C o TCON.



- ① TSEND_C en la CPU_1 crea una conexión y asigna una ID de conexión a dicha conexión (ID=1).
- ② TRCV_C en la CPU_2 crea una conexión y asigna la ID de conexión a dicha conexión en la CPU_2 (ID=1).
- ③ TSEND y TRCV en la CPU_1 utilizan la ID de conexión creada por TSEND_C en la CPU_1 (ID=1). TSEND y TRCV en la CPU_2 utilizan la ID de conexión creada por TRCV_C en la CPU_2 (ID=1).

Consulte también

Configurar la vía de conexión local/de interlocutor (Página 132)

10.2.2.2 Protocolos

El puerto PROFINET integrado de la CPU soporta múltiples estándares de comunicación a través de una red Ethernet:

- Transport Control Protocol (TCP)
- ISO on TCP (RFC 1006)
- User Datagram Protocol (UDP)

Tabla 10- 1 Protocolos e instrucciones de comunicación para cada uno

Protocolo	Ejemplos de uso	Entrada de datos en el área de recepción	Instrucciones de comunicación	Tipo de direccionamiento
TCP	Comunicación de CPU a CPU Transporte de tramas	Modo ad hoc	Sólo TRCV_C y TRCV	Asigna números de puerto a los dispositivos local (activo) e interlocutor (pasivo)
		Recepción de datos con la longitud especificada	TSEND_C, TRCV_C, TCON, TDISCON, TSEND y TRCV	
ISO on TCP	Comunicación de CPU a CPU Fragmentación y reensamblado de mensajes	Modo ad hoc	Sólo TRCV_C y TRCV	Asigna TSAPs a los dispositivos local (activo) e interlocutor (pasivo)
		Controlado por protocolo	TSEND_C, TRCV_C, TCON, TDISCON, TSEND y TRCV	
UDP	Comunicación de CPU a CPU Comunicaciones de programa de usuario	User Datagram Protocol	TUSEND y TURCV	Asigna números de puerto a los dispositivos local (activo) e interlocutor (pasivo), pero no es una conexión fija
Comunicación S7	Comunicación de CPU a CPU Leer/escribir datos de/en una CPU	Transmisión y recepción de datos con la longitud especificada	GET y PUT	Asigna TSAPs a los dispositivos local (activo) e interlocutor (pasivo)
PROFINET RT	Comunicación de CPU a dispositivo PROFINET IO	Transmisión y recepción de datos con la longitud especificada	Incorporado	Incorporado

10.2.2.3 Modo ad hoc

Generalmente, TCP e ISO on TCP reciben paquetes de datos con longitud especificada, que varía entre 1 y 8192 bytes. Sin embargo, las instrucciones de comunicación TRCV_C y TRCV también ofrecen un modo de comunicación "ad hoc" que puede recibir paquetes de datos con una longitud variable entre 1 y 1472 bytes.

Nota

Si se guardan los datos en un DB "optimizado" (sólo simbólico), únicamente es posible recibir datos en matrices de tipos de datos Byte, Char, USInt y SInt.

Para configurar la instrucción TRCV_C o la TRCV para el modo ad hoc, ajuste el parámetro LEN a 65535 (0xFFFF).

Si no se llama la instrucción TRCV_C o TRCV en modo ad hoc con frecuencia, es posible que se reciban varios paquetes en una misma llamada. Ejemplo: si tuviera que recibir cinco paquetes de 100 bytes en una llamada, TCP los transferiría como un solo paquete de 500 bytes, mientras que ISO on TCP reestructuraría los paquetes en cinco de 100 bytes cada uno.

10.2.2.4 TCP y ISO on TCP

Transport Control Protocol (TCP) es un protocolo estándar descrito por RFC 793: Transmission Control Protocol. El objetivo principal de TCP es ofrecer un servicio de conexión seguro y fiable entre pares de procesos. Este protocolo tiene las características siguientes:

- Protocolo de comunicación eficiente puesto que está vinculado estrechamente al hardware
- Adecuado para cantidades de datos medianas y grandes (hasta 8192 bytes)
- Ofrece numerosas prestaciones más a las aplicaciones, además de una recuperación de errores, control de flujo y fiabilidad considerables.
- Protocolo orientado a la conexión
- Puede utilizarse muy flexiblemente con sistemas de terceros que soporten únicamente TCP
- Apto para routing
- Son aplicables sólo las longitudes de datos estáticas.
- Los mensajes se acusan.
- Las aplicaciones se direccionan usando números de puerto.
- La mayoría de los protocolos de aplicación (p. ej. TELNET y FTP) utilizan TCP.
- Es necesario programar la gestión de datos debido a la interfaz de programación SEND / RECEIVE.

International Standards Organization (ISO) on Transport Control Protocol (TCP) (RFC 1006) (ISO on TCP) es un mecanismo que permite portar aplicaciones ISO a la red TCP/IP. Este protocolo tiene las características siguientes:

- Protocolo de comunicación eficiente vinculado estrechamente al hardware
- Adecuado para cantidades de datos medianas y grandes (hasta 8192 bytes)
- A diferencia de TCP, los mensajes tienen un indicador de fin y están orientados a los mensajes.
- Apto para routing; puede utilizarse en WAN
- Las longitudes de datos dinámicas son posibles.
- Es necesario programar la gestión de datos debido a la interfaz de programación SEND / RECEIVE.

Puesto que utiliza Transport Service Access Points (TSAPs), el protocolo TCP permite establecer varias conexiones con una sola dirección IP (hasta 64K conexiones). Gracias a RFC 1006, los TSAPs identifican unívocamente estas conexiones de puntos finales de comunicación a una dirección IP.

TSEND_C y TRCV_C

La instrucción TSEND_C combina las funciones de las instrucciones TCON, TDISCON y TSEND . La instrucción TRCV_C combina las funciones de las instrucciones TCON, TDISCON y TRCV. (Véase "TCON, TDISCON, TSEND y TRCV (Página 457)" para más información sobre estas instrucciones.)

El tamaño mínimo de los datos que pueden transmitirse (TSEND_C) o recibirse (TRCV_C) es de un byte; el tamaño máximo es de 8192 bytes. TSEND_C no soporta la transmisión de datos desde posiciones booleanas y TRCV_C no recibe datos en posiciones booleanas. Encontrará más información sobre cómo transferir datos con estas instrucciones en el apartado Coherencia de datos (Página 157).

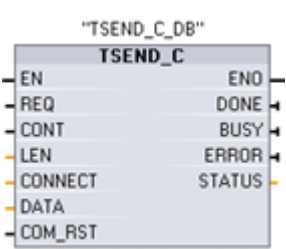
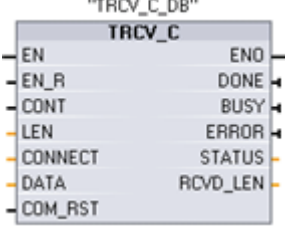
Nota

Inicializar los parámetros de comunicación

Después de introducir la instrucción TSEND_C o TRCV_C, utilice las "Propiedades" de la instrucción (Página 132) para configurar los parámetros de comunicación. Cuando se introducen los parámetros de los interlocutores en la ventana de inspección, STEP 7 introduce los datos correspondientes en el DB de la instrucción.

Si desea utilizar un DB multiinstancia, debe configurar manualmente el DB en ambas CPUs.

Tabla 10- 2 Instrucciones TSEND_C y TRCV_C

KOP / FUP	SCL	Descripción
 <p>"TSEND_C_DB" TSEND_C</p>	<pre>"TSEND_C_DB" (req:=_bool_in_, cont:=_bool_in_, len:=_uint_in_, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_, connect:=_struct_inout_, data:=_variant_inout_, com rst:=_bool_inout_);</pre>	<p>TSEND_C establece una conexión de comunicación TCP o ISO-on-TCP con un interlocutor, envía datos y puede deshacer la conexión. Una vez configurada y establecida la conexión, la CPU la mantiene y la vigila automáticamente.</p>
 <p>"TRCV_C_DB" TRCV_C</p>	<pre>"TRCV_C_DB" (en_r:=_bool_in_, cont:=_bool_in_, len:=_uint_in_, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_, rcvd_len=>_uint_out_, connect:=_struct_inout_, data:=_variant_inout_, com rst:=_bool_inout_);</pre>	<p>TRCV_C establece una conexión de comunicación TCP o ISO-on-TCP con una CPU interlocutora, recibe datos y puede deshacer la conexión. Una vez configurada y establecida la conexión, la CPU la mantiene y la vigila automáticamente.</p>

¹ STEP 7 crea el DB automáticamente al insertar la instrucción.

Tabla 10- 3 Tipos de datos TSEND_C y TRCV_C para los parámetros

Parámetro y tipo		Tipo de datos	Descripción
REQ (TSEND_C)	IN	Bool	El parámetro de control REQ inicia la tarea de transmisión con la conexión descrita en CONNECT cuando se detecta un flanco ascendente.
EN_R (TRCV_C)	IN	Bool	Parámetro de control habilitado para recibir: Si EN_R = 1, TRCV_C está preparado para recibir. La tarea de recepción se procesa.
CONT	IN	Bool	<ul style="list-style-type: none"> 0: Desconectar 1: Establecer y mantener la conexión
LEN	IN	UInt	Número máximo de bytes que deben enviarse (TSEND_C) o recibirse (TRCV_C): <ul style="list-style-type: none"> Predeterminado = 0: El parámetro DATA determina la longitud de datos transmitidos (TSEND_C) o recibidos (TRCV_C). Modo Ad hoc = 65535: Para la recepción se especifica una longitud de datos variable (TRCV_C).
CONNECT	IN_OUT	TCON_Param	Puntero hacia la descripción de la conexión
DATA	IN_OUT	Variant	<ul style="list-style-type: none"> Contiene la dirección y longitud de los datos que se van a enviar (TSEND_C). Contiene la dirección de inicio y la longitud máxima de los datos recibidos (TRCV_C).
COM_RST	IN_OUT	Bool	Permite reiniciar la instrucción: <ul style="list-style-type: none"> 0: Irrelevante 1: Reinicio completo del bloque de función; se deshace la conexión existente.
DONE	OUT	Bool	<ul style="list-style-type: none"> 0: Tarea no iniciada aún o en proceso. 1: Tarea finalizada sin errores.
BUSY	OUT	Bool	<ul style="list-style-type: none"> 0: Tarea finalizada. 1: Tarea no finalizada aún. No se puede iniciar una tarea nueva.
ERROR	OUT	Bool	Parámetros de estado con los valores siguientes: <ul style="list-style-type: none"> 0: No hay error 1: Ha ocurrido un error durante el procesamiento. STATUS proporciona información detallada sobre el tipo de error.
STATUS	OUT	Word	Información de estado, incluida información de error. (Véase la tabla "Parámetros de error y estado" a continuación.)
RCVD_LEN (TRCV_C)	OUT	Int	Cantidad de datos (en bytes) recibida realmente

Nota

La instrucción TSEND_C requiere una transición de bajo a alto en el parámetro de entrada REQ para iniciar una tarea de transmisión. El parámetro BUSY se pone a 1 durante el procesamiento. La finalización de la tarea de transmisión se indica poniendo a 1 los parámetros DONE o ERROR durante un ciclo. Durante este tiempo, se ignora cualquier transición de bajo a alto en el parámetro de entrada REQ.

Nota

El ajuste predeterminado del parámetro LEN (LEN = 0) utiliza el parámetro DATA para determinar la longitud de los datos que se están transmitiendo. Asegúrese de que los datos DATA transmitidos por la instrucción TSEND_C tengan el mismo tamaño que el parámetro DATA de la instrucción TRCV_C.

Operaciones TSEND_C

Las funciones siguientes describen el funcionamiento de la instrucción TSEND_C:

- Para establecer una conexión, ejecute TSEND_C con CONT = 1.
- Una vez establecida correctamente la conexión, TSEND_C activa el parámetro DONE durante un ciclo.
- Para deshacer la conexión, ejecute TSEND_C con CONT = 0. La conexión se interrumpirá inmediatamente. Esto afecta también la estación receptora. La conexión se cierra allí y pueden perderse los datos del búfer de recepción.
- Para enviar datos a través de una conexión establecida, ejecute TSEND_C cuando se produzca un flanco ascendente en REQ. Tras una operación de envío correcta, TSEND_C activa el parámetro DONE durante un ciclo.
- Para establecer una conexión y enviar datos, ejecute TSEND_C con CONT = 1 y REQ = 1. Tras una operación de envío correcta, TSEND_C activa el parámetro DONE durante un ciclo.

Operaciones TRCV_C

Las funciones siguientes describen el funcionamiento de la instrucción TRCV_C:

- Para establecer una conexión, ejecute TRCV_C con el parámetro CONT = 1.
- Para recibir datos, ejecute TRCV_C con el parámetro EN_R = 1. TRCV_C recibe los datos continuamente si los parámetros EN_R = 1 y CONT = 1.
- Para deshacer la conexión, ejecute TRCV_C con el parámetro CONT = 0. La conexión se deshace inmediatamente y pueden perderse datos.

TRCV_C utiliza los mismos modos de recepción que la instrucción TRCV. La tabla siguiente muestra cómo se introducen los datos en el área de recepción.

Tabla 10- 4 Entrada de datos en el área de recepción

Variante de protocolo	Entrada de datos en el área de recepción	Parámetro "connection_type"	Valor del parámetro LEN	Valor del parámetro RCVD_LEN (bytes)
TCP	Modo ad hoc	B#16#11	65535	1 a 1472
TCP	Recepción de datos con la longitud especificada	B#16#11	0 (recomendado) ó 1 a 8192, excepto 65535	1 a 8192
ISO on TCP	Modo ad hoc	B#16#12	65535	1 a 1472
ISO on TCP	Controlado por protocolo	B#16#12	0 (recomendado) ó 1 a 8192, excepto 65535	1 a 8192

Nota**Modo ad hoc**

El modo "ad hoc" sólo está disponible con las variantes de protocolo TCP e ISO on TCP. El modo "ad hoc" se establece asignando el valor "65535" al parámetro LEN. El área de recepción es idéntica al área especificada en el parámetro DATA. La longitud de los datos recibidos será transferida al parámetro RCVD_LEN.

Si se guardan los datos en un DB "optimizado" (sólo simbólico), únicamente es posible recibir datos en matrices de tipos de datos Byte, Char, USInt y SInt.

Nota**Importar al S7-1200 proyectos S7-300/400 de STEP 7 que contienen el modo "ad hoc"**

En los proyectos S7-300/400 de STEP 7, el modo "ad hoc" se selecciona asignando el valor "0" al parámetro LEN. En el S7-1200, el modo "ad hoc" se establece asignando el valor "65535" al parámetro LEN.

Si se importa al S7-1200 un proyecto S7-300/400 de STEP 7 que contiene el modo "ad hoc", el parámetro LEN debe modificarse a "65535".

Nota

Debido al procesamiento asíncrono de TSEND_C, es preciso conservar la coherencia de los datos en el área de emisión hasta que el parámetro DONE o ERROR adopte el valor TRUE.

Para TSEND_C, un estado TRUE en el parámetro DONE significa que los datos se han enviado correctamente. Sin embargo, no significa que la CPU interlocutora haya leído realmente el búfer de recepción.

Debido al procesamiento asíncrono de TRCV_C, los datos en el área de recepción sólo son coherentes si el parámetro DONE = 1.

Tabla 10- 5 Instrucciones TSEND_C y TRCV_C parámetros BUSY, DONEy ERROR

BUSY	DONE	ERROR	Descripción
TRUE	Irrelevante	Irrelevante	La tarea se está procesando.
FALSE	TRUE	FALSE	La tarea se ha completado correctamente.
FALSE	FALSE	TRUE	La tarea se ha finalizado con un error. La causa del error se indica en el parámetro STATUS.
FALSE	FALSE	FALSE	No se ha asignado ninguna tarea nueva.

Parámetros Error y Status

Tabla 10- 6 Códigos de condición TSEND_C y TRCV_C para ERROR y STATUS

ERROR	STATUS	Descripción
0	0000	Tarea ejecutada sin errores
0	7000	No se está procesando ninguna tarea
0	7001	Iniciar procesamiento de la tarea, estableciendo la conexión, esperando al interlocutor
0	7002	Enviando o recibiendo datos
0	7003	Deshaciendo la conexión
0	7004	Conexión establecida y vigilada, no se está procesando ninguna tarea
1	8085	El parámetro LEN es mayor que el valor máximo admisible.
1	8086	El parámetro CONNECT está fuera del rango admisible.
1	8087	Se ha alcanzado el número máximo de conexiones; no es posible establecer más conexiones.
1	8088	El parámetro LEN no es válido para el área de memoria especificada en DATA.
1	8089	El parámetro CONNECT no apunta a un bloque de datos.
1	8091	Se ha excedido la profundidad de anidamiento máxima.
1	809A	El parámetro CONNECT apunta a un campo que no concuerda con la longitud de la descripción de la conexión.
1	809B	La "local_device_id" indicada en la descripción de la conexión no concuerda con la CPU.
1	80A1	Error de comunicación: <ul style="list-style-type: none"> • La conexión indicada no se ha establecido todavía • La conexión indicada se está deshaciendo; no es posible transferir a través de esta conexión • La interfaz se está reiniciando
1	80A3	Se está intentando deshacer una conexión no existente
1	80A4	La dirección IP del interlocutor remoto de la conexión no es válida. Por ejemplo, la dirección IP del interlocutor remoto es igual a la dirección IP del interlocutor local.
1	80A5	ID de conexión en uso.
1	80A7	Error de comunicación: TDISCON se ha llamado antes de finalizar TSEND_C.
1	80B2	El parámetro CONNECT apunta a un bloque de datos generado con la palabra clave UNLINKED

ERROR	STATUS	Descripción
1	80B3	Parámetros incoherentes: <ul style="list-style-type: none"> • Error en la descripción de la conexión • El puerto local (parámetro local_tsap_id) ya existe en una descripción de conexión diferente. • La ID indicada en la descripción de la conexión no concuerda con la ID especificada como parámetro
1	80B4	Si se utiliza ISO on TCP (connection_type = B#16#12) para establecer una conexión pasiva, el código de condición 80B4 advierte de que el TSAP introducido no cumple uno de los siguientes requisitos para la dirección: <ul style="list-style-type: none"> • Si la longitud del TSAP local es 2 y el valor de la ID TSAP es E0 o E1 (hexadecimal) para el primer byte, el segundo byte deberá ser 00 ó 01. • Si la longitud del TSAP local es 3 o superior y el valor de la ID TSAP es E0 o E1 (hexadecimal) para el primer byte, el segundo byte deberá ser 00 ó 01 y todos los demás bytes deberán ser caracteres ASCII válidos. • Si la longitud del TSAP local es 3 o superior y el valor del primer byte de la ID TSAP no es E0 ni E1 (hexadecimal), todos los demás bytes de la ID TSAP deberán ser caracteres ASCII válidos. Los caracteres ASCII válidos son valores de byte comprendidos entre 20 y 7E (hexadecimal).
1	80B7	El tipo de datos y/o la longitud de los datos transmitidos no cabe en el área de la CPU interlocutora en la que deben escribirse.
1	80C3	Se están utilizando todos los recursos de conexión.
1	80C4	Error de comunicación temporal: <ul style="list-style-type: none"> • La conexión no se puede establecer en estos momentos • La interfaz está recibiendo nuevos parámetros • La instrucción TDISCON está deshaciendo la conexión configurada.
1	8722	Parámetro CONNECT: Área de origen no válida: el área no existe en el DB.
1	873A	Parámetro CONNECT: Imposible acceder a la descripción de la conexión (p. ej. porque el DB no está disponible).
1	877F	Parámetro CONNECT: Error interno, p. ej. referencia no válida a ANY
1	893A	El parámetro contiene el número de un DB que no está cargado.

Protocolos de conexión Ethernet

Toda CPU incorpora un puerto PROFINET que soporta la comunicación PROFINET estándar. Todas las instrucciones TSEND_C y TRCV_C y TSEND y TRCV soportan los protocolos TCP e ISO on TCP.

Encontrará más información en el apartado "Configuración de dispositivos: Configurar la vía de conexión local/interlocutor (Página 132)"

Consulte también

Parámetros de la conexión PROFINET (Página 134)

TCON, TDISCON, TSEND y TRCV

Comunicación Ethernet con los protocolos TCP e ISO on TCP

Nota

Instrucciones TSEND_C y TRCV_C

Para ayudar a simplificar la programación de la comunicación PROFINET/Ethernet, las instrucciones TSEND_C y TRCV_C combinan las funciones de las instrucciones TCON, TDISCON, TSEND y TRCV:

- TSEND_C combina las instrucciones TCON, TDISCON y TSEND.
- TRCV_C combina las instrucciones TCON, TDISCON y TRCV.

Las instrucciones siguientes controlan el proceso de comunicación:

- TCON establece la conexión TCP/IP entre el cliente y el servidor (CPU) PC.
- TSEND y TRCV permiten enviar y recibir datos.
- TDISCON deshace la conexión.

El tamaño mínimo de los datos que pueden transmitirse (TSEND) o recibirse (TRCV) es un byte; el tamaño máximo es 8192 bytes. TSEND no soporta la transmisión de datos desde posiciones booleanas y TRCV no recibe datos en posiciones booleanas. Encontrará más información sobre cómo transferir datos con estas instrucciones en el apartado Coherencia de datos (Página 157).

TCON, TDISCON, TSEND y TRCV funcionan de forma asíncrona, con lo que el procesamiento de la tarea abarca varias ejecuciones de la instrucción. Por ejemplo, una tarea para configurar y establecer una conexión se inicia ejecutando una instrucción TCON con el parámetro REQ = 1. Después se utilizan ejecuciones adicionales de TCON para vigilar la tarea y comprobar si ha finalizado con el parámetro DONE.

La tabla siguiente muestra la relación entre los parámetros BUSY, DONE y ERROR. Utilice la tabla para determinar el estado actual de la tarea.

Tabla 10- 7 Interacción entre los parámetros BUSY, DONE y ERROR

BUSY	DONE	ERROR	Descripción
TRUE	Irrelevante	Irrelevante	La tarea se está procesando.
FALSE	TRUE	FALSE	La tarea se ha ejecutado correctamente.
FALSE	FALSE	TRUE	La tarea se ha finalizado con un error. La causa del error se indica en el parámetro STATUS.
FALSE	FALSE	FALSE	No se ha asignado ninguna tarea nueva.

TCON y TDISCON

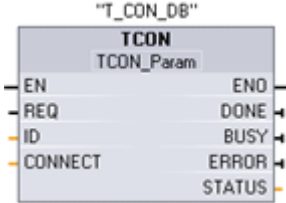
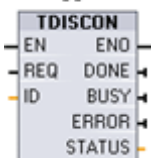
Nota

Inicializar los parámetros de comunicación

Después de introducir la instrucción TCON, utilice las "Propiedades" de la instrucción (Página 132) para configurar los parámetros de comunicación. Cuando se introducen los parámetros de los interlocutores en la ventana de inspección, STEP 7 introduce los datos correspondientes en el DB de instancia de la instrucción.

Si desea utilizar un DB multiinstancia, debe configurar manualmente el DB en ambas CPUs.

Tabla 10- 8 Instrucciones TCON y TDISCON

KOP / FUP		Descripción
	<pre>"TCON_DB" (req:=_bool_in_, ID:=_undef_in_, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_, connect:= struct inout);</pre>	TCP e ISO on TCP: TCON inicia una conexión entre la CPU y un interlocutor.
	<pre>"TDISCON_DB" (req:=_bool_in_, ID:=_word_in_, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_);</pre>	TCP e ISO on TCP: TDISCON deshace una conexión entre la CPU y un interlocutor.

¹ STEP 7 crea el DB automáticamente al insertar la instrucción.

Tabla 10- 9 Tipos de datos para los parámetros de TCON y TDISCON

Parámetro y tipo	Tipo de datos	Descripción
REQ	IN	Bool
ID	IN	CONN_OUC (Word)
CONNECT (TCON)	IN_OUT	TCON_Param

Parámetro y tipo		Tipo de datos	Descripción
DONE	OUT	Bool	<ul style="list-style-type: none"> 0: Tarea no iniciada aún o en proceso. 1: Tarea finalizada sin errores.
BUSY	OUT	Bool	<ul style="list-style-type: none"> 0: Tarea finalizada. 1: Tarea no finalizada aún. No se puede iniciar una tarea nueva.
ERROR	OUT	Bool	Parámetros de estado con los valores siguientes: <ul style="list-style-type: none"> 0: No hay error 1: Ha ocurrido un error durante el procesamiento. STATUS proporciona información detallada sobre el tipo de error.
STATUS	OUT	Word	Información de estado, incluida la información de error. (Consulte los códigos de condición de error y estado en la tabla siguiente.)

Ambos interlocutores ejecutan la instrucción TCON para configurar y establecer la conexión. El punto final activo y el punto final pasivo de la comunicación se especifican mediante parámetros. Una vez configurada y establecida la conexión, la CPU la mantiene y la vigila automáticamente.

Si la conexión se deshace p. ej. debido a una interrupción de la línea o por el interlocutor remoto, el interlocutor activo intenta restablecer la conexión configurada. No es necesario volver a ejecutar TCON.

Una conexión existente se deshace y la conexión configurada se elimina cuando se ejecuta la instrucción TDISCON o cuando la CPU pasa a estado operativo STOP. Para configurar y restablecer la conexión es preciso ejecutar TCON de nuevo.

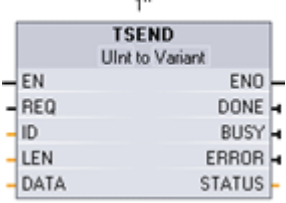
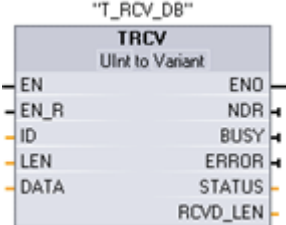
Tabla 10- 10 Códigos de condición ERROR y STATUS para TCON y TDISCON

ERROR	STATUS	Descripción
0	0000	La conexión se ha establecido correctamente.
0	7000	No se está procesando ninguna tarea
0	7001	Iniciar procesamiento de la tarea, estableciendo la conexión (TCON) o deshaciendo la conexión (TDISCON)
0	7002	Llamada intermedia (REQ es irrelevante), estableciendo la conexión (TCON) o deshaciendo la conexión (TDISCON)
1	8086	El parámetro ID está fuera del rango de direcciones admisible.
1	8087	TCON: se ha alcanzado el número máximo de conexiones; no es posible establecer más conexiones.
1	809B	TCON: la "local_device_id" indicada en la descripción de la conexión no concuerda con la CPU.
1	80A1	TCON: el usuario ya está utilizando el puerto o conexión.
1	80A2	TCON: el sistema está utilizando el puerto local o remoto.
1	80A3	Se está intentando restablecer una conexión existente (TCON) o deshacer una conexión no existente (TDISCON).
1	80A4	TCON: la dirección IP del punto final remoto de la conexión no es válida; puede ser que concuerde con la dirección IP local.

ERROR	STATUS	Descripción
1	80A5	TCON: ID de conexión en uso.
1 ()	80A7	TCON: error de comunicación: TDISCON se ha ejecutado antes de finalizar TCON. TDISCON debe deshacer primero por completo la conexión referenciada por la ID.
1	80B4	<p>TCON: si se utiliza ISO on TCP (connection_type = B#16#12) para establecer una conexión pasiva, el código de condición 80B4 advierte de que el TSAP introducido no cumple uno de los siguientes requisitos para la dirección:</p> <ul style="list-style-type: none"> • Si la longitud del TSAP local es 2 y el valor de la ID TSAP es E0 o E1 (hexadecimal) para el primer byte, el segundo byte deberá ser 00 ó 01. • Si la longitud del TSAP local es 3 o superior y el valor de la ID TSAP es E0 o E1 (hexadecimal) para el primer byte, el segundo byte deberá ser 00 ó 01 y todos los demás bytes deberán ser caracteres ASCII válidos. • Si la longitud del TSAP local es 3 o superior y el valor del primer byte de la ID TSAP no es E0 ni E1 (hexadecimal), todos los demás bytes de la ID TSAP deberán ser caracteres ASCII válidos. <p>Los caracteres ASCII válidos son valores de byte comprendidos entre 20 y 7E (hexadecimal).</p>
1	80B6	TCON: error de parametrización en el parámetro connection_type
1	80B7	TCON: el tipo de datos y/o la longitud de los datos transmitidos excede el área de la CPU interlocutora en la que se debe escribir.
1)	80B8	TCON: los parámetros de la descripción de la conexión local y el parámetro ID son diferentes.
1	80C3	TCON: se están utilizando todos los recursos de conexión.
1	80C4	<p>Error de comunicación temporal:</p> <ul style="list-style-type: none"> • La conexión no se puede establecer en estos momentos (TCON). • La instrucción TDISCON (TCON) está deshaciendo la conexión configurada. • La conexión se está estableciendo (TDISCON). • La interfaz está recibiendo nuevos parámetros (TCON y TDISCON).

TSEND y TRCV

Tabla 10- 11 Instrucciones TSEND y TRCV

KOP / FUP	SCL	Descripción
<p>"T_SEND_DB" 1¹</p> 	<pre>"TSEND_DB" (req:=_bool_in_, ID:=_word_in_, len:=_uint_in_, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_, data:=_variant_inout_);</pre>	<p>TCP e ISO on TCP: TSEND envía datos mediante una conexión entre la CPU y un interlocutor.</p>
<p>"T_RCV_DB"</p> 	<pre>"TRCV_DB" (en_r:=_bool_in_, ID:=_word_in_, len:=_uint_in_, ndr=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_, rcvd_len=>_uint_out_, data:=_variant_inout_);</pre>	<p>TCP e ISO on TCP: TRCV recibe datos mediante una conexión entre un interlocutor y la CPU.</p>

¹ STEP 7 crea el DB automáticamente al insertar la instrucción.

Tabla 10- 12 Tipos de datos para los parámetros de TSEND y TRCV

Parámetro y tipo	Tipo de datos	Descripción	
REQ	IN	Bool	TSEND: inicia la tarea de transmisión cuando se produce un flanco ascendente. Los datos se transfieren desde el área que indican los parámetros DATA y LEN.
EN_R	IN	Bool	TRCV: habilita la CPU para recibir, con EN_R = 1, TRCV está listo para recibir. La tarea de recepción se procesa.
ID	IN	CONN_OUC (Word)	Referencia a la conexión asociada. La ID debe ser idéntica al parámetro ID asociado en la descripción de la conexión local. Rango de valores: W#16#0001 a W#16#0FFF
LEN	IN	UInt	Número máximo de bytes que deben enviarse (TSEND) o recibirse (TRCV): <ul style="list-style-type: none"> • Predeterminado = 0: el parámetro DATA determina la longitud de los datos que deben enviarse (TSEND) o recibirse (TRCV). • Modo ad hoc = 65535: una longitud de datos variable está configurada para la recepción (TRCV).
DATA	IN_OUT	Variante	Puntero para enviar (TSEND) o recibir (TRCV) área de datos, el área de datos contiene la dirección y la longitud. La dirección hace referencia a la memoria I, Q, M o a un DB.

Parámetro y tipo		Tipo de datos	Descripción
DONE	OUT	Bool	TSEND: <ul style="list-style-type: none"> 0: Tarea no iniciada aún o en proceso. 1: Tarea finalizada sin errores.
NDR	OUT	Bool	TRCV: <ul style="list-style-type: none"> NDR = 0: Tarea no iniciada aún o en proceso. NDR = 1: Tarea finalizada correctamente.
BUSY	OUT	Bool	<ul style="list-style-type: none"> BUSY = 1: La tarea no ha finalizado aún. No se puede iniciar una tarea nueva. BUSY = 0: Tarea finalizada.
ERROR	OUT	Bool	ERROR = 1: Ha ocurrido un error durante el procesamiento. STATUS proporciona información detallada sobre el tipo de error.
STATUS	OUT	Word	Información de estado, incluida la información de error. (Consulte los códigos de condición de error y estado en la tabla siguiente.)
RCVD_LEN	OUT	Int	TRCV: cantidad de datos (en bytes) recibida realmente

Nota

La instrucción TSEND requiere una transición de bajo a alto en el parámetro de entrada REQ para iniciar una tarea de transmisión. El parámetro BUSY se pone a 1 durante el procesamiento. La finalización de la tarea de transmisión se indica poniendo a 1 los parámetros DONE o ERROR durante un ciclo. Durante este tiempo, se ignora cualquier transición de bajo a alto en el parámetro de entrada REQ.

Operaciones TRCV

La instrucción TRCV escribe los datos recibidos en un área de recepción especificada por las dos variables siguientes:

- Puntero al comienzo del área
- Longitud del área, o bien el valor suministrado en la entrada LEN si no es 0

Nota

El ajuste predeterminado del parámetro LEN (LEN = 0) utiliza el parámetro DATA para determinar la longitud de los datos que se están transmitiendo. Asegúrese de que los datos (DATA) transmitidos por la instrucción TSEND tienen el mismo tamaño que el parámetro DATA de la instrucción TRCV.

Tan pronto como se hayan recibido todos los datos de la tarea, TRCV los transferirá al área de recepción y pondrá NDR a 1.

Tabla 10- 13 Entrada de datos en el área de recepción

Variante de protocolo	Entrada de datos en el área de recepción	Parámetro "connection_type"	Valor del parámetro LEN	Valor del parámetro RCVD_LEN (bytes)
TCP	Modo ad hoc	B#16#11	65535	1 a 1472
TCP	Recepción de datos con la longitud especificada	B#16#11	0 (recomendado) ó 1 a 8192, excepto 65535	1 a 8192
ISO on TCP	Modo ad hoc	B#16#12	65535	1 a 1472
ISO on TCP	Controlado por protocolo	B#16#12	0 (recomendado) ó 1 a 8192, excepto 65535	1 a 8192

Nota

Modo ad hoc

El modo "ad hoc" sólo está disponible con las variantes de protocolo TCP e ISO on TCP. El modo "ad hoc" se establece asignando el valor "65535" al parámetro LEN. El área de recepción es idéntica al área especificada en el parámetro DATA. La longitud de los datos recibidos será transferida al parámetro RCVD_LEN. Inmediatamente después de recibir un bloque de datos, TRCV introduce los datos en el área de recepción y pone NDR a 1.

Si se guardan los datos en un DB "optimizado" (sólo simbólico), únicamente es posible recibir datos en matrices de tipos de datos Byte, Char, USInt y SInt.

Nota

Importar al S7-1200 proyectos S7-300/400 de STEP 7 que contienen el modo "ad hoc"

En los proyectos S7-300/400 de STEP 7, el modo "ad hoc" se selecciona asignando el valor "0" al parámetro LEN. En el S7-1200, el modo "ad hoc" se establece asignando el valor "65535" al parámetro LEN.

Si se importa al S7-1200 un proyecto S7-300/400 de STEP 7 que contiene el modo "ad hoc", el parámetro LEN debe modificarse a "65535".

Tabla 10- 14 Códigos de condición ERROR y STATUS para TSEND y TRCV

ERROR	STATUS	Descripción
0	0000	<ul style="list-style-type: none"> Tarea de transmisión finalizada sin errores (TSEND) Se han aceptado datos nuevos: La longitud actual de los datos recibidos se muestra en RCVD_LEN (TRCV).
0	7000	<ul style="list-style-type: none"> No se está procesando ninguna tarea (TSEND) El bloque no está listo para la recepción (TRCV)
0	7001	<ul style="list-style-type: none"> Inicio del procesamiento de la tarea, enviando los datos: Durante este procesamiento, el sistema operativo accede a los datos del área de emisión DATA (TSEND). El bloque está listo para la recepción, se ha activado la tarea de recepción (TRCV).

ERROR	STATUS	Descripción
0	7002	<ul style="list-style-type: none"> Ejecución intermedia de la instrucción (REQ es irrelevante), procesando la tarea: El sistema operativo accede a los datos del área de emisión DATA durante este procesamiento (TSEND). Ejecución intermedia de la instrucción, procesando la tarea de recepción: Los datos se escriben en el área de recepción durante este procesamiento. Por este motivo, un error puede generar datos incoherentes en el área de recepción (TRCV).
1	8085	<ul style="list-style-type: none"> El parámetro LEN es mayor que el valor máximo admisible (TSEND) y (TRCV). El parámetro LEN o DATA ha sido modificado desde la primera ejecución de la instrucción (TRCV).
1	8086	El parámetro ID está fuera del rango de direcciones admisible.
1	8088	El parámetro LEN excede el área de memoria especificada en DATA.
1	80A1	Error de comunicación: <ul style="list-style-type: none"> La conexión indicada no se ha establecido todavía (TSEND y TRCV). La conexión indicada se está deshaciendo. La transmisión o la tarea de recepción no es posible a través de esta conexión (TSEND y TRCV). La interfaz se está reiniciando (TSEND). La interfaz está recibiendo nuevos parámetros (TRCV).
1	80C3	Falta interna de recursos: Ya se está procesando un bloque con esta ID en una clase de prioridad diferente.
1	80C4	Error de comunicación temporal: <ul style="list-style-type: none"> La conexión con el interlocutor no se puede establecer en estos momentos. La interfaz está recibiendo nuevos ajustes de parámetros o la conexión se está estableciendo.

Protocolos de conexión Ethernet

Toda CPU incorpora un puerto PROFINET que soporta la comunicación PROFINET estándar. Las instrucciones TSEND_C, TRCV_C, TSEND y TRCV soportan los protocolos Ethernet para TCP e ISO on TCP.

Encontrará más información en el apartado "Configuración de dispositivos: Configurar la vía de conexión local/interlocutor (Página 132)"

Consulte también

Parámetros de la conexión PROFINET (Página 134)

10.2.2.5 UDP

UDP es un protocolo estándar descrito por RFC 768: User Datagram Protocol. UDP ofrece un mecanismo que le permite a una aplicación enviar un datagrama a otra; no obstante, el suministro de datos no está garantizado. Este protocolo tiene las características siguientes:

- Protocolo de comunicaciones rápido, puesto que está vinculado estrechamente al hardware
- Adecuado para cantidades de datos pequeñas y medianas (hasta 2048 bytes)
- UDP es un protocolo de control de transporte más sencillo que TCP, con una capa delgada que tiene un nivel bajo de overheads
- Puede utilizarse muy flexiblemente con distintos sistemas de terceros
- Apto para routing
- Utiliza números de puerto para direccionar los datagramas
- Los avisos no se acusan: esta aplicación se requiere para asumir la responsabilidad en cuanto a la recuperación de errores y a la seguridad
- Es necesario programar la gestión de datos debido a la interfaz de programación SEND / RECEIVE

UDP soporta la comunicación Broadcast. Para utilizar Broadcast es necesario configurar la parte de la dirección IP correspondiente a la configuración de ADDR. Ejemplo: una CPU que tenga una dirección IP 192.168.2.10 y una máscara de subred 255.255.255.0 utilizaría una dirección Broadcast de 192.168.2.255.

TUSEND y TURCV

Las instrucciones siguientes controlan el proceso de comunicación UDP:

- TCON establece la comunicación entre el PC cliente y servidor (CPU).
- TUSEND y TURCV permiten transmitir y recibir datos.
- TDISCON desconecta la comunicación entre el cliente y servidor.

Consulte TCON, TDISCON, TSEND y TRCV (Página 457) en la sección "TCP e ISO on TCP" para obtener más información sobre las instrucciones de comunicación TCON y TDISCON.

Tabla 10- 15 Instrucciones TUSEND y TURCV

KOP / FUP	SCL	Descripción
<p>"TUSEND_DB"</p>	<pre>"TUSEND_DB" (req:=_bool_in_, ID:=_word_in_, len:=_uint_in_, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_, data:=_variant_inout_);</pre>	<p>La instrucción TUSEND envía datos a través de UDP al interlocutor remoto especificado en el parámetro ADDR.</p> <p>Para iniciar la tarea de transmisión de datos llame la instrucción TUSEND con REQ = 1.</p>
<p>"TURCV_DB"</p>	<pre>"TURCV_DB" (en_r:=_bool_in_, ID:=_word_in_, len:=_uint_in_, ndr=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_word_out_, rcvd_len=>_uint_out_, data:=_variant_inout_);</pre>	<p>La instrucción TURCV recibe datos a través de UDP. El parámetro ADDR muestra la dirección del emisor. Tras ejecutar correctamente TURCV, el parámetro ADDR contiene la dirección del interlocutor remoto (emisor).</p> <p>TURCV no soporta el modo ad hoc.</p> <p>Para iniciar la tarea de recepción de datos llame la instrucción TURCV con EN_R = 1.</p>

¹ STEP 7 crea el DB automáticamente al insertar la instrucción.

TCON, TDISCON, TUSEND y TURCV funcionan de forma asíncrona, con lo que el procesamiento de la tarea abarca varias ejecuciones de la instrucción.

Tabla 10- 16 Tipos de datos TUSEND y TURCV para los parámetros

Parámetro y tipo	Tipo de datos	Descripción	
REQ (TUSEND)	IN	Bool	Inicia la tarea de transmisión cuando se produce un flanco ascendente. Los datos se transfieren desde el área que indican los parámetros DATA y LEN.
EN_R (TURCV)	IN	Bool	<ul style="list-style-type: none"> 0: La CPU no está habilitada para recibir. 1: Habilita la CPU para recibir. La instrucción TURCV está lista para recibir y la tarea de recepción se procesa.
ID	IN	Word	Referencia a la conexión asociada entre el programa de usuario y el nivel de comunicación del sistema operativo. La ID tiene que ser idéntica al parámetro asociado ID en la descripción de la conexión local. Rango de valores: W#16#0001 a W#16#0FFF.
LEN	IN	UInt	Número de bytes que deben enviarse (TUSEND) o recibirse (TURCV): <ul style="list-style-type: none"> Por defecto = 0. El parámetro DATA determina la longitud de los datos que deben enviarse o recibirse. En otro caso, rango de valores: 1 a 1472
DONE (TUSEND)	IN	Bool	Parámetro de estado DONE (TUSEND): <ul style="list-style-type: none"> 0: Tarea no iniciada aún o en proceso. 1: Tarea finalizada sin errores.

Parámetro y tipo		Tipo de datos	Descripción
NDR (TURCV)	OUT	Bool	Parámetro de estado NDR (TURCV): <ul style="list-style-type: none"> 0: Tarea no iniciada aún o en proceso. 1: Tarea finalizada correctamente.
BUSY	OUT	Bool	<ul style="list-style-type: none"> 1: Tarea no finalizada aún. No se puede iniciar una tarea nueva. 0: Tarea finalizada.
ERROR	OUT	Bool	Parámetros de estado con los valores siguientes: <ul style="list-style-type: none"> 0: No hay error 1: Ha ocurrido un error durante el procesamiento. STATUS proporciona información detallada sobre el tipo de error.
STATUS	OUT	Word	Información de estado, incluida la información de error. (Consulte los códigos de condición de error y estado en la tabla siguiente.)
DATA	IN_OUT	Variant	Dirección del área de emisión (TUSEND) o del área de recepción (TURCV): <ul style="list-style-type: none"> La memoria imagen de proceso de las entradas La memoria imagen de proceso de las salidas Un bit de marcas Un bloque de datos
ADDR	IN_OUT	Variant	Puntero a la dirección del receptor (para TUSEND) o del emisor (para TURCV) (por ejemplo, P#DB100.DBX0.0 byte 8). El puntero puede apuntar a cualquier área de memoria. Se requiere una estructura de 8 bytes como se indica a continuación: <ul style="list-style-type: none"> Los primeros 4 bytes contienen la dirección IP remota. Los 2 bytes siguientes especifican el número de puerto remoto. Los últimos 2 bytes están reservados.

El estado de la tarea se indica en los parámetros de salida BUSY y STATUS. STATUS equivale al parámetro de salida RET_VAL de las instrucciones que operan de forma asíncrona.

La tabla siguiente muestra la relación entre los parámetros BUSY, DONE (TUSEND), NDR (TURCV) y ERROR. Utilizando esta tabla se puede determinar el estado actual de la instrucción (TUSEND o TURCV) o bien el instante en que finaliza el proceso de envío (transmisión) / recepción.

Tabla 10- 17 Estado de los parámetros BUSY, DONE (TUSEND) / NDR (TURCV) y ERROR

BUSY	DONE / NDR	ERROR	Descripción
TRUE	Irrelevante	Irrelevante	La tarea se está procesando.
FALSE	TRUE	FALSE	La tarea se ha ejecutado correctamente.

BUSY	DONE / NDR	ERROR	Descripción
FALSE	FALSE	TRUE	La tarea se ha finalizado con un error. La causa del error se indica en el parámetro STATUS.
FALSE	FALSE	FALSE	A la instrucción no se le ha asignado una tarea (nueva).

¹ Debido al funcionamiento asíncrono de las instrucciones: Para TUSEND, es preciso conservar la coherencia de los datos en el área de emisión hasta que el parámetro DONE o ERROR adopta el valor TRUE. Para TURCV, los datos en el área de recepción sólo son coherentes si el parámetro NDR adopta el valor TRUE.

Tabla 10- 18 Códigos de condición TUSEND y TURCV para ERROR y STATUS

ERROR	STATUS	Descripción
0	0000	<ul style="list-style-type: none"> Tarea de transmisión finalizada sin errores (TUSEND). Se han aceptado datos nuevos. La longitud actual de los datos recibidos se muestra en RCVD_LEN (TURCV).
0	7000	<ul style="list-style-type: none"> No se está procesando ninguna tarea (TUSEND) El bloque no está listo para la recepción (TURCV)
0	7001	<ul style="list-style-type: none"> Inicio del procesamiento de la tarea, enviando los datos (TUSEND): Durante este procesamiento, el sistema operativo accede a los datos del área de emisión DATA. El bloque está listo para la recepción, se ha activado la tarea de recepción (TURCV).
0	7002	<ul style="list-style-type: none"> Ejecución intermedia de la instrucción (REQ es irrelevante), procesando la tarea (TUSEND): Durante este procesamiento, el sistema operativo accede a los datos del área de emisión DATA. Ejecución intermedia de la instrucción, procesando la tarea: Durante este procesamiento, la instrucción TURCV escribe datos en el área de recepción. Por este motivo, un error puede generar datos incoherentes en el área de recepción.
1	8085	El parámetro LEN excede el valor máximo admisible, tiene el valor 0 (TUSEND) o bien se ha cambiado el valor del parámetro LEN o DATA desde la primera ejecución de la instrucción (TURCV).
1	8086	El parámetro ID está fuera del rango de direcciones admisible.
1	8088	<ul style="list-style-type: none"> El parámetro LEN excede el área de memoria (TUSEND) o de recepción (TURCV) especificada en DATA. El área de recepción es demasiado pequeña (TURCV).
1	8089	El parámetro ADDR no apunta a un bloque de datos.
1	80A1	<p>Error de comunicación:</p> <ul style="list-style-type: none"> Aún no se ha establecido la conexión especificada entre el programa de usuario y la capa de comunicación del sistema operativo. Se está deshaciendo la conexión especificada entre el programa de usuario y la capa de comunicación del sistema operativo. La transmisión (TUSEND) o la tarea de recepción (TURCV) no es posible a través de esta conexión. La interfaz se está reiniciando.
1	80A4	La dirección IP del punto final remoto de la conexión no es válida; puede ser que concuerde con la dirección IP local (TUSEND).

ERROR	STATUS	Descripción
1	80B3	<ul style="list-style-type: none"> El protocolo definido (parámetro connection_type en la descripción de la conexión) no es UDP. Utilice la instrucción TSEND o TRCV. Parámetro ADDR: Ajustes no válidos para el número de puerto (TUSEND)
1	80C3	<ul style="list-style-type: none"> Ya se está procesando un bloque con esta ID en una clase de prioridad diferente. Falta interna de recursos
1	80C4	Error de comunicación temporal: <ul style="list-style-type: none"> La conexión entre el programa de usuario y la capa de comunicación del sistema operativo no se puede establecer en estos momentos (TUSEND). La interfaz está recibiendo nuevos parámetros (TUSEND). La conexión se está reiniciando (TURCV).

Protocolos de conexión Ethernet

Toda CPU incorpora un puerto PROFINET que soporta la comunicación PROFINET estándar. Las instrucciones TUSEND y TURCV soportan el protocolo Ethernet para UDP.

Encontrará más información en "Configurar la vía de conexión local/interlocutor" (Página 132), en el capítulo "Configuración de dispositivos".

Operaciones

Los dos interlocutores son pasivos en la comunicación UDP. La tabla siguiente muestra los valores iniciales típicos de los parámetros para el tipo de datos "TCON_Param". Los números de puerto (LOCAL_TSAP_ID) están escritos en un formato de 2 bytes. Se permiten todos los puertos excepto 161, 34962, 34963 y 34964.

Tabla 10- 19 "Valores de parámetros para el tipo de datos "TCON_Param"

Instrucción TCON	TCON "UDP Conn DB"																																																																																
	<table border="1"> <thead> <tr> <th>Nombre</th> <th>Tipo de datos</th> <th>Offset</th> <th>Valor de arranque</th> <th>Comentario</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Static</td> <td></td> <td></td> <td></td> </tr> <tr> <td>2</td> <td>Params</td> <td>TCON_Param</td> <td>0.0</td> <td></td> </tr> <tr> <td>3</td> <td>BLOCK_LENGTH</td> <td>UInt</td> <td>0.0</td> <td>64 byte length of SOT</td> </tr> <tr> <td>4</td> <td>ID</td> <td>CONN_OUC</td> <td>2.0</td> <td>1 reference to the connection</td> </tr> <tr> <td>5</td> <td>CONNECTION_TYPE</td> <td>USInt</td> <td>4.0</td> <td>19 17: TCP/IP, 18: ISO on TCP</td> </tr> <tr> <td>6</td> <td>ACTIVE_EST</td> <td>Bool</td> <td>5.0</td> <td>false active/passive connection establishment</td> </tr> <tr> <td>7</td> <td>LOCAL_DEVICE_ID</td> <td>USInt</td> <td>6.0</td> <td>1 1: local IE interface</td> </tr> <tr> <td>8</td> <td>LOCAL_TSAP_ID_LEN</td> <td>USInt</td> <td>7.0</td> <td>2 byte length of local TSAP id/port number</td> </tr> <tr> <td>9</td> <td>REM_SUBNET_ID_LEN</td> <td>USInt</td> <td>8.0</td> <td>0 byte length of remote subnet id</td> </tr> <tr> <td>10</td> <td>REM_STADDR_LEN</td> <td>USInt</td> <td>9.0</td> <td>0 byte length of remote IP address</td> </tr> <tr> <td>11</td> <td>REM_TSAP_ID_LEN</td> <td>USInt</td> <td>10.0</td> <td>0 byte length of remote port/TSAP id</td> </tr> <tr> <td>12</td> <td>NEXT_STADDR_LEN</td> <td>USInt</td> <td>11.0</td> <td>0 byte length of next station address</td> </tr> <tr> <td>13</td> <td>LOCAL_TSAP_ID</td> <td>Array[1..16] of Byte</td> <td>12.0</td> <td></td> </tr> <tr> <td>14</td> <td>LOCAL_TSAP_ID[1]</td> <td>Byte</td> <td></td> <td>B#16#07</td> </tr> <tr> <td>15</td> <td>LOCAL_TSAP_ID[2]</td> <td>Byte</td> <td></td> <td>B#16#D0</td> </tr> </tbody> </table>	Nombre	Tipo de datos	Offset	Valor de arranque	Comentario	1	Static				2	Params	TCON_Param	0.0		3	BLOCK_LENGTH	UInt	0.0	64 byte length of SOT	4	ID	CONN_OUC	2.0	1 reference to the connection	5	CONNECTION_TYPE	USInt	4.0	19 17: TCP/IP, 18: ISO on TCP	6	ACTIVE_EST	Bool	5.0	false active/passive connection establishment	7	LOCAL_DEVICE_ID	USInt	6.0	1 1: local IE interface	8	LOCAL_TSAP_ID_LEN	USInt	7.0	2 byte length of local TSAP id/port number	9	REM_SUBNET_ID_LEN	USInt	8.0	0 byte length of remote subnet id	10	REM_STADDR_LEN	USInt	9.0	0 byte length of remote IP address	11	REM_TSAP_ID_LEN	USInt	10.0	0 byte length of remote port/TSAP id	12	NEXT_STADDR_LEN	USInt	11.0	0 byte length of next station address	13	LOCAL_TSAP_ID	Array[1..16] of Byte	12.0		14	LOCAL_TSAP_ID[1]	Byte		B#16#07	15	LOCAL_TSAP_ID[2]	Byte		B#16#D0
Nombre	Tipo de datos	Offset	Valor de arranque	Comentario																																																																													
1	Static																																																																																
2	Params	TCON_Param	0.0																																																																														
3	BLOCK_LENGTH	UInt	0.0	64 byte length of SOT																																																																													
4	ID	CONN_OUC	2.0	1 reference to the connection																																																																													
5	CONNECTION_TYPE	USInt	4.0	19 17: TCP/IP, 18: ISO on TCP																																																																													
6	ACTIVE_EST	Bool	5.0	false active/passive connection establishment																																																																													
7	LOCAL_DEVICE_ID	USInt	6.0	1 1: local IE interface																																																																													
8	LOCAL_TSAP_ID_LEN	USInt	7.0	2 byte length of local TSAP id/port number																																																																													
9	REM_SUBNET_ID_LEN	USInt	8.0	0 byte length of remote subnet id																																																																													
10	REM_STADDR_LEN	USInt	9.0	0 byte length of remote IP address																																																																													
11	REM_TSAP_ID_LEN	USInt	10.0	0 byte length of remote port/TSAP id																																																																													
12	NEXT_STADDR_LEN	USInt	11.0	0 byte length of next station address																																																																													
13	LOCAL_TSAP_ID	Array[1..16] of Byte	12.0																																																																														
14	LOCAL_TSAP_ID[1]	Byte		B#16#07																																																																													
15	LOCAL_TSAP_ID[2]	Byte		B#16#D0																																																																													

La instrucción TUSEND envía datos a través de UDP al interlocutor remoto especificado en el tipo de datos "TADDR_Param". La instrucción TURCV recibe datos a través de UDP. Tras ejecutar correctamente la instrucción TURCV, el tipo de datos "TADDR_Param" muestra la dirección del interlocutor remoto (emisor).

Tabla 10- 20 "Valores de parámetros para el tipo de datos "TADDR_Param"

Instrucción TUSEND	TUSEND "UDP ADDR DB"																																																							
	<table border="1"> <thead> <tr> <th colspan="5">Send_UDP_ADDR</th> </tr> <tr> <th>Nome</th> <th>Tipo di dati</th> <th>Offset</th> <th>Valore di avvio</th> <th>Commento</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Static</td> <td></td> <td></td> <td></td> </tr> <tr> <td>2</td> <td>Addr_Data</td> <td>TADDR_Param</td> <td>0.0</td> <td></td> </tr> <tr> <td>3</td> <td>REM_IP_ADDR</td> <td>Array[1..4] of UInt</td> <td>0.0</td> <td>remote station address</td> </tr> <tr> <td>4</td> <td>REM_IP_ADDR[1]</td> <td>UInt</td> <td>0</td> <td></td> </tr> <tr> <td>5</td> <td>REM_IP_ADDR[2]</td> <td>UInt</td> <td>0</td> <td></td> </tr> <tr> <td>6</td> <td>REM_IP_ADDR[3]</td> <td>UInt</td> <td>0</td> <td></td> </tr> <tr> <td>7</td> <td>REM_IP_ADDR[4]</td> <td>UInt</td> <td>0</td> <td></td> </tr> <tr> <td>8</td> <td>REM_PORT_NR</td> <td>UInt</td> <td>4.0</td> <td>remote port number</td> </tr> <tr> <td>9</td> <td>RESERVED</td> <td>Word</td> <td>6.0</td> <td>unused; has to be 0</td> </tr> </tbody> </table>	Send_UDP_ADDR					Nome	Tipo di dati	Offset	Valore di avvio	Commento	1	Static				2	Addr_Data	TADDR_Param	0.0		3	REM_IP_ADDR	Array[1..4] of UInt	0.0	remote station address	4	REM_IP_ADDR[1]	UInt	0		5	REM_IP_ADDR[2]	UInt	0		6	REM_IP_ADDR[3]	UInt	0		7	REM_IP_ADDR[4]	UInt	0		8	REM_PORT_NR	UInt	4.0	remote port number	9	RESERVED	Word	6.0	unused; has to be 0
Send_UDP_ADDR																																																								
Nome	Tipo di dati	Offset	Valore di avvio	Commento																																																				
1	Static																																																							
2	Addr_Data	TADDR_Param	0.0																																																					
3	REM_IP_ADDR	Array[1..4] of UInt	0.0	remote station address																																																				
4	REM_IP_ADDR[1]	UInt	0																																																					
5	REM_IP_ADDR[2]	UInt	0																																																					
6	REM_IP_ADDR[3]	UInt	0																																																					
7	REM_IP_ADDR[4]	UInt	0																																																					
8	REM_PORT_NR	UInt	4.0	remote port number																																																				
9	RESERVED	Word	6.0	unused; has to be 0																																																				

10.2.2.6 T_CONFIG

La instrucción T_CONFIG cambia los parámetros de configuración IP del puerto PROFINET del programa de usuario, permitiendo una modificación o ajuste permanente de las siguientes funciones:

- Nombre de la estación
- Dirección IP
- Máscara de subred
- Dirección del router

Nota

Situado en la página "Dirección Ethernet" de las "Propiedades" de la CPU, el botón de opción "Asignar dirección IP por otra vía" (Página 475) permite cambiar la dirección IP online o mediante la instrucción "T_CONFIG" después de descargar el programa. Este método de asignación de la dirección IP sólo sirve para la CPU.

Situado en la página "Dirección Ethernet" de las "Propiedades" de la CPU, el botón de opción "Asignar nombre del dispositivo por otra vía" (Página 476) permite cambiar el nombre del dispositivo PROFINET online o mediante la instrucción "T_CONFIG" después de descargar el programa. Este método de asignación del nombre de dispositivo PROFINET solo sirve para la CPU.

ADVERTENCIA
<p>Después de utilizar T_CONFIG para cambiar un parámetro de configuración IP, la CPU reanuncia. La CPU pasará a modo STOP, luego arranque en caliente, y volverá al modo RUN.</p> <p>Los dispositivos de control pueden fallar y provocar condiciones no seguras, causando a su vez reacciones inesperadas de los equipos controlados. Las reacciones inesperadas podrían producir la muerte, lesiones corporales graves y/o daños materiales.</p> <p>Asegúrese de que el proceso pasa a un estado seguro cuando la CPU reanuncia como resultado de ejecutar la instrucción T_CONFIG.</p>

Tabla 10- 21 Instrucción T_CONFIG

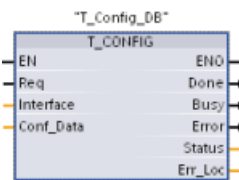
KOP / FUP	SCL	Descripción
	<pre>"T_CONFIG_DB" (req:=_bool_in_, interface:=_word_in_, conf_Data:=_variant_in_, done=>_bool_out_, busy=>_bool_out_, error=>_bool_out_, status=>_dword_out_, err_loc=>_word_out_);</pre>	<p>La instrucción T_CONFIG se puede usar para modificar los parámetros de configuración de IP desde el programa de usuario.</p> <p>T_CONFIG funciona de forma asíncrona. La ejecución abarca varias llamadas.</p>

Tabla 10- 22 Tipos de datos para los parámetros

Parámetro y tipo	Tipo de datos	Descripción	
REQ	Input	Bool	Inicia la instrucción con un flanco ascendente.
INTERFACE	Input	HW_Interface	ID de la interfaz de red
CONF_DATA	Input	Variant	Referencia a la estructura de los datos de configuración; CONF_DATA está definido por un tipo de datos de sistema (SDT).
DONE	Output	Bool	<ul style="list-style-type: none"> 0: Tarea no iniciada aún o en proceso. 1: Tarea finalizada sin errores.
BUSY	Output	Bool	<ul style="list-style-type: none"> 0: Tarea finalizada. 1: La tarea no ha finalizado aún. No se puede iniciar una tarea nueva.
ERROR	Output	Bool	<p>Parámetros de estado con los valores siguientes:</p> <ul style="list-style-type: none"> 0: No hay error 1: Ha ocurrido un error durante el procesamiento. <p>STATUS proporciona información detallada sobre el tipo de error.</p>
STATUS	Output	DWord	Información de estado, incluida la información de error. (Consulte los códigos de condición de error y estado en la tabla siguiente.)
ERR_LOC	Output	DWord	Localización del fallo (ID de campo e ID de subcampo del parámetro de error)

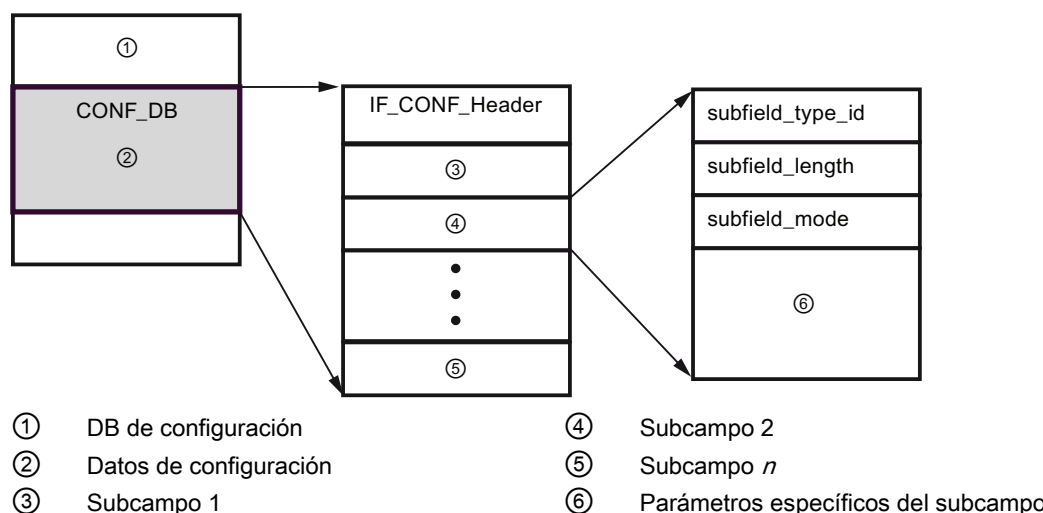
La información de la configuración de IP se deposita en el bloque de datos CONF_DATA, junto con un puntero de Variant en el parámetro CONF_DATA, del que se hace referencia más arriba. La ejecución correcta de la instrucción T_CONFIG finaliza con la entrega de los datos de configuración IP a la interfaz de red. Los errores se asignan al parámetro de salida STATUS.

Tabla 10- 23 Códigos de condición de ERROR y STATUS

ERROR	STATUS	Descripción
0	00000000	No hay error
0	00700000	La tarea no ha finalizado (BUSY = 1).
0	00700100	Comienza la ejecución de la tarea
0	00700200	Llamada intermedia (REQ irrelevante)
1	C08xyy00	Fallo general
1	C0808000	Los parámetros LADDR para identificar la interfaz no son válidos.
1	C0808100	A los parámetros LADDR para identificar la interfaz se les ha asignado una interfaz de hardware que no se soporta.
1	C0808200	Error de parámetro CONF_DATA: El tipo de datos del puntero de Variant no concuerda con el tipo de datos Byte.
1	C0808300	Error de parámetro CONF_DATA: El puntero de área no está en el DB del puntero de Variant.
1	C0808400	Error de parámetro CONF_DATA: El puntero de Variant tiene una longitud incorrecta.
1	C0808600	Reservado
1	C0808700	Incoherencia en la longitud del bloque de datos CONF_DATA en relación con la configuración de IP
1	C0808800	Los parámetros de la field_type_id del bloque de datos CONF_DATA no son válidos. (Sólo está permitida la field_type_id = 0.)
1	C0808900	Los parámetros de la field_type_id del bloque de datos CONF_DATA no son válidos o han sido utilizados varias veces.
1	C0808A00	Longitud LEN de los parámetros de la configuración de IP o errores subfield_cnt
1	C0808B00	El parámetro ID de la configuración de IP no es válido o no se soporta.
1	C0808C00	El subbloque de la configuración de IP está posicionado incorrectamente (subbloque incorrecto, orden incorrecto o utilizado de manera múltiple).
1	C0808D00	La longitud de una instrucción LEN de un subbloque no es válida.
1	C0808E00	El valor del parámetro en modo de subbloques no es válido.
1	C0808F00	Conflicto de subbloques entre la configuración de IP y un subbloque previo.
1	C0809000	Los parámetros del subcampo están protegidos contra escritura (por ejemplo: los parámetros están especificados por configuración o el modo PNIO está deshabilitado).
1	C0809100	Reservado
1	C0809400	Un parámetro en la configuración de IP de subbloques no ha sido definido o no se puede utilizar.
1	C0809500	Hay incoherencia entre un parámetro de la configuración de IP de subbloques y otros parámetros.
1	C080C200	No se puede ejecutar la instrucción. Este error puede ocurrir si, por ejemplo, se ha perdido la comunicación con la interfaz.
1	C080C300	No hay suficientes recursos. Este error puede ocurrir si, por ejemplo, se llama la instrucción de manera múltiple con diferentes parámetros
1	C080C400	Fallo de comunicación. Este error puede producirse temporalmente y será necesario repetir el programa de usuario.
1	C080D200	La interfaz PROFINET no soporta la ejecución de la instrucción.

Bloque de datos CONF_DATA

El diagrama siguiente muestra cómo se guardan en el DB de configuración los datos de configuración que deben transferirse.



Los datos de configuración del CONF_DB se componen de un campo que contiene un encabezado (IF_CONF_Header) y varios subcampos. IF_CONF_Header ofrece los siguientes elementos:

- field_type_id (tipo de datos UInt): Cero
- field_id (tipo de datos UInt): Cero
- subfield_cnt (tipo de datos UInt): Número de subcampos

Cada subcampo, a su vez, consta de un encabezado (subfield_type_id, subfield_length, subfield_mode) y de los parámetros específicos del subcampo. Cada subcampo debe estar conformado por un número de bytes par. subfield_mode admite el valor 1.

Nota

Actualmente solo se permite un campo (IF_CONF_Header). Sus parámetros field_type_id y field_id deben tener el valor cero. Otros campos con diferentes valores para field_type_id y field_id están sujetos a futuras extensiones.

En el campo IF_CONF_Header, actualmente sólo se permiten dos subcampos, "addr" (dirección IP) y "nos" (Name of station).

Tabla 10- 24 Subcampos soportados

subfield_type_id	Tipo de datos	Significado
30	IF_CONF_V4	Parámetros de IP: Dirección IP, máscara de subred, dirección del router
40	IF_CONF_NOS	Nombre de dispositivo PROFINET IO (Name of station)

Tabla 10- 25 Elementos del tipo de datos IF_CONF_V4

Nombre	Tipo de datos	Valor de arranque	Descripción		
Id	UInt	30	subfield_type_id		
len	UInt	18	subfield_length		
mode	UInt	1	subfield_mode (1: permanente)		
InterfaceAddress	IP_V4	-	Dirección de interfaz		
ADDR	Array [1..4] of Byte				
ADDR[1]	Byte			b#16#C8	Dirección IP high byte: 200
ADDR[2]	Byte			b#16#0C	Dirección IP high byte: 12
ADDR[3]	Byte			b#16#01	Dirección IP low byte: 1
ADDR[4]	Byte	b#16#90	Dirección IP low byte: 144		
SubnetMask	IP_V4	-	Máscara de subred		
ADDR	Array [1..4] of Byte				
ADDR[1]	Byte			b#16#FF	Máscara de subred high byte: 255
ADDR[2]	Byte			b#16#FF	Máscara de subred high byte: 255
ADDR[3]	Byte			b#16#FF	Máscara de subred low byte: 255
ADDR[4]	Byte	b#16#00	Máscara de subred low byte: 0		
DefaultRouter	IP_V4	-	Router predeterminado		
ADDR	Array [1..4] of Byte				
ADDR[1]	Byte			b#16#C8	Router high byte: 200
ADDR[2]	Byte			b#16#0C	Router high byte: 12
ADDR[3]	Byte			b#16#01	Router low byte: 1
ADDR[4]	Byte	b#16#01	Router low byte: 1		

Tabla 10- 26 Elementos del tipo de datos IF_CONF_NOS

Nombre	Tipo de datos	Valor de arranque	Descripción
id	UInt	40	subfield_type_id
len	UInt	246	subfield_length
mode	UInt	1	subfield_mode (1: permanente)
Nos (Name of station)	Array[1..240] of Byte	0	Nombre de la estación: Hay que ocupar el ARRAY desde el primer byte. Si el ARRAY es más largo que el nombre que se le va a asignar a la estación, es necesario introducir un byte cero detrás del nombre real de la estación (de conformidad con IEC 61158-6-10). De lo contrario, nos será rechazado y la instrucción "T_CONFIG (Página 470)" introduce el código de error DW#16#C0809400 en STATUS. Si ocupa el primer byte con cero se borra el nombre de la estación.

El nombre de la estación está sujeto a las siguientes limitaciones:

- Un componente dentro del nombre de la estación, p. ej. una cadena de caracteres entre dos puntos, no debe exceder 63 caracteres.
- Sin caracteres especiales como diéresis, corchetes, guión bajo, barra oblicua, espacio en blanco, etc. El único carácter especial permitido es el guión.

- El nombre de la estación no debe comenzar ni terminar con el carácter especial "-".
- El nombre de la estación no debe comenzar con un número.
- Para el nombre de la estación no está permitido el formato n.n.n.n (n = 0, ... 999).
- El nombre de la estación no debe comenzar con la cadena "port-xyz" o "port-xyz-abcde" (a, b, c, d, e, x, y, z = 0, ... 9).

Nota

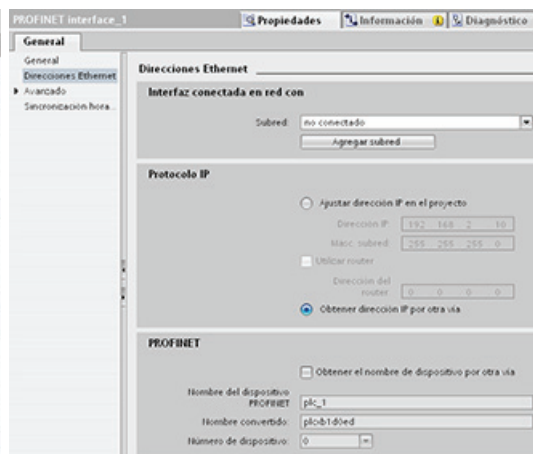
También es posible crear un ARRAY "nos" que tenga menos de 240 bytes, pero no menos de 2 bytes. En este caso, es necesario ajustar de manera correspondiente la variable "len" (longitud del subcampo).

Cómo cambiar los parámetros IP

En el ejemplo siguiente, en el subcampo "addr", se han cambiado la "InterfaceAddress" (dirección IP), la "SubnetMask", y el "DefaultRouter" (router IP). En la página "Dirección Ethernet" de las "Propiedades" de la CPU, hay que pulsar el botón de opción "Asignar dirección IP por otra vía" para cambiar la dirección IP mediante la instrucción "T_CONFIG" después de descargar el programa.

Tabla 10- 27 Cómo cambiar los parámetros IP

CONF_DATA_1			
	Nombre	Tipo de datos	Valor de arranq...
1	Static		
2	Conf_data	Struct	
3	header	IF_CONF_Header	
4	FieldType	UInt	0
5	FieldId	UInt	0
6	SubfieldCount	UInt	1
7	addr	IF_CONF_v4	
8	Id	UInt	30
9	Length	UInt	18
10	Mode	UInt	1
11	InterfaceAddress	IP_V4	
12	ADDR	array [1..4] of Byte	
13	ADDR[1]	Byte	192
14	ADDR[2]	Byte	168
15	ADDR[3]	Byte	2
16	ADDR[4]	Byte	30
17	SubnetMask	IP_V4	
18	ADDR	array [1..4] of Byte	
19	ADDR[1]	Byte	255
20	ADDR[2]	Byte	255
21	ADDR[3]	Byte	255
22	ADDR[4]	Byte	0
23	DefaultRouter	IP_V4	
24	ADDR	array [1..4] of Byte	
25	ADDR[1]	Byte	192
26	ADDR[2]	Byte	168
27	ADDR[3]	Byte	2
28	ADDR[4]	Byte	1

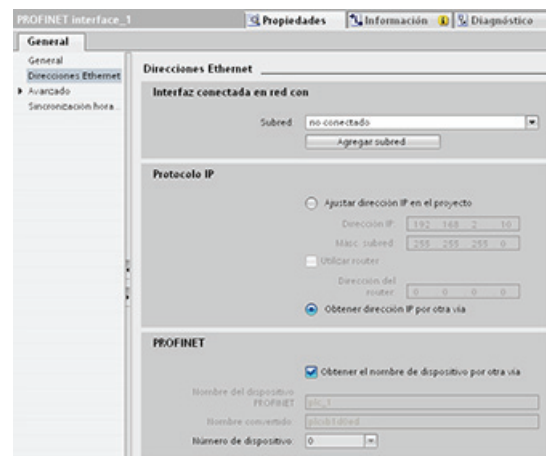


Cómo cambiar los parámetros IP y los nombres de dispositivo PROFINET IO

En el ejemplo siguiente, se han cambiado los subcampos "addr" y "nos" (Name of station). En la página "Dirección Ethernet" de las "Propiedades" de la CPU, hay que pulsar el botón de opción "Asignar nombre del dispositivo por otra vía" para cambiar el nombre del dispositivo PROFINET mediante la instrucción "T_CONFIG" después de descargar el programa.

Tabla 10- 28 Cómo cambiar los parámetros IP y los nombres de dispositivo PROFINET IO

CONF_DATA_2			
	Nombre	Tipo de datos	Valor de arranq...
1	Static		
2	Conf_data	Struct	
3	header	IF_CONIF_Header	
4	FieldType	UInt	0
5	Fieldid	UInt	0
6	SubfieldCount	UInt	2
7	addr	IF_CONIF_V4	
8	Id	UInt	30
9	Length	UInt	18
10	Mode	UInt	1
11	InterfaceAddress	IP_V4	
12	ADDR	array [1..4] of Byte	
13	SubnetMask	IP_V4	
14	ADDR	array [1..4] of Byte	
15	DefaultRouter	IP_V4	
16	ADDR	array [1..4] of Byte	
17	nos	IF_CONIF_IOS	
18	Id	UInt	40
19	Length	UInt	246
20	Mode	UInt	1
21	IOS	array [1..240] of Byte	



10.2.2.7 Parámetros comunes para instrucciones

Parámetro de entrada REQ

Muchas de las instrucciones de la comunicación abierta utilizan la entrada REQ para iniciar la operación en una transición de "low" a "high". El estado lógico de la entrada REQ debe ser "high" (TRUE) durante una ejecución de la instrucción, aunque puede permanecer TRUE durante un tiempo cualquiera. La instrucción no inicia ninguna operación diferente hasta que no sea ejecutada con la entrada REQ puesta a FALSE, de manera que pueda inicializar el histórico de la entrada REQ. Esto es necesario para que la instrucción pueda detectar la transición de "low" a "high" para iniciar la siguiente operación.

Al insertar una de estas instrucciones en su programa, STEP 7 le solicitará identificar el DB de instancia. Utilice un DB unívoco para cada llamada de la instrucción. Esto garantiza que cada instrucción procese correctamente las entradas, p. ej. REQ.

Parámetro de entrada ID

Esta es una referencia de la "ID Local (hex) en la "Vista de red" de "Dispositivos y redes" de STEP 7 y es la ID de la red que usted desea utilizar para este bloque de comunicación. La ID debe ser idéntica al parámetro ID asociado en la descripción de la conexión local.

Parámetros de salida DONE, NDR, ERROR y STATUS

Estas instrucciones ponen a disposición salidas que describen el estado de finalizado:

Tabla 10- 29 Parámetros de salida de las instrucciones de la comunicación abierta

Parámetro	Tipo de datos	Valor predeterminado	Descripción
DONE	Bool	FALSE	Si adopta el estado lógico TRUE durante una ejecución indica que la última petición se ha finalizado sin errores; de lo contrario, FALSE.
NDR	Bool	FALSE	Si adopta el estado lógico TRUE durante una ejecución indica que la acción solicitada se ha finalizado sin errores y que se han recibido datos nuevos; de lo contrario, FALSE.
BUSY	Bool	FALSE	Si adopta el estado lógico TRUE mientras está activo indica que: <ul style="list-style-type: none"> • La tarea no ha finalizado aún. • No se puede iniciar una tarea nueva. Adopta el estado lógico FALSE cuando la tarea ha finalizado.
ERROR	Bool	FALSE	Si adopta el estado lógico TRUE durante una ejecución indica que la última petición se ha finalizado con errores. El código de error aplicable aparece en STATUS; de lo contrario, FALSE.
STATUS	Word	0	Resultado del estado: <ul style="list-style-type: none"> • Si se activa el bit DONE o NDR, STATUS se pone a 0 o a un código de información. • Si se activa el bit ERROR, STATUS se pone a un código de error. • Si no se activa ninguno de estos bits, la instrucción devuelve resultados de estado que describen el estado actual de la función. STATUS conserva su valor durante la ejecución de la función.

Nota

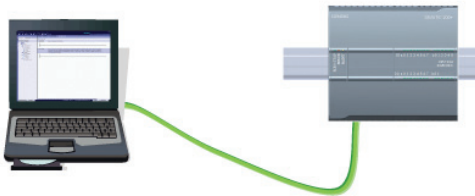
Observe que DONE, NDR y ERROR sólo están activados durante una ejecución.

TSAPs y números de puerto restringidos para la comunicación ISO y TCP pasiva

Si la instrucción "TCON" se utiliza para configurar y establecer una conexión pasiva, las siguientes direcciones de puerto están restringidas y no se deben utilizar:

- TSAP ISO (pasivo):
 - 01.00, 01.01, 02.00, 02.01, 03.00, 03.01
 - 10.00, 10.01, 11.00, 11.01, ... BF.00, BF.01
- Puerto TCP (pasivo): 5001, 102, 123, 20, 21, 25, 34962, 34963, 34964, 80
- Puerto UDP (pasivo): 161, 34962, 34963, 34964

10.2.3 Comunicación con una programadora



Una CPU puede comunicarse con una programadora con STEP 7 en una red.

Al configurar la comunicación entre una CPU y una programadora debe considerarse lo siguiente:

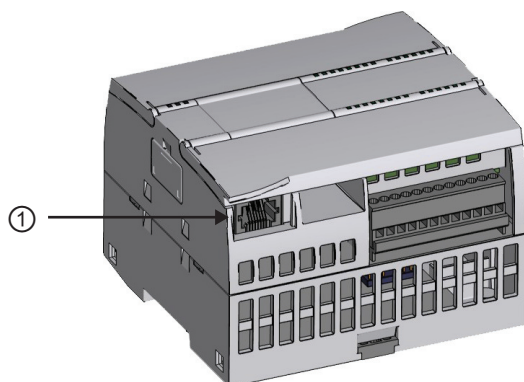
- Configuración/instalación: Es preciso configurar el hardware.
- Para la comunicación entre dos interlocutores no se requiere un switch Ethernet. Un switch Ethernet se requiere sólo si la red comprende más de dos dispositivos.

10.2.3.1 Establecer la conexión de hardware

Las interfaces PROFINET establecen las conexiones físicas entre una programadora y una CPU. Puesto que la CPU ofrece la función "auto-crossover", es posible utilizar un cable Ethernet estándar o cruzado ("crossover") para la interfaz. Para conectar una programadora directamente a una CPU no se requiere un switch Ethernet.

Para crear la conexión de hardware entre una programadora y una CPU, proceda del siguiente modo:

1. Monte la CPU (Página 47).
2. Conecte el cable Ethernet al puerto PROFINET que se muestra abajo.
3. Conecte el cable Ethernet a la programadora.



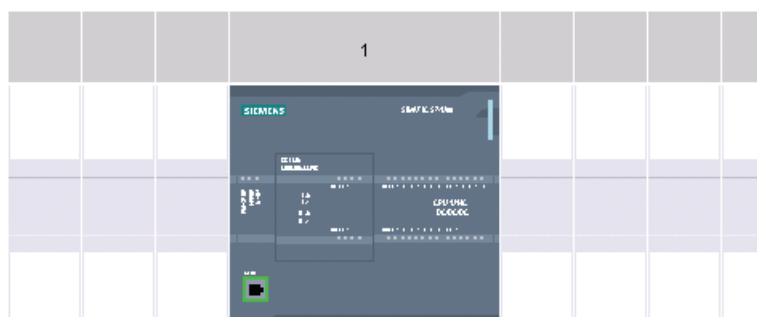
① Puerto PROFINET

Hay una descarga de tracción opcional disponible para reforzar la conexión PROFINET.

10.2.3.2 Configurar los dispositivos

Si ya se ha creado un proyecto con una CPU, ábralo en STEP 7.

En caso contrario, cree un proyecto e inserte una CPU (Página 124) en el rack. En el proyecto que aparece abajo, una CPU se muestra en la "Vista de dispositivos".



10.2.3.3 Asignar direcciones IP (Internet Protocol)

Asignar direcciones IP

En una red PROFINET todo dispositivo debe tener también una dirección IP (Internet Protocol o Protocolo Internet). Esta dirección permite al dispositivo transferir datos a través de una red enrutada y más compleja:

- Si usted dispone de programadoras u otros dispositivos de red que utilicen una tarjeta adaptadora integrada conectada a la LAN corporativa o una tarjeta adaptadora Ethernet-USB conectada a una red aislada, es necesario asignarles direcciones IP. Para más información, consulte el apartado "Asignar direcciones IP a los dispositivos de programación y red" (Página 137).
- También es posible asignar una dirección IP a una CPU o a un dispositivo de red online. Esto es especialmente útil al configurar los dispositivos por primera vez. Para más información, consulte el apartado "Asignar una dirección IP a una CPU online" (Página 137).
- Tras configurar la CPU o el dispositivo de red en su proyecto, usted puede configurar parámetros para la interfaz PROFINET, para incluir la dirección IP correspondiente. Encontrará más información al respecto en "Configurar una dirección IP para una CPU en el proyecto". (Página 139)

10.2.3.4 Comprobar la red PROFINET

Tras finalizar la configuración es necesario cargar el proyecto en la CPU. Todas las direcciones IP se configuran al cargar el proyecto en el dispositivo.

La función de la CPU "Cargar en dispositivo" y su cuadro de diálogo "Carga avanzada en dispositivo" permiten visualizar todos los dispositivos de red accesibles y verificar si se han asignado direcciones IP unívocas a todos ellos. Para más información, consulte "Comprobar la red PROFINET" (Página 144)

10.2.4 Comunicación entre dispositivos HMI y el PLC



La CPU soporta conexiones PROFINET con dispositivos HMI. Los siguientes requisitos deben considerarse al configurar la comunicación entre CPUs y HMIs:

Configuración/instalación:

- El puerto PROFINET de la CPU debe configurarse para poder establecer una conexión con el HMI.
- El HMI se debe instalar y configurar.

- La información de configuración del HMI forma parte del proyecto de la CPU y se puede configurar y cargar desde el proyecto.
- Para la comunicación entre dos interlocutores no se requiere un switch Ethernet. Un switch Ethernet se requiere sólo si la red comprende más de dos dispositivos.

Nota

El switch Ethernet de 4 puertos CSM1277 montado en un rack puede utilizarse para conectar las CPUs y los dispositivos HMI. El puerto PROFINET de la CPU no contiene un dispositivo de conmutación Ethernet.

Funciones soportadas:

- El HMI puede leer/escribir datos en la CPU.
- Es posible disparar mensajes, según la información consultada de la CPU.
- Diagnóstico del sistema

Tabla 10- 30 Pasos necesarios para configurar la comunicación entre un dispositivo HMI y una CPU

Paso	Tarea
1	<p>Establecer la conexión de hardware</p> <p>Una interfaz PROFINET establece la conexión física entre un dispositivo HMI y una CPU. Puesto que la función "auto-crossover" está integrada en la CPU, es posible utilizar un cable Ethernet estándar o cruzado ("crossover") para la interfaz. Para conectar un HMI a una CPU no se requiere un switch Ethernet.</p> <p>Para más información, consulte "Comunicación con una programadora: Establecer la conexión de hardware". (Página 479)</p>
2	<p>Configurar los dispositivos</p> <p>Para más información, consulte "Comunicación con una programadora: Configurar los dispositivos". (Página 479)</p>
3	<p>Configurar las conexiones de red lógicas entre un dispositivo HMI y una CPU</p> <p>Encontrará más información en "Comunicación entre HMI y PLC: Configurar las conexiones de red lógicas entre dos dispositivos (Página 482)".</p>
4	<p>Configurar una dirección IP en el proyecto</p> <p>Utilice el mismo proceso de configuración. No obstante, es preciso configurar direcciones IP para el HMI y la CPU.</p> <p>Encontrará más información en "Configuración de dispositivos: Configurar una dirección IP para una CPU en el proyecto". (Página 141)</p>
5	<p>Comprobar la red PROFINET</p> <p>La configuración debe cargarse en cada CPU y cada dispositivo HMI.</p> <p>Encontrará más información en "Configuración de dispositivos: Comprobar la red PROFINET". (Página 144)</p>

10.2.4.1 Configurar las conexiones de red lógicas entre dos dispositivos

Tras configurar el rack con la CPU podrá configurar las conexiones de red.

En el portal "Dispositivos y redes", utilice la "Vista de red" para crear las conexiones de red entre los dispositivos del proyecto. En primer lugar, haga clic en la ficha "Conexiones" y luego seleccione el tipo de conexión en la lista desplegable ubicada justo a la derecha (p. ej. una conexión ISO on TCP).

Para crear una conexión PROFINET, haga clic en el cuadro verde (PROFINET) del primer dispositivo y trace una línea hasta el cuadro PROFINET del segundo dispositivo. Suelte el botón del ratón para crear la conexión PROFINET.

Para más información, consulte el apartado "Configuración de dispositivos: Crear una conexión de red". (Página 131)

10.2.5 Comunicación entre PLCs



Una CPU puede comunicarse con otra CPU utilizando las instrucciones TSEND_C y TRCV_C.

Considere lo siguiente al configurar la comunicación entre dos CPUs:

- Configuración/instalación: Es preciso configurar el hardware.
- Funciones soportadas: Leer/escribir datos en una CPU interlocutora
- Para la comunicación entre dos interlocutores no se requiere un switch Ethernet. Un switch Ethernet se requiere sólo si la red comprende más de dos dispositivos.

Tabla 10- 31 Pasos necesarios para configurar la comunicación entre dos CPUs

Paso	Tarea
1	<p>Establecer la conexión de hardware</p> <p>Una interfaz PROFINET establece la conexión física entre dos CPUs. Puesto que la función "auto-crossover" está integrada en la CPU, es posible utilizar un cable Ethernet estándar o cruzado ("crossover") para la interfaz. Para conectar dos CPUs no se requiere un switch Ethernet.</p> <p>Para más información, consulte "Comunicación con una programadora: Establecer la conexión de hardware". (Página 479)</p>
2	<p>Configurar los dispositivos</p> <p>Hay que configurar dos CPUs en el proyecto.</p> <p>Para más información, consulte "Comunicación con una programadora: Configurar los dispositivos". (Página 479)</p>
3	<p>Configurar las conexiones de red lógicas entre dos CPUs</p> <p>Encontrará más información en "Comunicación entre PLC y PLC: Configurar las conexiones de red lógicas entre dos dispositivos (Página 483)".</p>

Paso	Tarea
4	<p>Configurar una dirección IP en el proyecto</p> <p>Utilice el mismo proceso de configuración. No obstante, es preciso configurar direcciones IP para dos CPUs (p. ej. PLC_1 y PLC_2).</p> <p>Encontrará más información en "Configuración de dispositivos: Configurar una dirección IP para una CPU en el proyecto". (Página 141)</p>
5	<p>Configurar los parámetros de transmisión y recepción</p> <p>Las instrucciones TSEND_C y TRCV_C deben configurarse en ambas CPUs para habilitar la comunicación entre ellas.</p> <p>Encontrará más información en el apartado "Configurar la comunicación entre dos CPUs: Configurar los parámetros de transmisión y recepción". (Página 484)</p>
6	<p>Comprobar la red PROFINET</p> <p>La configuración debe cargarse en cada una de las CPUs.</p> <p>Encontrará más información en "Configuración de dispositivos: Comprobar la red PROFINET (Página 144)".</p>

10.2.5.1 Configurar las conexiones de red lógicas entre dos dispositivos

Tras configurar el rack con la CPU podrá configurar las conexiones de red.

En el portal "Dispositivos y redes", utilice la "Vista de red" para crear las conexiones de red entre los dispositivos del proyecto. En primer lugar, haga clic en la ficha "Conexiones" y luego seleccione el tipo de conexión en la lista desplegable ubicada justo a la derecha (p. ej. una conexión ISO on TCP).

Para crear una conexión PROFINET, haga clic en el cuadro verde (PROFINET) del primer dispositivo y trace una línea hasta el cuadro PROFINET del segundo dispositivo. Suelte el botón del ratón para crear la conexión PROFINET.

Para más información, consulte el apartado "Configuración de dispositivos: Crear una conexión de red". (Página 131)

10.2.5.2 Configurar la vía de conexión local/interlocutor entre dos dispositivos

Configurar los parámetros generales

Los parámetros de comunicación se especifican en el cuadro de diálogo de configuración "Propiedades" de la instrucción de comunicación. Este diálogo aparece en el lado inferior de la página cuando se ha seleccionado alguna parte de la instrucción.

Encontrará más información en "Configuración de dispositivos: Configurar la vía de conexión local/interlocutor (Página 132)"

En el área "Detalles de dirección" del diálogo "Parámetros de la conexión" se definen los TSAPs o puertos que deben utilizarse. El TSAP o puerto de una conexión en la CPU se introduce en el campo "TSAP local". El TSAP o puerto asignado a la conexión en la CPU interlocutora se introduce en el campo "TSAP del interlocutor".

10.2.5.3 Configurar los parámetros de transmisión y recepción

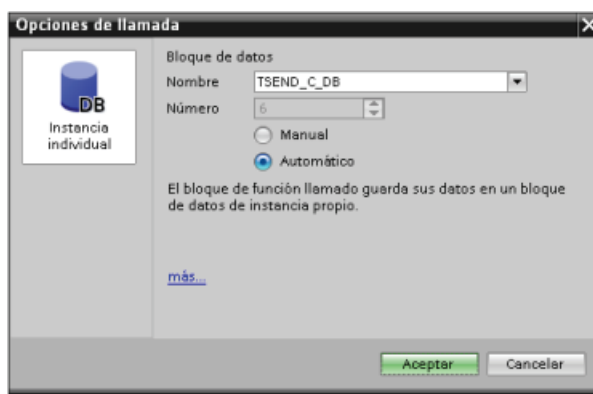
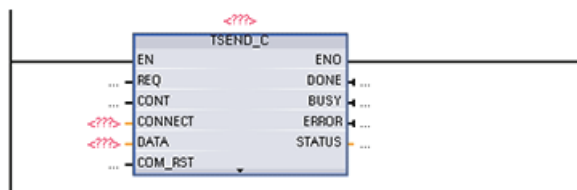
Los bloques de comunicación (por ejemplo TSEND_C y TRCV_C) sirven para establecer conexiones entre dos CPUs. Para que las CPUs puedan intervenir en la comunicación PROFINET es preciso configurar parámetros para transmitir y recibir mensajes. Estos parámetros determinan cómo deben funcionar las comunicaciones al transmitir o recibir mensajes a/de un dispositivo de destino.

Configurar los parámetros de transmisión de la instrucción TSEND_C

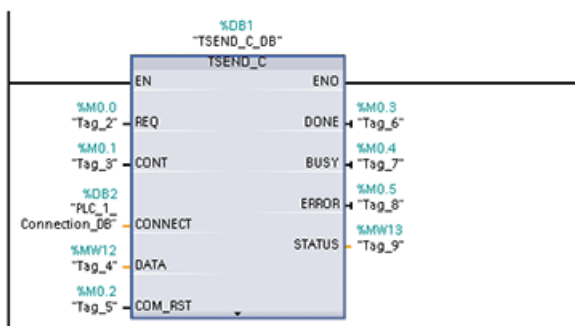
Instrucción TSEND_C

La instrucción TSEND_C (Página 451) crea una conexión con un interlocutor. La conexión se configura, establece y vigila automáticamente hasta que la instrucción ordene que sea desconectada. La instrucción TSEND_C combina las funciones de las instrucciones TCON, TDISCON y TSEND.

En la "Configuración de dispositivos" de STEP 7 es posible configurar cómo la instrucción TSEND_C debe transmitir los datos. Para comenzar, inserte la instrucción en el programa desde la carpeta "Comunicación" de la Task Card "Instrucciones". La instrucción TSEND_C se visualizará junto con el diálogo "Opciones de llamada" en el que se asigna un DB para almacenar los parámetros de la instrucción.



Como muestra la figura siguiente, es posible asignar posiciones de memoria a las entradas y salidas en la memoria de variables:



Configurar los parámetros generales

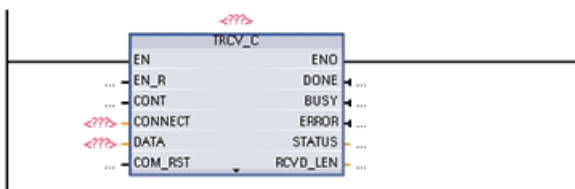
Los parámetros se configuran en el diálogo "Propiedades" de la instrucción TSEND_C. Este diálogo aparece en el lado inferior de la página cuando se ha seleccionado alguna parte de la instrucción TSEND_C.

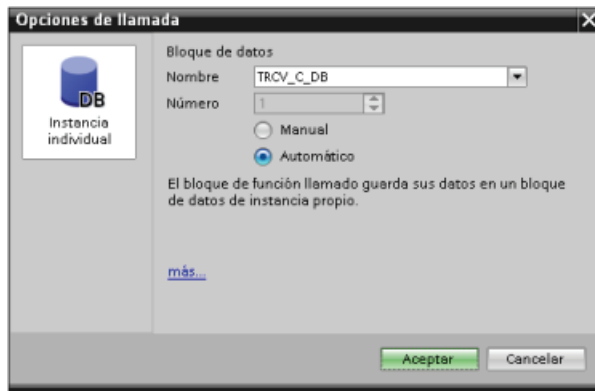
Configurar los parámetros de recepción de la instrucción TRCV_C

Instrucción TRCV_C

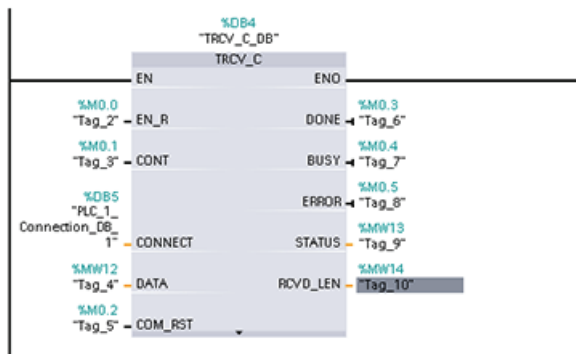
La instrucción TRCV_C (Página 451) crea una conexión con un interlocutor. La conexión se configura, establece y vigila automáticamente hasta que la instrucción ordene que sea desconectada. La instrucción TRCV_C combina las funciones de las instrucciones TCON, TDISCON y TRCV.

Desde la configuración de la CPU en STEP 7 es posible configurar cómo la instrucción TRCV_C debe recibir los datos. Para comenzar, inserte la instrucción en el programa desde la carpeta "Comunicación" de la Task Card "Instrucciones". La instrucción TRCV_C se visualizará junto con el diálogo "Opciones de llamada" en el que se asigna un DB para almacenar los parámetros de la instrucción.





Como muestra la figura siguiente, es posible asignar posiciones de memoria a las entradas y salidas en la memoria de variables:



Configurar los parámetros generales

Los parámetros se configuran en el diálogo "Propiedades" de la instrucción TRCV_C. Este diálogo aparece en el lado inferior de la página cuando se ha seleccionado alguna parte de la instrucción TRCV_C.

10.2.6 Configurar una CPU y un dispositivo PROFINET IO

Agregar un dispositivo PROFINET IO


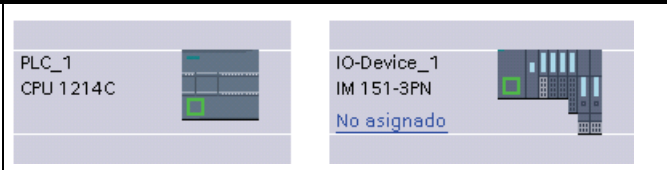
En el portal "Dispositivos y redes", utilice el catálogo de hardware para agregar dispositivos PROFINET IO.

Nota

Para agregar un dispositivo PROFINET IO se puede utilizar STEP 7 Professional o Basic V11 o superior.

Por ejemplo, expanda las carpetas siguientes del catálogo de hardware para agregar un dispositivo IO ET200S: E/S distribuidas, ET200S, módulos de interfaz y PROFINET. A continuación, seleccione el módulo de interfaz en la lista de dispositivos ET200S (clasificados por referencia) y agregue el dispositivo IO ET200S.

Tabla 10- 32 Agregar un dispositivo IO ET200S a la configuración de dispositivos

Insertar el dispositivo IO	Resultado
 <p>PLC_1 CPU 1214C</p> <p>6ES7 151-3AA23-0AB0</p>	 <p>PLC_1 CPU 1214C</p> <p>IO-Device_1 IM 151-3PN No asignado</p>

Ahora, el dispositivo PROFINET IO puede conectarse a la CPU:

1. Haga clic con el botón derecho del ratón en el enlace "No asignado" del dispositivo y seleccione "Asignar a controlador IO nuevo" del menú contextual para visualizar el cuadro de diálogo "Seleccionar controlador IO".
2. Seleccione la CPU S7-1200 (en el ejemplo "PLC_1") en la lista de controladores del proyecto.
3. Haga clic en "Aceptar" para crear la conexión de red.

Configurar conexiones de red lógicas

Tras configurar el rack con la CPU podrá configurar las conexiones de red.

En el portal "Dispositivos y redes", utilice la "Vista de redes" para crear las conexiones de red entre los dispositivos del proyecto. Para crear una conexión PROFINET, haga clic en el cuadro verde (PROFINET) del primer dispositivo y trace una línea hasta el cuadro PROFINET del segundo dispositivo. Suelte el botón del ratón para crear la conexión PROFINET.

Para más información, consulte el apartado "Configuración de dispositivos: Crear una conexión de red (Página 131)".

Asignar CPUs y nombres de dispositivo

Las conexiones de red entre los dispositivos también asignan el dispositivo PROFINET IO a la CPU, algo necesario para que la CPU pueda controlar dicho dispositivo. Para modificar la asignación, haga clic en el nombre del PLC que se muestra en el dispositivo PROFINET IO. Se abre un cuadro de diálogo que permite desconectar el dispositivo PROFINET IO de la CPU actual y reasignarlo o bien dejarlo sin asignación, como se prefiera.

Los dispositivos de la red PROFINET deben tener asignados nombres antes de poder ser conectados a la CPU. Utilice la "Vista de redes" para asignar nombres a los dispositivos PROFINET, en caso de que éstos no hayan sido asignados o bien si el nombre del dispositivo debe ser modificado. Haga clic con el botón derecho del ratón en el dispositivo PROFINET IO y seleccione "Asignar nombre de dispositivo" para hacerlo.

A cada dispositivo PROFINET IO se le debe asignar el mismo nombre en el proyecto de STEP 7 y, utilizando la herramienta "Online y diagnóstico", en la memoria de configuración del dispositivo PROFINET IO (p. ej. en la memoria de configuración de un módulo de interfaz ET200 S). Si hace falta un nombre o éste no coincide en ninguna ubicación, el modo de intercambio de datos PROFINET IO no se ejecutará. Encontrará más información en "Herramientas online y diagnóstico: Asignar un nombre a un dispositivo PROFINET online (Página 712)".

Asignar direcciones IP

En una red PROFINET todo dispositivo debe tener también una dirección IP (Internet Protocol o Protocolo Internet). Esta dirección permite al dispositivo transferir datos a través de una red enrutada y más compleja:

- Si usted dispone de programadoras u otros dispositivos de red que utilicen una tarjeta adaptadora integrada conectada a la LAN corporativa o una tarjeta adaptadora Ethernet-USB conectada a una red aislada, es necesario asignarles direcciones IP. Para más información, consulte el apartado "Asignar direcciones IP a los dispositivos de programación y red" (Página 137).
- También es posible asignar una dirección IP a una CPU o a un dispositivo de red online. Esto es especialmente útil al configurar los dispositivos por primera vez. Para más información, consulte el apartado "Asignar una dirección IP a una CPU online" (Página 139).
- Tras configurar la CPU o el dispositivo de red en su proyecto, usted puede configurar parámetros para la interfaz PROFINET, para incluir la dirección IP correspondiente. Encontrará más información al respecto en "Configurar una dirección IP para una CPU en el proyecto". (Página 141)

Configurar el tiempo de ciclo IO

Un dispositivo PROFINET IO recibe datos nuevos de la CPU dentro de un intervalo de tiempo "Ciclo IO". El tiempo de actualización puede configurarse por separado para cada dispositivo y determina el intervalo de tiempo en el que los datos se transmiten de la CPU al dispositivo y viceversa.

STEP 7 calcula automáticamente el tiempo de actualización del "Ciclo IO" en la configuración predeterminada de cada dispositivo de la red PROFINET, tomando en consideración el volumen de datos que deben intercambiarse y el número de dispositivos asignados a dicho controlador. Si no desea que el tiempo de actualización se calcule automáticamente, es posible cambiar este ajuste.

Los parámetros del "Ciclo IO" se especifican en el cuadro de diálogo de configuración "Propiedades" del dispositivo PROFINET IO. Este cuadro de diálogo aparece en la parte inferior de la página cuando se ha seleccionado alguna parte de la instrucción.

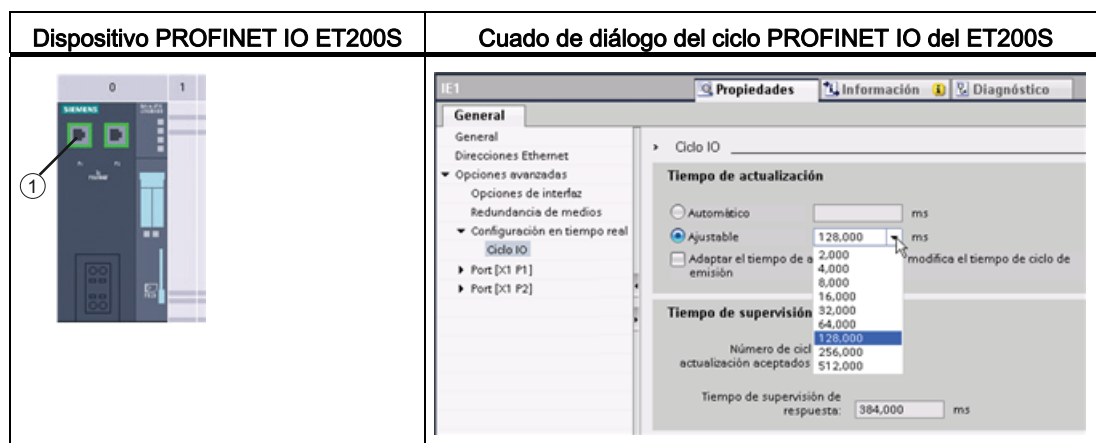
En la "Vista de dispositivos" del dispositivo PROFINET IO, haga clic en el puerto PROFINET. En el cuadro de diálogo "Interfaz PROFINET", acceda a los parámetros del "Ciclo IO" con las siguientes selecciones de menú:

- "Opciones avanzadas"
- "Configuración en tiempo real"
- "Ciclo IO"

Defina el "Tiempo de actualización" del ciclo IO con las selecciones siguientes:

- Para que se calcule automáticamente un tiempo de actualización adecuado, seleccione "Automático".
- Para ajustar uno mismo el tiempo de actualización, seleccione "Ajustable" e introduzca el tiempo de actualización necesario en ms.
- Para garantizar la coherencia entre el tiempo de ciclo de emisión y el tiempo de actualización, active la opción "Ajustar el tiempo de actualización si se modifica el tiempo de ciclo de emisión". Esta opción asegura que el tiempo de actualización no sea inferior al tiempo de ciclo de emisión.

Tabla 10- 33 Configurar el tiempo de ciclo PROFINET IO del ET200S



① Puerto PROFINET

10.2.7 Diagnóstico

Bloque de organización de alarma de diagnóstico (OB82)

Si un módulo con capacidad de diagnóstico tiene una alarma de diagnóstico habilitada y detecta un cambio en su estado, envía una petición de alarma de diagnóstico a la CPU en las situaciones siguientes:

- El módulo ha detectado un problema (p. ej. una rotura de hilo) o bien un componente necesita mantenimiento o bien ambos (evento entrante).
- El problema ha sido solucionado o ha desaparecido y ningún otro componente requiere mantenimiento (evento saliente).

Si el OB82 no existe, los errores se escriben en el búfer de diagnóstico. La CPU no realiza ninguna acción ni pasa a STOP.

Si el OB82 existe, el sistema operativo lo puede llamar en respuesta a un evento entrante. Es necesario crear el OB82. Este OB permite configurar el procesamiento de errores locales, así como una reacción más detallada ante eventos entrantes.

Si está utilizando una CPU apta para DPV1, es posible obtener información adicional sobre la alarma a través de la instrucción RALRM, la cual ofrece información más detallada que la información de inicio del OB82.

Alarmas de acceso a la periferia

Estos errores se escriben en el búfer de diagnóstico. La CPU no realiza ninguna acción ni pasa a STOP. Los errores que se escriben en el búfer de diagnóstico incluyen:

- Fallos del módulo
- Módulo inapropiado
- Falta el módulo

Errores de acceso IO

Estos errores se escriben en el búfer de diagnóstico. La CPU no realiza ninguna acción ni pasa a STOP.

10.2.8 Instrucciones de E/S descentralizadas

Consulte "E/S descentralizadas (PROFINET, PROFIBUS o AS-i)" (Página 285) para obtener información sobre cómo utilizar las instrucciones de E/S descentralizadas con estas redes de comunicación.

10.2.9 Instrucciones de diagnóstico

Consulte "Diagnóstico (PROFINET o PROFIBUS)": "Instrucciones de diagnóstico" (Página 311) para obtener información sobre cómo utilizar estas instrucciones con estas redes de comunicación.

10.2.10 Eventos de diagnóstico de E/S descentralizadas

Consulte "Diagnóstico (PROFINET o PROFIBUS)": "Eventos de diagnóstico de E/S descentralizadas" (Página 311) para obtener información sobre cómo utilizar esta información de diagnóstico con estas redes de comunicación.

10.3 PROFIBUS

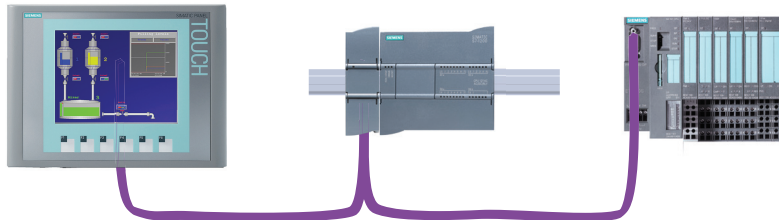
Un sistema PROFIBUS utiliza un maestro de bus para consultar dispositivos esclavos descentralizados según el sistema MULTIDROP en un bus serie RS485. Un esclavo PROFIBUS es cualquier dispositivo periférico (transductor E/S, válvula, accionamiento del motor u otro dispositivo de medición) que procese información y envíe su salida al maestro. El esclavo conforma una estación pasiva en la red debido a que no tiene derechos de acceso al bus y sólo puede acusar mensajes recibidos o bien enviar mensajes de respuesta al maestro sobre petición. Todos los esclavos PROFIBUS tienen la misma prioridad y toda la comunicación de red se inicia desde el maestro.

Un maestro PROFIBUS conforma una "estación activa" en la red. PROFIBUS DP define dos clases de maestro. Un maestro clase 1 (por lo general un controlador central programable (PLC) o un equipo dotado de un software especial) procesa la comunicación normal o intercambia datos con los esclavos que tiene asignados. Un maestro clase 2 (por lo general un dispositivo de configuración, p. ej. un portátil o una consola de programación utilizada para la puesta en marcha, mantenimiento o con fines de diagnóstico) es un dispositivo especial utilizado principalmente para poner en marcha esclavos y para fines de diagnóstico.

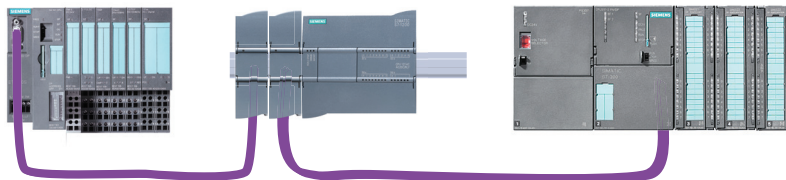
El S7-1200 se conecta a una red PROFIBUS como esclavo DP con el módulo de comunicación CM 1242-5. El módulo CM 1242-5 (esclavo DP) puede ser el interlocutor de maestros DP V0/V1. En la figura de abajo el S7-1200 es un esclavo DP de un controlador S7-300.



El S7-1200 se conecta a una red PROFIBUS como maestro DP con el módulo de comunicación CM 1243-5. El módulo CM 1243-5 (maestro DP) puede ser el interlocutor de esclavos DP V0/V1. En la figura de abajo el S7-1200 es un maestro que controla a un esclavo ET200S DP.



Si un CM 1242-5 y un CM 1243-5 están instalados conjuntamente, un S7-1200 puede ejecutar ambos simultáneamente como un esclavo de un sistema maestro DP de nivel superior y como un maestro de un sistema maestro DP subordinado, respectivamente.



En V3.0 se pueden configurar tres CM PROFIBUS como máximo por estación, en la que puede haber cualquier combinación de CM maestros DP o esclavos CP. Los maestros DP en una implementación V3.0 pueden controlar un máximo de 32 esclavos.

En V2.2 se pueden configurar tres CM PROFIBUS como máximo por estación, en la que sólo uno puede ser el maestro DP. Un maestro DP en una implementación V2.2 puede controlar un máximo de 16 esclavos.

10.3.1 Módulos de comunicación para PROFIBUS

10.3.1.1 Conexión a PROFIBUS

Conexión de la S7-1200 a PROFIBUS DP

Con ayuda de los siguientes módulos de comunicación se puede conectar la S7-1200 a un sistema de bus de campo PROFIBUS:

- CM 1242-5
Función de esclavo DP
- CM 1243-5
Función de maestro DP clase 1

En caso de montaje de un CM 1242-5 y un CM 1243-5, una S7-1200 puede realizar al mismo tiempo las siguientes funciones:

- esclavo de un sistema maestro DP de rango superior
y
- maestro de un sistema maestro DP subordinado

10.3.1.2 Servicios de comunicación de los CMs PROFIBUS

Protocolo de bus

Los CM PROFIBUS utilizan el protocolo PROFIBUS DP V1.

Interlocutores de comunicación PROFIBUS de la S7-1200

Con los dos CM PROFIBUS se hace posible a la S7-1200 transmitir datos a los siguientes interlocutores.

- CM 1242-5
El CM 1242-5 (esclavo DP) puede ser interlocutor de los siguientes maestros DP-V0/V1:
 - SIMATIC S7-1200, S7-300, S7-400, S7-Modular Embedded Controller
 - Módulos maestros DP de la periferia descentralizada SIMATIC ET200
 - Estaciones PC SIMATIC
 - SIMATIC NET IE/PB Link
 - Autómatas programables de diversos fabricantes
- CM 1243-5
El CM 1243-5 (maestro DP) puede ser interlocutor de los siguientes esclavos DP V0/V1:
 - Periferia descentralizada SIMATIC ET200
 - CPU S7-1200 con CM 1242-5
 - CPU S7-200 con módulo DP PROFIBUS EM 277
 - Convertidores de frecuencia SINAMICS
 - Accionamientos y actuadores de diversos fabricantes
 - Sensores de diversos fabricantes
 - CPU S7-300/400 con interfaz PROFIBUS
 - CPU S7-300/400 con CP PROFIBUS (p. ej. CP 342-5)
 - Estaciones PC SIMATIC con CP PROFIBUS

Tipos de comunicación bajo DP-V1

Se dispone de los siguientes tipos de comunicación bajo DP-V1:

- Comunicación cíclica (CM 1242-5 y CM 1243-5)

Ambos módulos PROFIBUS son compatibles con la comunicación cíclica para la transmisión de datos de proceso entre esclavo DP y maestro DP.

De la comunicación cíclica se encarga el sistema operativo de la CPU. Para ello no se necesitan bloques de software. Los datos de E/S se leen o se escriben directamente en la memoria imagen del proceso de la CPU.

- Comunicación acíclica (sólo CM 1243-5)

El módulo maestro DP es compatible asimismo con la comunicación acíclica mediante bloques de software:

- Para el tratamiento de alarmas se dispone de la instrucción "RALRM".
- Para la transmisión de datos de configuración y diagnóstico se dispone de las instrucciones "RDREC" y "WRREC".

Funciones no soportadas del CM 1243-5:

SYNC/FREEZE

Get_Master_Diag

Otros servicios de comunicación del CM 1243-5

El módulo maestro DP CM 1243-5 es compatible con los siguientes servicios de comunicación adicionales:

- Comunicación S7

- Servicios PUT/GET

El maestro DP actúa como cliente y servidor para peticiones de otros controladores S7 o PC a través de PROFIBUS.

- Comunicación PG/OP

Las funciones de PG permiten cargar datos de configuración y programas de usuario desde una programadora (PG) así como la transmisión de datos de diagnóstico a una programadora.

Interlocutores posibles para la comunicación OP son HMI Panels, SIMATIC Panel PC con WinCC flexible o sistemas SCADA compatibles con la comunicación S7.

10.3.1.3 Otras propiedades de los CMs PROFIBUS

Configuración y sustitución de módulos

La configuración de los módulos, las redes y las conexiones se realiza en STEP 7 a partir de la versión V11.0.

Para la configuración en sistemas de terceros existe un archivo GSD para el CM 1242-5 (esclavo DP) en el CD suministrado con el módulo, así como en las páginas de Siemens Automation Customer Support en Internet.

Los datos de configuración de los CM PROFIBUS se almacenan en la respectiva CPU local. Gracias a ello, en caso de recambio estos módulos de comunicación se pueden sustituir fácilmente.

Por cada estación se pueden configurar como máximo tres CM PROFIBUS y de ellos.

Conexiones eléctricas

- Alimentación eléctrica
 - El CM 1242-5 se alimenta a través del bus de fondo de panel de la estación SIMATIC.
 - El CM 1243-5 posee una conexión propia para la alimentación eléctrica con 24 V DC.

- PROFIBUS

La interfaz RS485 de la conexión PROFIBUS es una conexión hembra Sub-D de 9 polos.

Las redes PROFIBUS ópticas se pueden conectar opcionalmente a través de un Optical Bus Terminal OBT o de un Optical Link Module OLM.

Información complementaria

Encontrará información detallada sobre los CM PROFIBUS en los manuales de los dispositivos. Los encontrará en Internet, en las páginas de Siemens Industrial Automation Customer Support, con la siguiente ID de artículo:

- CM 1242-5:
49852105 (<http://support.automation.siemens.com/WW/view/es/49852105>)
- CM 1243-5:
49851842 (<http://support.automation.siemens.com/WW/view/es/49851842>)

10.3.1.4 Ejemplos de configuración para PROFIBUS

A continuación encontrará ejemplos para la configuración en donde se utiliza el CM 1242-5 como esclavo PROFIBUS y el CM 1243-5 como maestro PROFIBUS.

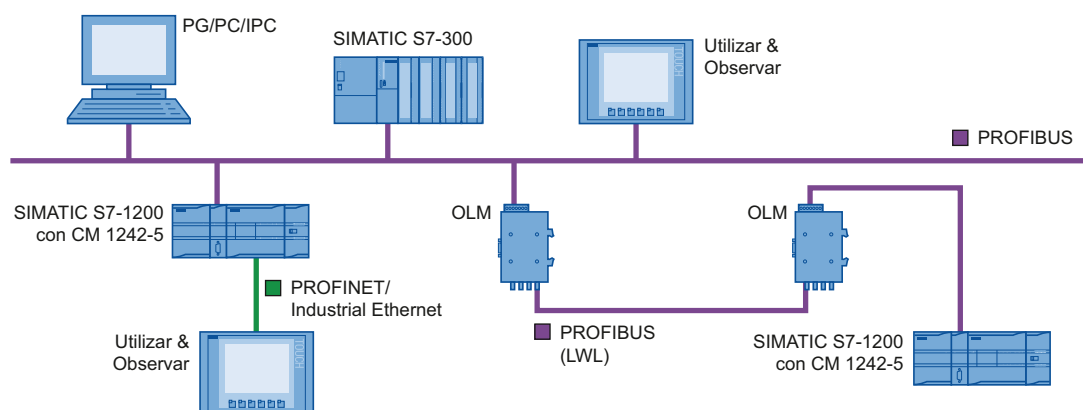


Figura 10-1 Ejemplo de configuración con CM 1242-5 como esclavo PROFIBUS

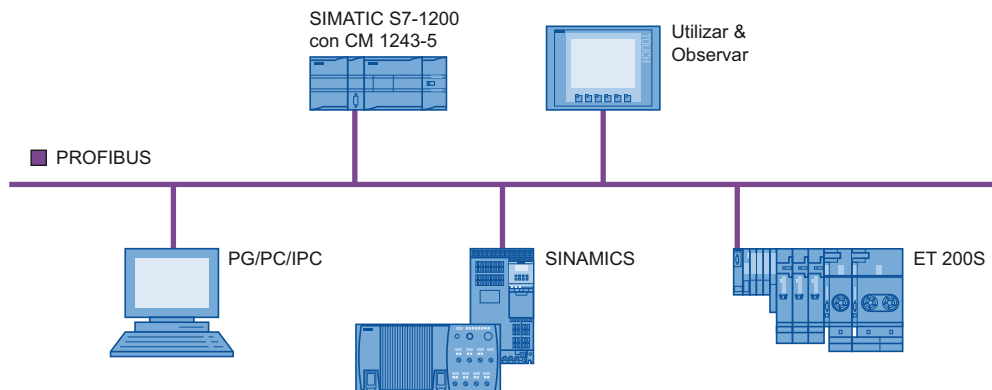


Figura 10-2 Ejemplo de configuración con CM 1243-5 como maestro PROFIBUS

10.3.2 Configurar un maestro DP y un dispositivo esclavo

10.3.2.1 Agregar el módulo CM 1243-5 (maestro DP) y un esclavo DP

En el portal "Dispositivos y redes", utilice el catálogo de hardware para agregar módulos PROFIBUS a la CPU. Estos módulos se conectan a la izquierda de la CPU. Para insertar un módulo en la configuración de hardware, selecciónelo en el catálogo de hardware y haga doble clic en él, o bien arrástrelo hasta el slot resaltado.

Tabla 10- 34 Agregar un módulo PROFIBUS CM 1243-5 (maestro DP) a la configuración de dispositivos

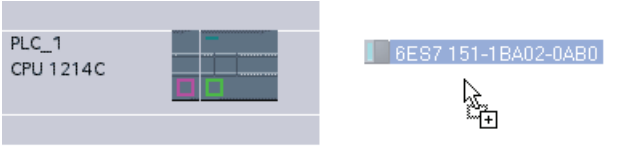

Módulo	Seleccionar el módulo	Insertar el módulo	Resultado
CM 1243-5 (maestro DP)			

Asimismo, utilice el catálogo de hardware para agregar esclavos DP. Por ejemplo, para agregar un esclavo DP ET200 S, en el catálogo de hardware, expanda las siguientes carpetas:

- E/S descentralizada
- ET200 S
- Módulos de interfaz
- PROFIBUS

A continuación, seleccione "6ES7 151-1BA02-0AB0" (IM151-1 HF) en la lista de referencias y agregue el esclavo DP ET200 S como se muestra en la figura de abajo.

Tabla 10- 35 Agregar un esclavo DP ET200 S a la configuración de dispositivos

Inserte el esclavo DP	Resultado
	

10.3.2.2 Configurar las conexiones de red lógicas entre dos dispositivos PROFIBUS

Tras configurar el módulo CM 1243-5 (maestro DP) podrá proceder a configurar sus conexiones de red.

En el portal "Dispositivos y redes", utilice la "Vista de red" para crear las conexiones de red entre los dispositivos del proyecto. Para crear la conexión PROFIBUS, seleccione la casilla (PROFIBUS) lila en el primer dispositivo. Arrastre una línea hasta la casilla PROFIBUS del segundo dispositivo. Suelte el botón del ratón para crear la conexión PROFIBUS.

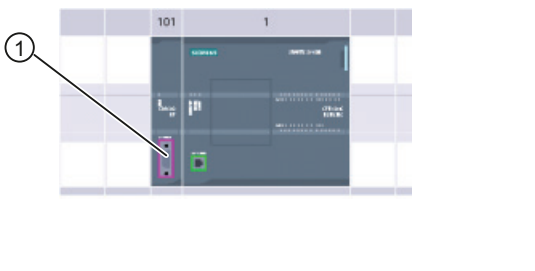
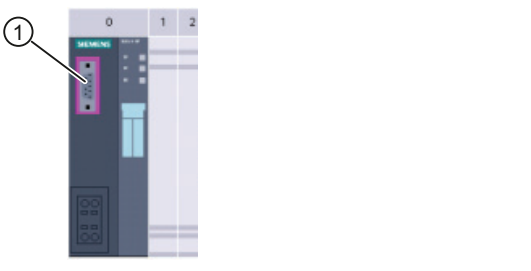
Para más información, consulte el apartado "Configuración de dispositivos: Crear una conexión de red". (Página 131)

10.3.2.3 Asignar direcciones PROFIBUS al módulo CM 1243-5 y al esclavo DP

Configurar la interfaz PROFIBUS

Tras configurar las conexiones de red lógicas entre dos dispositivos PROFIBUS, puede proceder a configurar los parámetros de las interfaces PROFIBUS. Para tal fin, haga clic en la casilla PROFIBUS lila en el módulo CM 1243-5; seguidamente, la ficha "Propiedades" de la ventana de inspección mostrará la interfaz PROFIBUS. La interfaz PROFIBUS del esclavo DP se configura del mismo modo.

Tabla 10- 36 Configurar las interfaces PROFIBUS del módulo CM 1243-5 (maestro DP) y del esclavo DP ET200 S

Módulo CM 1243-5 (maestro DP)	Esclavo DP ET200 S
	

① Puerto PROFIBUS

Asignar la dirección PROFIBUS

En una red PROFIBUS a cada dispositivo se le asigna una dirección PROFIBUS. Esta dirección tiene un rango de 0 a 127, con las excepciones siguientes:

- Dirección 0: Reservada para la configuración de red y/o herramientas de programación asignadas al bus
- Dirección 1: Reservada por Siemens para el primer maestro
- Dirección 126: Reservada para dispositivos de fábrica que no disponen de un ajuste por interruptor y deben ser predireccionados a través de la red
- Dirección 127: Reservada para transmitir mensajes a todos los dispositivos de la red y no puede ser asignada a dispositivos operativos.

Por lo tanto, las direcciones que se pueden utilizar para dispositivos operativos PROFIBUS están comprendidas entre 2 y 125.

En la ventana de propiedades, seleccione la entrada de configuración "Dirección PROFIBUS". STEP 7 muestra el cuadro de diálogo de configuración de la dirección PROFIBUS, mediante el cual se asigna la dirección PROFIBUS del dispositivo.

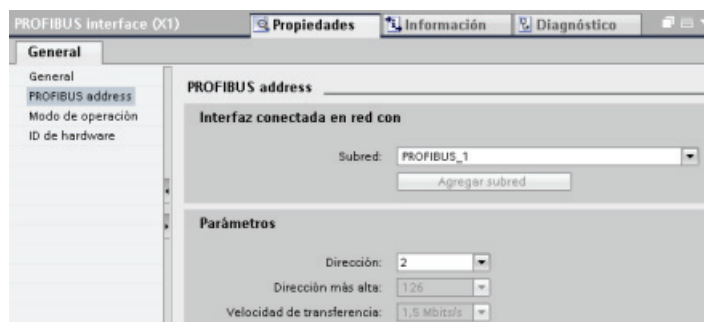


Tabla 10- 37 Parámetros de la dirección PROFIBUS

Parámetro	Descripción	
Subred	Nombre de la subred a la que está conectada el dispositivo. Haga clic en el botón "Agregar nueva subred" para crear una subred nueva. El ajuste predeterminado es "no conectado". Son posibles dos tipos de conexión: <ul style="list-style-type: none"> • El ajuste predeterminado "no conectado" ofrece una conexión local. • Una subred se requiere cuando la red comprende dos o más dispositivos. 	
Parámetros	Dirección	Dirección PROFIBUS asignada al dispositivo
	Dirección más alta	La dirección PROFIBUS más alta está basada en las estaciones activas en PROFIBUS (por ejemplo, maestro DP). Los esclavos DP pasivos tienen independientemente direcciones PROFIBUS entre 1 y 125 incluso si la dirección PROFIBUS más alta está ajustada p. ej. en 15. La dirección PROFIBUS más alta es relevante para el envío del token (envío de los derechos de transmisión). El token sólo se envía a estaciones activas. Al especificar la dirección PROFIBUS más alta se optimiza el bus.

Parámetro		Descripción
	Velocidad de transferencia	Velocidad de transferencia de la red PROFIBUS configurada: Las velocidades de transferencia de PROFIBUS abarcan un rango de 9,6 Kbits/s a 12 Mbits/s. El ajuste de la velocidad de transferencia depende de las propiedades de los nodos PROFIBUS utilizados. La velocidad de transferencia no debe exceder la velocidad soportada por el nodo más lento. La velocidad de transferencia se ajusta normalmente para el maestro en la red PROFIBUS. Todos los esclavos DP utilizan automáticamente la misma velocidad de transferencia (auto-baud).

10.3.3 Instrucciones de E/S descentralizadas

Consulte "E/S descentralizadas (PROFINET, PROFIBUS o AS-i)" (Página 285) para obtener información sobre cómo utilizar las instrucciones de E/S descentralizadas con estas redes de comunicación.

10.3.4 Instrucciones de diagnóstico

Consulte "Diagnóstico (PROFINET o PROFIBUS)": "Instrucciones de diagnóstico" (Página 311) para obtener información sobre cómo utilizar estas instrucciones con estas redes de comunicación.

10.3.5 Eventos de diagnóstico de E/S descentralizadas

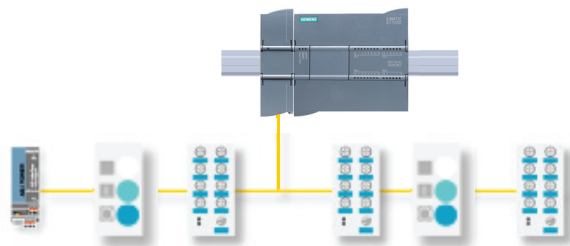
Consulte "Diagnóstico (PROFINET o PROFIBUS)": "Eventos de diagnóstico de E/S descentralizadas" (Página 311) para obtener información sobre cómo utilizar esta información de diagnóstico con estas redes de comunicación.

10.4 AS-i

El maestro AS-i CM 1243-2 del S7-1200 permite conectar una red AS-i a una CPU S7-1200.

La interfaz actuador/sensor (o bien AS-i) es un sistema de conexión de red con maestro individual que se utiliza en el nivel más bajo de sistemas de automatización. El CM 1243-2 asume la función de maestro AS-i en la red. Mediante un único cable AS-i, los sensores y actuadores (dispositivos esclavos AS-i) se pueden conectar a la CPU a través del CM 1243-2. El CM 1243-2 gestiona toda la coordinación de la red AS-i y transmite datos e información de estado de los actuadores y sensores a la CPU a través de las direcciones de E/S asignadas al CM 1243-2. Se puede acceder a valores binarios o analógicos dependiendo del tipo de esclavo. Los esclavos AS-i son los canales de entrada y salida del sistema AS-i y sólo están activos cuando el CM 1243-2 los llama.

En la siguiente figura, el S7-1200 es un maestro AS-i que controla el panel de operador AS-i y los dispositivos esclavos digitales/analógicos del módulo de E/S.



10.4.1 Configuración de un maestro AS-i y un dispositivo esclavo

El maestro AS-i CM 1243-2 está integrado en el sistema de automatización S7-1200 como módulo de comunicación.

Encontrará información detallada sobre el maestro AS-i CM 1243-2 en el manual "Maestro AS-i CM 1243-2 y módulo de desacoplamiento de datos AS-i DCM 1271 para SIMATIC S7-1200" (<http://support.automation.siemens.com/WW/view/en/50414115/133300>).

10.4.1.1 Agregar el módulo maestro AS-i CM 1243-2 y esclavo AS-i

Utilice el catálogo de hardware para agregar módulos maestros AS-i CM1243-2 a la CPU. Estos módulos se conectan por el lado izquierdo de la CPU, siendo posible utilizar un máximo de tres módulos maestros AS-i CM1243-2. Para insertar un módulo en la configuración de hardware, selecciónelo en el catálogo de hardware y haga doble clic en él, o bien arrástrelo hasta el slot resaltado.

Tabla 10- 38 Agregar un módulo maestro AS-i CM1243-2 a la configuración de dispositivos

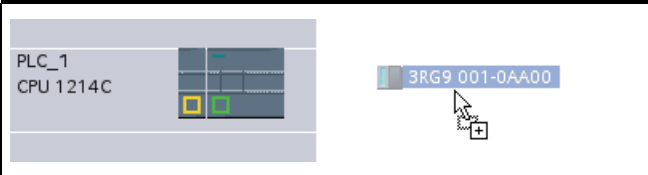
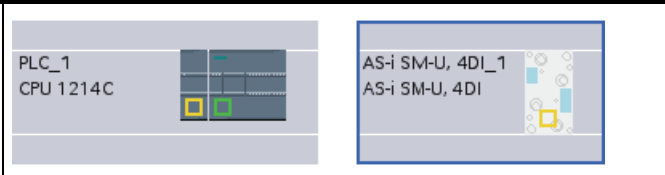
Módulo	Seleccionar el módulo	Insertar el módulo	Resultado
CM 1243-2 maestro AS-i			

Asimismo, utilice el catálogo de hardware para agregar esclavos AS-i. Por ejemplo, para agregar un esclavo "módulo E/S, compacto, digital, de entrada" en el catálogo de hardware, expanda las siguientes carpetas:

- Dispositivos de campo
- Esclavos ASInterface

A continuación, seleccione "3RG9 001-0AA00" (AS-i SM-U, 4DI) en la lista de referencias y agregue el esclavo "módulo E/S, compacto, digital, de entrada" como se muestra en la figura de abajo.

Tabla 10- 39 Agregar un esclavo AS-i a la configuración de dispositivos

Insertar el esclavo AS-i	Resultado
	

10.4.1.2 Configuración de las conexiones de red lógicas entre dos dispositivos AS-i

Después de configurar el maestro AS-i CM1243-2, ya puede configurar las conexiones de red.

En el portal "Dispositivos y redes", utilice la "Vista de red" para crear las conexiones de red entre los dispositivos del proyecto. Para crear la conexión AS-i, seleccione la casilla (AS-i) amarilla en el primer dispositivo. Arrastre una línea hasta la casilla AS-i del segundo dispositivo. Suelte el botón del ratón para crear la conexión AS-i.

Para más información, consulte el apartado "Configuración de dispositivos: Crear una conexión de red". (Página 131)

10.4.1.3 Configurar las propiedades del maestro AS-i CM1243-2

Para configurar los parámetros para la interfaz AS-i, haga clic en la casilla AS-i amarilla del módulo maestro AS-i CM1243-2; en la ficha "Propiedades" de la ventana de inspección se mostrará la interfaz AS-i.

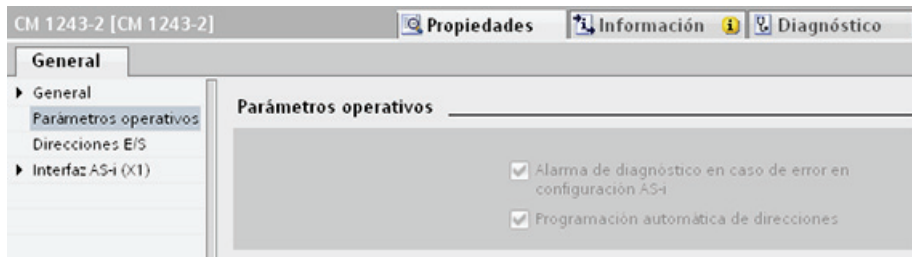
En la ventana de inspección de STEP 7 puede ver, configurar y cambiar la información general, las direcciones y los parámetros operativos:

Tabla 10- 40 Propiedades del módulo maestro AS-i CM1243-2

Propiedad	Descripción
General	Nombre del maestro AS-i CM1243-2
Parámetros operativos	Parámetros para la respuesta del maestro AS-i
Direcciones de E/S	Área para las direcciones de E/S de esclavos
Interfaz AS-i (X1)	Red AS-i asignada

Nota

Las opciones "Alarma de diagnóstico en caso de error en la configuración AS-i" y "Programación automática de direcciones" siempre están activas y, por consiguiente, aparecen en gris.



10.4.1.4 Asignar una dirección AS-i a un esclavo AS-i

Configurar la interfaz AS-i del esclavo

Para configurar los parámetros para la interfaz AS-i, haga clic en la casilla amarilla AS-i del esclavo AS-i; en la ficha "Propiedades" de la ventana de inspección se mostrará la interfaz AS-i.



Asignar la dirección de esclavo AS-i

En una red AS-i, a cada dispositivo se le asigna una dirección de esclavo AS-i. Dicha dirección puede estar comprendida entre 0 y 31; no obstante, el 0 está reservado exclusivamente para nuevos dispositivos esclavos.

Las direcciones de esclavo van de 1(A o B) a 31(A o B) para un total de 62 dispositivos esclavos. Cualquier dirección que se encuentre en el rango de 1 - 31 puede ser asignada a un dispositivo esclavo, es decir, no es relevante si los esclavos comienzan por la dirección 21 o si el primer esclavo tiene asignada realmente la dirección 1.

Asigne aquí la dirección de esclavo AS-i.

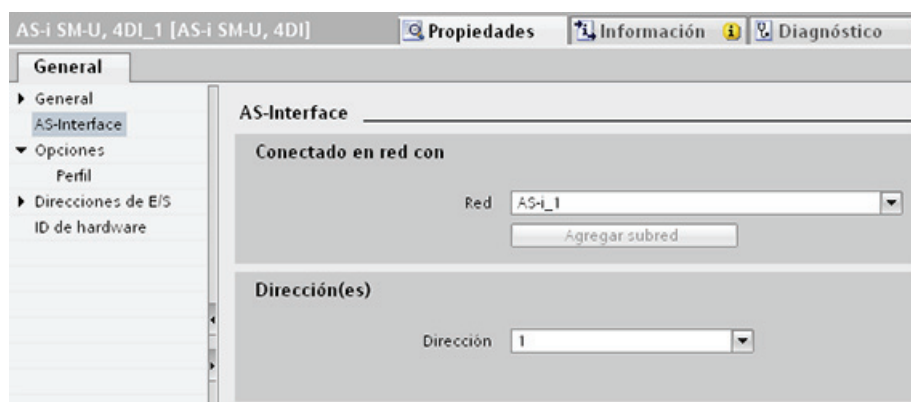


Tabla 10- 41 Parámetros de la interfaz AS-i

Parámetro	Descripción
Red	Nombre de la red a la que está conectado el dispositivo
Direcciones	Dirección AS-i asignada para el dispositivo esclavo dentro del rango de 1(A o B) a 31(A o B) para un total de 62 dispositivos esclavos

10.4.2 Intercambio de datos entre el programa de usuario y los esclavos AS-i

10.4.2.1 Configuración básica de STEP 7

El maestro AS-i reserva un área de datos de 62 bytes en el área E/S de la CPU. Se accede a los datos digitales en bytes; para cada esclavo, hay un byte de datos de entrada y un byte de datos de salida.

La asignación de las conexiones AS-i de los esclavos digitales AS-i a los bits de datos del byte asignado se indica en la ventana de inspección del maestro AS-i CM 1243-2.

Dirección I	Dirección O	Dirección A...	ID HW
		0	335
2	2	1A	336
33	33	1B	337
3	3	2A	338
34	34	2B	339
4	4	3A	340
35	35	3B	341
5	5	4A	342
36	36	4B	343
6	6	5A	344
37	37	5B	345
7	7	6A	346

Se accede a los datos de los esclavos AS-i en el programa de usuario utilizando las direcciones E/S visualizadas con las operaciones de lógica binaria adecuadas (p. ej. "Y") o bien asignaciones de bits.

Nota

El parámetro "Asignación de sistema" se activa automáticamente si no se configuran los esclavos AS-i con STEP 7.

Si no se configuran esclavos, hay que informar al maestro AS-i CM 1243-2 acerca de la configuración de bus real utilizando la función online "ES > DEBE".

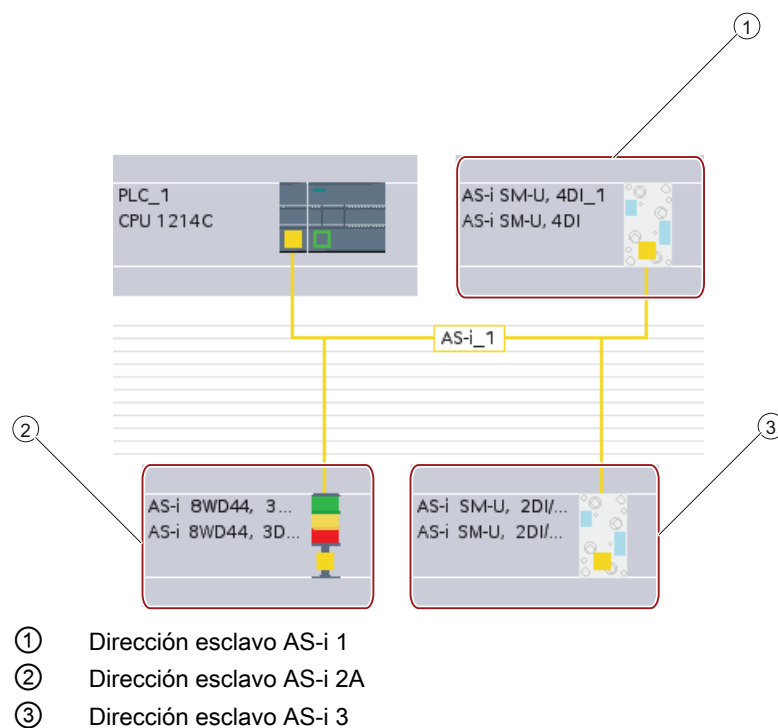
Más información

Encontrará información detallada sobre el maestro AS-i CM 1243-2 en el manual "Maestro AS-i CM 1243-2 y módulo de desacoplamiento de datos AS-i DCM 1271 para SIMATIC S7-1200" (<http://support.automation.siemens.com/WW/view/en/50414115/133300>).

10.4.2.2 Configurar esclavos con STEP 7

Transferir valores digitales AS-i

La CPU accede a las entradas y salidas digitales de los esclavos AS-i a través del maestro AS-i CM 1243-2 durante el funcionamiento cíclico. Accede a los datos utilizando las direcciones E/S o bien mediante una transferencia de registros.

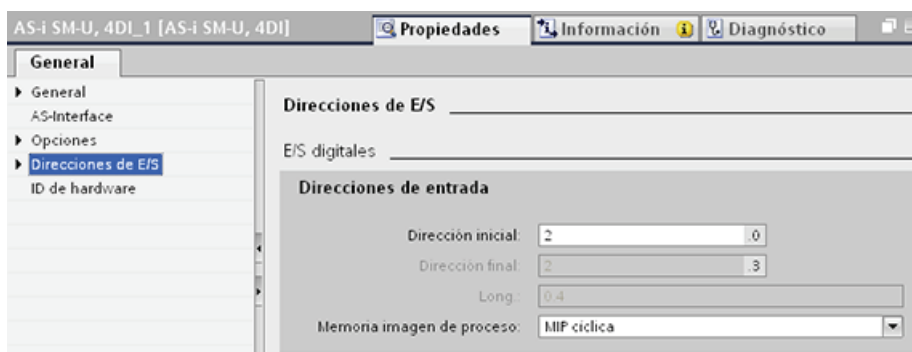


En este caso, se accede a los datos digitales en bytes (es decir, cada esclavo digital AS-i tiene asignado un byte). Al configurar los esclavos AS-i en STEP 7, la dirección E/S para acceder a los datos desde el programa de usuario se muestra en la ventana de inspección del esclavo AS-i correspondiente.

El módulo de entradas digitales (AS-i SM-U, 4DI) en la red AS-i superior ha recibido la dirección de esclavo 1. Al hacer clic en el módulo de entradas digitales, la ficha "AS-Interface" de "Propiedades" muestra la dirección del esclavo, como se muestra a continuación:

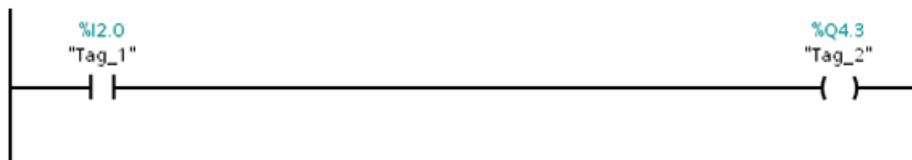


El módulo de entradas digitales (AS-i SM-U, 4DI) en la red AS-i anterior ha recibido la dirección de E/S 2. Al hacer clic en el módulo de entradas digitales, la ficha "Direcciones de E/S" de "Propiedades" muestra la dirección de E/S, como se muestra a continuación:



Para acceder a los datos de los esclavos AS-i en el programa de usuario, se utilizan las direcciones E/S con las operaciones de lógica binaria adecuadas (p. ej. "Y") o bien asignaciones de bits. El siguiente programa sencillo muestra cómo funciona la asignación:

La entrada 2.0 se interroga en este programa. En el sistema AS-i, dicha entrada pertenece al esclavo1 (byte de entrada 2, bit 0). La salida 4.3, que se activa entonces, corresponde al esclavo AS-i 3 (byte de salida 4, bit 3)



Transferir valores analógicos AS-i

Se puede acceder a los datos analógicos de un esclavo AS-i desde la memoria imagen de proceso de la CPU siempre que se haya configurado dicho esclavo AS-i en STEP 7 como esclavo analógico.

Si no se ha configurado el esclavo analógico en STEP 7, solo será posible acceder a los datos del esclavo AS-i utilizando funciones acíclicas (interfaz de registro). En el programa de usuario de la CPU, las llamadas AS-i se leen y escriben utilizando las instrucciones E/S descentralizadas RDREC (leer registro) y WRREC (escribir registro).

Nota

Si se ha especificado una configuración de los esclavos AS-i con STEP 7 y se ha descargado en la estación S7, esta es transferida por la CPU al maestro AS-i CM 1243-2 durante el arranque de la estación S7. Cualquier configuración existente que se haya determinado mediante la función online "Asignación de sistema" (Página 503) ("ES -> DEBE") se sobrescribirá.

Más información

Encontrará información detallada sobre el maestro AS-i CM 1243-2 en el manual "Maestro AS-i CM 1243-2 y módulo de desacoplamiento de datos AS-i DCM 1271 para SIMATIC S7-1200" (<http://support.automation.siemens.com/WW/view/en/50414115/133300>).

10.4.3 Instrucciones de E/S descentralizadas

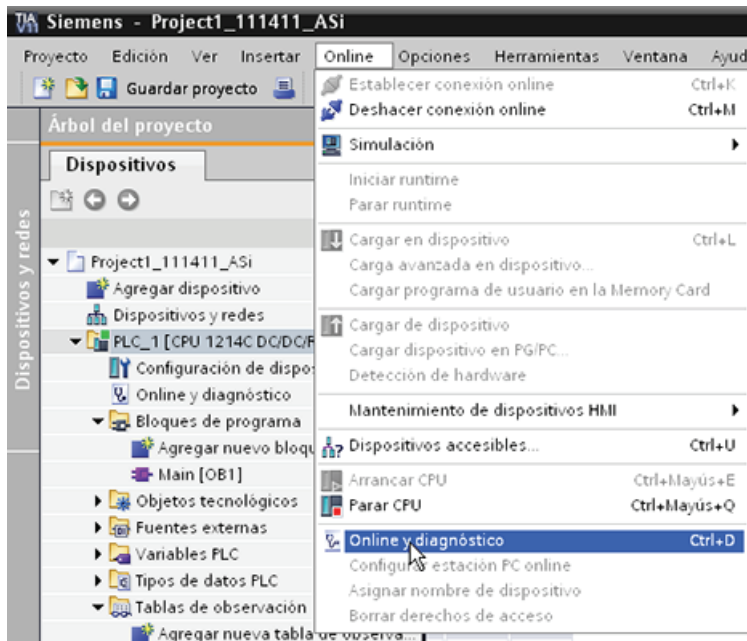
Consulte "E/S descentralizadas (PROFINET, PROFIBUS o AS-i)" (Página 285) para obtener información sobre cómo utilizar las instrucciones de E/S descentralizadas con estas redes de comunicación.

10.4.4 Trabajar con herramientas online AS-i

Cambiar los modos de operación AS-i online

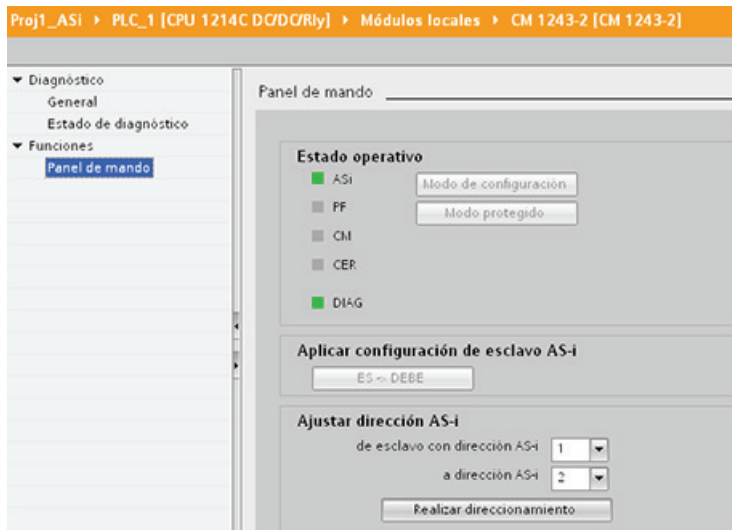
Hay que pasar a online para ver y modificar los modos de operación AS-i.

Para cambiar al modo online, vaya a "Configuración de dispositivos" con el módulo maestro AS-i CM 1243-2 seleccionado y, seguidamente, haga clic en el botón "Establecer conexión online" de la barra de herramientas. A continuación, seleccione el comando "Online y diagnóstico" del menú "Online".



Existen dos modos de operación AS-i:

- Modo de protección:
 - No es posible modificar el dispositivo esclavo AS-i ni las direcciones E/S de la CPU.
 - El LED verde "CM" está apagado.
- Modo de configuración:
 - Es posible realizar los cambios necesarios en el dispositivo esclavo AS-i y en las direcciones E/S de la CPU.
 - El LED verde "CM" está encendido.



En el campo "Ajustar dirección AS-i", puede cambiar la dirección del esclavo AS-i. A un esclavo nuevo que aún no tenga dirección se le asigna siempre la dirección 0. El maestro lo reconoce como esclavo nuevo aunque no tenga asignada una dirección y no lo incluye en la comunicación normal hasta que no le sea asignada una dirección.

Error de configuración

Si el LED amarillo "CER" está encendido, existe un error en la configuración del dispositivo esclavo AS-i. Seleccione el botón "ES > DEBE" para sobrescribir la configuración del dispositivo esclavo del módulo maestro AS-i CM 1243-2 con la configuración del dispositivo esclavo de la red de campo AS-i.

10.5 Comunicación S7

10.5.1 Instrucciones GET y PUT

Las instrucciones GET y PUT se pueden utilizar para comunicarse con las CPU S7 a través de conexiones PROFINET y PROFIBUS.

- Acceso a datos en una CPU remota: una CPU S7-1200 sólo puede utilizar direcciones absolutas en el campo de entrada ADDR_x para direccionar variables de CPU remotas (S7-200/300/400/1200).
- Acceder a los datos en un DB estándar: una CPU S7-1200 sólo puede utilizar direcciones absolutas en el campo de entrada ADDR_x para direccionar variables de DB en un DB estándar de una CPU S7 remota.
- Acceder a los datos en un DB optimizado: una CPU S7-1200 no puede acceder a variables de DB en un DB optimizado de una CPU S7-1200 remota.
- Acceso a datos en una CPU local: una CPU S7-1200 puede usar direcciones tanto absolutas como simbólicas como entradas de los campos de entrada RD_x o SD_x de las instrucciones GET o PUT.

Tabla 10- 42 Instrucciones GET y PUT

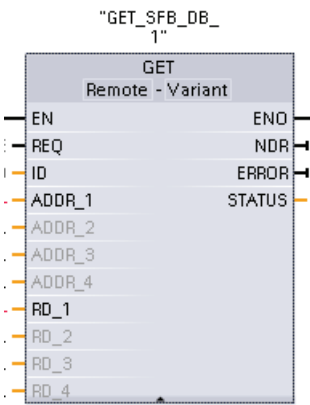
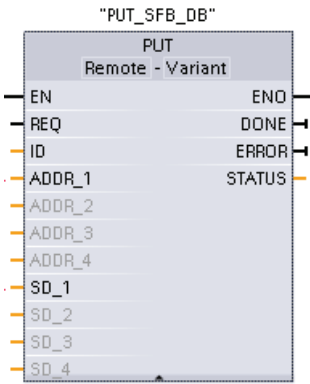
KOP / FUP	SCL	Descripción
<p>"GET_SFB_DB_1"</p> 	<pre>"GET_DB" (req:=_bool_in_, ID:=_word_in_, ndr=>_bool_out_, error=>_bool_out_, status=>_word_out_, addr_1:=_remote_inout_, [...addr_4:=_remote_inout_,] rd_1:=_variant_inout_ [...rd_4:=_variant_inout_]);</pre>	<p>Utilice la instrucción GET para leer datos desde una CPU S7 remota. La CPU remota puede estar tanto en modo RUN como STOP.</p> <p>STEP 7 crea automáticamente el DB al introducir la instrucción.</p>
<p>"PUT_SFB_DB"</p> 	<pre>"PUT_DB" (req:=_bool_in_, ID:=_word_in_, done=>_bool_out_, error=>_bool_out_, status=>_word_out_, addr_1:=_remote_inout_, [...addr_4:=_remote_inout_,] sd_1:=_variant_inout_, [...sd_4:=_variant_inout_]);</pre>	<p>Utilice la instrucción PUT para escribir datos en una CPU S7 remota. La CPU remota puede estar tanto en modo RUN como STOP.</p> <p>STEP 7 crea automáticamente el DB al introducir la instrucción.</p>

Tabla 10- 43 Tipos de datos para los parámetros

Parámetro y tipo	Tipo de datos	Descripción
REQ	Input	Bool
ID	Input	CONN_PRG (Word)
NDR (GET)	Output	Bool
DONE (PUT)	Output	Bool

Parámetro y tipo		Tipo de datos	Descripción
ERROR STATUS	Output Output	Bool Word	<ul style="list-style-type: none"> • ERROR=0 Valor de STATUS: – 0000H: ni advertencia ni error – <> 0000H: Advertencia, STATUS suministra información detallada • ERROR=1 Existe un error. STATUS suministra información detallada sobre la naturaleza del error.
ADDR_1	InOut	Remoto	Puntero a las áreas de memoria de la CPU remota que almacena los datos que deben leerse (GET) o que se envían (PUT).
ADDR_2	InOut	Remoto	
ADDR_3	InOut	Remoto	
ADDR_4	InOut	Remoto	
RD_1 (GET) SD_1 (PUT)	InOut	Variante	Puntero a las áreas de memoria de la CPU local que almacena los datos que deben leerse (GET) o enviarse (PUT).
RD_2 (GET) SD_2 (PUT)	InOut	Variante	Tipos de datos permitidos: Bool (sólo se permite un único bit), Byte, Char, Word, Int, DWord, DInt o Real.
RD_3 (GET) SD_3 (PUT)	InOut	Variante	Nota: Si el puntero accede a un DB, se debe especificar la dirección absoluta, p. ej.:
RD_4 (GET) SD_4 (PUT)	InOut	Variante	P# DB10.DBX5.0 Byte 10 En este caso, 10 representa el número de bytes que deben leerse (GET) o transmitirse (PUT).

Asegúrese de que la longitud (número de bytes) y los tipos de datos para los parámetros ADDR_x (CPU remota) y RD_x o SD_x (CPU local) concuerden. El número tras el identificador "Byte" es el número de bytes referenciado por el parámetro ADDR_x, RD_x o SD_x.

Nota

El número total de bytes recibidos en una instrucción GET o el número total de bytes enviados en una instrucción PUT es limitado. Las limitaciones se basan en el número de las cuatro posibles direcciones y áreas de memoria que se utilizan:

- Si sólo se utilizan ADDR_1 y RD_1/SD_1, una instrucción GET puede recibir 222 bytes y una instrucción PUT puede enviar 212 bytes.
- Si se utilizan ADDR_1, RD_1/SD_1, ADDR_2 y RD_2/SD_2, una instrucción GET puede recibir un total de 218 bytes y una instrucción PUT puede enviar un total de 196 bytes.
- Si se utilizan ADDR_1, RD_1/SD_1, ADDR_2, RD_2/SD_2, ADDR_3 y RD_3/SD_3, una instrucción GET puede recibir un total de 214 bytes y una instrucción PUT puede enviar un total de 180 bytes.
- Si se utilizan ADDR_1, RD_1/SD_1, ADDR_2, RD_2/SD_2, ADDR_3, RD_3/SD_3, ADDR_4 y RD_4/SD_4, una instrucción GET puede recibir un total de 210 bytes y una instrucción PUT puede enviar un total de 164 bytes.

La suma del número de bytes de cada uno de los parámetros de dirección y área de memoria debe ser menor o igual que los límites definidos. Si se superan esos límites, las instrucciones GET o PUT devuelven un error.

Con un flanco ascendente del parámetro REQ, la operación de lectura (GET) o de escritura (PUT) carga los parámetros ID, ADDR_1, y RD_1 (GET) o SD_1 (PUT).

- Para GET: La CPU remota devuelve los datos solicitados a las áreas de recepción (RD_x), comenzando por el siguiente ciclo. Una vez la operación de lectura ha finalizado sin errores, el parámetro NDR se pone a 1. Sólo se puede iniciar una nueva operación cuando la operación anterior haya finalizado.
- Para PUT: La CPU local empieza a enviar los datos (SD_x) a la posición de memoria (ADDR_x) en la CPU remota. Una vez la operación de escritura ha finalizado sin errores, la CPU remota devuelve un acuse de ejecución. El parámetro DONE de la instrucción PUT se pone a 1. Sólo se puede iniciar una nueva operación de escritura cuando la operación anterior haya finalizado.

Nota

Para garantizar la coherencia de los datos, antes de acceder a los datos o inicializar otra operación de lectura o escritura, compruebe siempre que la operación haya finalizado (NDR = 1 para GET o DONE = 1 para PUT).

Los parámetros ERROR y STATUS suministran información sobre el estado de la operación de lectura (GET) o escritura (PUT).

Tabla 10- 44 Información de error

ERROR	STATUS (decimal)	Descripción
0	11	<ul style="list-style-type: none"> • La nueva tarea no se puede ejecutar debido a que aún no ha finalizado una tarea anterior. • La tarea se está procesando ahora en una clase de prioridad más baja.
0	25	Se ha iniciado la comunicación. La tarea se está procesando.
1	1	Problemas de comunicación, p. ej.: <ul style="list-style-type: none"> • No se ha cargado la descripción de la conexión (local o remota) • Conexión interrumpida (p. ej.: cable, CPU desactivada o CM/CB/CP en modo STOP) • Conexión con el interlocutor aún no establecida
1	2	Acuse negativo por parte del dispositivo interlocutor. No se puede ejecutar la tarea.
1	4	Errores en los punteros del área de emisión (RD_x para GET o SD_x para PUT) relacionados con la longitud de los datos o el tipo de datos.
1	8	Error de acceso a la CPU interlocutora
1	10	No es posible acceder a la memoria de usuario local (p. ej. intentar acceder a un DB borrado)
1	12	Al llamar el SFB: <ul style="list-style-type: none"> • Se especificó un DB de instancia que no pertenece a GET o PUT • No se especificó un DB de instancia sino un DB compartido • No se ha encontrado ningún DB de instancia (al cargar un nuevo DB de instancia)

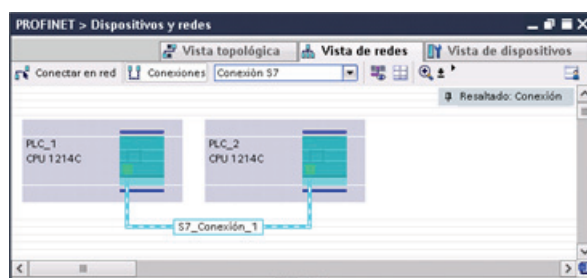
ERROR	STATUS (decimal)	Descripción
1	20	<ul style="list-style-type: none"> Se ha excedido el número máximo de tareas/instancias simultáneas Las instancias se sobrecargaron al arrancar la CPU Este estado puede presentarse al ejecutar por primera vez las instrucciones GET o PUT
1	27	No hay una instrucción GET o PUT que concuerde con la CPU.

10.5.2 Crear una conexión S7

El tipo de conexión seleccionado crea una conexión con un interlocutor. La conexión se configura, establece y vigila automáticamente.

En el portal "Dispositivos y redes", utilice la "Vista de redes" para crear las conexiones de red entre los dispositivos del proyecto. En primer lugar, haga clic en la ficha "Conexiones" y luego seleccione el tipo de conexión en la lista desplegable ubicada justo a la derecha (p. ej. una conexión S7). Haga clic en el cuadro verde (PROFINET) del primer dispositivo y trace una línea hasta el cuadro PROFINET del segundo dispositivo. Suelte el botón del ratón para crear la conexión PROFINET.

Encontrará más información en "Crear una conexión de red" (Página 131).



Haga clic en el botón "Resaltado: conexión" para acceder al cuadro de diálogo de configuración "Propiedades" de la instrucción de comunicación.

10.5.3 Configurar la vía de conexión local/interlocutor entre dos dispositivos

Configurar los parámetros generales

Los parámetros de comunicación se especifican en el cuadro de diálogo de configuración "Propiedades" de la instrucción de comunicación. Este diálogo aparece en el lado inferior de la página cuando se ha seleccionado alguna parte de la instrucción.

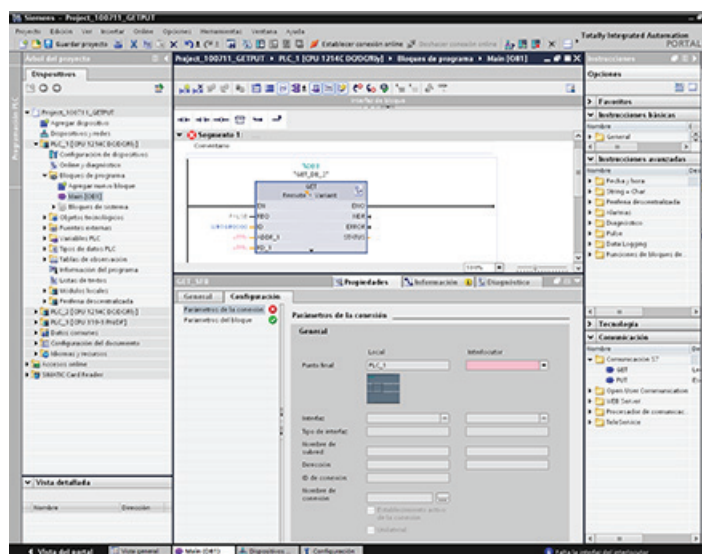
Encontrará más información en "Configuración de dispositivos: Configurar la vía de conexión local/interlocutor (Página 132)"

En el área "Detalles de dirección" del diálogo "Parámetros de la conexión" se definen los TSAPs o puertos que deben utilizarse. El TSAP o puerto de una conexión en la CPU se introduce en el campo "TSAP local". El TSAP o puerto asignado a la conexión en la CPU interlocutora se introduce en el campo "TSAP del interlocutor".

10.5.4 Asignación de parámetros de conexión GET/PUT

La asignación de parámetros de conexión mediante las instrucciones GET/PUT es una ayuda al usuario para configurar conexiones de comunicación S7 de CPU a CPU.

Tras insertar un bloque GET o PUT, se inicia la asignación de parámetros de conexión de las instrucciones GET/PUT.



La ventana de inspección muestra las propiedades de la conexión cuando se selecciona cualquier parte de la instrucción. Los parámetros de comunicación se especifican en la ficha "Configuración" de "Propiedades" de la instrucción de comunicación.

10.5.4.1 Parámetros de la conexión

La página "Parámetros de conexión" permite al usuario configurar la conexión S7 necesaria y configurar el parámetro "ID de conexión" referenciado por el parámetro de bloque GET/PUT "ID". El contenido de la página ofrece información sobre el punto final local y permite que el usuario defina la interfaz local. El usuario también puede definir el punto final del interlocutor.

La página "Parámetros del bloque" permite al usuario configurar los parámetros de bloque adicionales.

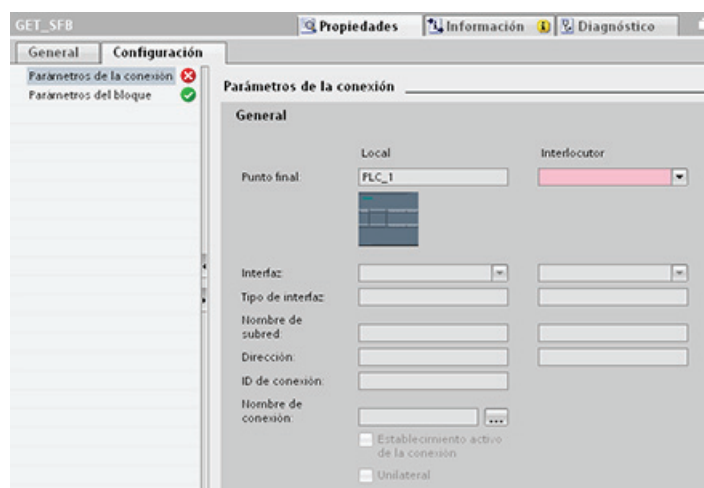


Tabla 10- 45 Parámetro de conexión: definiciones generales

Parámetro	Definición
Parámetro de conexión: General	
Punto final	"Punto final local": nombre asignado a la CPU local "Punto final del interlocutor": nombre asignado a la CPU interlocutora (remota) Nota: en la lista desplegable "Punto final del interlocutor", el sistema ofrece todos los interlocutores S7 potenciales del proyecto actual además de la opción "no especificado". Un interlocutor no especificado representa a un interlocutor que no está actualmente en el proyecto STEP 7 (por ejemplo, un dispositivo interlocutor de terceros).
Interfaz	Nombre asignado a las interfaces Nota: El usuario puede modificar la conexión cambiando las interfaces local y del interlocutor.
Tipo de interfaz	Tipo de interfaz
Nombre de subred	Nombre asignado a las subredes
Dirección	Direcciones IP asignadas Nota: El usuario puede especificar la dirección remota de un dispositivo de terceros para un interlocutor "no especificado".
ID de conexión	Número de ID: es generado automáticamente por la asignación de parámetros de conexión GET/PUT
Nombre de conexión	Ubicación de los datos de las CPU local e interlocutora: es generado automáticamente por la asignación de parámetros de conexión GET/PUT
Iniciativa local	Casilla de verificación para seleccionar la CPU local como la conexión activa
Unilateral	Casilla de verificación para especificar una conexión de uno o dos sentidos; solo lectura Nota: en una conexión GET/PUT PROFINET, tanto el dispositivo local como el interlocutor pueden actuar de servidor o cliente. Esto hace posible una conexión de dos sentidos y la casilla de verificación "Unilateral" está deseleccionada. En una conexión GET/PUT PROFIBUS, en algunos casos el dispositivo interlocutor solo puede actuar de servidor (por ejemplo un S7-300) y la casilla de verificación "Unilateral" está marcada.

Parámetro de ID de conexión

Hay tres formas de cambiar las ID de conexión definidas por el sistema:

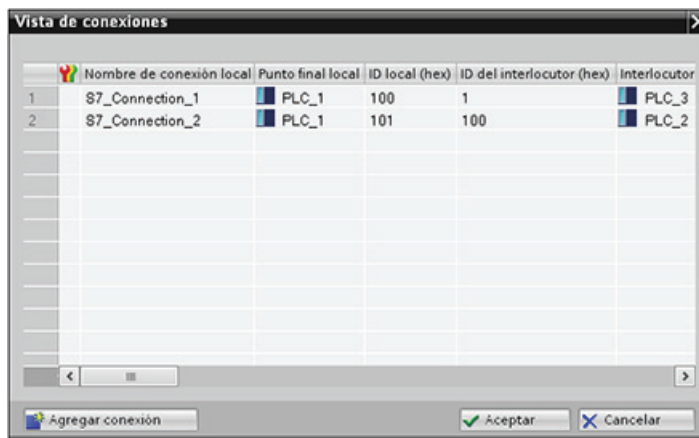
1. El usuario puede cambiar la ID actual directamente en el bloque GET/PUT. Si la ID nueva pertenece a una conexión ya existente, la conexión se modifica.
2. El usuario puede cambiar la ID actual directamente en el bloque GET/PUT, pero la ID nueva no existe. el sistema crea una nueva conexión S7.
3. El usuario puede cambiar la ID actual en el cuadro de diálogo "Vista de conexiones": la entrada del usuario se sincroniza con el parámetro de ID en el bloque GET/PUT correspondiente.

Nota

El parámetro "ID" del bloque GET/PUT no es un nombre de conexión, sino una expresión numérica que se escribe como el ejemplo siguiente: W#16#1

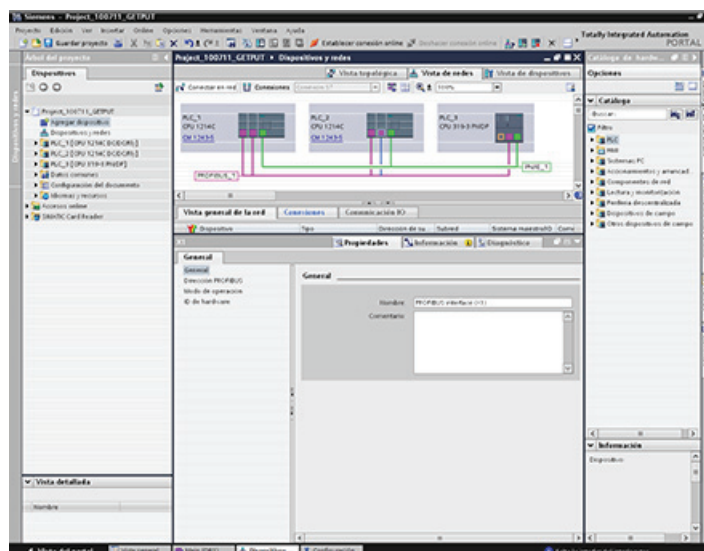
Parámetro de nombre de conexión

El nombre de la conexión se puede editar mediante un control de usuario especial, el cuadro de diálogo "Vista de conexiones". Este cuadro de diálogo ofrece todas las conexiones S7 disponibles que se pueden seleccionar como alternativa para la comunicación GET/PUT actual. El usuario puede crear una conexión completamente nueva en esta tabla. Haga clic en el botón a la derecha del campo "Nombre de conexión" para iniciar el cuadro de diálogo "Vista de conexiones".



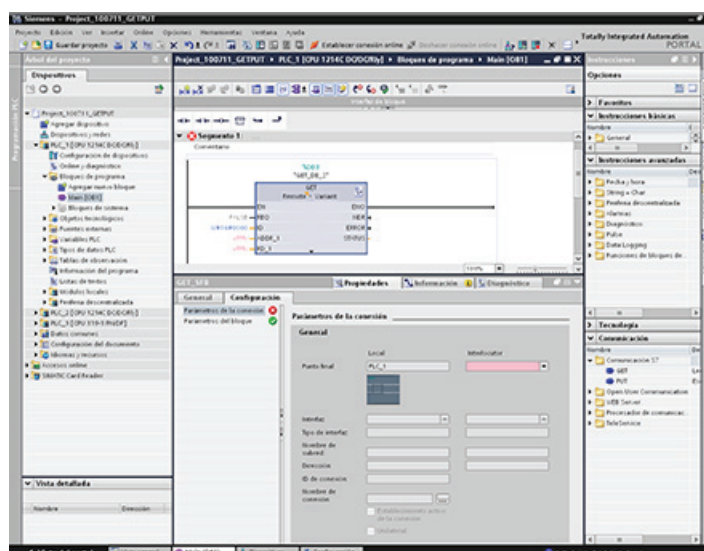
10.5.4.2 Configurar una conexión S7 de CPU a CPU

Dada la configuración de PLC_1, PLC_2 y PLC_3 mostrada en la figura siguiente, inserte bloques GET o PUT para "PLC_1".



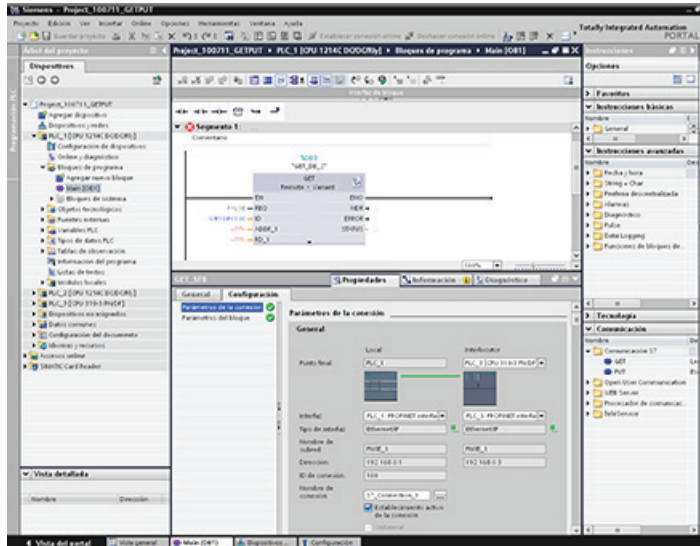
Para la instrucción GET o PUT, la ficha "Propiedades" se muestra automáticamente en la ventana de inspección con las siguientes selecciones de menú:

- "Configuración"
- "Parámetros de la conexión"



Configurar una conexión S7 PROFINET

Para el "Punto final del interlocutor", seleccione "PLC_3".



El sistema reacciona con los cambios siguientes:

Tabla 10- 46 Parámetro de conexión: valores generales

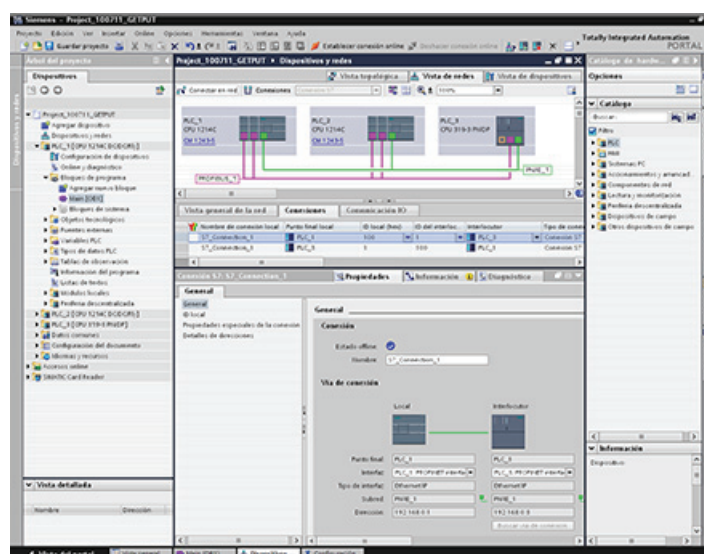
Parámetro	Definición
Parámetro de conexión: General	<p>Punto final</p> <p>"Punto final local" contiene "PLC_1" de solo lectura. El campo "Punto final del interlocutor" contiene "PLC_3[CPU319-3PN/DP]":</p> <ul style="list-style-type: none"> El color cambia de rojo a blanco. Aparece la imagen del dispositivo "interlocutor". Aparece una línea de conexión entre las imágenes de los dispositivos PLC_1 y PLC_3 (línea Ethernet verde).
Interfaz	<p>La "Interfaz local" contiene "CPU1214C DC/DC/DC, interfaz PROFINET (R0/S1)".</p> <p>La "Interfaz del interlocutor" contiene: "CPU319-3PN/DP, interfaz PROFINET (R0/S2)".</p>
Tipo de interfaz	<p>El "Tipo de interfaz local" contiene "Ethernet/IP"; el control es de solo lectura.</p> <p>El "Tipo de interfaz del interlocutor" contiene "Ethernet/IP"; el control es de solo lectura.</p> <p>Las imágenes del tipo de interfaz se muestran a la derecha, junto al "Tipo de interfaz" local y del interlocutor (icono Ethernet verde).</p>
Nombre de subred	<p>El "Nombre de subred local" contiene "PN/IE_1"; el control es de solo lectura.</p> <p>El "Nombre de subred del interlocutor" contiene "PN/IE_1"; el control es de solo lectura.</p>
Dirección	<p>La "Dirección local" contiene la dirección IP local; el control es de solo lectura.</p> <p>La "Dirección del interlocutor" contiene la dirección IP del interlocutor; el control es de solo lectura.</p>

Parámetro	Definición
ID de conexión	"ID de conexión" contiene "100". En el editor de programas, en el [OB1] principal, el valor "ID de conexión" del bloque GET/PUT también contiene "100".
Nombre de conexión	El "Nombre de conexión" contiene el nombre predeterminado de la conexión (por ejemplo, "S7_Connection_1"); el control está habilitado.
Iniciativa local	Marcada y habilitada para seleccionar la CPU local como conexión activa.
Unilateral	Solo lectura y sin marcar. Nota: tanto "PLC_1" (una CPU S7-1200 1214CDC/DC/Rly) como "PLC_3" (una CPU S7-300 319-3PN/DP) pueden actuar de servidor y cliente en una conexión GET/PUT PROFINET, lo que hace posible una conexión en ambos sentidos.

El icono GET/PUT del árbol Vista de propiedades también cambia de rojo a verde.

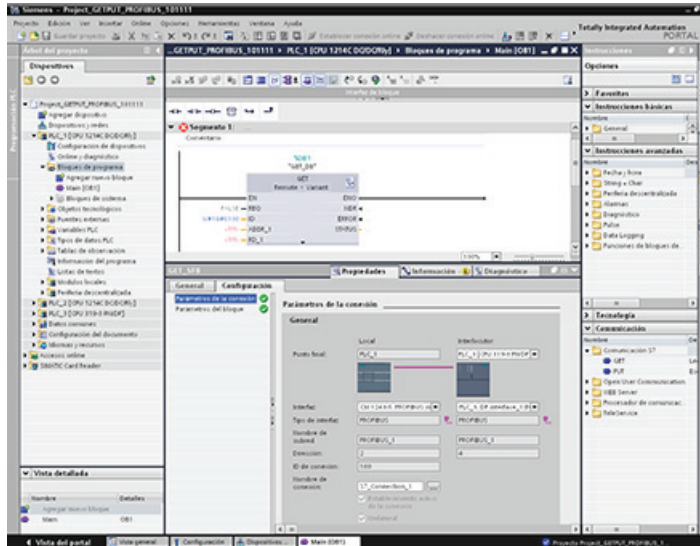
Conexión S7 PROFINET completada

En la "Vista de redes" se muestra una conexión S7 bilateral en la tabla "Conexiones" entre "PLC_1" y "PLC_3".



Configurar una conexión S7 PROFIBUS

Para el "Punto final del interlocutor", seleccione "PLC_3".



El sistema reacciona con los cambios siguientes:

Tabla 10- 47 Parámetro de conexión: valores generales

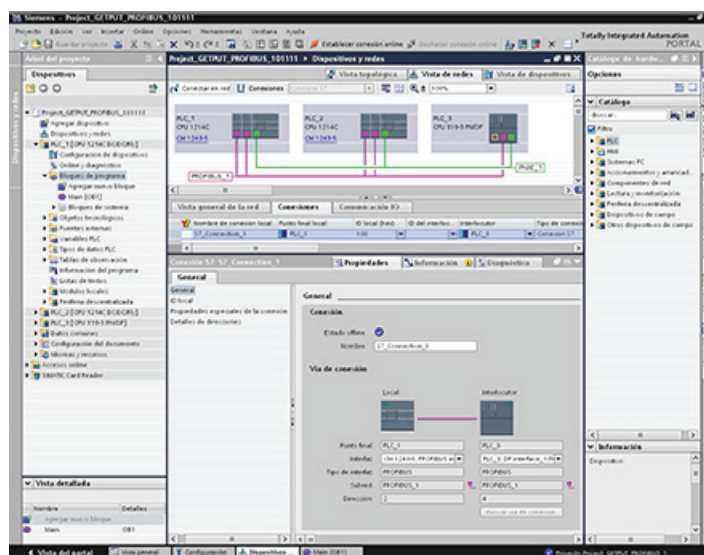
Parámetro	Definición
Parámetro de conexión: General	<p>Punto final</p> <p>"Punto final local" contiene "PLC_1" de solo lectura. El campo "Punto final del interlocutor" contiene "PLC_3[CPU319-3PN/DP]":</p> <ul style="list-style-type: none"> El color cambia de rojo a blanco. Aparece la imagen del dispositivo "interlocutor". Aparece una línea de conexión entre las imágenes de los dispositivos PLC_1 y PLC_3 (línea PROFIBUS púrpura).
Interfaz	<p>La "Interfaz local" contiene "CPU1214C DC/DC/DC, interfaz PROFIBUS (R0/S1)".</p> <p>La "Interfaz del interlocutor" contiene: "CPU319-3PN/DP, interfaz PROFIBUS (R0/S2)".</p>
Tipo de interfaz	<p>El "Tipo de interfaz local" contiene "PROFIBUS"; el control es de solo lectura.</p> <p>El "Tipo de interfaz del interlocutor" contiene "PROFIBUS"; el control es de solo lectura.</p> <p>Las imágenes del tipo de interfaz se muestran a la derecha, junto al "Tipo de interfaz" local y del interlocutor (icono PROFIBUS púrpura).</p>
Nombre de subred	<p>El "Nombre de subred local" contiene "PROFIBUS_1"; el control es de solo lectura.</p> <p>El "Nombre de subred del interlocutor" contiene "PROFIBUS_1"; el control es de solo lectura.</p>
Dirección	<p>La "Dirección local" contiene la dirección IP local; el control es de solo lectura.</p> <p>La "Dirección del interlocutor" contiene la dirección IP del interlocutor; el control es de solo lectura.</p>

Parámetro	Definición
ID de conexión	"ID de conexión" contiene "100". En el editor de programas, en el [OB1] principal, el valor "ID de conexión" del bloque GET/PUT también contiene "100".
Nombre de conexión	El "Nombre de conexión" contiene el nombre predeterminado de la conexión (por ejemplo, "S7_Connection_1"); el control está habilitado.
Iniciativa local	De solo lectura, marcada y habilitada para seleccionar la CPU local como conexión activa.
Unilateral	De solo lectura y marcada. Nota: "PLC_3" (una CPU S7-300 319-3PN/DP) solo puede actuar de servidor (no puede ser cliente) en una conexión GET/PUT PROFIBUS, lo que permite únicamente una conexión de sentido único.

El icono GET/PUT del árbol Vista de propiedades también cambia de rojo a verde.

Conexión S7 PROFIBUS completada

En la "Vista de redes" se muestra una conexión S7 unilateral en la tabla "Conexiones" entre "PLC_1" y "PLC_3".



Servidor web

El servidor web para el S7-1200 ofrece a la página web acceso a datos de la CPU y datos de proceso de la CPU.

Páginas web estándar

El S7-1200 incluye páginas web estándar a las que se accede desde el navegador web del PC (Página 525):

- Introduction (Página 530) - punto de acceso a las páginas web estándar
- Start Page (Página 531) - información general sobre la CPU
- Identification (Página 532) - información detallada sobre la CPU incluyendo la referencia, el número de serie y el número de versión
- Module Information (Página 533) - información sobre los módulos en el rack local
- Communication (Página 535) - información sobre las direcciones de las redes, propiedades físicas de las interfaces de comunicación y una estadística de la comunicación
- Diagnostic Buffer (Página 532) - el búfer de diagnóstico
- Variable Status (Página 537) - E/S y variables de la CPU, accesibles con la dirección o el nombre de variable PLC
- Data Logs (Página 539) - archivos de registro almacenados internamente en la CPU o en una Memory Card
- Update Firmware (Página 541) - actualiza el firmware de la CPU

Estas páginas están integradas en el S7-1200. Para más información acerca de las páginas web estándar y cómo acceder a las mismas, consulte el apartado Páginas web estándar (Página 525).

Páginas web definidas por el usuario

El S7-1200 también ofrece ayuda para crear páginas web definidas por el usuario que pueden acceder a los datos de la CPU. Es posible desarrollar estas páginas con el software de diseño HTML de su elección, incluyendo comandos "AWP" (Automation Web Programming) predefinidos en su código HTML para acceder a los datos de la CPU. Consulte el apartado Páginas web definidas por el usuario (Página 543) para obtener información detallada sobre el desarrollo de páginas web definidas por el usuario y sobre la configuración y programación correspondiente en STEP 7.

Requisitos de navegador web

Los siguientes navegadores web soportan el servidor web:

- Internet Explorer 8.0 o superior
- Mozilla Firefox 3.0 o superior
- Opera 11.0 o superior


En la sección Limitaciones (Página 583) aparecen las restricciones que rigen para navegadores web que pueden interferir con la visualización de páginas web definidas por el usuario.

11.1 Habilitar el servidor web

El servidor web se habilita en STEP 7 desde la configuración de dispositivos de la CPU con la que se desea establecer la conexión.

Para habilitar el servidor web, proceda del siguiente modo:

1. Seleccione la CPU en la vista de configuración de dispositivos.
2. En la ventana de inspección, seleccione "Servidor web" de las propiedades de la CPU.
3. Seleccione la casilla de verificación para "Activar servidor web en el módulo".
4. Si requiere un acceso seguro al servidor web, seleccione la casilla de verificación "Permitir acceso sólo con HTTPS" (Permitir acceso sólo con HTTPS).

 ADVERTENCIA
<p>El acceso no autorizado a la CPU o el cambio de variables de PLC a valores no válidos puede afectar al funcionamiento del proceso y puede producir la muerte, lesiones corporales graves y/o daños materiales.</p> <p>Puesto que la habilitación del servidor web permite a usuarios "admin" realizar cambios de modo de operación, escrituras en datos de PLC y actualizaciones de firmware, Siemens recomienda observar estas prácticas de seguridad:</p> <ul style="list-style-type: none">• Permitir el acceso al servidor web sólo con el protocolo HTTPS.• Proteger con contraseña la CPU (Página 168) utilizando una contraseña segura. Las contraseñas seguras tienen ocho caracteres como mínimo, constan de letras, números y caracteres especiales, no son palabras que puedan encontrarse en un diccionario, ni son nombres ni identificadores que puedan derivarse a partir de información personal. Mantenga la contraseña en secreto y cámbiela con frecuencia.• Realice comprobación de errores y de rango de las variables usadas en la lógica del programa, puesto que los usuarios de páginas web pueden cambiar variables de PLC a valores no válidos.

Después de descargar la configuración del dispositivo es posible utilizar las páginas web estándar para acceder a la CPU. Si se selecciona "Enable" para "Automatic update", las páginas web estándar se actualizan cada diez segundos.

Si ha creado páginas web definidas por el usuario, es posible acceder a éstas desde el menú de la página web estándar.

Nota

Si se está ejecutando una "Carga en RUN" (Página 728), las páginas web estándar y personalizadas no actualizan valores de datos ni permiten escribir valores de datos hasta que no haya finalizado la carga. Cualquier intento de escribir valores de datos durante la carga es rechazado.

11.2 Páginas web estándar

11.2.1 Acceder a las páginas web estándar desde el PC

Para acceder a las páginas web estándar del S7-1200, proceda del siguiente modo:

1. Asegúrese de que el S7-1200 y el PC se encuentren dentro de una red Ethernet común o que estén interconectados directamente con un cable Ethernet estándar.
2. Abra un navegador web e introduzca la URL "http://ww.xx.yy.zz", siendo "ww.xx.yy.zz" la dirección IP de la CPU S7-1200.

El navegador web abre la página Introduction.

Nota

Si el acceso a Internet impide la conexión directa a una dirección IP, consulte al administrador de TI. Es posible que su entorno web o sistema operativo también imponga otras restricciones (Página 583).

Alternativamente, el navegador web también se puede direccionar a una página web estándar específica. Para tal fin, introduzca la URL del siguiente modo:

"http://ww.xx.yy.zz/<page>.html", siendo <page> una de las páginas web estándar:

- start (Página 531) - información general sobre la CPU
- identification (Página 532) - información detallada sobre la CPU incluyendo la referencia, el número de serie y el número de versión
- module (Página 533) - información sobre los módulos en el rack local
- communication (Página 535) - información sobre las direcciones de las redes, propiedades físicas de las interfaces de comunicación y una estadística de la comunicación
- diagnostic (Página 532) - el búfer de diagnóstico
- variable (Página 537) - E/S y variables de la CPU, accesibles con la dirección o el nombre de variable PLC

- datalog (Página 539) - archivos de registros almacenados internamente en la CPU o en una Memory Card
- updatefirmware (Página 541) - página que permite actualizar el firmware de la CPU desde un archivo
- index (Página 530) - página de introducción para acceder a las páginas web estándar

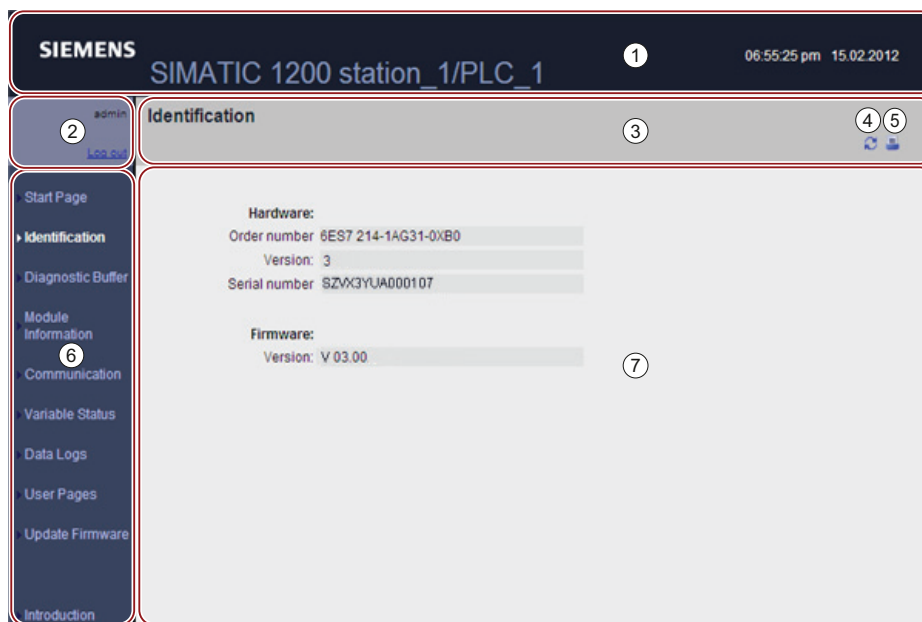
Por ejemplo, si se introduce "http://ww.xx.yy.zz/communication.html", el navegador mostrará la página Communication.

Acceso seguro

Se puede utilizar https:// en vez de http:// para acceder de modo seguro a las páginas web estándar. Al conectar al S7-1200 con https://, el sitio web cifra la sesión con un certificado digital. Los datos se transmiten de forma segura y no están accesibles para quien quiera verlos. Por lo general, aparece una advertencia de seguridad indicando que confirmando con "Sí" se puede acceder a las páginas web estándar. Para evitar que la advertencia de seguridad aparezca en cada acceso seguro, es posible importar el certificado de software de Siemens a su navegador web (Página 586).

11.2.2 Representación de las páginas web estándar

Todas las páginas web estándar tienen una estructura común con enlaces de navegación y controles de página, como aparece a continuación:



- ① Encabezado del servidor web
- ② Iniciar o cerrar sesión
- ③ Encabezado estándar de la página web con el nombre de la página visualizada. Este ejemplo muestra la página de identificación de la CPU. Algunas de las páginas web estándar, como la de información de módulo, también muestran aquí una ruta de navegación si se puede acceder a varias pantallas de ese tipo.
- ④ Icono de actualización: para páginas con actualización automática, activa o desactiva la función de actualización automática; en páginas sin actualización automática actualiza la página con datos actuales
- ⑤ Icono de impresión: prepara y visualiza una versión de impresión de la información disponible en la página visualizada
- ⑥ Área de navegación para pasar a otra página
- ⑦ Área de contenido de la página web estándar específica visualizada. Este ejemplo muestra la página Identification de la CPU.

Nota

Imprimir páginas web estándar

Al imprimir el contenido de una página web estándar, tenga en cuenta que el contenido impreso puede diferir en algunos casos de la página visualizada. Así, por ejemplo, es posible que una copia impresa del búfer de diagnóstico contenga entradas de diagnóstico nuevas que no aparecen en la visualización de la página Diagnostic buffer. Si la actualización automática no está habilitada, la visualización de la página muestra los eventos de diagnóstico que existían en el momento en que se visualizó la página inicialmente, mientras que la copia impresa contiene los eventos de diagnóstico que había en el momento de ejecutar la función de impresión.

Inicio de sesión

No es necesario iniciar sesión para ver los datos en las páginas web estándar. Para ejecutar ciertas acciones como cambiar el modo de operación del controlador, escribir valores en la memoria o actualizar el firmware de la CPU es necesario iniciar sesión como "admin".




El cuadro de inicio de sesión se encuentra cerca de la esquina izquierda superior de cada página.

Para iniciar sesión como "admin", proceda del siguiente modo:

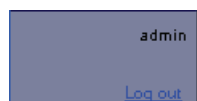
1. Introduzca "admin" en el campo Nombre.
2. Introduzca la contraseña de la CPU si ha configurado una en el campo "Contraseña", de lo contrario, pulse la tecla Intro.

De este modo, habrá iniciado sesión como "admin".

 ADVERTENCIA
<p>El acceso no autorizado a la CPU o el cambio de variables de PLC a valores no válidos puede afectar al funcionamiento del proceso y puede producir la muerte, lesiones corporales graves y/o daños materiales.</p> <p>Puesto que la habilitación del servidor web permite a usuarios "admin" realizar cambios de modo de operación, escrituras en datos de PLC y actualizaciones de firmware, Siemens recomienda observar estas prácticas de seguridad:</p> <ul style="list-style-type: none">• Permitir el acceso al servidor web sólo con el protocolo HTTPS.• Proteger con contraseña la CPU (Página 168) utilizando una contraseña segura. Las contraseñas seguras tienen ocho caracteres como mínimo, constan de letras, números y caracteres especiales, no son palabras que puedan encontrarse en un diccionario, ni son nombres ni identificadores que puedan derivarse a partir de información personal. Mantenga la contraseña en secreto y cámbiela con frecuencia.• Realice comprobación de errores y de rango de las variables usadas en la lógica del programa, puesto que los usuarios de páginas web pueden cambiar variables de PLC a valores no válidos.

Si se producen errores al iniciar sesión, vuelva a la página Introducción (Página 530) y descargue el certificado de seguridad de Siemens (Página 586). A continuación podrá iniciar una sesión sin errores.

Cierre de sesión



Para cerrar la sesión "admin", simplemente haga clic en el link "Cerrar sesión" en cualquiera de las páginas.

Se puede acceder y observar las páginas web estándar sin haber iniciado sesión, pero no es posible ejecutar las acciones que están restringidas a "admin". La descripción de todas las páginas web estándar especifica las acciones que requieren un inicio de sesión como "admin", si las hubiera.

11.2.3 Introducción

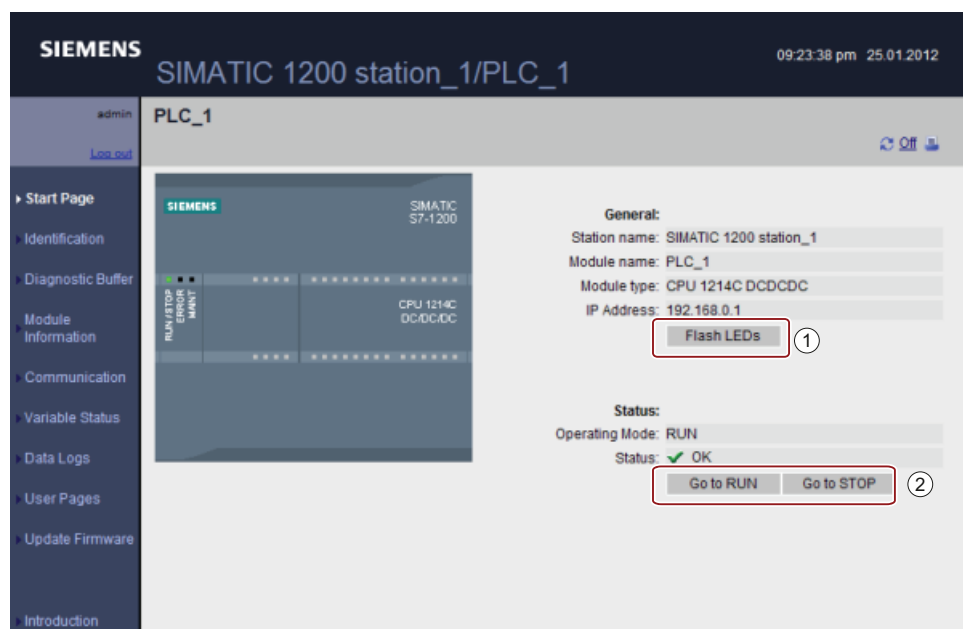
La página de introducción es la pantalla de bienvenida de las páginas web estándar del S7-1200.



Desde esta página, haga clic en "Intro" para acceder a las páginas web estándar S7-1200. En la parte superior de la pantalla hay enlaces de sitios web afines de Siemens, así como un enlace para cargar el certificado de seguridad de Siemens (Página 586).

11.2.4 Arranque

La página de arranque muestra una representación de la CPU con la que se ha realizado la conexión e indica la información general sobre la CPU. Si el usuario inicia sesión como "admin", también puede modificar el modo de operación de la CPU y encender los LED.

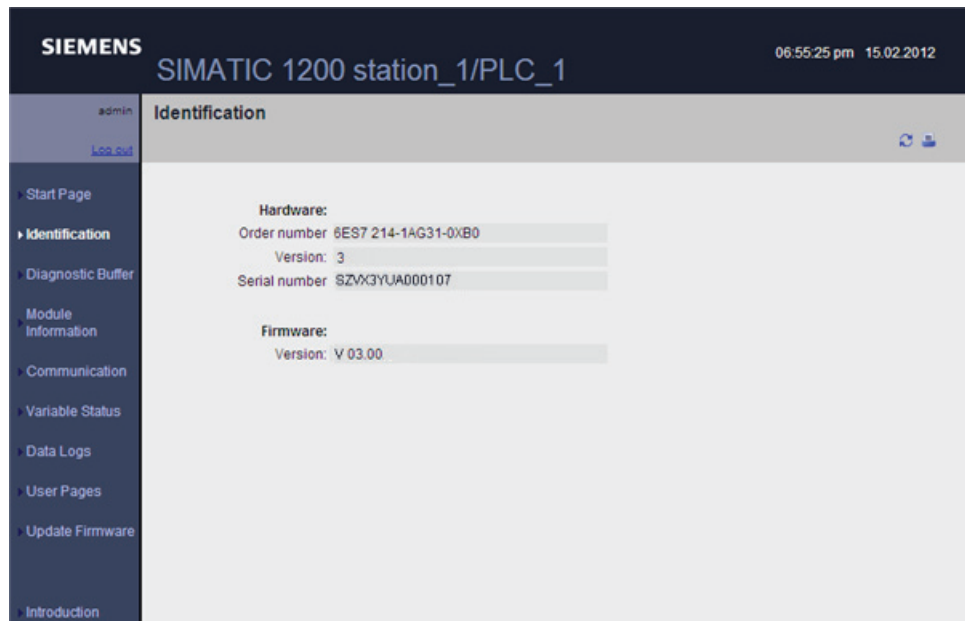


① y ② Los botones para encender los LED y cambiar el modo de operación sólo aparecen en la página de arranque si el usuario ha iniciado sesión como "admin".

11.2.5 Identificación

La página de identificación muestra características identificativas de la CPU:

- Número de serie
- Referencias
- Información de la versión



La página de identificación no cambia con un inicio de sesión como "admin".

11.2.6 Búfer de diagnóstico

La página Diagnostic buffer muestra los eventos de diagnóstico. Con el selector se puede definir qué rango de entradas del búfer de diagnóstico debe visualizarse, es decir, 1 - 25 ó 26 - 50. La parte superior de la página muestra las entradas con la fecha y la hora de la CPU indicando el instante en que ocurrieron los eventos. Las horas son horas del sistema obtenidas del reloj en tiempo real (Página 88) de la CPU.

En la parte superior de la página se puede seleccionar cualquier entrada individual y visualizar información detallada de la misma en la parte inferior de la página.

Number	Time	Date	Event
1	01:55:06.490 pm	26.01.2012	CPU info: Follow-on operating mode change
2	01:55:06.488 pm	26.01.2012	CPU info: Communication initiated request: WARM RESTART
3	01:55:06.488 pm	26.01.2012	CPU info: New startup information
4	01:54:57.888 pm	26.01.2012	CPU info: New startup information
5	01:54:57.788 pm	26.01.2012	CPU info: Communication initiated request: STOP
6	08:53:29:180 pm	25.01.2012	CPU info: Follow-on operating mode change
7	08:53:29:178 pm	25.01.2012	CPU info: Communication initiated request: WARM RESTART
8	08:53:29:178 pm	25.01.2012	CPU info: New startup information
9	08:53:26:978 pm	25.01.2012	CPU info: New startup information
10	08:53:23:478 pm	25.01.2012	CPU info: New startup information
11	08:53:22:262 pm	25.01.2012	CPU info: New startup information
12	08:53:22:162 pm	25.01.2012	CPU info: Communication initiated request: STOP
13	08:52:43:600 pm	25.01.2012	CPU info: Follow-on operating mode change

La página Diagnostic buffer no cambia con un inicio de sesión como "admin".

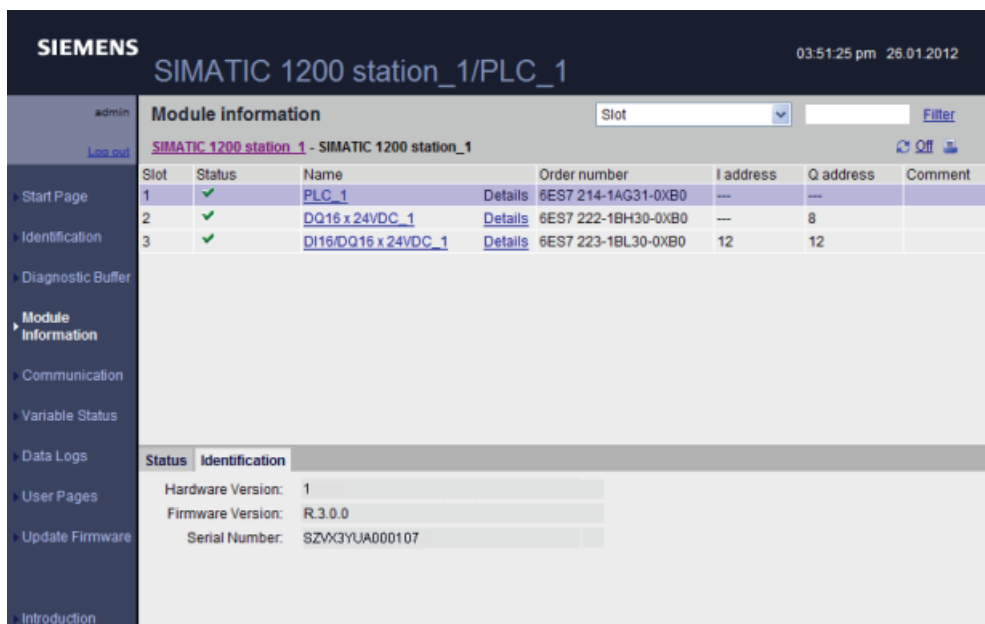
11.2.7 Información del módulo

La página de información del módulo ofrece información sobre todos los módulos en el rack local. La sección superior de la pantalla muestra un resumen de los módulos y la sección inferior muestra el estado y la identificación del módulo seleccionado.

Indicador de estado

Slot	Status	Name	Order number	I address	Q address	Comment
1	✓	PLC_1	6ES7 214-1AG31-0XB0	---	---	
2	✓	DQ16 x 24VDC_1	6ES7 222-1BH30-0XB0	---	8	
3	✓	DI16/DQ16 x 24VDC_1	6ES7 223-1BL30-0XB0	12	12	

Indicador de identificación

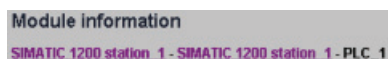


Navegación jerárquica

Se puede seleccionar un enlace en la sección superior para navegar hasta la información de ese módulo en particular. Los módulos con submódulos tienen enlaces para cada submódulo. El tipo de información que se visualiza varía en función del módulo seleccionado. Por ejemplo, el cuadro de diálogo de información del módulo muestra inicialmente el nombre de la estación SIMATIC 1200, un indicador de estado y un comentario. Si se navega hasta la CPU, la información del módulo muestra el nombre de las entradas/salidas digitales y analógicas que ofrece el modelo de CPU (p. ej. "DI14/DO10", "AI2"), información de direccionamiento de las E/S, indicadores de estado, números de slot y comentarios.

Slot	Status	Name	Order number	I address	Q address	Comment
1.16	✓	HSC_1	6ES7 214-1AG31-0XB0	1000	---	
1.17	✓	HSC_2	6ES7 214-1AG31-0XB0	1004	---	
1.18	✓	HSC_3	6ES7 214-1AG31-0XB0	1008	---	
1.19	✓	HSC_4	6ES7 214-1AG31-0XB0	1012	---	
1.20	✓	HSC_5	6ES7 214-1AG31-0XB0	1016	---	
1.21	✓	HSC_6	6ES7 214-1AG31-0XB0	1020	---	
1.2	✓	AI2_1	6ES7 214-1AG31-0XB0	64	---	
1.1	✓	DI14/DO10_1	6ES7 214-1AG31-0XB0	0	0	
1.32	✓	Pulse_1	6ES7 214-1AG31-0XB0	---	1000	
1.33	✓	Pulse_2	6ES7 214-1AG31-0XB0	---	1002	

Al navegar, la página de información del módulo muestra la ruta que se ha seguido. Se puede hacer clic en cualquier enlace de esta ruta para regresar a un nivel superior.



Clasificación de campos

Cuando la lista muestra varios módulos, se puede hacer clic en el encabezado de la columna de un campo para clasificarlo en orden ascendente o descendente.

Name ▼	
AJ2_1	Details
DI14/DQ10_1	Details
HSC_1	Details
HSC_2	Details
HSC_3	Details
HSC_4	Details
HSC_5	Details
HSC_6	Details
Pulse_1	Details
Pulse_2	Details

Filtrar la información del módulo

Es posible filtrar cualquier campo de la lista de información del módulo. En la lista desplegable seleccione el nombre del campo para el que desea filtrar los datos. Introduzca texto en el cuadro de texto asociado y haga clic en el enlace "Filter". La lista se actualiza para mostrar módulos que se correspondan con los criterios de filtrado.

Información de estado

La ficha Estado ubicada en la sección inferior de la página de información del módulo muestra una descripción del estado actual del módulo seleccionado en la sección superior.

Identificación

La ficha Identificación muestra el número de serie y los números de revisión del módulo seleccionado.

La página de información del módulo no cambia con un inicio de sesión como "admin".

11.2.8 Comunicación

La página de comunicación muestra los parámetros de la CPU conectada y una estadística de la comunicación. La ficha Parámetros muestra la dirección MAC de la CPU, la dirección IP y la configuración IP de la CPU, así como las propiedades físicas. La ficha Estadística muestra una estadística de envío y recepción de la comunicación.

Comunicación: indicador de parámetros



Comunicación: indicador de estadísticas



La página de comunicación no cambia con un inicio de sesión como "admin".

11.2.9 Estado de las variables

La página Variable Status permite ver cualquier dato de memoria o E/S en la CPU. Se puede introducir una dirección directa (como I0.0), un nombre de variable PLC o una variable de un bloque de datos determinado. Para variables de bloque de datos se pone el nombre del bloque entre comillas dobles. Para cada valor de observación se puede seleccionar un formato de visualización de datos. Se puede seguir introduciendo y especificando tantos valores como se desee dentro de los límites de la página. Los valores de observación se visualizan automáticamente y se actualizan por defecto, a menos que se haga clic en el icono "Off" en el área superior derecha de la página. Cuando la actualización automática está desactivada se puede hacer clic en "On" para activarla nuevamente.

Iniciando sesión como "admin" también se pueden modificar valores de datos. Introduzca los valores que desee ajustar en el campo "Modify Value" apropiado. Haga clic en el botón "Go" junto a un valor para escribirlo en la CPU. También es posible introducir varios valores y escribirlos conjuntamente en la CPU, para ello haga clic en "Modify All Values".

The screenshot shows the 'Variable Status' page for a SIMATIC 1200 station. The page title is 'SIMATIC 1200 station_1/PLC_1' and the user is logged in as 'admin'. The page contains a table of variables with the following columns: Address, Display Format, Monitor Value, Modify Value, and Modify. The variables listed are Q0.1, I0.1, Conveyor_speed, Mixer_on, Start_Mixer, MW10, "Data_block_1".flag1, "Data_block_1".location, and Tag_1. The Modify Value column contains input fields for each variable, and the Modify column contains 'Go' buttons. A red box highlights the Modify Value and Modify columns, with a circled '1' next to it. A 'Modify All Values' button is located at the bottom right of the table area.

Address	Display Format	Monitor Value	Modify Value	Modify
Q0.1	BOOL	true	<input type="text"/>	Go
I0.1	BOOL	false	<input type="text"/>	Go
Conveyor_speed	DEC	145	<input type="text"/>	Go
Mixer_on	BOOL	false	<input type="text"/>	Go
Start_Mixer	BOOL	false	<input type="text"/>	Go
MW10	DEC	12	<input type="text"/>	Go
"Data_block_1".flag1	BIN	2#1	<input type="text"/>	Go
"Data_block_1".location	STRING	-	<input type="text"/>	Go
Tag_1	FLOATING_POINT	0.0	<input type="text"/>	Go
New variable	BIN		<input type="text"/>	

① La función "Modificar valor" sólo es visible y accesible si el usuario ha iniciado sesión como "admin".

Si sale de la página Variable Status y regresa, la página no conserva las entradas. Es posible marcar la página y regresar a la marca para ver las mismas entradas. Si no se marca la página, deberá introducir nuevamente las variables.

Nota

Al utilizar la página Variable Status, tenga en cuenta lo siguiente:

- La página Variable Status no permite modificar una cadena de caracteres de más de 198 caracteres.
 - En caso de utilizar notación exponencial para introducir un valor del tipo de datos Real o LReal en la página Variable Status:
 - Para introducir un valor de número real (Real o LReal) con un exponente positivo (como +3,402823e+25), introduzca el valor en uno de estos formatos:
+3,402823e25
+3,402823e+25
 - Para introducir un valor de número real (Real o LReal) con un exponente negativo, como +3,402823e-25, debe introducir el valor como se indica a continuación:
+3,402823e-25
 - Asegúrese de que la mantisa del valor real en notación exponencial incluye un punto decimal. Si no se incluye un punto decimal, el valor se convertirá en un entero inesperado. Así, por ejemplo, introduzca -1.0e8 en lugar de -1e8.
 - La página Variable Status admite únicamente 15 dígitos para un valor de tipo LReal (independientemente de la posición de la coma). Si se introducen más de 15 dígitos, se genera un error de redondeo.
-

Limitaciones de la página Variable Status:

- En cada página se pueden introducir como máximo 50 variables.
 - El número máximo de caracteres de la URL que se corresponde con la página Variable Status es 2083. La URL que representa la página de variables actual se puede ver en la barra de direcciones del navegador.
 - Para el formato de visualización de caracteres, la página muestra valores hexadecimales si los valores de la CPU son caracteres ASCII no válidos y el navegador no los puede interpretar.
-

Nota

Si el nombre de la variable presenta caracteres especiales que no se admiten como entrada en la página Variable Status, existe la posibilidad de poner el nombre de la variable entre comillas dobles. En la mayoría de los casos, la página Variable Status reconocerá entonces el nombre de la variable.

11.2.10 Registros

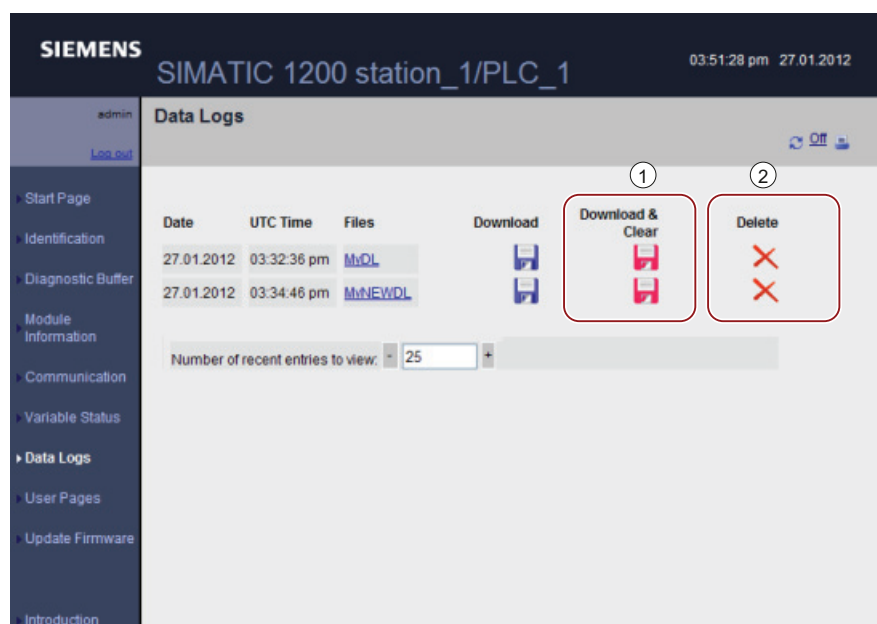
La página Data Logs permite visualizar o cargar un determinado número de entradas de registro. Iniciando sesión como "admin", también es posible borrar o bien eliminar estas entradas después de cargarlas. El servidor web carga registros de datos en el PC en formato de archivo separado por comas (CSV).

La página Data Logs muestra hasta 40 archivos de registro. Si existen más de 40 archivos de registro, la página Data Logs muestra los primeros 40 que se crearon.

Nota

El sello de tiempo para los registros se muestra en la hora del sistema, no en la hora local

La CPU escribe los sellos de tiempo de los registros en la hora del sistema (Página 88) y la página estándar "Data Logs" del servidor web muestra los sellos de tiempo de los registros con la hora del sistema.



- ① La opción "Cargar y borrar" sólo está disponible si el usuario ha iniciado sesión como "admin".
- ② La opción "Eliminar" sólo está disponible si el usuario ha iniciado sesión como "admin".

Nota

El archivo de registro es de formato separado por comas (CSV) para sistemas británicos y estadounidenses. Para abrir el archivo con Excel en sistemas que no son ni estadounidenses ni británicos, debe importar el archivo a Excel aplicando una configuración específica (Página 587).

Entradas recientes: cargar un número concreto de entradas de datos recientes

Especifique el número de entradas recientes que deben cargarse y haga clic en el nombre del registro de datos para iniciar la carga del número de entradas especificado. En el archivo de resultados .csv, las entradas están ordenadas por orden descendente de entrada. Windows le pedirá que abra o guarde el archivo de registro.

De forma predeterminada se visualizan 25 entradas recientes. Este valor se puede modificar en el campo "Number of recent entries to view" (número máximo de entradas recientes visualizadas) introduciendo un número o utilizando el botón + o - para incrementar o decrementar el valor.

Nota

El parámetro Records de la instrucción DataLogCreate (Página 330) especifica el número máximo de entradas de un archivo de registro.

Cargar un archivo de registro que contiene todas las entradas de datos

Para cargar un archivo de registro completo, haga clic en el icono de carga correspondiente al archivo de registro deseado. Windows le pedirá que abra o guarde el archivo de registro.

En el archivo de resultados .csv se incluyen todas las entradas, en orden ascendente por número de entrada, a no ser que el registro esté lleno y las entradas más antiguas (número de entrada más bajo) hayan sido sobrescritas por entradas posteriores (número de entrada más alto).

Cargar y borrar un archivo de registro

Para cargar un archivo de registro y luego borrar todas las entradas, el usuario debe haber iniciado sesión como "admin". A continuación haga clic en el icono "Cargar y borrar" correspondiente al archivo de registro deseado. Windows le pedirá que abra o guarde el archivo de registro.

Una vez finalizada la carga, se inserta una línea "//END" nueva después de la entrada del encabezado del archivo de registro almacenado en el PLC. De este modo se borra efectivamente el registro de datos para el futuro procesamiento interno del PLC, aunque las cargas posteriores de dicho archivo provocarán la inserción de nuevas entradas antes de la primera línea "//END".

Nota

Marca "//END" del archivo de registro .csv

La marca de fin "//END" del archivo .csv sólo se utiliza para las primeras entradas ((entradas máx.) -1) con el fin de señalar el fin lógico del archivo. Detrás del fin lógico, el archivo puede contener datos que Excel puede interpretar como entradas adicionales. Es recomendable buscar el primer "//END" y borrarlo junto con todas las entradas que haya debajo. Si no hay ninguna marca de fin lógico, es posible clasificar las filas de datos por el número de entrada.

Eliminar un archivo de registro

Para eliminar un archivo de registro, el usuario debe haber iniciado sesión como "admin". A continuación haga clic en el icono Borrar correspondiente al archivo de registro deseado. El servidor web elimina el archivo de registro seleccionado.

Información adicional

Encontrará información sobre la programación con instrucciones de registro de datos en el apartado Registrar datos (Página 328).

11.2.11 Update Firmware

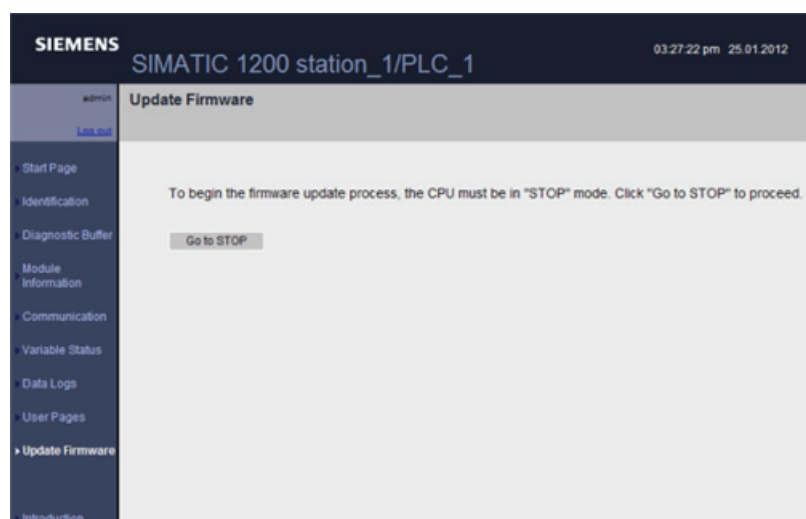
La página Update Firmware permite a un usuario "admin" actualizar el firmware de la CPU desde un archivo.

Nota

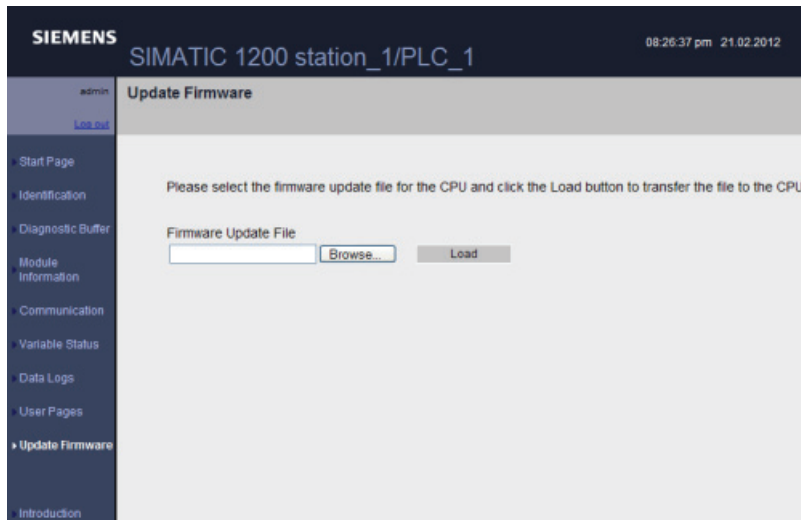
Con la función Actualizar el firmware sólo se pueden actualizar CPU S7-1200 de la versión 3.0 o posteriores.

El servidor web utiliza el protocolo "https" para actualizar el firmware.

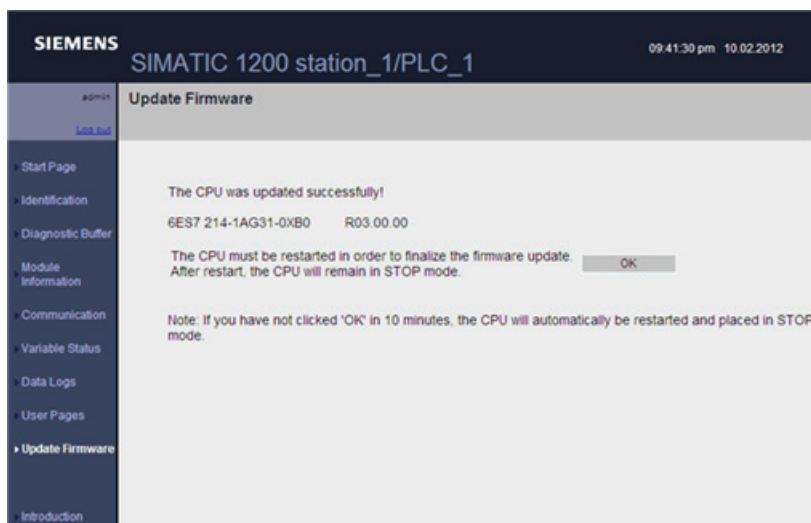
Para actualizar el firmware, la CPU debe estar en modo STOP. Si la CPU no está en modo STOP, el servidor web solicita que se conmute la CPU a modo STOP.



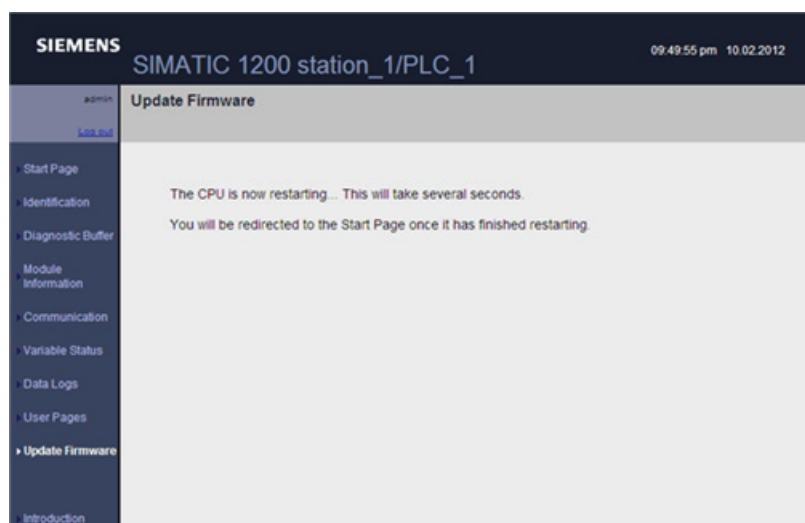
Cuando la CPU esté en modo STOP, se puede seleccionar un archivo desde el que cargar la actualización de la versión de firmware. Las actualizaciones de firmware están disponibles en el sitio web del Customer Support (<http://support.automation.siemens.com>). Una vez descargada la actualización de firmware adecuada desde el sitio web al PC, se puede navegar hasta el archivo del PC y seleccionarlo para actualizar la CPU.



Durante la actualización, la página Update Firmware muestra un mensaje que indica que la actualización está en curso. Al finalizar la actualización, la página Update Firmware muestra la referencia y el número de versión del firmware actualizado y le solicita que permita un reinicio de la CPU.



Si no se responde en diez minutos, la CPU se reinicia automáticamente.

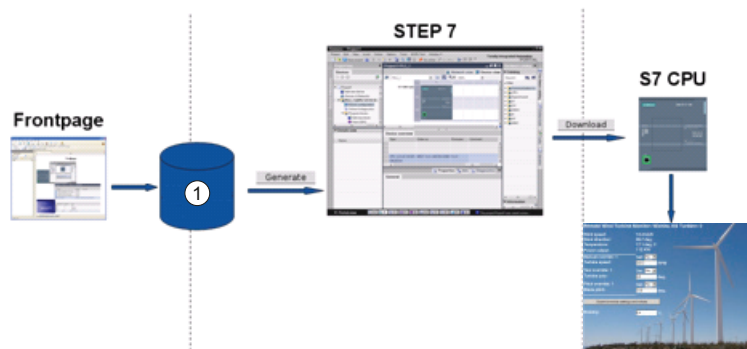


11.3 Páginas web definidas por el usuario

El servidor web de S7-1200 también ofrece medios para crear páginas HTML específicas para la aplicación que incorporan datos del PLC. Cree estas páginas utilizando el editor de HTML deseado y cárguelas en la CPU en la que serán accesibles desde el menú de la página web estándar. Este proceso incluye varias tareas:

- Crear páginas HTML con un editor de HTML como Microsoft Frontpage. (Página 544)
- Incluir comandos AWP en comentarios HTML con el código HTML (Página 545): los comandos AWP son un conjunto fijo de comandos que suministra Siemens para acceder a la información de la CPU.
- Configurar STEP 7 para leer y procesar las páginas HTML. (Página 560)
- Generar bloques desde las páginas HTML (Página 560)
- Programar STEP 7 para controlar el uso de las páginas HTML (Página 561)
- Compilar y cargar los bloques en la CPU (Página 563)
- Acceder a las páginas web definidas por el usuario desde el PC (Página 563)

Este proceso se describe a continuación:



① Archivos HTML con comandos AWP incrustados

11.3.1 Creación de páginas HTML

El paquete de software deseado puede utilizarse para crear páginas HTML con el fin de usarlas con el servidor web. Hay que asegurarse de que el código HTML cumpla los estándares HTML de W3C (World Wide Web Consortium). STEP 7 no realiza verificación alguna de la sintaxis HTML.

Se puede usar un paquete de software que permita diseñar en WYSIWYG o en modo de esquema de trazado, pero tiene que ser posible editar el código HTML en HTML puro. La mayor parte de las herramientas de diseño web ofrece este tipo de edición; si no es así, siempre se puede usar un simple editor de textos para editar el código HTML. Incluya la línea siguiente en su página HTML para configurar la fuente para la página a UTF-8:

```
<meta http-equiv="content-type" content="text/html; charset=utf-8">
```

Asegúrese también de guardar el archivo desde el editor con codificación de caracteres UTF-8:

Utilice STEP 7 para compilar todo en las páginas HTML en bloques de datos de STEP 7. Estos bloques de datos constan de un bloque de datos de control que controla la visualización de las páginas web, así como de uno o varios bloques de datos de fragmentos que contienen las páginas web compiladas. Hay que tener en cuenta que los conjuntos extendidos de páginas HTML, en especial los que contienen una gran cantidad de imágenes, requieren mucho espacio de memoria de carga (Página 564) para los DB de fragmentos. Si la memoria de carga interna de la CPU no es suficiente para las páginas web definidas por el usuario, se puede utilizar una Memory Card (Página 111) para ofrecer memoria de carga externa.

Para programar el código HTML con el objetivo de usar datos del S7-1200, hay que incluir comandos AWP (Página 545) como comentarios HTML. Al acabar, guarde las páginas HTML en el PC y anote la ruta de carpetas en la que se han guardado.

Actualizar páginas web definidas por el usuario

Las páginas web definidas por el usuario no se actualizan automáticamente. Es decisión del usuario si desea programar o no el HTML para actualizar la página. En páginas que muestran datos del PLC resulta muy útil actualizar los datos periódicamente. En páginas HTML que sirven para introducir datos, la actualización puede interferir con los datos que introduce el usuario. Si desea que la página entera se actualice automáticamente, es posible agregar esta línea al encabezado HTML, siendo "10" el número de segundos entre actualizaciones:

```
<meta http-equiv="Refresh" content="10">
```

También se puede utilizar JavaScripts u otras técnicas HTML para controlar la actualización de la página o de los datos. Para tal fin, consulte la documentación acerca de HTML y JavaScript.

11.3.2 Comandos AWP soportados por el servidor web del S7-1200

El servidor web del S7-1200 ofrece comandos AWP que se incrustan en las páginas web definidas por el usuario como comentarios HTML para los fines siguientes:

- Leer variables (Página 546)
- Escribir variables (Página 547)
- Leer variables especiales (Página 549)
- Escribir variables especiales (Página 551)
- Definir tipos de enumeración (Página 553)
- Asignar variables a tipos de enumeración (Página 554)
- Crear bloques de datos de fragmentos (Página 555)

Sintaxis general

A excepción del comando para leer una variable, los comandos AWP tienen la sintaxis siguiente:

```
<!-- AWP_ <command name and parameters> -->
```

Utilice los comandos AWP en combinación con comandos típicos de formas HTML para escribir en variables de la CPU.

Las descripciones de los comandos AWP en las páginas siguientes utilizan las convenciones siguientes:

- Los elementos encerrados en corchetes [] son opcionales.
- Los elementos entre corchetes angulares < > son valores de parámetros que deben especificarse.
- Las comillas son parte literal del comando. Deben estar presentes tal y como se indica.
- Los caracteres especiales en nombres de variables o de bloques de datos, dependiendo de su uso, deben escaparse o encerrarse entre comillas (Página 557).

Utilice un editor de textos o un modo de edición HTML para insertar los comandos AWP en las páginas.

Resumen de comandos AWP

Los detalles para utilizar cada comando AWP están reunidos en los apartados siguientes, no obstante aquí se muestra un resumen corto de los comandos:

Leer variables

`:=<Varname>:`

Escribir variables

`<!-- AWP_In_Variable Name='<Varname1>' [Use='<Varname2>'] ... -->`

Este comando AWP sólo declara que la variable en la cláusula de nombre es escribible. El código HTML permite operaciones de escritura en la variable a partir del nombre mediante `<input>`, `<select>` u otras instrucciones HTML dentro de una forma HTML.

Leer variables especiales

`<!-- AWP_Out_Variable Name='<Type>:<Name>' [Use='<Varname>'] -->`

Escribir variables especiales

`<!-- AWP_In_Variable Name='<Type>:<Name>' [Use='<Varname>'] -->`

Definir tipos de enumeración

`<!--
AWP_Enum_Def Name='<Enum type name>' Values='<Value>, <Value>, ... '
-->`

Referenciar tipos de enumeración

`<!-- AWP_Enum_Ref Name='<VarName>' Enum='<EnumType>' -->`

Crear fragmentos

`<!-- AWP_Start_Fragment Name='<Name>' [Type=<Type>] [ID=<id>] -->`

Importar fragmentos

`<!-- AWP_Import_Fragment Name='<Name>' -->`

11.3.2.1 Leer variables

Las páginas web definidas por el usuario pueden leer variables (variables PLC) de la CPU.

Sintaxis

`:=<Varname>:`

Parámetros

<code><Varname></code>	La variable que debe leerse puede ser un nombre de variable PLC del programa STEP 7, una variable de bloque de datos, E/S o de la memoria direccionable. En las direcciones de memoria o de E/S o en los alias (Página 557) no utilice comillas para el nombre de la variable. Para variables PLC utilice comillas dobles para el nombre de la variable. Para variables de bloque de datos ponga el nombre del bloque sólo entre comillas dobles. El nombre de la variable debe quedar por fuera de las comillas. Observe que debe utilizarse el nombre y no el número del bloque de datos.
------------------------------	---

Ejemplos

```
:= "Conveyor_speed" := "My_Data_Block".flag1:  
:= I0.0:  
:= MW100:
```

Ejemplo de lectura de una variable con alias

```
<!-- AWP_Out_Variable Name='flag1' Use=' "My_Data_Block".flag1' -->  
:= flag1:
```

Nota

Encontrará información sobre cómo definir alias para variables PLC y variables de bloque de datos en el apartado Utilizar un alias para una referencia de variable (Página 552).

Si el nombre de la variable o del bloque de datos incluye caracteres especiales hay que utilizar comillas adicionales o caracteres de escape, tal como se describe en el apartado Procesamiento de nombres de variable que contienen caracteres especiales (Página 557).

11.3.2.2 Escribir variables

Las páginas definidas por el usuario pueden escribir datos en la CPU. Esto se realiza utilizando un comando AWP para identificar una variable en la CPU que sea escribible desde la página HTML. La variable debe especificarse por el nombre de variable PLC o el nombre de variable de bloque de datos. Es posible declarar múltiples nombres de variable en una instrucción. Para escribir los datos en la CPU, utilice el comando estándar HTTP POST.

Una aplicación típica es diseñar una forma en la página HTML con campos de entrada de texto u opciones en una lista de selección que se correspondan con variables de CPU escribibles. Como ocurre con todas las páginas definidas por el usuario, a continuación deben generarse los bloques desde STEP 7, como los que están incluidos en el programa de STEP 7. Cuando un usuario admin accede a esta página e introduce datos en los campos de entrada o selecciona una opción de una lista de selección, el servidor web convierte la entrada al tipo de datos correcto para la variable y escribe el valor en la variable de la CPU. Tenga en cuenta que la cláusula de nombre para campos de entrada HTML y listas de selección HTML utiliza una sintaxis típica para la cláusula de nombre del comando AWP_In_Variable. Por lo general debe ponerse el nombre entre comillas simples y, si se referencia un bloque de datos, el nombre del mismo entre comillas dobles.

Para más detalles sobre la gestión de formas, consulte la documentación de HTML.

Sintaxis

```
<!-- AWP_In_Variable Name='<Varname1>' [Use='<Varname2>'] ... -->
```

Parámetros

<Varname1>	Si no está disponible ninguna cláusula de uso, entonces Varname1 es la variable que debe escribirse. Puede tratarse de un nombre de variable PLC del programa STEP 7 o de una variable de un bloque de datos específico. Si está disponible una cláusula de uso, Varname1 es un nombre alternativo para la variable referenciada en <Varname2> (Página 552). Se trata de un nombre local dentro de la página HTML.
<Varname2>	Si está disponible una cláusula de uso, entonces Varname2 es la variable que debe escribirse. Puede tratarse de un nombre de variable PLC del programa STEP 7 o de una variable de un bloque de datos específico.

Tanto para las cláusulas de nombre como para las cláusulas de uso, el nombre entero debe ponerse entre comillas simples. Dentro de las comillas simples, utilice comillas dobles para una variable PLC y para un nombre de bloque de datos. El nombre del bloque de datos está dentro de las comillas dobles pero no el nombre de la variable de bloque de datos. Observe que en el caso de variables de bloque de datos debe utilizarse el nombre y no el número del bloque de datos.

Ejemplos utilizando un campo de entrada HTML

```
<!-- AWP_In_Variable Name='\"Target_Level\"' -->
<form method="post">
<p>Input Target Level: <input name='\"Target_Level\"' type="text" />
</p>
</form>
```

```
<!-- AWP_In_Variable Name='\"Data_block_1\".Braking' -->
<form method="post">
<p>Braking: <input name='\"Data_block_1\".Braking' type="text" />
%</p>
</form>
```

```
<!-- AWP_In_Variable Name='\"Braking\"' Use='\"Data_block_1\".Braking' -
->
<form method="post">
<p>Braking: <input name='\"Braking\"' type="text" /> %</p>
</form>
```

Ejemplo utilizando una lista de selección HTML

```
<!-- AWP_In_Variable Name='\"Data_block_1\".ManualOverrideEnable'-->
<form method="post">
<select name='\"Data_block_1\".ManualOverrideEnable'>
<option value='\"Data_block_1\".ManualOverrideEnable:> </option>
<option value=1>Yes</option>
<option value=0>No</option>
</select><input type="submit" value="Submit setting" /></form>
```

Nota

Solamente un usuario admin puede escribir datos en la CPU. Los comandos se ignoran si el usuario no ha iniciado sesión como admin.

Si el nombre de la variable o del bloque de datos incluye caracteres especiales hay que utilizar comillas adicionales o caracteres de escape, tal como se describe en el apartado "Procesamiento de nombres de variable que contienen caracteres especiales (Página 557)".

11.3.2.3 Leer variables especiales

El servidor web ofrece la posibilidad de leer valores desde el PC para almacenarlos en variables especiales en el encabezado de respuesta HTTP. Es posible, por ejemplo, que se quiera leer un nombre de ruta desde una variable PLC para redireccionar la URL a otra ubicación utilizando la variable especial HEADER:Location.

Sintaxis

```
<!-- AWP_Out_Variable Name='<Type>:<Name>' [Use='<Varname>'] -->
```

Parámetros

<Type>	Se refiere al tipo de variable especial y puede ser uno de los siguientes: HEADER COOKIE_VALUE COOKIE_EXPIRES
<Name>	En la documentación HTTP encontrará una lista con todos los nombres de variables HEADER. A continuación aparece una lista con algunos ejemplos: Status: código de respuesta Location: ruta para redireccionar Retry-After: tiempo previsto en que el servicio no está disponible para el cliente solicitante Para los tipos COOKIE_VALUE y COOKIE_EXPIRES, <Name> es el nombre de una cookie específica. COOKIE_VALUE:name: valor de la cookie indicada COOKIE_EXPIRES:name: tiempo de vencimiento en segundos de la cookie indicada La cláusula de nombre debe ponerse entre comillas simples o dobles. Si no se ha especificado ninguna cláusula de uso, el nombre de la variable especial corresponde a un nombre de variable PLC. Ponga toda la cláusula de nombre entre comillas simples y la variable PLC entre comillas dobles. El nombre de la variable especial y el de la variable PLC deben coincidir.
<Varname>	Nombre de la variable PLC o variable de bloque de datos para la variable que debe leerse El Varname debe ponerse entre comillas simples. Dentro de las comillas simples, utilice comillas dobles para una variable PLC o un nombre de bloque de datos. El nombre del bloque de datos está dentro de las comillas dobles pero no el nombre de la variable de bloque de datos. Observe que en el caso de variables de bloque de datos debe utilizarse el nombre y no el número del bloque de datos.

Ejemplos

```
<!-- AWP_Out_Variable Name="HEADER:Status" -->
```

En este ejemplo, la variable especial HTTP "HEADER:Status" recibe el valor de la variable PLC "HEADER:Status". El nombre en la tabla de variables PLC debe coincidir exactamente con el nombre de la variable especial si no está especificada ninguna cláusula de uso.

```
<!-- AWP_Out_Variable Name='HEADER:Status' Use='Estado' -->
```

En este ejemplo, la variable especial HTTP "HEADER:Status" recibe el valor de la variable PLC "Status".

Si el nombre de la variable o del bloque de datos incluye caracteres especiales hay que utilizar comillas adicionales o caracteres de escape, tal como se describe en el apartado Procesamiento de nombres de variable que contienen caracteres especiales (Página 557).

11.3.2.4 Escribir variables especiales

El servidor web ofrece la posibilidad de escribir valores en la CPU de variables especiales en el encabezado de peticiones HTTP. Así, por ejemplo, se puede almacenar en STEP 7 información acerca de la cookie asociada a una página web definida por el usuario, acerca del usuario que acceda a la página o a la información del encabezado. El servidor web ofrece acceso a variables especiales específicas que se pueden escribir en la CPU, siempre y cuando el usuario haya iniciado sesión como admin.

Sintaxis

```
<!-- AWP_In_Variable Name='<Type>:<Name>' [Use='<Varname>']-->
```

Parámetros

<Type>	Se refiere al tipo de variable especial y puede ser uno de los siguientes: HEADER SERVER COOKIE_VALUE
<Name>	Variable específica de uno de los tipos definidos previamente, como se muestra en estos ejemplos: HEADER:Accept: tipos de contenido que son aceptables HEADER:User-Agent: información acerca del agente usuario que origina la petición. SERVER:current_user_id: id del usuario actual; 0 si ningún usuario ha iniciado sesión SERVER:current_user_name: nombre del usuario actual COOKIE_VALUE:<name>: valor de la cookie indicada Ponga la cláusula de nombre entre comillas simples. Si no se ha especificado ninguna cláusula de uso, el nombre de la variable especial corresponde a un nombre de variable PLC. Ponga toda la cláusula de nombre entre comillas simples y la variable PLC entre comillas dobles. El nombre de la variable especial y el de la variable PLC deben coincidir. En la documentación HTTP encontrará una lista con todos los nombres de variables HEADER.
<Varname>	El nombre de variable en el programa STEP 7 en que desea escribir la variable especial, la cual puede ser un nombre de variable PLC o una variable de bloque de datos. El Varname debe ponerse entre comillas simples. Dentro de las comillas simples, utilice comillas dobles para una variable PLC o un nombre de bloque de datos. El nombre del bloque de datos está dentro de las comillas dobles pero no el nombre de la variable de bloque de datos. Observe que en el caso de variables de bloque de datos debe utilizarse el nombre y no el número del bloque de datos.

Ejemplos

```
<!-- AWP_In_Variable Name='SERVER:current_user_id' -->
```

En este ejemplo, la página web escribe el valor de la variable especial HTTP "SERVER:current_user_id" en la variable PLC con el nombre "SERVER:current_user_id".

```
<!-- AWP_In_Variable Name=SERVER:current_user_id' Use='my_userid' -->
```

En este ejemplo, la página web escribe el valor de la variable especial HTTP "SERVER:current_user_id" en la variable PLC con el nombre "my_userid".

Nota

Solamente un usuario admin puede escribir datos en la CPU. Los comandos se ignoran si el usuario no ha iniciado sesión como admin.

Si el nombre de la variable o del bloque de datos incluye caracteres especiales hay que utilizar comillas adicionales o caracteres de escape, tal como se describe en el apartado "Procesamiento de nombres de variable que contienen caracteres especiales (Página 557)".

11.3.2.5 Utilizar un alias para una referencia de variable

Se puede utilizar un alias en la página web definida por el usuario para una In_Variable o una Out_Variable. Así, por ejemplo, es posible utilizar un nombre simbólico diferente en su página HTML del utilizado en la CPU o bien equiparar una variable en la CPU con una variable especial. La cláusula de uso AWP ofrece esta posibilidad.

Sintaxis

```
<-- AWP_In_Variable Name='<Varname1>' Use='<Varname2>' -->
<-- AWP_Out_Variable Name='<Varname1>' Use='<Varname2>' -->
```

Parámetros

<Varname1>	El alias o el nombre de la variable especial Varname1 debe ponerse entre comillas simples o dobles.
<Varname2>	Nombre de la variable PLC a la que desea asignar un alias. La variable puede ser una variable PLC, una variable de bloque de datos o una variable especial. El Varname2 debe ponerse entre comillas simples. Dentro de las comillas simples, utilice comillas dobles para una variable PLC, una variable especial o un nombre de bloque de datos. El nombre del bloque de datos está dentro de las comillas dobles pero no el nombre de la variable de bloque de datos. Observe que en el caso de variables de bloque de datos debe utilizarse el nombre y no el número del bloque de datos.

Ejemplos

```
<-- AWP_In_Variable Name='SERVER:current_user_id'
Use=' "Data_Block_10".server_user' -->
```

En este ejemplo, la variable especial SERVER:current_user_id se escribe en la variable "server_user" del bloque de datos "Data_Block_10".

```
<-- AWP_Out_Variable Name='Weight'
Use=' "Data_Block_10".Tank_data.Weight' -->
```

En este ejemplo, el valor del elemento de estructura del bloque de datos Data_Block_10.Tank_data.Weight puede referenciarse simplemente con "Weight" en el resto de la página web definida por el usuario.

```
<-- AWP_Out_Variable Name='Weight' Use=' "Raw_Milk_Tank_Weight"' -->
```

En este ejemplo, el valor de la variable PLC "Raw_Milk_Tank_Weight" puede referenciarse simplemente con "Weight" en el resto de la página web definida por el usuario.

Si el nombre de la variable o del bloque de datos incluye caracteres especiales hay que utilizar comillas adicionales o caracteres de escape, tal como se describe en el apartado Procesamiento de nombres de variable que contienen caracteres especiales (Página 557).

11.3.2.6 Definir tipos de enumeración

Es posible definir tipos de enumeración en las páginas web definidas por el usuario y asignar los elementos en un comando AWP.

Sintaxis

```
<!-- AWP_Enum_Def Name='<Enum type name>' Values='<Value>,
<Value>,... ' -->
```

Parámetros

<Enum type name>	Nombre del tipo de enumeración, entre comillas simples o dobles.
<Value>	<constant>:<name> La constante indica el valor numérico para la asignación del tipo de enumeración. El número total es ilimitado. El nombre es el valor asignado al elemento de enumeración.

Tenga en cuenta que la cadena completa de valores de enumeración asignados está encerrada entre comillas sencillas y cada elemento individual del tipo de enumeración asignado está encerrado entre comillas dobles. El alcance de una definición de tipo de enumeración es global en las páginas web definidas por el usuario. Si ha configurado las páginas web definidas por el usuario en carpetas de lenguaje (Página 576), la definición del tipo de enumeración es global para todas las páginas de la carpeta de lenguaje.

Ejemplo

```
<!-- AWP_Enum_Def Name='AlarmEnum' Values='0:"No alarms", 1:"Tank is full", 2:"Tank is empty"' -->
```

11.3.2.7 Referenciar variables de CPU con un tipo de enumeración

Es posible asignar una variable en la CPU a un tipo de enumeración. Esta variable se puede utilizar en otro lugar de la página web definida por el usuario en una operación de lectura (Página 546) o una operación de escritura (Página 547). En una operación de lectura, el servidor web reemplaza el valor numérico leído desde la CPU por el valor de texto de enumeración correspondiente. En una operación de escritura, el servidor web reemplaza el valor de texto por el valor entero de la enumeración correspondiente al texto antes de escribir el valor en la CPU.

Sintaxis

```
<!-- AWP_Enum_Ref Name='<Varname>' Enum="<EnumType>" -->
```

Parámetros

<Varname>	Nombre de la variable de PLC o de la variable de bloque de datos que se desea asociar con el tipo de enumeración o el alias para una variable de PLC (Página 552), de estar declarado. El Varname debe ponerse entre comillas simples. Dentro de las comillas simples, utilice comillas dobles para una variable PLC o un nombre de bloque de datos. Observe que en el caso de variables de bloque de datos debe utilizarse el nombre y no el número del bloque de datos. El nombre del bloque de datos está dentro de las comillas dobles pero no el nombre de la variable de bloque de datos.
<EnumType>	Nombre del tipo de enumeración que debe ponerse entre comillas simples o dobles

El alcance de una referencia de tipo de enumeración es el fragmento actual.

Ejemplo de declaración

```
<!-- AWP_Enum_Ref Name='Alarm' Enum="AlarmEnum" -->
```

Ejemplo de utilización en una lectura de variable

```
<!-- AWP_Enum_Def Name='AlarmEnum' Values='0:"No alarms", 1:"Tank is full", 2:"Tank is empty"' -->
<!-- AWP_Enum_Ref Name='Alarm' Enum="AlarmEnum" -->
...
<p>The current value of "Alarm" is := "Alarm":</p>
```

Si el valor de "Alarm" en la CPU es 2, la página HTML visualiza 'The current value of "Alarm" is Tank is empty' debido a que la definición del tipo de enumeración (Página 553) asigna la cadena de texto "Tank is empty" al valor numérico 2.

Ejemplo de utilización en una escritura de variable

```

<!-- AWP_Enum_Def Name='AlarmEnum' Values='0:"No alarms", 1:"Tank is
full", 2:"Tank is empty"' -->
<!-- AWP_In_Variable Name='"Alarm"' -->
<!-- AWP_Enum_Ref Name='"Alarm"' Enum='AlarmEnum' -->
...
<form method="POST">
<p><input type="hidden" name='"Alarm"' value="Tank is full" /></p>
<p><input type="submit" value='Set Tank is full' /></p>
</form>

```

Dado que la definición del tipo de enumeración (Página 553) asigna "Tank is full" al valor numérico 1, el valor 1 se escribe en la variable de PLC denominada "Alarm" en la CPU.

Tenga en cuenta que la cláusula de nombre en la declaración AWP_In_Variable debe corresponderse exactamente con la cláusula de nombre en la declaración AWP_Enum_Ref.

Ejemplo de utilización en una escritura de variable aplicando un alias

```

<!-- AWP_Enum_Def Name='AlarmEnum' Values='0:"No alarms", 1:"Tank is
full", 2:"Tank is empty"' -->
<!-- AWP_In_Variable Name='"Alarm"'
Use='Data_block_4.Motor1.Alarm'-->
<!-- AWP_Enum_Ref Name='"Alarm"' Enum='AlarmEnum' -->
...
<form method="POST">
<p><input type="hidden" name='"Alarm"' value="Tank is full" /></p>
<p><input type="submit" value='Set Tank is full' /></p>
</form>

```

Dado que la definición del tipo de enumeración (Página 553) asigna "Tank is full" al valor numérico 1, el valor 1 se escribe en el alias "Alarm" que se corresponde con la variable PLC denominada "Motor1.Alarm" en el bloque de datos "Data_Block_4" de la CPU.

Si el nombre de la variable o del bloque de datos incluye caracteres especiales hay que utilizar comillas adicionales o caracteres de escape, tal como se describe en el apartado Procesamiento de nombres de variable que contienen caracteres especiales (Página 557).

11.3.2.8 Crear fragmentos

STEP 7 convierte y almacena páginas web definidas por el usuario como un DB de control y DBs de fragmentos al hacer clic en "Generar bloques" en las Propiedades de la CPU para el servidor web. Se pueden configurar fragmentos específicos para páginas específicas o para secciones de páginas específicas. Estos fragmentos se pueden identificar a través de un nombre y un número con el comando AWP "Start_Fragment". Todos los elementos de la página que siguen al comando AWP_Start_Fragment pertenecen a ese fragmento hasta que se lance otro AWP_Start_Command o hasta que se alcance el final del archivo.

Sintaxis

```

<!-- AWP_Start_Fragment Name='<Name>' [Type=<Type>] [ID=<id>] -->

```

Parámetros

<Name>	Cadena de texto: nombre del DB de fragmentos Los nombres de fragmentos deben empezar con una letra o un guión bajo y estar formados por letras, números y guiones bajos. El nombre del fragmento es una expresión regular con la forma: [a-zA-Z_][a-zA-Z_0-9]*
<Type>	"manual" o "automático" manual: El programa de STEP 7 debe solicitar este fragmento y puede responder en correspondencia. El funcionamiento del fragmento se debe controlar con STEP 7 y con las variables del DB de control. automático: El servidor web procesa el fragmento automáticamente. Si no se especifica el tipo de parámetro, el ajuste predeterminado es "automático".
<id>	Número de identificación entero. Si no se especifica el parámetro ID, el servidor web asigna un número predeterminado. Para fragmentos manuales, ajuste un número reducido para la ID. La ID es el medio mediante el cual el programa de STEP 7 controla un fragmento manual.
<Mode>	"visible" u "oculto" visible: Los contenidos del fragmento se mostrarán en la página web definida por el usuario. oculto: Los contenidos del fragmento no se mostrarán en la página web definida por el usuario. Si no se especifica el tipo de parámetro, el ajuste predeterminado es "visible".

Fragmentos manuales

Si se crea un fragmento manual para una página web definida por el usuario o para una parte de una página, el programa de STEP 7 debe controlar el momento en que se envía el fragmento. El programa de STEP 7 debe configurar los parámetros adecuados en el DB de control para una página definida por el usuario mediante control manual y luego debe llamar la instrucción WWW con el DB de control según se modifique. Consulte el apartado Control avanzado de páginas web definidas por el usuario (Página 580) para comprender la estructura del DB de control y aprender a manejar las páginas individuales y los fragmentos.

11.3.2.9 Importar fragmentos

Es posible crear un fragmento determinado a partir de una parte del código HTML y luego importar este fragmento a otro lugar del grupo de páginas web definidas por el usuario. Por ejemplo, tome un grupo de páginas web definidas por el usuario que tenga una página de inicio y luego varias páginas HTML accesibles a través de los enlaces de la página de inicio. Supongamos que cada página individual debe visualizar el logotipo de la empresa en la página. Esto se puede realizar creando un fragmento (Página 555) que cargue la imagen del logotipo de la empresa. A continuación, cada página HTML puede importar este fragmento para visualizar el logotipo de la empresa. Para este fin, utilice el comando AWP Import_Fragment. El código HTML del fragmento sólo existe en un fragmento, no obstante este DB de fragmentos se puede importar tantas veces como se requiera en cualquier número de páginas web de su elección.

Sintaxis

```
<!-- AWP_Import_Fragment Name='<Name>' -->
```

Parámetros

<Name>	Cadena de texto: nombre del DB de fragmentos que se debe importar
--------	---

Ejemplo

Extracto del código HTML que crea un fragmento para visualizar una imagen:

```
<!-- AWP_Start_Fragment Name='My_company_logo' --><p></p>
```

Extracto del código HTML en otro archivo .html que importa el fragmento que visualiza la imagen del logotipo:

```
<!-- AWP_Import_Fragment Name='My_company_logo' -->
```

Ambos archivos .html (el que crea el fragmento y el que lo importa) se encuentran en la estructura de carpetas que se define al configurar las páginas definidas por el usuario en STEP 7 (Página 560).

11.3.2.10 Combinar definiciones

Al declarar variables para usarlas en las páginas web definidas por el usuario se puede combinar una declaración de variable y un alias para la variable (Página 552). Asimismo, se pueden declarar varias In_Variables en una instrucción y varias Out_Variables en una instrucción.

Ejemplos

```
<!-- AWP_In_Variable Name='"Level"', Name='"Weight"', Name='"Temp"'  
-->  
<--! AWP_Out_Variable Name='HEADER:Status', Use='"Status"',  
Name='HEADER:Location', Use="Location",  
Name='COOKIE_VALUE:name', Use="my_cookie" -->  
<!-- AWP_In_Variable Name='Alarm' Use='"Data_block_10".Alarm' -->
```

11.3.2.11 Procesamiento de nombres de variable que contienen caracteres especiales

Al especificar nombres de variables en páginas web definidas por el usuario, hay que observar cuidadosamente si los nombres de variable contienen caracteres con significados especiales.

Leer variables

Para leer una variable (Página 546) se utiliza la sintaxis siguiente:

```
:=<Varname>:
```

Las reglas siguientes rigen para leer variables:

- Para nombres de variables de la tabla de variables PLC, ponga el nombre de la variable entre comillas dobles.
- Para nombres de variables de bloque de datos, ponga el nombre del bloque de datos entre comillas dobles. La variable debe quedar por fuera de las comillas.
- Para nombres de variables que son direcciones E/S directas, direcciones de memoria o alias no utilice comillas para la variable de lectura.
- Para nombres de variables o variables de bloque de datos que contienen una barra inversa, ponga otra barra inversa delante de la primera.
- Si un nombre de variable o variable de bloque de datos contiene dos puntos, el signo menos, el signo más o un signo &, defina un alias que no contenga caracteres especiales para la variable de lectura y lea la variable utilizando el alias. Ponga una barra inversa delante de dos puntos en nombres de variables dentro de una cláusula de uso.

Tabla 11- 1 Ejemplos de lectura de variables

Nombre del bloque de datos	Nombre de la variable	Comando de lectura
n/a	ABC:DEF	<code><!--AWP_Out_Variable Name='special_tag' Use ='"ABC:DEF"' -->:=special_tag:</code>
n/a	T\	<code>:= "T\\" :</code>
n/a	A \B 'C :D	<code><!--AWP_Out_Variable Name='another_special_tag' Use='"A \\B \'C :D"' -->:=another special tag:</code>
n/a	a<b	<code><!--AWP_Out_Variable Name='a_less_than_b' Use=' "a<b" ' -->:=a less than b:</code>
Data_block_1	Tag_1	<code>:= "Data_block_1".Tag_1:</code>
Data_block_1	ABC:DEF	<code><!-- AWP_Out_Variable Name='special_tag' Use=' "Data_block_1".ABC\ :DEF' -->:=special_tag:</code>
DB A' B C D\$ E	Tag	<code>:= "DB A' B C D\$ E".Tag:</code>
DB:DB	Tag:Tag	<code><!--AWP_Out_Variable Name='my_tag' Use ='"DB:DB".Tag\ :Tag' -->:=my_tag:</code>

Cláusulas de nombre y uso

Los comandos AWP_In_Variable, AWP_Out_Variable, AWP_Enum_Def, AWP_Enum_Ref, AWP_Start_Fragment y AWP_Import_Fragment tienen cláusulas de nombre. Los comandos de formas HTML como <input> y <select> también tienen cláusulas de nombre. AWP_In_Variable y AWP_Out_Variable pueden tener adicionalmente cláusulas de uso. Indistintamente del comando, la sintaxis para las cláusulas de nombre y uso es igual en lo referente al procesamiento de caracteres especiales.

- El texto que se suministra para una cláusula de nombre o uso debe ponerse entre comillas simples. Si el nombre entre comillas es un nombre de variable PLC o de bloque de datos, utilice comillas simples para toda la cláusula.
- Dentro de una cláusula de nombre o uso, los nombres de bloque de datos y de variables PLC deben ponerse entre comillas dobles.
- Si un nombre de variable o de bloque de datos incluye un carácter de comilla simple o una barra inversa, omita dicho carácter con una barra inversa. La barra inversa es el carácter "Escape" en el compilador de comandos AWP.

Tabla 11- 2 Ejemplos de cláusulas de nombre

Nombre del bloque de datos	Nombre de la variable	Opciones de la cláusula de nombre
n/a	ABC'DEF	Name="ABC\ 'DEF" '
n/a	A \B 'C :D	Name="A \\B \ 'C :D" '
Data_block_1	Tag_1	Name="Data_block_1".Tag_1'
Data_block_1	ABC'DEF	Name="Data_block_1".ABC\ 'DEF'
Data_block_1	A \B 'C :D	Name="Data_block_1".A \\B \ 'C :D'
DB A' B C D\$ E	Tag	Name="DB A\ ' B C D\$ E".Tag'

Las cláusulas de uso se rigen por las mismas convenciones que las cláusulas de nombre.

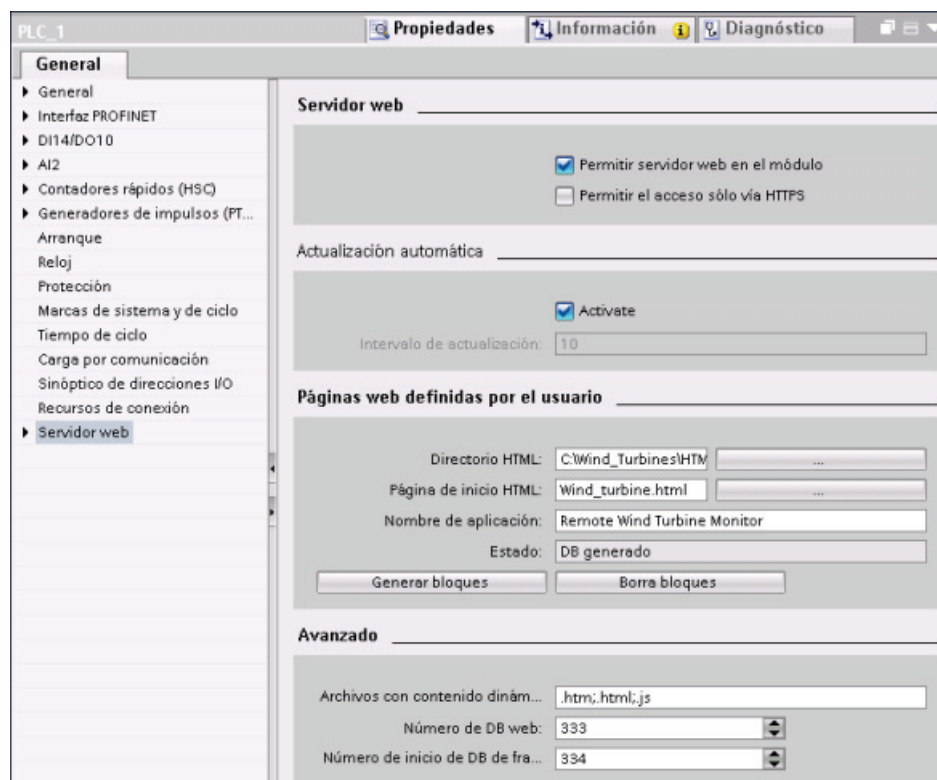
Nota

Indistintamente de qué caracteres se utilicen en la página HTML, configure la fuente de la página HTML a UTF-8 y guárdela desde el editor con codificación de caracteres UTF-8.

11.3.3 Configurar el uso de las páginas web definidas por el usuario

Para configurar las páginas web definidas por el usuario desde STEP 7, proceda del siguiente modo:

1. Seleccione la CPU en la vista de configuración de dispositivos.
2. Visualice las propiedades del "Web server" en la ventana de inspección de la CPU.



3. De no estarlo, seleccione la casilla de verificación para "Enable Web server".
4. Seleccione la casilla de verificación para "Permit access only with HTTPS" (Permitir acceso sólo con HTTPS) para asegurarse de que el servidor web utiliza comunicación cifrada y para aumentar la seguridad de la CPU accesible mediante web.
5. Introduzca o navegue hasta el nombre de la carpeta en el PC donde guardó la página HTML predeterminada (página de arranque).
6. Introduzca el nombre de la página predeterminada.
7. Asigne un nombre a la aplicación (opcional). El nombre de la aplicación se emplea para agrupar o categorizar con más detalle las páginas web. Cuando el nombre de la aplicación existe, la URL aparece con el formato siguiente:
http://ww.xx.yy.zz/awp/<nombre aplicación>/<nombre página>.html.
8. Especifique las extensiones de nombres de archivos que se analizarán para detectar comandos AWP. De forma predeterminada, STEP 7 analiza archivos con extensiones .htm, .html o .js. Si dispone de otras extensiones de archivos, añádalas.

9. Conserve el número de DB predeterminado de la web o introduzca un número de su elección. Este es el número del DB de control que controla la visualización de las páginas web.
10. Conserve el número inicial predeterminado del DB de fragmentos o introduzca un número de su elección. Este es el primer DB de fragmentos que contiene las páginas web.

Generar bloques de programa

Cuando se hace clic en el botón "Generar bloques", STEP 7 genera bloques de datos a partir de las páginas HTML del directorio fuente HTML especificado y un bloque de datos de control para operar las páginas web. Es posible configurar estos atributos para su aplicación según los requerimientos específicos (Página 561). STEP 7 genera también un conjunto de bloques de datos de fragmentos para garantizar la representación de todas las páginas HTML. Al generar los bloques de datos, STEP 7 actualiza las propiedades para visualizar el número del bloque de datos de control y el número del primer bloque de datos de fragmentos. Tras generar los bloques de datos, las páginas web definidas por el usuario forman parte del programa de STEP 7. Los bloques que pertenecen a estas páginas aparecen en una carpeta del servidor web, que se encuentra en los bloques de programa del árbol de navegación del proyecto, en la carpeta de bloques de programa.

Borrar bloques de programa

Para borrar bloques de datos generados previamente, haga clic en el botón "Borrar bloques de datos". STEP 7 borra el bloque de datos de control y todos los bloques de datos de fragmentos del proyecto asociados a páginas web definidas por el usuario.

11.3.4 Programar la instrucción WWW para páginas web definidas por el usuario

El programa de usuario de STEP 7 debe incluir y ejecutar instrucciones WWW para que las páginas web definidas por el usuario sean accesibles desde las páginas web estándar. El bloque de datos de control es el parámetro de entrada de la instrucción WWW y especifica el contenido de las páginas, según se representan en los bloques de datos de fragmentos, así como la información de estado y control. STEP 7 crea el bloque de datos de control al hacer clic en el botón "Crear bloques" en la configuración de las páginas web definidas por el usuario (Página 560).

Programar la instrucción WWW

El programa STEP 7 debe ejecutar la instrucción WWW para que las páginas web definidas por el usuario sean accesibles desde las páginas web estándar. También puede resultar conveniente que las páginas web definidas por el usuario sólo estén disponibles en determinadas circunstancias que están sujetas a las preferencias y a los requerimientos de la aplicación. En este caso, la lógica del programa puede controlar cuándo se invocará la instrucción WWW.

Tabla 11- 3 Instrucción WWW

KOP / FUP	SCL	Descripción
	<pre>ret_val := WWW(ctrl_db:=_uint_in_);</pre>	Permite acceder a las páginas web definidas por el usuario desde las páginas web estándar

Hay que especificar el parámetro de entrada del bloque de datos de control (CTRL_DB) que se corresponda con el número de DB entero del DB de control. Este número de DB de control (denominado número de DB de la web) aparece en las propiedades del servidor web de la CPU tras crear los bloques para las páginas web definidas por el usuario. Introduzca el número entero de DB como parámetro CTRL_DB para la instrucción WWW. El valor de retorno (RET_VAL) contiene el resultado de la función. Observe que la instrucción WWW se ejecuta de modo asíncrono y que la salida RET_VAL puede tener un valor inicial de 0 a pesar de que un error puede ocurrir más tarde. El programa puede comprobar el estado del DB de control para garantizar que la aplicación se ha iniciado correctamente o bien puede comprobar RET_VAL llamando posteriormente la instrucción WWW.

Tabla 11- 4 Valor de retorno

RET_VAL	Descripción
0	No hay error
16#00yx	x: La petición representada por el bit respectivo está en estado de espera: x=1: petición 0 x=2: petición 1 x=4: petición 2 x=8: petición 3 Los valores x pueden ser una operación lógica O para representar los estados de espera de varias peticiones. Si x = 6, por ejemplo, entonces las peticiones 1 y 2 están en espera. y: 0: sin error; 1: existe un error y "last_error" ha sido depositado en el DB de control
16#803a	El DB de control no está cargado.
16#8081	El DB de control es de un tipo, formato o versión incorrecta.
16#80C1	No hay recursos disponibles para inicializar la aplicación web.

Utilización del DB de control

STEP 7 crea el bloque de datos de control al hacer clic en "Generar bloques" y visualiza el número del DB de control en las propiedades de las páginas web definidas por el usuario. El DB de control también aparece en la carpeta de bloques de programa del árbol de navegación del proyecto.

Generalmente, el programa de STEP 7 emplea el DB de control directamente tal y como se creó en el proceso de generación de bloques, sin ninguna manipulación adicional. No obstante, el programa de usuario de STEP 7 puede establecer comandos globales en el DB de control para desactivar el servidor web o para volver a activarlo posteriormente. Asimismo, para las páginas definidas por el usuario creadas como DB de fragmentos manuales (Página 560), el programa de usuario de STEP 7 debe controlar el comportamiento de dichas páginas mediante una tabla de petición en el DB de control. Encontrará más información acerca de estas tareas avanzadas en el apartado Control avanzado de páginas web definidas por el usuario (Página 580).

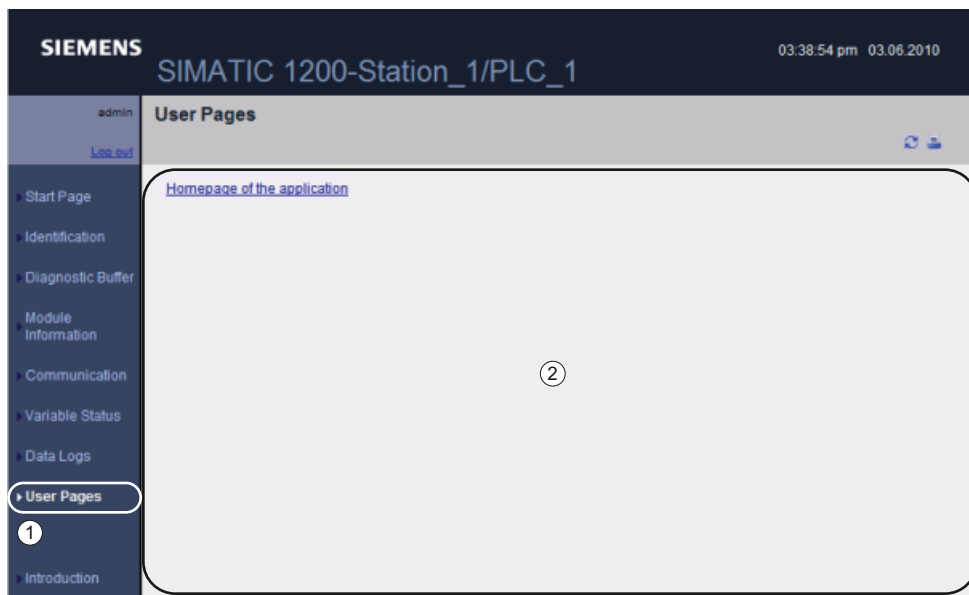
11.3.5 Cargar los bloques de programa en la CPU

Tras generar los bloques para páginas web definidas por el usuario, éstos pasan a ser parte del programa de STEP 7 como es el caso de otros bloques de programa. Siga el proceso normal para cargar los bloques de programa en la CPU. Tenga en cuenta que solo es posible descargar bloques de programa para páginas web personalizadas cuando la CPU está en STOP.

11.3.6 Acceder a las páginas web definidas por el usuario desde el PC

A las páginas web definidas por el usuario se accede desde las páginas web estándar (Página 525). Las páginas web estándar visualizan un enlace para "Páginas de usuario" en el menú ubicado en la parte izquierda, donde aparecen los enlaces para las demás páginas. Cuando se hace clic en el enlace "Páginas de usuario", el navegador web va a la página que permite acceder a la página predeterminada. En la parte interior del área de contenidos definidos por el usuario, la navegación depende de cómo han sido diseñadas las páginas específicas.

El servidor web muestra el contenido de la página definida por el usuario dentro del marco de visualización de la página web estándar. Es decir, el marco del encabezado se encuentra en el extremo superior y los cuadros de inicio de sesión y de navegación ubicados a la izquierda permanecen en posiciones fijas.



- ① Enlace para la página predeterminada de las páginas definidas por el usuario
- ② Área de contenido de las páginas web definidas por el usuario

11.3.7 Limitaciones específicas de las páginas web definidas por el usuario

Las limitaciones para páginas web estándar (Página 583) también se aplican a las páginas web definidas por el usuario. Además, las páginas web definidas por el usuario presentan algunas características específicas.

Espacio de memoria de carga

Las páginas web definidas por el usuario se convierten en bloques de datos al hacer clic en "Generar bloques". Esta acción requiere mucho espacio en la memoria de carga. Si se ha instalado una Memory Card, se puede contar con la capacidad de dicha Memory Card como espacio de memoria de carga externa para las páginas web definidas por el usuario.

Si no se ha instalado ninguna Memory Card, estos bloques emplean el espacio de la memoria de carga interna, que está limitado según el modelo de CPU.

Es posible comprobar la cantidad de memoria de carga empleada y la cantidad disponible en las herramientas online y de diagnóstico de STEP 7. También se pueden consultar las propiedades de los bloques individuales que genera STEP 7 desde las páginas web definidas por el usuario y comprobar el consumo de memoria de carga.

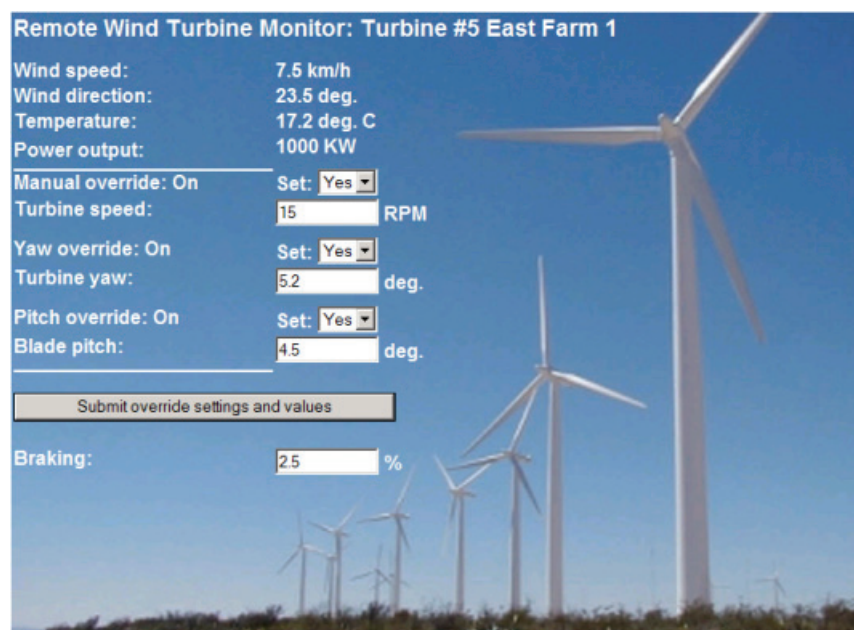
Nota

Si hay que reducir el espacio necesario para las páginas web definidas por el usuario, se deberá reducir el uso de imágenes, si procede.

11.3.8 Ejemplo de una página web definida por el usuario

11.3.8.1 Página web para vigilar y controlar una turbina de viento

Una página web definida por el usuario podría ser, por ejemplo, una página web utilizada para vigilar y controlar a distancia una turbina de viento:



Remote Wind Turbine Monitor: Turbine #5 East Farm 1

Wind speed:	7.5 km/h
Wind direction:	23.5 deg.
Temperature:	17.2 deg. C
Power output:	1000 KW
Manual override: On	Set: Yes
Turbine speed:	15 RPM
Yaw override: On	Set: Yes
Turbine yaw:	52 deg.
Pitch override: On	Set: Yes
Blade pitch:	4.5 deg.

Submit override settings and values

Braking: 25 %

Nota

La página en este ejemplo está en inglés, pero obviamente se puede utilizar cualquier idioma para desarrollar sus propias páginas HTML.

Descripción

En esta aplicación, cada turbina de un parque de turbinas de viento está equipada con un S7-1200 para ejecutar las funciones de control. En el programa de STEP 7, cada turbina de viento tiene un bloque de datos con información específica de la turbina de viento y de la ubicación.

La página web definida por el usuario permite un acceso remoto a la turbina desde un PC. Un usuario puede conectarse con páginas web estándar de la CPU de una turbina de viento en particular y acceder a la página web definida por el usuario para la vigilancia remota de la turbina de viento. Asimismo, un usuario admin puede conmutar la turbina a modo manual y controlar las variables para la velocidad, el viraje y el cabeceo de la turbina desde la página web. Un usuario admin también puede ajustar un valor de frenado indistintamente de si la turbina está bajo control manual o automático.

El programa de STEP 7 comprobaría los valores booleanos para corregir el control automático y, de estar ajustada la función correspondiente, aplicaría los valores especificados por el usuario para la velocidad, el viraje y el cabeceo de la turbina. De lo contrario, el programa ignoraría estos valores.

Archivos utilizados

Este ejemplo de página web definida por el usuario consta de tres archivos:

- **Wind_turbine.html:** Ésta es la página HTML que implementa la visualización que aparece arriba, utilizando comandos AWP para acceder a los datos del controlador.
- **Wind_turbine.css:** Esta es la hoja de estilo en cascada que contiene los estilos de formato de la página HTML. El uso de una hoja de estilo en cascada es opcional pero puede simplificar el diseño de la página HTML.
- **Wind_turbine.jpg:** Esta es la imagen de fondo utilizada por la página HTML. El uso de imágenes en páginas web definidas por el usuario es, por supuesto, opcional y requiere espacio adicional en la memoria de la CPU.

Estos archivos no se suministran con la instalación, pero se describen a modo de ejemplo.

Implementación

La página HTML utiliza comandos AWP para leer valores del PLC (Página 546) para los campos de visualización y escribir valores en el PLC (Página 547) para datos provenientes de las entradas del usuario. Esta página también utiliza comandos AWP para definir tipos de enumeración (Página 553) y referenciar (Página 554) el manejo de ajustes ON/OFF.

Esta primera parte de la página visualiza una línea de encabezado que incluye el número de la turbina de viento y la ubicación.

Remote Wind Turbine Monitor: Turbine #5 East Farm 1

La siguiente parte de la página muestra las condiciones atmosféricas a las que está sometida la turbina de viento. Estos campos son alimentados por E/S ubicadas en el sitio donde está instalada la turbina y que suministran la velocidad y la dirección del viento, así como la temperatura actual.

Wind speed:	7.5 km/h
Wind direction:	23.5 deg.
Temperature:	17.2 deg. C

A continuación, la página muestra la salida de potencia de la turbina con base en la lectura del S7-1200.

Power output:	1000 KW
---------------	---------

Las siguientes secciones permiten controlar la turbina manualmente, corrigiendo el control automático normal del S7-1200. Están disponibles las siguientes modalidades:

- Corrección manual: activa la corrección manual de la turbina. El programa de usuario de STEP 7 exige que el ajuste de corrección manual sea TRUE (verdadero) antes de habilitar el uso de cualquier ajuste manual para la velocidad, el viraje o el cabeceo de la turbina.
- Corrección de viraje: activa la corrección manual del viraje ajustado y un ajuste manual del mismo. El programa de usuario de STEP 7 exige que tanto la corrección manual como de viraje sean TRUE (verdadero) para aplicar el ajuste de viraje.
- Corrección de cabeceo: activa la corrección manual del cabeceo de las hojas. El programa de usuario de STEP 7 exige que tanto la corrección manual como de cabeceo sean TRUE (verdadero) para aplicar el ajuste de cabeceo de las hojas.

Manual override: On	Set: Yes
Turbine speed:	15 RPM

Yaw override: On	Set: Yes
Turbine yaw:	52 deg.

Pitch override: On	Set: Yes
Blade pitch:	4.5 deg.

La página HTML dispone de un botón para enviar los ajustes de corrección al controlador.

Submit override settings and values

El campo de entrada de usuario para frenado permite ajustar manualmente un porcentaje de frenado. El programa de usuario de STEP 7 no requiere una corrección manual para aceptar el valor de frenado.

Braking:	25 %
----------	------

Además, la página HTML utiliza un comando AWP para escribir la variable especial (Página 551) que contiene la ID del usuario que está accediendo a una variable de la lista de variables PLC a través de la página.

11.3.8.2 Leer y visualizar datos del controlador

La página HTML para la vigilancia remota de la turbina de viento utiliza numerosos comandos AWP para leer datos del controlador (Página 546) y visualizarlos en la página. Por ejemplo, tenga en cuenta el código HTML para visualizar la salida de potencia como se indica en esta parte de la página web a modo de ejemplo:

Power output:	1000 KW
---------------	---------

Ejemplo de código HTML

El extracto siguiente de la página HTML para la vigilancia remota de la turbina de viento muestra el texto "Power Output:" en la celda izquierda de una fila de la tabla y lee la variable para la salida de potencia y la visualiza en la celda derecha de la fila de la tabla junto con el texto de las unidades, KW.

El comando AWP := "Data_block_1".PowerOutput: ejecuta la operación de lectura. Observe que los bloques de datos están referenciados por el nombre y no por el número de bloque de datos (es decir, "Data_block_1" y no "DB1").

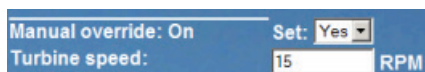
```

<tr style="height:2%;">
<td>
<p>Power output:</p>
</td>
<td>
<p style="margin-bottom:5px;"> := "Data_block_1".PowerOutput: KW</p>
</td>
</tr>

```

11.3.8.3 Utilizar un tipo de enumeración

%La página HTML para el monitoreo remoto de la turbina de viento utiliza tipos de enumeración para las tres instancias donde la página HTML visualiza "ON" u "OFF" para un valor booleano y donde el usuario especifica un valor booleano. El tipo de enumeración para "ON" resulta en un valor de 1 y el tipo de enumeración para "OFF" resulta en un valor de 0. Por ejemplo, tenga en cuenta el código HTML para leer y escribir el ajuste de activación de corrección manual en el valor "Data_block_1".ManualOverrideEnable utilizando un tipo de enumeración:



Ejemplo de código HTML

Los extractos siguientes de la página HTML para la vigilancia remota de la turbina de viento muestran cómo declarar un tipo de enumeración denominado "OverrideStatus" con valores de "Off" y "On" para 0 y 1 respectivamente y, a continuación, el ajuste de una referencia de tipo de enumeración a OverrideStatus para la variable booleana ManualOverrideEnable en el bloque de datos denominado "Data_block_1". Tenga en cuenta que para cada declaración AWP_Enum_Ref debe haber una declaración AWP_In_Variable correspondiente para la variable de bloque de datos o variable PLC si la página HTML escribe en la variable utilizando un tipo de enumeración.

```

<!-- AWP_In_Variable Name= "Data_block_1".ManualOverrideEnable' -->

<!-- AWP_Enum_Def Name="OverrideStatus" Values='0:"Off",1:"On"' -->

<!-- AWP_Enum_Ref Name=' "Data_block_1".ManualOverrideEnable'
Enum="OverrideStatus" -->

```

Donde la página HTML incluye un campo de visualización en una celda de la tabla para el estado actual de ManualOverrideEnable. Se utiliza solamente un comando normal de lectura de variables, pero utilizando el tipo de enumeración referenciado y declarado anteriormente, la página muestra "Off" u "On" en vez de 0 ó 1.

```

<td style="width:24%; border-top-style: Solid; border-top-width:
2px; border-top-color: #ffffff;">
<p>Manual override: := "Data_block_1".ManualOverrideEnable:</p>
</td>

```

La página HTML dispone de una lista de selección desplegable para que el usuario modifique el valor de ManualOverrideEnable. La lista de selección utiliza el texto "Yes" y "No" para visualizarlo en las listas de selección. Al utilizar el tipo de enumeración, "Yes" se correlaciona con el valor "On" del tipo de enumeración y "No" se correlaciona con el valor "Off". Una selección vacía no modifica el valor de ManualOverrideEnable.


```
<select name='"Data_block_1".ManualOverrideEnable'>
<option value=':"Data_block_1".ManualOverrideEnable:'> </option>
<option value="On">Yes</option>
<option selected value="Off">No</option>
</select>
```

La lista de selección está incluida dentro de una forma en la página HTML. Si el usuario hace clic en el botón de envío, la página envía la forma que escribe un valor de "1" en el ManualOverrideEnable booleano de Data_block_1, siempre y cuando el usuario haya seleccionado "Yes" o bien "0", siempre y cuando el usuario haya seleccionado "No".

11.3.8.4 Escribir una entrada del usuario en el controlador

La página HTML de vigilancia remota de la turbina de viento dispone de varios comandos AWP para escribir datos en el controlador (Página 547). La página HTML declara AWP_In_Variables para variables booleanas, de modo que un usuario con derechos de administrador puede conmutar la turbina de viento a control manual y activar la corrección manual para la velocidad de la turbina, la corrección de viraje y/o la corrección de cabeceo de hojas. Esta página también utiliza AWP_In_Variables para permitirle a un usuario admin especificar posteriormente valores en coma flotante para la velocidad, el viraje y el cabeceo de la turbina, así como para el porcentaje de frenado. La página utiliza un comando de envío de formas HTTP para escribir las AWP_In_Variables en el controlador.

Por ejemplo, tenga en cuenta el código HTML para ajustar manualmente el valor de frenado:



Ejemplo de código HTML

El extracto siguiente de la página HTML para la vigilancia remota de la turbina de viento declara primero una AWP_In_Variable para "Data_block_1" que permite a la página HTML escribir en cualquier variable del bloque de datos "Data_block_1". La página muestra el texto "Braking:" en la celda izquierda de una fila de la tabla. En la celda derecha de la fila de la tabla se encuentra el campo que acepta entradas del usuario para la variable "Braking" de "Data_block_1". Este valor de entrada de usuario está dentro de una forma HTML que utiliza el método HTTP "POST" para enviar los datos de texto introducidos a la CPU. A continuación, la página lee el valor de frenado actual del controlador y lo visualiza en el campo de entrada de datos.

Un usuario admin puede utilizar posteriormente esta página para escribir un valor de frenado en el bloque de datos de la CPU que controla el frenado.

```
<!-- AWP_In_Variable Name='"Data_block_1"' -->
...
<tr style="vertical-align: top; height: 2%;">
<td style="width: 22%;"><p>Braking:</p></td>
<td>
<form method="POST">
<p><input name='"Data_block_1".Braking' size="10" type="text"> %</p>
</form>
</td>
</tr>
```

Nota

Tenga en cuenta que si una página definida por el usuario tiene un campo de entrada de datos para una variable de escritura del bloque de datos del tipo String, el usuario debe poner el string entre comillas simples cuando introduzca el valor del string en el campo.

Nota

Observe que si se declara un bloque de datos entero en una declaración de AWP_In_Variable, por ejemplo <!-- AWP_In_Variable Name="Data_block_1" -->, entonces toda variable dentro de ese bloque de datos se puede escribir desde la página web definida por el usuario. Utilice esta opción si pretende que todas las variables de un bloque de datos sean de escritura. De lo contrario, si desea que sólo las variables de bloque de datos específicas se puedan escribir desde la página web definida por el usuario, declárelo específicamente con una declaración, por ejemplo <!-- AWP_In_Variable Name="Data_block_1".Braking' -->

11.3.8.5 Escribir una variable especial

La página web para la vigilancia remota de la turbina de viento escribe la variable especial SERVER:current_user_id en una variable PLC de la CPU. En este caso, el valor de la variable PLC contiene la ID del usuario que esté accediendo a la página web para la vigilancia remota de la turbina de viento. Actualmente, el usuario admin tiene una ID de usuario de 1, de modo que el valor de la variable PLC se pone a 1.

La variable especial se escribe en el PLC a través de la página web y no requiere una interfaz de usuario.

Ejemplo de código HTML

```
<!-- AWP_In_Variable Name="SERVER:current_user_id" Use="User_ID"-->
```

11.3.8.6 Referencia: Lista HTML de la página web para la vigilancia remota de la turbina de viento

Wind_turbine.html

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd"><!--
Este programa de prueba simula una página web para monitorizar y
controlar a distancia una turbina de viento.
Variables PLC y variables de bloque de datos requeridas en STEP 7:

Variable PLC:
User_ID: Int

Bloques de datos:
Data_block_1

Variables en Data_Block_1:
```

```

Location: String
TurbineNumber: Int
WindSpeed: Real
WindDirection: Real
Temperature: Real
PowerOutput: Real
ManualOverrideEnable: Bool
TurbineSpeed: Real
YawOverride: Bool
Yaw: Real
PitchOverride: Bool
Pitch: Real
Braking: Real

```

La página web definida por el usuario muestra los valores actuales de los datos PLC y proporciona una lista de selección para establecer los tres valores booleanos utilizando una asignación de tipo de enumeración. El botón correspondiente envía los valores booleanos seleccionados y los campos de entrada de datos para la velocidad de la turbina, el viraje y el cabeceo. El valor de frenado se puede establecer sin utilizar el botón de envío.

Para utilizar esta página, no se requiere un programa STEP 7. Teóricamente, el programa STEP 7 sólo actuaría sobre los valores de velocidad de la turbina, viraje y cabeceo si se hubiesen establecido los valores booleanos asociados. El único requisito de STEP 7 consiste en llamar a la instrucción WWW con el número de DB de los bloques de datos generados para esta página.

```

-->
<!-- AWP_In_Variable Name='Data_block_1' -->
<!-- AWP_In_Variable Name='Data_block_1'.ManualOverrideEnable' -->
<!-- AWP_In_Variable Name='Data_block_1'.PitchOverride' -->
<!-- AWP_In_Variable Name='Data_block_1'.YawOverride' -->
<!-- AWP_In_Variable Name="SERVER:current_user_id" Use="User_ID"-->
<!-- AWP_Enum_Def Name="OverrideStatus" Values='0:"Off",1:"On"' -->
<!-- AWP_Enum_Ref Name='Data_block_1'.ManualOverrideEnable'
Enum="OverrideStatus" -->
<!-- AWP_Enum_Ref Name='Data_block_1'.PitchOverride'
Enum="OverrideStatus" -->
<!-- AWP_Enum_Ref Name='Data_block_1'.YawOverride'
Enum="OverrideStatus" -->

```

```

<html>
<head>
<meta http-equiv="content-type" content="text/html; charset=utf-8"><link rel="stylesheet" href="Wind_turbine.css">
<title>Monitorización remota de la turbina de viento</title>
</head>
<body>
<table style="background-image: url('./Wind_turbine.jpg'); width: 904px; height: 534px;" cellpadding="0" cellspacing="2"><tr style="height: 2%;"><td colspan="2">
<h2>Remote Wind Turbine Monitor: Turbine
#:="Data_block_1".TurbineNumber: :="Data_block_1".Location:</h2>
</td>

```

```

<tr style="height: 2%;"><td style="width: 24%;"><p>Wind
speed:</p></td>
<td><p> := "Data_block_1".WindSpeed: km/h</p></td>
</tr>

<tr style="height: 2%;">
<td style="width: 24%;"><p>Wind direction:</p></td>
<td><p> := "Data_block_1".WindDirection: deg.</p></td>
</tr>

<tr style="height: 2%;"><td style="width:
24%;"><p>Temperature:</p></td>
<td><p> := "Data_block_1".Temperature: deg. C</p></td>
</tr>

<tr style="height: 2%;">
<td style="width: 24%;"><p>Power output:</p></td>
<td style="margin-bottom: 5px;"> := "Data_block_1".PowerOutput:
KW</p>
</td>
</tr>

<form method="POST" action="">
<tr style="height: 2%; " >
<td style="width=24%; border-top-style: Solid; border-top-width:
2px; border-top-color: #ffffff;">
<p>Manual override: := "Data_block_1".ManualOverrideEnable:</p>
</td>
<td class="Text">Set:

<select name=' "Data_block_1".ManualOverrideEnable'>
<option value=' := "Data_block_1".ManualOverrideEnable:'> </option>
<option value="On">Yes</option>
<option value="Off">No</option>
</select>

</td>
</tr>

<tr style="vertical-align: top; height: 2%;"><td style="width:
24%;"><p>Turbine speed:</p></td>
<td>
<p style="margin-bottom: 5px;"><input
name=' "Data_block_1".TurbineSpeed' size="10"
value=' := "Data_block_1".TurbineSpeed:' type="text"> RPM</p>
</td>
</tr>

<tr style="vertical-align: top; height: 2%;">
<td style="width: 24%;">
<p>Yaw override: := "Data_block_1".YawOverride: </p>
</td>
<td class="Text">Set:

```

```

<select name='"Data_block_1".YawOverride'>
<option value=':="Data_block_1".YawOverride:'> </option>
<option value="On">Yes</option>
<option value="Off">No</option>
</select>

</td>
</tr>

<tr style="vertical-align: top; height: 2%;">
<td style="width: 24%;">
<p>Turbine yaw:</p>
</td>
<td>
<p style="margin-bottom:5px;"><input name='"Data_block_1".Yaw'
size="10" value=':="Data_block_1".Yaw:' type="text"> deg.</p>
</td>
</tr>

<tr style="vertical-align: top; height: 2%;">
<td style="width: 24%;">
<p>Pitch override: :="Data_block_1".PitchOverride: </p>
</td>
<td class="Text">Set:

<select name='"Data_block_1".PitchOverride'>
<option value=':="Data_block_1".PitchOverride:'> </option>
<option value="On">Yes</option>
<option value="Off">No</option>
</select>

</td>
</tr>

<tr style="vertical-align: top; height: 2%;">
<td style="width=24%; border-bottom-style: Solid; border-bottom-
width: 2px; border-bottom-color: #ffffff;">
<p>Blade pitch:</p>
</td>
<td>
<p style="margin-bottom:5px;"><input name='"Data_block_1".Pitch'
size="10" value=':="Data_block_1".Pitch:' type="text"> deg.</p>
</td>

</tr>
<tr style="height: 2%;">
<td colspan="2"><br>
<input type="submit" value="Submit override settings and values">
</td>
</tr>
</form>

<tr style="vertical-align: top; height: 2%;">
<td style="width: 24%;"><p>Braking:</p></td>
<td>

```

```

<form method="POST" action="">
<p> <input name="'Data_block_1'.Braking' size="10"
value=':="Data_block_1".Braking:' type="text"> %</p>
</form>
</td>
</tr>
<tr><td></td></tr>

</table>
</body>
</html>

```

Wind_turbine.css

```

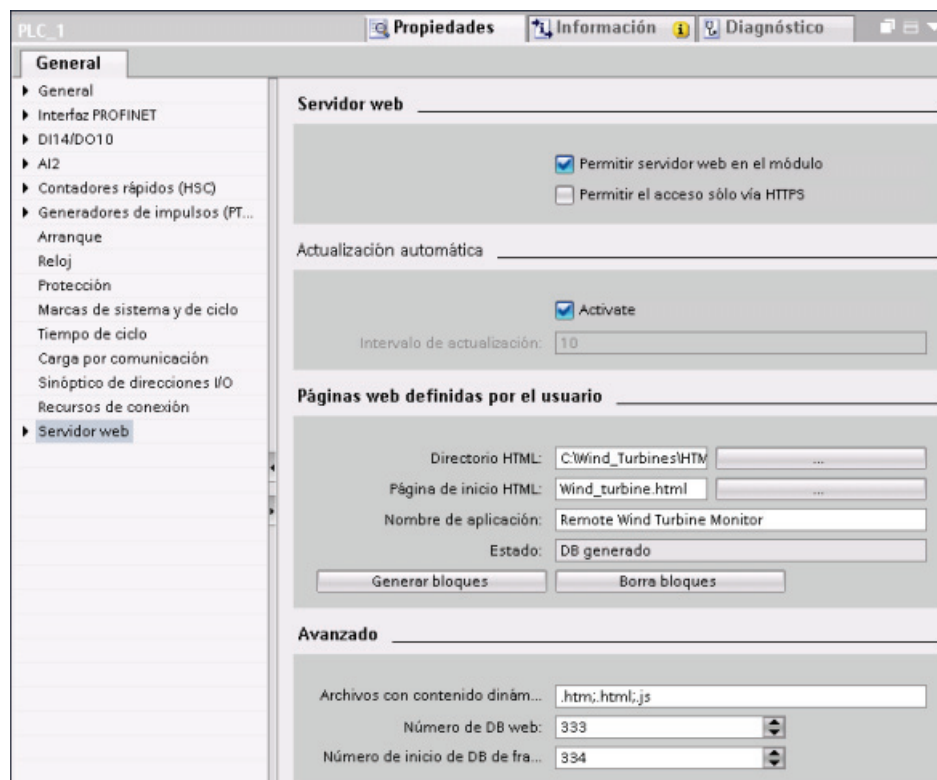
H2 {
font-family: Arial;
font-weight: bold;
font-size: 14.0pt;
color: #FFFFFF;
margin-top:0px;
margin-bottom:10px;
}
P {
font-family: Arial;
font-weight: bold;
color: #FFFFFF;
font-size: 12.0pt;
margin-top:0px;
margin-bottom:0px;
}
TD.Text {
font-family: Arial;
font-weight: bold;
color: #FFFFFF;
font-size: 12.0pt;
margin-top:0px;
margin-bottom:0px;
}

```

11.3.8.7 Configuración en STEP 7 de la página web de ejemplo

Para incluir la página HTML para la vigilancia remota de la turbina de viento como página web definida por el usuario para el S7-1200, hay que configurar los datos acerca de la página HTML en STEP 7 y crear bloques de datos desde la página HTML.

Acceda a las propiedades de la CPU del S7-1200 que controla la turbina de viento e introduzca la información de configuración en las propiedades de las páginas web definidas por el usuario del servidor web:



Campos de configuración

- Directorio HTML: Este campo especifica el nombre de ruta plenamente habilitado para la carpeta donde está ubicada dentro del PC la página predeterminada (página principal o página de inicio). El botón "..." permite navegar hasta la carpeta requerida.
- Página HTML predeterminada: este campo especifica el nombre de archivo, la página predeterminada o la página de inicio de la aplicación HTML. El botón "..." permite seleccionar el archivo requerido. Para este ejemplo, WindTurbine.html es la página HTML predeterminada. El ejemplo de la vigilancia remota de la turbina de viento consiste sólo de una página, pero en otras aplicaciones definidas por el usuario la página predeterminada puede llamar otras páginas a través de enlaces. En el código HTML, la página predeterminada debe referenciar otras páginas en relación con la carpeta de origen HTML.
- Nombre de la aplicación: Este campo opcional contiene el nombre que el navegador web incluye en el campo de dirección cuando visualiza la página. Para este ejemplo, el nombre es "Remote Wind Turbine Monitor", pero puede ser cualquier otro.

Ningún otro campo requiere configuración.

Pasos finales

Para utilizar la vigilancia remota de la turbina de viento según se ha configurado, genere los bloques, programe la instrucción WWW (Página 561) con el número del DB generado como parámetro de entrada, cargue los bloques de programa y conmute la CPU a RUN.

Cuando un operador accede posteriormente a las páginas web estándar del S7-1200 que controla la turbina de viento, la página de vigilancia remota de la turbina de viento es accesible desde el enlace "User Pages" de la barra de navegación. Esta página ofrece ahora los medios para vigilar y controlar la turbina de viento.

11.3.9 Configurar páginas web definidas por el usuario en varios idiomas

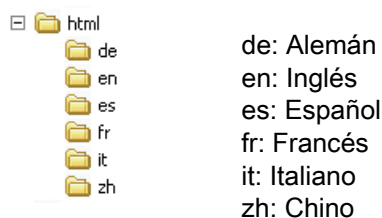
El servidor web ofrece los medios para diseñar páginas web definidas por el usuario en los idiomas siguientes:

- Alemán (de)
- Inglés (en)
- Español (es)
- Francés (fr)
- Italiano (it)
- Chino (zh)

Esto se realiza configurando las páginas HTML en una estructura de carpetas (Página 576) que se corresponda con los idiomas y configurando una cookie específica denominada "siemens_automation_language" de las páginas (Página 577). El servidor web responde a esta cookie y cambia a la página predeterminada en la carpeta de idiomas correspondiente.

11.3.9.1 Crear la estructura de carpetas

Para ofrecer páginas web definidas por el usuario en varios idiomas, debe crearse una estructura de carpetas en el directorio HTML. Los nombres de carpeta de dos letras son específicos y deben nombrarse como se indica a continuación:



En el mismo nivel, también se pueden incluir otras carpetas que necesite la página, por ejemplo, carpetas para imágenes o scripts.

Es posible incluir cualquier subconjunto de carpetas de idiomas. No es necesario incluir los seis idiomas. Dentro de las carpetas de idiomas, cree y programe las páginas HTML en el idioma correspondiente.

11.3.9.2 Programar el cambio de idioma

El servidor web permite un cambio de idioma mediante el uso de una cookie denominada "siemens_automation_language". Esta es una cookie que está definida e integrada en las páginas HTML y que es interpretada por el servidor web para visualizar el idioma correspondiente a partir de la carpeta de idioma del mismo nombre. La página HTML debe incluir una JavaScript para configurar esta cookie a uno de los identificadores de idioma predefinidos: "de", "en", "es", "fr", "it" o "zh".

Si, por ejemplo, la página HTML configura la cookie en "de", el servidor web cambia a la carpeta "de" y visualiza la página con el nombre predeterminado de la página HTML según se haya definido en la configuración de STEP 7 (Página 579).

Ejemplo

El ejemplo siguiente utiliza una página HTML predeterminada con el nombre "langswitch.html" en cada una de las carpetas de idiomas. En el directorio HTML también hay una carpeta con el nombre "script". La carpeta script incluye un archivo de JavaScript llamado "lang.js". Cada página langswitch.html utiliza esta JavaScript para configurar la cookie de idioma, "siemens_automation_language".

HTML para "langswitch.html" en la carpeta "en"

El encabezado de la página HTML ajusta el idioma a inglés, configura el juego de caracteres a UTF-8 y define la ruta al archivo de JavaScript lang.js.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Language" content="en">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Language switching english page</title>
<script type="text/javascript" src="script/lang.js" ></script>
```

El cuerpo del archivo utiliza una lista de selección para que el usuario elija entre alemán e inglés. Inglés ("en") está preseleccionado como idioma. Cuando el usuario cambia el idioma, la página llama la función de JavaScript DoLocalLanguageChange() con el valor o la opción seleccionada.

```
<!-- Language Selection -->
<table>
  <tr>
    <td align="right" valign="top" nowrap>
      <!-- change language immediately on selection change -->
      <select name="Language"
        onchange="DoLocalLanguageChange(this)"
        size="1">
        <option value="de" >German</option>
        <option value="en" selected >English</option>
      </select>
    </td>
  </tr>
</table><!-- Language Selection End-->
```

HTML para "langswitch.html" en la carpeta "de"

El encabezado de la página langswitch.html en alemán es el mismo que en inglés, excepto que el idioma ajustado es alemán.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Language" content="de"><meta http-
equiv="Content-Type" content="text/html; charset=utf-8">
<title>Sprachumschaltung Deutsche Seite</title>
<script type="text/javascript" src="script/lang.js" ></script>
</head>
```

El HTML de la página en alemán es idéntico al de la página en inglés, excepto que el valor predeterminado del idioma seleccionado es alemán ("de").

```
<!-- Language Selection -->
<table>
  <tr>
    <td align="right" valign="top" nowrap>
      <!-- change language immediately on change of the selection -
->
      <select name="Language"
        onchange="DoLocalLanguageChange(this) "
        <size="1">
          <option value="de" selected >Deutsch</option>
          <option value="en" >Englisch</option>
        </select>
      </td>
    </tr>
  </table><!-- Language Selection End-->
```

JavaScript "lang.js" en la carpeta "script"

La función "DoLocalLanguageChange()" se encuentra en el archivo lang.js. Esta función llama la función "SetLangCookie()" y luego vuelve a cargar la ventana que está visualizando la página HTML.

La función "SetLangCookie()" define una asignación que asigna el valor de la lista de selección a la cookie "siemens_automation_language" del documento. Además, se define la ruta para la aplicación, de modo que la página conmutada y no la solicitada recibe el valor de la cookie.

De manera opcional, la página puede ajustar un valor de vencimiento para la cookie en la sección de comentarios.

```

function DoLocalLanguageChange(oSelect) {
    SetLangCookie(oSelect.value);
    top.window.location.reload();
}
function SetLangCookie(value) {
    var strval = "siemens_automation_language=";
    // Esta es la cookie con la que el servidor web
    // detecta la lengua deseada
    // Este nombre es requerido por el servidor web.
    strval = strval + value;
    strval = strval + "; path=/ ";
    // Ajustar la ruta de la aplicación, pues en otro caso
    // la ruta se ajustará a la página solicitante
    // y esta página no obtendrá la cookie.
    /* OPTIONAL
    tiempo
    Utilice vencimiento si esta cookie debe existir más
    tiempo
    que la sesión actual del navegador:
    var now = new Date();
    var endttime = new Date(now.getTime() + expiration);
    strval = strval + "; expires=" +
        endttime.toGMTString() + ";";
    */
    document.cookie = strval;
}

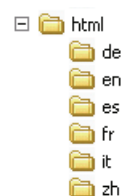
```

11.3.9.3 Configurar STEP 7 para utilizar una estructura de página multilingüe

El procedimiento para configurar páginas web definidas por el usuario en varios idiomas es similar al proceso general para configurar páginas web definidas por el usuario (Página 560). Si las carpetas se han configurado para idiomas, no obstante se debe configurar el directorio HTML para la carpeta que contiene las carpetas de idiomas individuales. El directorio HTML no se debe configurar de modo que sea una de las carpetas de idiomas.

Al seleccionar la página HTML predeterminada, se navega hasta la carpeta de idiomas y se selecciona la página HTML que va a ser la página de arranque. Si posteriormente se generan bloques y éstos se cargan en la CPU, el servidor web visualiza la página de arranque en la carpeta de idiomas que ha sido configurada.

Por ejemplo, si la estructura de carpetas indicada aquí estuviese en C:\, la configuración del directorio HTML sería C:\html, y si inglés estuviese definido para visualizar la página de arranque, sería necesario navegar hasta en\langswitch.html para acceder a la configuración de la página HTML predeterminada.



11.3.10 Control avanzado de páginas web definidas por el usuario

Al generar bloques de datos para las páginas web definidas por el usuario, STEP 7 crea un DB de control que se utiliza para controlar la visualización de las páginas definidas por el usuario, así como la interacción con éstas. STEP 7 crea también un juego de DBs de fragmentos que representa las páginas individuales. En condiciones normales, no es indispensable conocer la estructura del DB de control ni saber cómo manejarlo.

Si desea activar y desactivar una aplicación web o bien manipular fragmentos manuales individuales, utilice para tal fin las variables de DB de control y la instrucción WWW.

Estructura del DB de control

El DB de control es una estructura de datos extensiva que es accesible al programar el programa de usuario de STEP 7. Aquí se describen sólo algunas de las variables de DB de control.

Estructura Commandstate

"Commandstate" es una estructura que contiene comandos y estados globales del servidor web.

Comandos globales en la estructura "Commandstate"

Los comandos globales rigen en general para el servidor web. El servidor web se puede activar o bien reiniciar desde los parámetros del DB de control.

Variable de bloque	Tipo de datos	Descripción
init	BOOL	Evalúa el DB de control e inicializa la aplicación web
deactivate	BOOL	Desactiva la aplicación web

Estados globales en la estructura Commandstate

Los estados globales rigen en general para el servidor web y contienen información de estado acerca de la aplicación web.

Variable de bloque	Tipo de datos	Descripción
initializing	BOOL	La aplicación web está leyendo un DB de control
error	BOOL	No ha sido posible inicializar la aplicación web
deactivating	BOOL	La aplicación web está finalizando
deactivated	BOOL	La aplicación ha finalizado
initialized	BOOL	La aplicación web ha sido inicializada

Tabla de petición

La tabla de petición es una matriz de estructuras que contiene comandos y estados que se aplican a determinados DBs de fragmentos. Si ha creado fragmentos con el comando AWP_Start_Fragment (Página 555) de tipo "manual", el programa de usuario de STEP 7 debe controlar estas páginas con el DB de control. Los estados de petición sólo se pueden leer y ofrecen información sobre el fragmento actual. Es posible utilizar los comandos de petición para controlar el fragmento actual.

Variable de bloque	Tipo de datos	Descripción
requesttab	ARRAY [1 .. 4] OF STRUCT	Matriz de estructuras para controlar individualmente un DB de fragmentos. El servidor web puede procesar hasta cuatro fragmentos a la vez. El índice de matriz para un fragmento particular es arbitrario cuando el servidor web está procesando varios fragmentos o fragmentos de varias sesiones del navegador.

Elementos de la variable STRUCT requesttab

Variable de bloque	Tipo de datos	Descripción
page_index	UINT	Número de la página web actual
fragment_index	UINT	Número del fragmento actual - puede ajustarse para un fragmento diferente
// Comandos de petición		
continue	BOOL	Activa la transmisión para la página/fragmento actual y continúa con el siguiente fragmento
repeat	BOOL	Activa la retransmisión para la página/fragmento actual y continúa con el mismo fragmento
abort	BOOL	Cierra la conexión http sin transmitir
finish	BOOL	Transmite este fragmento; la página está completa - no procesa fragmentos adicionales
// Estados de petición		Los estados de petición sólo se pueden leer
idle	BOOL	Ninguna tarea pero activo
waiting	BOOL	El fragmento está en espera de ser activado
sending	BOOL	El fragmento está transmitiendo
aborting	BOOL	el usuario ha interrumpido la petición actual

Funcionamiento

Cada vez que el programa hace cambios en el DB de control debe llamar la instrucción WWW aplicando el número del DB de control modificado como parámetro. Los comandos globales y de petición tienen efecto cuando el programa de usuario de STEP 7 ejecuta la instrucción WWW (Página 561).

El programa de usuario de STEP 7 puede definir el fragment_index explícitamente, lo cual hace que el servidor web procese el fragmento especificado con un comando de petición. De lo contrario, el servidor web procesa el fragmento actual para la página actual cuando se ejecuta la instrucción WWW.

Los métodos disponibles para utilizar el fragment_index incluyen:

- Procesar el fragmento actual: no modifique el fragment_index y active el comando "continue".
- Omitir el fragmento actual: ajuste el fragment_index a 0 y active el comando "continue".
- Reemplazar el fragmento actual por otro fragmento: ajuste en el fragment_index la nueva ID de fragmento y active el comando "continue".

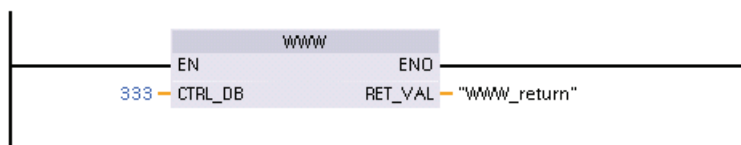
Para comprobar los estados globales o de petición que estén sujetos a cambio, el programa de usuario de STEP 7 debe llamar la instrucción WWW para evaluar los valores actuales de estos estados. Un caso típico puede ser llamar la instrucción WWW periódicamente hasta que ocurra un estado específico.

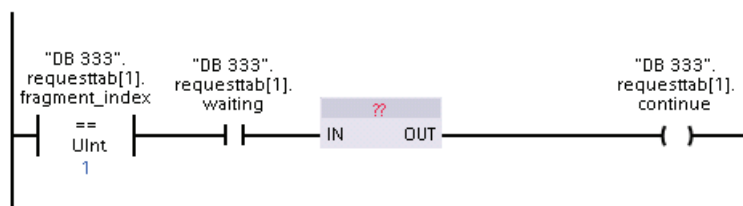
Nota

Si el programa de usuario de STEP 7 activa más de un comando de petición, la instrucción WWW procesa sólo uno de acuerdo con la siguiente prioridad: interrumpir, finalizar, repetir, continuar. La instrucción WWW borra todos los comandos de petición después del procesamiento.

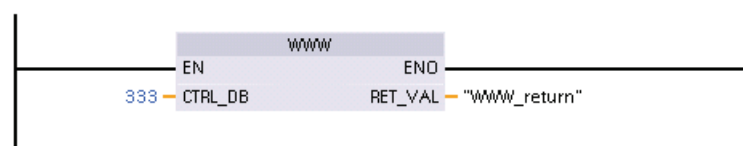
Ejemplos

El ejemplo siguiente muestra un programa de usuario de STEP 7 comprobando que un fragmento con una ID de 1 esté en estado de espera, después de una llamada previa de la instrucción WWW. También puede esperar a que ocurran otras condiciones específicas de la aplicación. A continuación ejecuta el procesamiento que se requiera para el fragmento, por ejemplo configurar variables de bloque de datos, realizar cálculos u otras tareas específicas de la aplicación. Después, activa la marca "continue" para que el servidor web ejecute este fragmento.





Cuando el programa llama la instrucción WWW con este DB de control modificado, la página web definida por el usuario a la que pertenece este fragmento puede visualizarse desde el navegador web.



Tenga en cuenta que se trata de un ejemplo simplificado. Realmente, el fragmento podría estar en cualquiera de los cuatro STRUCTS requesttab de la matriz.

11.4 Limitaciones

Los factores de TI siguientes pueden afectar al uso del servidor web:

- Normalmente, hay que usar la dirección IP de la CPU para acceder a las páginas web estándar o a las páginas web definidas por el usuario. Si el navegador web no permite la conexión directa a una dirección IP, consulte al administrador de TI. Si las políticas locales soportan DNS, el usuario se puede conectar a la dirección IP a través de una entrada DNS de dicha dirección.
- Los cortafuegos, la configuración del proxy y otras restricciones específicas del sitio también pueden restringir el acceso a la CPU. Consulte al administrador de TI para resolver estos problemas.
- Las páginas web estándar emplean JavaScripts y cookies. Si el navegador web tiene deshabilitadas las cookies o JavaScripts, hay que proceder a su habilitación. Si no se pueden habilitar, algunas de las funciones se verán restringidas (Página 584). El uso de JavaScripts y cookies en las páginas web definidas por el usuario es opcional. En caso de usarlas, es necesario habilitarlas en el servidor web.
- El servidor web soporta la capa de sockets seguros (SSL). Se puede acceder a las páginas web estándar y a las páginas web definidas por el usuario con una URL `http://ww.xx.yy.zz` o `https://ww.xx.yy.zz`, donde "ww.xx.yy.zz" representa la dirección IP de la CPU.
- Siemens ofrece un certificado de seguridad para el acceso seguro al servidor web. En la página web estándar de introducción (Página 530), se puede cargar e importar el certificado en las opciones de Internet del navegador web (Página 586). Si se elige no importar el certificado, aparece una solicitud de verificación de seguridad cada vez que se accede al servidor web `https://`.

Número de conexiones

El servidor web soporta un máximo de 30 conexiones HTTP activas. Esas 30 conexiones pueden usarse de varias formas, según el navegador web utilizado y el número de objetos diferentes por página (archivos .css, imágenes, archivos .html adicionales). Algunas conexiones persisten mientras se esté viendo la página y otras se liberan tras la conexión inicial.

Si, por ejemplo, se usa Mozilla Firefox 8, que soporta un máximo de seis conexiones persistentes, se podrán usar cinco navegadores o instancias de pestaña de navegador antes de que el servidor web empiece a cortar conexiones. Si una página no usa las seis conexiones, se pueden tener más instancias de pestañas de navegador o navegadores.

Tenga en cuenta que el número de conexiones activas puede afectar al rendimiento de presentación de páginas.

11.4.1 Funciones restringidas al deshabilitar JavaScript

Las páginas web estándar se implementan mediante HTML, JavaScript y cookies. Si el sitio restringe el uso de JavaScript y cookies, hay que habilitarlos para que las páginas funcionen de manera adecuada. Si no se puede habilitar JavaScript para el navegador web, las funciones controladas por JavaScript no pueden ejecutarse.

General

Las páginas no se actualizan dinámicamente. Hay que actualizar la página manualmente con el icono de actualización (Página 527) para ver los datos más recientes.

Página del búfer de diagnóstico

- Visualización de los detalles de eventos: Con JavaScript, seleccione una fila en el búfer de diagnóstico para ver los detalles en la sección inferior. Sin JavaScript, hay que hacer clic en el hipervínculo del campo de evento de una entrada del búfer de diagnóstico para ver los datos del evento en la sección inferior.
- Cambio del intervalo de las entradas del búfer de diagnóstico que se van a visualizar: Con JavaScript, hay que usar la lista desplegable situada en la parte superior para seleccionar el rango de las entradas del búfer de diagnóstico que se desea visualizar y la página se actualiza automáticamente. Sin JavaScript, hay que usar la lista desplegable situada en la parte superior para seleccionar el rango de las entradas del búfer de diagnóstico que se desea visualizar, pero hay que hacer clic en el enlace "Ir" para actualizar la página del búfer de diagnóstico con el rango seleccionado en la lista desplegable.

Observe que los hipervínculos "Ir" y del campo de evento sólo son visibles si JavaScript no está habilitado. Estos elementos no son necesarios y por lo tanto no están presentes si JavaScript está habilitado.

Nota

El navegador Opera V11.0 no soporta el botón "Ir" o entradas de diagnóstico hipervinculados. Con Opera V11.0 no se puede acceder a los detalles del evento o modificar el rango si se ha deshabilitado el JavaScript.

Página de información del módulo

- No se pueden filtrar los datos.
- No se pueden clasificar los campos.

Página de estado de las variables

- Tras introducir cada variable, hay que destacar manualmente la fila de nueva variable para introducir la variable nueva.
- La selección de un formato de visualización no modifica automáticamente la visualización de los valores de los datos al formato seleccionado. Hay que hacer clic en el botón "Valor de observación" para actualizar la pantalla y aplicar el nuevo formato.

Página de registros

- No se puede hacer clic en un nombre de archivo del área de entradas recientes para abrir o guardar un archivo de registro. No obstante, el icono de carga se puede utilizar para ejecutar las mismas funciones.
- La página de registros no se actualiza.
- Los botones "+" y "-" para aumentar y reducir el número de entradas recientes no funcionan.
- La entrada directa de un valor en el número de entradas recientes no ajusta el número de entradas. Si se intenta introducir un valor en este campo desde Mozilla Firefox, la pantalla se pondrá blanca. Vuelva a seleccionar "Data Logs" del panel de navegación para restaurar la pantalla de Data Logs. El campo Número de entradas recientes permanece invariable.

Conviene tener en cuenta que se puede salir y volver a entrar en la página de registros de datos para obtener las 25 entradas más recientes.

11.4.2 Funciones restringidas cuando se no se permiten cookies

Si el navegador web no permite cookies, el usuario no puede iniciar sesión como "admin".

11.4.3 Importar el certificado de seguridad Siemens

El usuario puede importar el certificado de seguridad Siemens en las opciones de Internet para omitir la verificación de seguridad al introducir `https://ww.xx.yy.zz` en el navegador web, siendo "ww.xx.yy.zz" la dirección IP de la CPU. Si utiliza una URL `http://` en vez de una URL `https://`, entonces no es necesario cargar e instalar el certificado.

Cargar el certificado

Utilice el link "Cargar certificado" de la página de introducción (Página 530) para cargar el certificado de seguridad Siemens en su PC. Este procedimiento varía en función del navegador web utilizado:

[download certificate](#)

Importar el certificado en Internet Explorer

1. Haga clic en el link "Cargar certificado" de la página de introducción. A continuación aparece el diálogo "File Download - Security Warning".
2. En el diálogo "File Download - Security Warning", haga clic en "Open" para abrir el archivo. Entonces aparece el diálogo "Certificate".
3. En el diálogo "Certificate", haga clic en el botón "Install Certificate" para iniciar el asistente de importación de certificados.
4. Siga las instrucciones de los diálogos del asistente de importación de certificados dejando que el sistema operativo seleccione automáticamente el almacenamiento del certificado.

Importar el certificado en Mozilla Firefox

1. Haga clic en el link "download certificate" en la página de introducción. A continuación aparece el diálogo "Opening MiniWebCA_Cer.crt".
2. Haga clic en "Save file" en el diálogo "Opening MiniWebCA_Cer.crt". Entonces aparece el diálogo "Downloads".
3. En el diálogo "Downloads", haga doble clic en "MiniWebCA_Cer.crt". Si ha intentado realizar la carga más de una vez, aparecerán varias copias. Simplemente haga doble clic en una de las entradas "MiniWebCA_Cer.crt".
4. Haga clic en "OK" si se solicita abrir un archivo ejecutable.
5. Haga clic en "Open" en el diálogo "Open File - Security Warning", en caso de que aparezca. Entonces aparece el diálogo "Certificate".
6. En el diálogo "Certificate", haga clic en el botón "Install Certificate".
7. Siga las instrucciones de los diálogos del asistente de importación de certificados dejando que el sistema operativo seleccione automáticamente el almacenamiento del certificado.
8. Si aparece el diálogo "Security Warning", haga clic en "Yes" para confirmar la instalación del certificado.

Otros navegadores

Siga las convenciones de su navegador web para importar e instalar el certificado Siemens. Una vez que se haya instalado el certificado de seguridad Siemens "S7-Controller Family" en las opciones de Internet para el contenido del navegador web, ya no será necesario confirmar un aviso de seguridad al acceder al servidor web con `https:// ww.xx.yy.zz`.

11.4.4 Importar registros de datos en formato CSV a versiones de Microsoft Excel que no son estadounidenses ni británicas

Los archivos de registro tienen el formato separado por comas (CSV). Estos archivos se pueden abrir directamente con Excel desde la página de Data Logs si su sistema utiliza una versión de Excel estadounidense o británica. Sin embargo, en otros países este formato no está muy difundido porque las comas están presentes frecuentemente en la notación numérica.

Para abrir un archivo de registro guardado previamente, proceda del siguiente modo en caso de utilizar otras versiones de Excel que no sean estadounidenses o británicas.

1. Abra Excel y cree un libro vacío.
2. Seleccione el comando "Importar datos" del menú "Datos > Obtener datos externos".
3. Navegue hasta el archivo de registros que desea abrir y selecciónelo. Se inicia el asistente de importación de texto.
4. En el asistente de importación de texto, cambie la opción predeterminada "Tipo de datos original" de "Longitud fija" a "Delimitado".
5. Haga clic en el botón "Siguiente".
6. En el diálogo del paso 2, seleccione la casilla de verificación "Coma" para cambiar el tipo de delimitador de "Tabulador" a "Coma".
7. Haga clic en el botón "Siguiente".
8. En el diálogo del paso 3 se puede cambiar el formato de fecha de MDY (month/day/year) a otro formato.
9. Lleve a cabo los pasos restantes del asistente de importación de texto para importar el archivo.

Procesador de comunicaciones

12.1 Utilizar las interfaces de comunicación serie

Dos módulos de comunicación (CM) y una placa de comunicación (CB) ofrecen la interfaz para la comunicación PtP:

- CM 1241 RS232 (Página 869)
- CM 1241 RS422/485 (Página 870)
- CB 1241 RS485 (Página 866)

Se pueden conectar hasta tres CMs (de cualquier tipo) y una CB para un total de cuatro interfaces de comunicación. Instale el CM a la izquierda de la CPU o de otro CM. Instale la CB en la parte frontal de la CPU. Encontrará información detallada acerca del montaje y desmontaje de módulos en el capítulo "Montaje" del Manual de sistema S7-1200 (Página 53).

Las interfaces de comunicación serie tienen las siguientes características:

- Cuentan con un puerto aislado
- Soportan protocolos punto a punto
- Se configuran y programan mediante instrucciones avanzadas y funciones de librería
- Muestran la actividad de transmisión y recepción mediante LED
- Muestran un LED de diagnóstico (sólo CM)
- Reciben alimentación de la CPU: No necesita conexión a una fuente de alimentación externa.

Consulte los datos técnicos de las interfaces de comunicación (Página 858).

Indicadores LED

Los módulos de comunicación tienen tres indicadores LED:

- LED de diagnóstico (DIAG): este LED parpadea en color rojo hasta ser direccionado por la CPU. Tras el arranque de la CPU, ésta detecta si hay CMs y los direcciona. El LED de diagnóstico comienza a parpadear en color verde. Esto indica que la CPU ha direccionado el CM, pero aún no ha suministrado la configuración correspondiente. La CPU carga la configuración en los CMs configurados cuando el programa se carga en la CPU. Una vez cargado el programa en la CPU, el LED de diagnóstico del módulo de comunicación debería encenderse en color verde.
- LED de transmisión (Tx): el LED de transmisión se enciende cuando el puerto de comunicación envía datos.
- LED de recepción (Rx): este LED se enciende cuando el puerto de comunicación recibe datos.

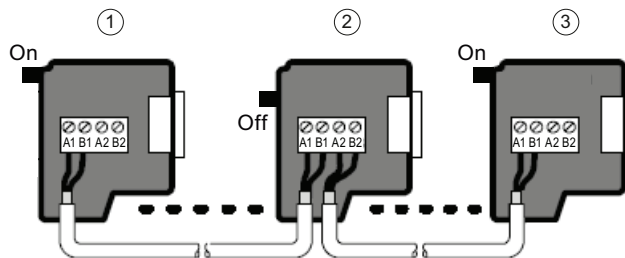
La placa de comunicación cuenta con un LED de transmisión (TxD) y uno de recepción (RxD). No tiene ningún LED de diagnóstico.

12.2 Polarizar y terminar un conector de red RS485

Siemens ofrece un conector de red RS485 (Página 881) que permite conectar fácilmente varios aparatos a una red RS485. El conector posee dos juegos de terminales para fijar los cables de entrada y salida. También incluye interruptores para polarizar y terminar selectivamente la red.

Nota

Sólo se terminan y polarizan los dos extremos de la red RS485. Los dispositivos que están entre los dos dispositivos terminadores no se terminan ni polarizan. Pantalla de protección pelada: aprox. 12 mm (1/2 pulg.) deben hacer contacto con la guía de metal en todos los puntos.



- ① Selector en posición ON: terminado y polarizado
- ② Selector en posición OFF: no terminado ni polarizado
- ③ Selector en posición ON: terminado y polarizado

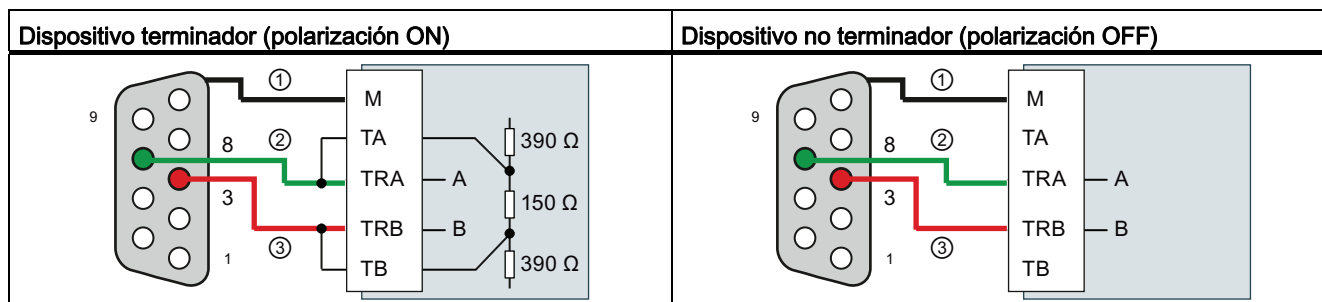
Tabla 12- 1 Terminación y polarización para el conector RS485

Dispositivo terminador (polarización ON)	Dispositivo no terminador (polarización OFF)

- ① Número de pin
- ② Conector de bus
- ③ Pantalla de cable

El CB 1241 ofrece resistores internos para terminar y polarizar la red. Para terminar y polarizar la conexión, conecte TRA con TA y TRB con TB con el fin de incluir los resistores internos en el circuito. El CB 1241 no tiene ningún conector de 9 pines. La tabla siguiente muestra las conexiones con un conector de 9 pines del interlocutor.

Tabla 12- 2 Terminación y polarización para el CB 1241

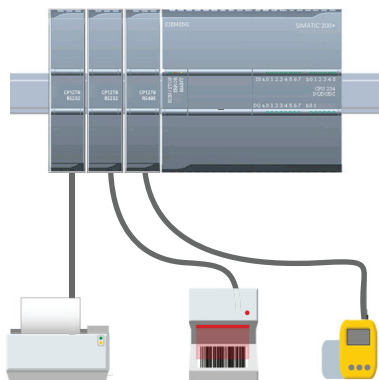


- ① Conecte M a la pantalla de protección
- ② A = TxD/RxD - (cable verde / pin 8)
- ③ B = TxD/RxD + (cable rojo / pin 3)

12.3 Comunicación punto a punto (PtP)

La CPU soporta la siguiente comunicación punto a punto (PtP) para protocolos serie basados en caracteres. PtP ofrece una libertad y flexibilidad máximas, pero requiere una implementación exhaustiva en el programa de usuario.

- PtP (Página 591)
- USS (Página 634)
- Modbus (Página 651)



PtP ofrece numerosas posibilidades, a saber:

- Posibilidad de enviar directamente información a un dispositivo externo, p. ej. una impresora
- Posibilidad de recibir información de otros dispositivos, p. ej. lectores de código de barras, lectores RFID, cámaras o sistemas de visión de terceros y muchos dispositivos más
- Posibilidad de intercambiar información (enviar y recibir datos) con otros dispositivos tales como equipos GPS, cámaras o sistemas de visión de terceros, módems radio y muchos más

La comunicación PtP es una comunicación serie que utiliza UARTs estándar para soportar distintas velocidades de transferencia y opciones de paridad. Los módulos de comunicación RS232 y RS422/485 y la placa de comunicación RS485 ofrecen las interfaces eléctricas para realizar la comunicación PtP.

12.3.1 Instrucciones de comunicación punto a punto

12.3.1.1 Parámetros comunes de las instrucciones de comunicación punto a punto

Tabla 12- 3 Parámetros de entrada comunes para las instrucciones PTP

Parámetro	Descripción
REQ	<p>Muchas de las instrucciones PtP utilizan la entrada REQ para iniciar la operación en una transición de "low" a "high". El estado lógico de la entrada REQ debe ser "high" (TRUE) durante una ejecución de la instrucción. No obstante, la entrada REQ puede permanecer TRUE durante un tiempo cualquiera. La instrucción no inicia ninguna operación diferente hasta que no sea llamada con la entrada REQ puesta a FALSE, de manera que pueda inicializar el histórico de la entrada REQ. Esto es necesario para que la instrucción pueda detectar la transición de "low" a "high" para iniciar la siguiente operación.</p> <p>Al insertar una instrucción PtP en su programa, STEP 7 le solicitará identificar el DB instancia. Utilice un DB unívoco para cada llamada de la instrucción PtP. Esto garantiza que cada instrucción procese correctamente la entradas, p. ej. REQ.</p>
PORT	<p>Se asigna una dirección de puerto durante la configuración del dispositivo de comunicación. Después de la configuración, es posible seleccionar un nombre simbólico predeterminado para el puerto en la lista desplegable auxiliar de parámetros. El valor de puerto CM o CB asignado es la propiedad de configuración del dispositivo "identificador de hardware". El nombre simbólico del puerto se asigna en la ficha "Constantes" de la tabla de variables PLC.</p>
Resolución de tiempos de bit	<p>Numerosos parámetros se definen en un número de tiempos de bit a la velocidad de transferencia configurada. Si el parámetro se define en tiempos de bit, puede ser independiente de la velocidad de transferencia. Todos los parámetros definidos en unidades de tiempos de bit pueden especificarse hasta un número máximo de 65535. No obstante, el tiempo máximo que puede medir el CM o la CB es ocho segundos.</p>

Los parámetros de salida DONE, NDR, ERROR y STATUS de las instrucciones PtP ponen a disposición de las operaciones PtP el estado de ejecución finalizada.

Tabla 12- 4 Parámetros de salida DONE, NDR, ERROR y STATUS

Parámetro	Tipo de datos	Valor predeterminado	Descripción
DONE	Bool	FALSE	Adopta el estado lógico TRUE durante una ejecución para indicar que la última petición se ha finalizado sin errores; de lo contrario, FALSE.
NDR	Bool	FALSE	Adopta el estado lógico TRUE durante una ejecución para indicar que la acción solicitada se ha finalizado sin errores y que se han recibido datos nuevos; de lo contrario, FALSE.

Parámetro	Tipo de datos	Valor predeterminado	Descripción
ERROR	Bool	FALSE	Adopta el estado lógico TRUE durante una ejecución para indicar que la última petición se ha finalizado con errores. El código de error aplicable aparece en STATUS; de lo contrario, FALSE.
STATUS	Word	0	<p>Resultado del estado:</p> <ul style="list-style-type: none"> • Si se activa el bit DONE o NDR, STATUS se pone a 0 o a un código de información. • Si se activa el bit ERROR, STATUS se pone a un código de error. • Si no se activa ninguno de estos bits, la instrucción devuelve resultados de estado que describen el estado actual de la función. <p>STATUS conserva su valor durante la ejecución de la función.</p>

Nota

Los parámetros DONE, NDR y ERROR sólo están activados durante una ejecución. La lógica del programa debe guardar temporalmente los valores de estado de la salida en flip-flops de datos, de modo que sea posible detectar cambios de estado en posteriores ciclos del programa.

Tabla 12- 5 Códigos de condición comunes

STATUS (W#16#....)	Descripción
0000	Sin error
7000	La función no está ocupada
7001	La función está ocupada con la primera llamada.
7002	La función está ocupada con las llamadas posteriores (sondeos tras la primera llamada).
8x3A	Puntero no permitido en el parámetro x
8070	Se está utilizando toda la memoria de instancia interna, hay demasiadas instrucciones simultáneas en curso
8080	Número de puerto no permitido.
8081	Timeout, error del módulo u otro error interno
8082	Ha fallado la parametrización porque se está parametrizando en segundo plano.
8083	Desbordamiento del búfer: El CM o la CB ha devuelto un mensaje recibido con una longitud superior a la que permite el parámetro de longitud.
8090	Error interno: Longitud de mensaje incorrecta, submódulo incorrecto o mensaje no permitido Póngase en contacto con el Customer Support.
8091	Error interno: Versión incorrecta en el mensaje de parametrización Póngase en contacto con el Customer Support.
8092	Error interno: Longitud de registro incorrecta en el mensaje de parametrización Póngase en contacto con el Customer Support.

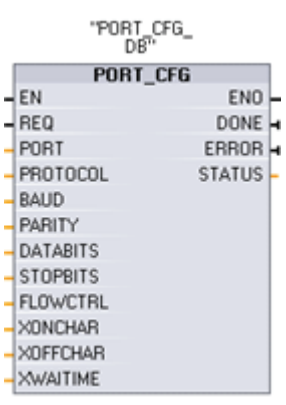
12.3 Comunicación punto a punto (PtP)

Tabla 12- 6 Clases de error comunes

Descripción de la clase	Clases de error	Descripción
Configuración del puerto	80Ax	Permite definir errores comunes de configuración del puerto
Configuración de la transmisión	80Bx	Permite definir errores comunes de configuración de la transmisión
Configuración de la recepción	80Cx	Permite definir errores comunes de configuración de la recepción
Tiempo de ejecución de la transmisión	80Dx	Permite definir errores comunes en tiempo de ejecución de la transmisión
Tiempo de ejecución de la recepción	80Ex	Permite definir errores comunes en tiempo de ejecución de la recepción
Procesamiento de señales	80Fx	Permite definir errores comunes en relación con el procesamiento de señales

12.3.1.2 Instrucción PORT_CFG

Tabla 12- 7 Instrucción PORT_CFG (configuración de puerto)

KOP / FUP	SCL	Descripción
	<pre>"PORT_CFG_DB" (REQ:=_bool_in_, PORT:=_uint_in_, PROTOCOL:=_uint_in_, BAUD:=_uint_in_, PARITY:=_uint_in_, DATABITS:=_uint_in_, STOPBITS:=_uint_in_, FLOWCTRL:=_uint_in_, XONCHAR:=_char_in_, XOFFCHAR:=_char_in_, WAITTIME:=_uint_in_, DONE=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_);</pre>	<p>La instrucción PORT_CFG permite cambiar los parámetros de puerto (p. ej. la velocidad de transferencia) desde el programa.</p> <p>La configuración estática inicial del puerto puede ajustarse en las propiedades de la configuración de dispositivos. Como alternativa, es posible utilizar los valores predeterminados. La instrucción PORT_CFG puede ejecutarse en el programa con el fin de modificar la configuración.</p>

1 STEP 7 crea el DB automáticamente al insertar la instrucción.

Los cambios de la configuración de PORT_CFG no se almacenan de forma permanente en la CPU. Los parámetros ajustados en la configuración de dispositivos se restablecen cuando la CPU cambia de RUN a STOP y tras desconectar y volver a conectar la alimentación. Encontrará más información en Configurar los puertos de comunicación (Página 611) y Gestionar el control de flujo (Página 613).

Tabla 12- 8 Tipos de datos para los parámetros

Parámetro y tipo		Tipo de datos	Descripción
REQ	IN	Bool	Activa el cambio de la configuración cuando se detecta un flanco ascendente en esta entrada. (Valor predeterminado: False)
PORT	IN	PORT	Tras haber instalado y configurado un dispositivo de comunicación CM o CB, el identificador de puerto aparece en la lista desplegable de parámetros disponible en la conexión del cuadro PUERTO. El valor de puerto CM o CB asignado es la propiedad de configuración del dispositivo "identificador de hardware". El nombre simbólico del puerto se asigna en la ficha "Constantes del sistema" de la tabla de variables PLC. (Valor predeterminado: 0)
PROTOCOL	IN	UInt	0 - Protocolo de comunicación punto a punto (valor predeterminado) 1..n - Definición futura para protocolos específicos
BAUD	IN	UInt	Velocidad de transferencia del puerto (valor predeterminado: 0): 1 = 300 baudios, 2 = 600 baudios, 3 = 1200 baudios, 4 = 2400 baudios, 5 = 4800 baudios, 6 = 9600 baudios, 7 = 19200 baudios, 8 = 38400 baudios, 9 = 57600 baudios, 10 = 76800 baudios, 11 = 115200 baudios
PARITY	IN	UInt	Paridad del puerto (Valor predeterminado: 0): 1 = Sin paridad, 2 = Paridad par, 3 = Paridad impar, 4 = Paridad Mark, 5 = Paridad Space
DATABITS	IN	UInt	Bits por carácter (valor predeterminado): 1 = 8 bits de datos, 2 = 7 bits de datos
STOPBITS	IN	UInt	Bits de parada (valor predeterminado: 0): 1 = 1 bit de parada, 2 = 2 bits de parada
FLOWCTRL	IN	UInt	Control de flujo (valor predeterminado: 0): 1 = Sin control de flujo, 2 = XON/XOFF, 3 = RTS de hardware siempre ON, 4 = RTS de hardware conmutado
XONCHAR	IN	Char	Determina el carácter que se utiliza como carácter XON. Generalmente, es un carácter DC1 (11H). Este parámetro se evalúa sólo si está habilitado el control de flujo. (Valor predeterminado: 0)
XOFFCHAR	IN	Char	Determina el carácter que se utiliza como carácter XOFF. Generalmente, es un carácter DC3 (13H). Este parámetro se evalúa sólo si está habilitado el control de flujo. (Valor predeterminado: 0)
XWAITIME	IN	UInt	Determina cuánto tiempo se debe esperar un carácter XON tras recibir un carácter XOFF o cuánto tiempo se debe esperar la señal CTS tras habilitar RTS (0 a 65535 ms). Este parámetro se evalúa sólo si está habilitado el control de flujo. (Valor predeterminado: 2000)
DONE	OUT	Bool	TRUE durante una ejecución tras haberse finalizado la última petición sin error
ERROR	OUT	Bool	TRUE durante una ejecución tras haberse finalizado la última petición con un error
STATUS	OUT	Word	Código de condición de ejecución (valor predeterminado: 0)


12.3 Comunicación punto a punto (PtP)

Tabla 12- 9 Códigos de condición

STATUS (W#16#....)	Descripción
80A0	El protocolo indicado no existe.
80A1	La velocidad de transferencia indicada no existe.
80A2	La opción de paridad indicada no existe.
80A3	El número de bits de datos indicado no existe.
80A4	El número de bits de parada indicado no existe.
80A5	El tipo de control de flujo indicado no existe.
80A6	El tiempo de espera es 0 y el control de flujo está habilitado
80A7	XON y XOFF son valores no permitidos (p. ej. el mismo valor)

12.3.1.3 Instrucción SEND_CFG

Tabla 12- 10 Instrucción SEND_CFG (configuración de transmisión)

KOP / FUP	SCL	Descripción
	<pre>"SEND_CFG_DB" (REQ:=_bool_in_, PORT:=_uint_in_, RTSONDLY:=_uint_in_, RTSOFFDLY:=_uint_in_, BREAK:=_uint_in_, IDLELINE:=_uint_in_, DONE=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_);</pre>	<p>La instrucción SEND_CFG permite configurar dinámicamente los parámetros de transmisión serie de un puerto de comunicación PtP. Todos los mensajes en cola de espera en un CM o una CB se rechazarán cuando se ejecute SEND_CFG.</p>

¹ STEP 7 crea el DB automáticamente al insertar la instrucción.

La configuración estática inicial del puerto puede ajustarse en las propiedades de la configuración de dispositivos. Como alternativa, es posible utilizar los valores predeterminados. La instrucción SEND_CFG puede ejecutarse en el programa con el fin de modificar la configuración.

Los cambios de la configuración de SEND_CFG no se almacenan de forma permanente en la CPU. Los parámetros ajustados en la configuración de dispositivos se restablecen cuando la CPU cambia de RUN a STOP y tras desconectar y volver a conectar la alimentación. Véase Configurar los parámetros de transmisión (Página 614).

Tabla 12- 11 Tipos de datos para los parámetros

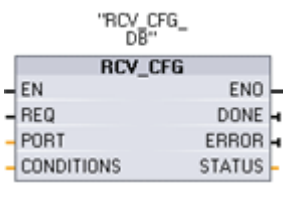
Parámetro y tipo		Tipo de datos	Descripción
REQ	IN	Bool	Activa el cambio de configuración cuando se produce un flanco ascendente en esta entrada. (Valor predeterminado: False)
PORT	IN	PORT	Tras haber instalado y configurado un dispositivo de comunicación CM o CB, el identificador de puerto aparece en la lista desplegable de parámetros disponible en la conexión del cuadro PUERTO. El valor de puerto CM o CB asignado es la propiedad de configuración del dispositivo "identificador de hardware". El nombre simbólico del puerto se asigna en la ficha "Constantes del sistema" de la tabla de variables PLC. (Valor predeterminado: 0)
RTSONDLY	IN	UInt	Número de milisegundos que debe esperarse tras la habilitación de RTS antes de cualquier transmisión de datos Tx. Este parámetro sólo es válido si está habilitado el control de flujo por hardware. El rango válido es de 0 a 65535 ms. Un valor 0 desactiva la función. (Valor predeterminado: 0)
RTSOFFDLY	IN	UInt	Número de milisegundos que se debe esperar después de la transmisión de datos Tx antes de que se inhíba RTS: Este parámetro sólo es válido si está habilitado el control de flujo por hardware. El rango válido es de 0 a 65535 ms. Un valor 0 desactiva la función. (Valor predeterminado: 0)
BREAK	IN	UInt	Este parámetro indica que se enviará una pausa tras el inicio de cada mensaje durante el número de tiempos de bit indicado. El máximo es 65535 tiempos de bit hasta ocho segundos como máximo. Un valor 0 desactiva la función. (Valor predeterminado: 12)
IDLELINE	IN	UInt	Este parámetro indica que la línea permanecerá inactiva durante el número de tiempos de bit indicado hasta el inicio de cada mensaje. El máximo es 65535 tiempos de bit hasta ocho segundos como máximo. Un valor 0 desactiva la función. (Valor predeterminado: 12)
DONE	OUT	Bool	TRUE durante una ejecución tras haberse finalizado la última petición sin error
ERROR	OUT	Bool	TRUE durante una ejecución tras haberse finalizado la última petición con un error
STATUS	OUT	Word	Código de condición de ejecución (valor predeterminado: 0)

Tabla 12- 12 Códigos de condición

STATUS (W#16#....)	Descripción
80B0	No se permite configurar una alarma de transmisión. Póngase en contacto con el Customer Support.
80B1	El tiempo de pausa excede el valor máximo permitido.
80B2	El tiempo de inactividad excede el valor máximo permitido.

12.3.1.4 Instrucción RCV_CFG

Tabla 12- 13 Instrucción RCV_CFG (configuración de recepción)

KOP / FUP	SCL	Descripción
	<pre>"RCV_CFG_DB" (REQ:=_bool_in_, PORT:=_uint_in_, CONDITIONS:=_struct_in_, DONE=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_);</pre>	<p>La instrucción RCV_CFG permite configurar dinámicamente los parámetros de recepción serie de un puerto de comunicación PtP. Esta instrucción configura las condiciones que indican el inicio y fin de un mensaje recibido. Todos los mensajes en cola de espera en un CM o una CB se rechazarán cuando se ejecute RCV_CFG.</p>

¹ STEP 7 crea el DB automáticamente al insertar la instrucción.

La configuración estática inicial del puerto de comunicación puede ajustarse en las propiedades de la configuración de dispositivos. Como alternativa, es posible utilizar los valores predeterminados. La instrucción RCV_CFG puede ejecutarse en el programa con el fin de modificar la configuración.

Los cambios de la configuración de RCV_CFG no se almacenan de forma permanente en la CPU. Los parámetros ajustados en la configuración de dispositivos se restablecen cuando la CPU cambia de RUN a STOP y tras desconectar y volver a conectar la alimentación. Encontrará más información en Configurar los parámetros de recepción (Página 614).

Tabla 12- 14 Tipos de datos para los parámetros

Parámetro y tipo	Tipo de datos	Descripción	
REQ	IN	Bool	Activa el cambio de configuración cuando se produce un flanco ascendente en esta entrada. (Valor predeterminado: False)
PORT	IN	PORT	Tras haber instalado y configurado un dispositivo de comunicación CM o CB, el identificador de puerto aparece en la lista desplegable de parámetros disponible en la conexión del cuadro PUERTO. El valor de puerto CM o CB asignado es la propiedad de configuración del dispositivo "identificador de hardware". El nombre simbólico del puerto se asigna en la ficha "Constantes del sistema" de la tabla de variables PLC. (Valor predeterminado: 0)
CONDITIONS	IN	CONDITIONS	La estructura de datos CONDITIONS indica las condiciones de inicio y fin del mensaje, como se describe a continuación.
DONE	OUT	Bool	TRUE durante un ciclo tras haberse finalizado la última petición sin error
ERROR	OUT	Bool	TRUE durante un ciclo tras haberse finalizado la última petición con un error
STATUS	OUT	Word	Código de condición de ejecución (valor predeterminado: 0)

Condiciones de inicio de la instrucción RCV_PTP

La instrucción RCV_PTP utiliza la configuración indicada por la instrucción RCV_CFG para determinar el inicio y fin de los mensajes de comunicación punto a punto. Las condiciones de inicio determinan el inicio de un mensaje. Una o más condiciones de inicio pueden determinar el inicio de un mensaje. Si se especifica más de una condición de inicio, todas las condiciones deberán cumplirse antes de iniciar el mensaje.

En el apartado "Configurar los parámetros de recepción (Página 616)" encontrará una descripción de las condiciones de inicio del mensaje.

Estructura de los tipos de datos del parámetro CONDITIONS, 1ª parte (condiciones de inicio)

Tabla 12- 15 Estructura de CONDITIONS para condiciones de inicio

Parámetro y tipo		Tipo de datos	Descripción
STARTCOND	IN	UInt	Indica la condición de inicio (valor predeterminado: 1) <ul style="list-style-type: none"> • 01H - Carácter inicial • 02H - Cualquier carácter • 04H - Line Break • 08H - Idle Line • 10H - Secuencia 1 • 20H - Secuencia 2 • 40H - Secuencia 3 • 80H - Secuencia 4
IDLETIME	IN	UInt	Número de tiempos de bit necesario para el timeout por "idle line". (Valor predeterminado: 40). Se utiliza únicamente para una condición de "idle line". 0 a 65535
STARTCHAR	IN	Byte	Carácter de inicio utilizado con la condición de carácter de inicio. (Valor predeterminado: B#16#2)
STRSEQ1CTL	IN	Byte	Ignorar/comparar el control de cada carácter de la secuencia 1: (Valor predeterminado: B#16#0) Estos son los bits de habilitación de cada carácter de la secuencia de inicio. <ul style="list-style-type: none"> • 01H - Carácter 1 • 02H - Carácter 2 • 04H - Carácter 3 • 08H - Carácter 4 • 10H - Carácter 5 Si se inhibe el bit asociado a un carácter, cualquier carácter será válido en esta posición de la secuencia.
STRSEQ1	IN	Char[5]	Caracteres de inicio de la secuencia 1 (5 caracteres). Valor predeterminado: 0
STRSEQ2CTL	IN	Byte	Ignorar/comparar el control de cada carácter de la secuencia 2. Valor predeterminado: B#16#0)
STRSEQ2	IN	Char[5]	Caracteres de inicio de la secuencia 2 (5 caracteres). Valor predeterminado: 0
STRSEQ3CTL	IN	Byte	Ignorar/comparar el control de cada carácter de la secuencia 3. Valor predeterminado: B#16#0
STRSEQ3	IN	Char[5]	Caracteres de inicio de la secuencia 3 (5 caracteres). Valor predeterminado: 0

Parámetro y tipo		Tipo de datos	Descripción
STRSEQ4CTL	IN	Byte	Ignorar/comparar el control de cada carácter de la secuencia 4. Valor predeterminado: B#16#0
STRSEQ4	IN	Char[5]	Caracteres de inicio de la secuencia 4 (5 caracteres), valor predeterminado: 0

Ejemplo

Considere el siguiente mensaje recibido en código hexadecimal: "68 10 aa 68 bb 10 aa 16" y las secuencias de inicio configuradas que muestra la tabla siguiente. Las secuencias de inicio se comienzan a evaluar al recibirse correctamente el primer carácter 68H. Tras recibirse correctamente el cuarto carácter (el segundo 68H), se cumple la condición de inicio 1. Una vez cumplidas las condiciones de inicio, se iniciará la evaluación de las condiciones de fin.

El procesamiento de la secuencia de inicio puede cancelarse debido a distintos errores de paridad, trama o tiempo excedido entre caracteres. Debido a estos errores, no se recibe el mensaje puesto que no se ha cumplido la condición de inicio.

Tabla 12- 16 Condiciones de inicio

Condición de inicio	Primer carácter	Primer carácter +1	Primer carácter +2	Primer carácter +3	Primer carácter +4
1	68H	xx	xx	68H	xx
2	10H	aaH	xx	xx	xx
3	dcH	aaH	xx	xx	xx
4	e5H	xx	xx	xx	xx

Condiciones de fin de la instrucción RCV_PTP

Las condiciones de fin configuradas determinan el fin de un mensaje. La primera aparición de una o más condiciones de fin configuradas determina el fin de un mensaje. La sección "Condiciones de fin del mensaje" del apartado "Configurar los parámetros de recepción (Página 616)" describe las condiciones de fin que se pueden configurar en la instrucción RCV_CFG.

Las condiciones de fin se pueden configurar tanto en las propiedades de la interfaz de comunicación en la configuración de dispositivos como desde la instrucción RCV_CFG. Cada vez que la CPU realiza una transición de STOP a RUN, los parámetros de recepción (condiciones de inicio y fin) vuelven a los ajustes de la configuración de dispositivos. Si el programa de usuario de STEP 7 ejecuta la instrucción RCV_CFG, los ajustes cambiarán a las condiciones de RCV_CFG.

Estructura de los tipos de datos del parámetro CONDITIONS, 2ª parte (condiciones de fin)

Tabla 12- 17 Estructura de CONDITIONS para condiciones de fin

Parámetro	Tipo de parámetro	Tipo de datos	Descripción
ENDCOND	IN	UInt 0	Este parámetro define la condición de fin del mensaje: <ul style="list-style-type: none"> • 01H -Tiempo de respuesta • 02H -Tiempo del mensaje • 04H - Tiempo excedido entre caracteres • 08H - Longitud máxima • 10H - N + LEN + M • 20H - Secuencia
MAXLEN	IN	UInt 1	Longitud máxima del mensaje: Se utiliza sólo si está seleccionada la condición de fin "Longitud máxima". 1 a 1024 bytes
N	IN	UInt 0	Posición de byte dentro del campo de longitud en el mensaje. Se utiliza sólo con la condición de fin N + LEN + M. 1 a 1022 bytes
LENGTHSIZE	IN	UInt 0	Tamaño del campo de byte (1, 2 ó 4 bytes). Se utiliza sólo con la condición de fin N + LEN + M.
LENGTHM	IN	UInt 0	Indica el número de caracteres posteriores al campo de longitud que no se incluyen en el valor del campo de longitud. Se utiliza sólo con la condición de fin N + LEN + M. 0 a 255 bytes
RCVTIME	IN	UInt 200	Indica cuánto tiempo debe esperarse hasta que se reciba el primer carácter. La recepción se terminará con un error si no se recibe correctamente ningún carácter dentro del tiempo indicado. Se utiliza sólo con la condición de fin "Tiempo de respuesta". (de 0 a 65535 tiempos de bit con un máximo de 8 segundos) Este parámetro no es una condición de fin del mensaje dado que la evaluación finaliza cuando se recibe el primer carácter de una respuesta. Es una condición de fin sólo en el sentido de que finaliza una operación de recepción debido a que no se recibe ninguna respuesta cuando se espera una. Se debe seleccionar una condición de fin distinta.
MSGTIME	IN	UInt 200	Indica cuánto tiempo debe esperarse hasta la recepción completa del mensaje una vez recibido el primer carácter. Este parámetro se utiliza sólo si está seleccionada la condición de fin "Tiempo de mensaje excedido". (de 0 a 65535 milisegundos)
CHARGAP	IN	UInt 12	Indica el número de tiempos de bit entre caracteres. Si el número de tiempos de bit entre caracteres excede el valor especificado, se cumplirá la condición de fin. Se utiliza sólo con la condición de fin "Tiempo excedido entre caracteres". (de 0 a 65535 tiempos de bit hasta 8 segundos máx.)

12.3 Comunicación punto a punto (PtP)

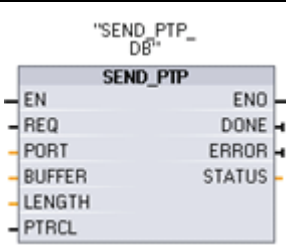
Parámetro	Tipo de parámetro	Tipo de datos	Descripción
ENDSEQ1CTL	IN	Byte B#16#0	Ignorar/comparar el control de cada carácter de la secuencia 1: Estos son los bits de habilitación de cada carácter de la secuencia de fin. El carácter 1 es el bit 0, el carácter 2 es el bit 1, ..., el carácter 5 es el bit 4. Si se inhibe el bit asociado a un carácter, cualquier carácter será válido en esta posición de la secuencia.
ENDSEQ1	IN	Char[5] 0	Caracteres de inicio de la secuencia 1 (5 caracteres)

Tabla 12- 18 Códigos de condición

STATUS (W#16#....)	Descripción
80C0	Se ha seleccionado una condición de inicio no permitida
80C1	Se ha seleccionado una condición de fin no permitida o no se ha seleccionado ninguna condición de fin
80C2	Hay una alarma de recepción habilitada, pero esto no es posible.
80C3	La condición de fin "Longitud máxima" está habilitada y la longitud máxima es 0 o > 1024.
80C4	La longitud calculada está habilitada y N es >= 1023.
80C5	La longitud calculada está habilitada y la longitud no es 1, 2 ó 4.
80C6	La longitud calculada está habilitada y el valor M es > 255.
80C7	La longitud calculada está habilitada y es > 1024.
80C8	El timeout de respuesta está habilitado y es igual a cero.
80C9	El tiempo excedido entre caracteres está habilitado y es igual a cero.
80CA	El timeout por "idle line" está habilitado y es igual a cero.
80CB	La secuencia de fin está habilitada, pero todos los caracteres son irrelevantes.
80CC	La secuencia de inicio (una cualquiera de 4) está habilitada, pero todos los caracteres son irrelevantes.

12.3.1.5 Instrucción SEND_PTP

Tabla 12- 19 Instrucción SEND_PTP (enviar datos punto a punto)

KOP / FUP	SCL	Descripción
	<pre>"SEND_PTP_DB" (REQ:=_bool_in_, PORT:=_uint_in_, BUFFER:=_variant_in_, LENGTH:=_uint_in_, PTRCL:=_bool_in_, DONE=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_);</pre>	<p>La instrucción SEND_PTP inicia la transferencia de datos y transmite el búfer asignado a la interfaz de comunicación. El programa de la CPU se sigue ejecutando hasta que el CM o la CB envía los datos a la velocidad de transferencia asignada. Sólo una operación de transmisión puede estar pendiente en un momento dado. El CM o la CB devuelve un error si una segunda instrucción SEND_PTP se ejecuta mientras el CM o la CB está transmitiendo un mensaje.</p>

1 STEP 7 crea el DB automáticamente al insertar la instrucción.

Tabla 12- 20 Tipos de datos para los parámetros

Parámetro y tipo		Tipo de datos	Descripción
REQ	IN	Bool	Activa la transmisión solicitada cuando se produce un flanco ascendente en esta entrada de habilitación de la transmisión. Esto inicia la transferencia del contenido del búfer a la interfaz de comunicación punto a punto. (Valor predeterminado: False)
PORT	IN	PORT	Tras haber instalado y configurado un dispositivo de comunicación CM o CB, el identificador de puerto aparece en la lista desplegable de parámetros disponible en la conexión del cuadro PUERTO. El valor de puerto CM o CB asignado es la propiedad de configuración del dispositivo "identificador de hardware". El nombre simbólico del puerto se asigna en la ficha "Constantes del sistema" de la tabla de variables PLC. (Valor predeterminado: 0)
BUFFER	IN	Variante	Este parámetro apunta a la dirección inicial del búfer de transmisión. (Valor predeterminado: 0) Nota: no se soportan los datos booleanos ni las matrices booleanas.
LENGTH	IN	UInt	Longitud de la trama transmitida en bytes (valor predeterminado: 0) Al transmitir una estructura compleja, utilice siempre la longitud 0.
PTRCL	IN	Bool	Este parámetro selecciona el búfer para la comunicación punto a punto normal o los protocolos suministrados por Siemens que están implementados en el CM o la CB acoplada. (Valor predeterminado: False) FALSE = operaciones punto a punto controladas por el programa de usuario. (única opción válida)
DONE	OUT	Bool	TRUE durante un ciclo tras haberse finalizado la última petición sin error
ERROR	OUT	Bool	TRUE durante un ciclo tras haberse finalizado la última petición con un error
STATUS	OUT	Word	Código de condición de ejecución (valor predeterminado: 0)

El estado lógico de las salidas DONE y ERROR es FALSE durante la operación de transmisión. Una vez finalizada la operación de transmisión, la salida DONE o ERROR se pone a TRUE para mostrar el estado de la transmisión. Mientras que DONE o ERROR sea TRUE, la salida STATUS será válida.

La instrucción devuelve el estado 16#7001 si la interfaz de comunicación acepta los datos transmitidos. Las ejecuciones posteriores de SEND_PTP devuelven 16#7002 si el CM o la CB aún está ocupada transmitiendo datos. Una vez finalizada la operación de transmisión, el CM o la CB devuelve el estado de transmisión como 16#0000 (si no han ocurrido errores). Las ejecuciones posteriores de SEND_PTP con REQ puesto a "low" devuelven el estado 16#7000 (no ocupado).

12.3 Comunicación punto a punto (PtP)

Los diagramas siguientes muestran la relación entre los valores de salida y REQ. Para esto se supone que la instrucción se llama periódicamente para comprobar el estado del proceso de transmisión. En el diagrama siguiente se supone que la instrucción se llama en cada ciclo (representado por los valores de STATUS).

REQ							
DONE							
ERROR							
STATUS	7000H	7001H	7002H	7002H	7002H	0000H	7000H

El diagrama siguiente muestra cómo los parámetros DONE y STATUS son válidos sólo durante un ciclo si la línea REQ se impulsa (durante un ciclo) para iniciar la operación de transmisión.

REQ								
DONE								
ERROR								
STATUS	7000H	7001H	7002H	7002H	7002H	0000H	7000H	7000H

El diagrama siguiente muestra la relación de los parámetros DONE, ERROR y STATUS cuando se presenta un error.

REQ								
DONE								
ERROR								
STATUS	7000H	7001H	7002H	7002H	7002H	80D1H	7000H	7000H

Los valores DONE, ERROR y STATUS sólo son válidos hasta que se ejecute nuevamente SEND_PTP con el mismo DB de instancia.

Tabla 12- 21 Códigos de condición

STATUS (W#16#....)	Descripción
80D0	Nueva petición estando activo el transmisor
80D1	Transmisión cancelada debido a que no se ha detectado ningún CTS durante el tiempo de espera
80D2	Transmisión cancelada debido a que no se ha recibido DSR del dispositivo DCE
80D3	Transmisión cancelada debido a desbordamiento de la cola de espera (transmisión de más de 1024 bytes)
80D5	Señal de bias inverso (condición de rotura de hilo)
833A	El DB para el parámetro BUFFER no existe.

Interacción de los parámetros LENGTH y BUFFER para SEND_PTP

El tamaño mínimo de los datos que pueden transmitirse con la instrucción SEND_PTP es un byte. El parámetro BUFFER determina el tamaño de los datos por transmitir. No puede utilizarse el tipo de datos Bool ni matrices de Bool para el parámetro BUFFER.

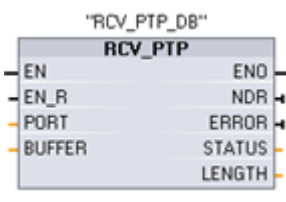
El parámetro LENGTH se puede ajustar siempre en 0 para garantizar que SEND_PTP envíe la estructura completa de datos representada por el parámetro BUFFER. Si sólo desea enviar una parte de una estructura de datos del parámetro BUFFER, LENGTH se puede ajustar como se indica a continuación:

Tabla 12- 22 Parámetros LENGTH y BUFFER

LENGTH	BUFFER	Descripción
= 0	No utilizado	Los datos completos se envían según se define en el parámetro BUFFER. No es necesario especificar el número de bytes transmitidos si LENGTH = 0.
> 0	Tipo de datos simple	El valor de LENGTH debe contener el conteo de bytes de este tipo de datos. Por ejemplo, para un valor Word, el valor LENGTH debe ser dos. Para Dword o Real, el valor LENGTH debe ser cuatro. De lo contrario, no se transferirá nada y se devolverá el error 8088H.
	Estructura	El valor LENGTH puede contener un conteo de bytes menor que la longitud total de bytes de la estructura, en cuyo caso sólo los primeros bytes de LENGTH de la estructura se envían desde BUFFER. Dado que la organización interna de bytes de una estructura no siempre se puede determinar, es posible que se obtengan resultados inesperados. En este caso, utilice un LENGTH de 0 para enviar la estructura completa.
	Matriz	El valor LENGTH debe contener un conteo de bytes que sea menor que la longitud total de bytes de la matriz y que sea múltiplo del conteo de bytes del elemento de datos. Por ejemplo, el parámetro LENGTH para una matriz de Words debe ser un múltiplo de dos y para una matriz de Reals, un múltiplo de cuatro. Cuando se especifica LENGTH, el número de elementos de matriz contenidos en bytes de LENGTH se transfiere. Si su BUFFER, por ejemplo, contiene una matriz de 15 Dwords (60 bytes totales) y se especifica un valor de 20 para LENGTH, entonces las primeras cinco Dwords de la matriz se transfieren. El valor de LENGTH debe ser un múltiplo del conteo de bytes del elemento de datos. De lo contrario, STATUS = 8088H, ERROR = 1 y la transmisión no se efectuará.
	String	El parámetro LENGTH contiene el número de caracteres que debe transmitirse. Sólo se transmiten los caracteres del String. Los bytes de longitud máxima y real del String no se transmiten.

12.3.1.6 Instrucción RCV_PTP

Tabla 12- 23 Instrucción RCV_PTP (recepción punto a punto)

KOP / FUP	SCL	Descripción
	<pre>"RCV_PTP_DB" (EN_R:=_bool_in_, PORT:=_uint_in_, BUFFER:=_variant_in_, NDR=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_, LENGTH=>_uint_out_);</pre>	<p>RCV_PTP comprueba si se han recibido mensajes en el CM o la CB. Si hay un mensaje disponible, se transfiere del CM o de la CB a la CPU. Un error devuelve el valor de STATUS apropiado.</p>

¹ STEP 7 crea el DB automáticamente al insertar la instrucción.

Tabla 12- 24 Tipos de datos para los parámetros

Parámetro y tipo	Tipo de datos	Descripción	
EN_R	IN	Bool	Si esta entrada es TRUE y un mensaje está disponible, el mensaje se transferirá del CM o de la CB a BUFFER. Si EN_R es FALSE, se comprobará si el CM o la CB ha recibido mensajes y se actualizarán las salidas NDR, ERROR y STATUS, pero el mensaje no se transferirá a BUFFER. (Valor predeterminado: 0)
PORT	IN	PORT	Tras haber instalado y configurado un dispositivo de comunicación CM o CB, el identificador de puerto aparece en la lista desplegable de parámetros disponible en la conexión del cuadro PUERTO. El valor de puerto CM o CB asignado es la propiedad de configuración del dispositivo "identificador de hardware". El nombre simbólico del puerto se asigna en la ficha "Constantes del sistema" de la tabla de variables PLC. (Valor predeterminado: 0)
BUFFER	IN	Variante	Este parámetro apunta a la dirección inicial del búfer de recepción. Este búfer debería ser lo suficientemente grande para recibir el mensaje de longitud máxima. No se soportan los datos booleanos ni las matrices booleanas. (Valor predeterminado: 0)
NDR	OUT	Bool	TRUE durante una ejecución cuando se dispone de datos nuevos y la operación ha finalizado sin errores.
ERROR	OUT	Bool	TRUE durante una ejecución tras haberse finalizado la operación con un error.
STATUS	OUT	Word	Código de condición de ejecución (valor predeterminado: 0)
LENGTH	OUT	UInt	Longitud del mensaje devuelto (en bytes) (valor predeterminado: 0)

El valor de STATUS es válido si NDR o ERROR es TRUE. El valor STATUS proporciona el motivo para finalizar la recepción en el CM o la CB. Por lo general, este valor es positivo, lo cual indica que la recepción se realizó correctamente y que el proceso de recepción finalizó normalmente. Si el valor de STATUS es negativo (es decir, si está activado el bit más significativo del valor hexadecimal), esto indica que la recepción se ha finalizado debido a un error p. ej. de paridad, trama o desbordamiento.

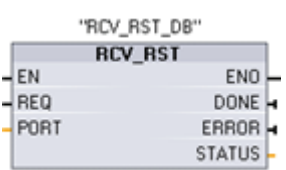
Cada interfaz de comunicación PtP puede almacenar como máximo 1024 bytes en un búfer. Esto podría ser un mensaje grande o varios mensajes pequeños. Si más de un mensaje está disponible en el CM o en la CB, la instrucción RCV_PTP devolverá el mensaje más antiguo disponible. La siguiente ejecución de la instrucción RCV_PTP devolverá el siguiente mensaje más antiguo disponible.

Tabla 12- 25 Códigos de condición

STATUS (W#16#...)	Descripción
0000	No hay ningún búfer disponible
80E0	Mensaje terminado debido a que el búfer de recepción está lleno
80E1	Mensaje terminado debido a un error de paridad
80E2	Mensaje terminado debido a un error de trama
80E3	Mensaje terminado debido a un error de desbordamiento
80E4	Mensaje terminado porque la longitud calculada excede el tamaño del búfer
80E5	Señal de bias inverso (condición de rotura de hilo)
0094	Mensaje terminado porque se ha recibido la longitud máxima de caracteres
0095	Mensaje terminado debido a timeout del mensaje
0096	Mensaje terminado porque se ha excedido el tiempo entre caracteres
0097	Mensaje terminado debido a timeout de respuesta
0098	Mensaje terminado porque se ha cumplido la condición de longitud "N+LEN+M"
0099	Mensaje terminado porque se ha cumplido la condición de secuencia de fin
833A	El DB para el parámetro BUFFER no existe.

12.3.1.7 Instrucción RCV_RST

Tabla 12- 26 Instrucción RCV_RST (inicializar receptor)

KOP / FUP	SCL	Descripción
	<pre>"RCV_RST_DB" (REQ:=_bool_in_, PORT:=_uint_in_, DONE=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_);</pre>	<p>La instrucción RCV_RST borra los búferes de recepción en el CM o la CB.</p>

¹ STEP 7 crea el DB automáticamente al insertar la instrucción.

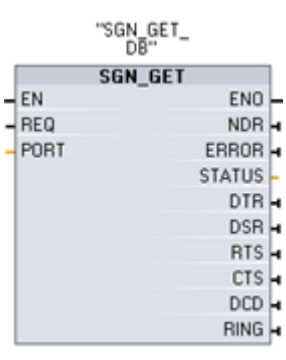
12.3 Comunicación punto a punto (PtP)

Tabla 12- 27 Tipos de datos para los parámetros

Parámetro y tipo		Tipo de datos	Descripción
REQ	IN	Bool	Activa la inicialización del receptor cuando se produce un flanco ascendente en esta entrada de habilitación (valor predeterminado: False)
PORT	IN	PORT	Tras haber instalado y configurado un dispositivo de comunicación CM o CB, el identificador de puerto aparece en la lista desplegable de parámetros disponible en la conexión del cuadro PUERTO. El valor de puerto CM o CB asignado es la propiedad de configuración del dispositivo "identificador de hardware". El nombre simbólico del puerto se asigna en la ficha "Constantes del sistema" de la tabla de variables PLC. (Valor predeterminado: 0)
DONE	OUT	Bool	Si es TRUE durante un ciclo, indica que la última petición se ha finalizado sin errores.
ERROR	OUT	Bool	Si es TRUE, indica que la última petición se ha finalizado con errores. Además, si esta salida es TRUE, la salida STATUS contendrá los códigos de error respectivos.
STATUS	OUT	Word	Código de error (valor predeterminado: 0)

12.3.1.8 Instrucción SGN_GET

Tabla 12- 28 Instrucción SGN_GET (consultar señales RS232)

KOP / FUP	SCL	Descripción
	<pre>"SGN_GET_DB" (REQ:= _bool_in_, PORT:= _uint_in_, NDR=> _bool_out_, ERROR=> _bool_out_, STATUS=> _word_out_, DTR=> _bool_out_, DSR=> _bool_out_, RTS=> _bool_out_, CTS=> _bool_out_, DCD=> _bool_out_, RING=> _bool_out_);</pre>	<p>La instrucción SGN_GET lee los estados actuales de las señales de comunicación RS232.</p> <p>Esta función es válida sólo para el CM RS232.</p>

¹ STEP 7 crea el DB automáticamente al insertar la instrucción.

Tabla 12- 29 Tipos de datos para los parámetros

Parámetro y tipo		Tipo de datos	Descripción
REQ	IN	Bool	Los valores de las señales RS232 se consultan cuando se produce un flanco ascendente en esta entrada (valor predeterminado: False)
PORT	IN	PORT	Tras haber instalado y configurado un dispositivo de comunicación CM o CB, el identificador de puerto aparece en la lista desplegable de parámetros disponible en la conexión del cuadro PUERTO. El valor de puerto CM o CB asignado es la propiedad de configuración del dispositivo "identificador de hardware". El nombre simbólico del puerto se asigna en la ficha "Constantes del sistema" de la tabla de variables PLC.

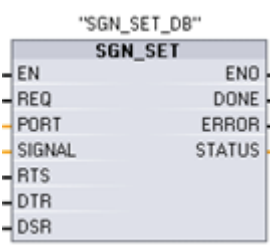
Parámetro y tipo		Tipo de datos	Descripción
NDR	OUT	Bool	TRUE durante un ciclo cuando se dispone de datos nuevos y la operación ha finalizado sin errores
ERROR	OUT	Bool	TRUE durante un ciclo tras haberse finalizado la operación con un error
STATUS	OUT	Word	Código de condición de ejecución (valor predeterminado: 0)
DTR	OUT	Bool	Terminal de datos disponible, módulo listo (salida). Valor predeterminado: False
DSR	OUT	Bool	Equipo de datos listo, interlocutor listo (entrada). Valor predeterminado: False
RTS	OUT	Bool	Petición de transmitir, módulo listo para transmitir (salida). Valor predeterminado: False
CTS	OUT	Bool	Listo para transmitir; el interlocutor puede recibir datos (entrada). Valor predeterminado: False
DCD	OUT	Bool	Detección de portadora de datos, nivel de señal de recepción (siempre False, no se soporta)
RING	OUT	Bool	Indicación de timbre, señalización de una llamada entrante (siempre False, no se soporta)

Tabla 12- 30 Códigos de condición

STATUS (W#16#....)	Descripción
80F0	El CM o la CB es un RS485; no hay señales disponibles

12.3.1.9 Instrucción SGN_SET

Tabla 12- 31 Instrucción SGN_SET (activar señales RS232)

KOP / FUP	SCL	Descripción
	<pre>"SGN_SET_DB" (REQ:=_bool_in_, PORT:=_uint_in_, SIGNAL:=_byte_in_, RTS:=_bool_in_, DTR:=_bool_in_, DSR:=_bool_in_, DONE=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_);</pre>	<p>La instrucción SGN_SET activa los estados de las señales de comunicación RS232. Esta función es válida sólo para el CM RS232.</p>

¹ STEP 7 crea el DB automáticamente al insertar la instrucción.

12.3 Comunicación punto a punto (PtP)

Tabla 12- 32 Tipos de datos para los parámetros

Parámetro y tipo		Tipo de datos	Descripción
REQ	IN	Bool	Inicia la activación de las señales RS232 cuando se produce un flanco ascendente en esta entrada (valor predeterminado: False)
PORT	IN	PORT	Tras haber instalado y configurado un dispositivo de comunicación CM o CB, el identificador de puerto aparece en la lista desplegable de parámetros disponible en la conexión del cuadro PUERTO. El valor de puerto CM o CB asignado es la propiedad de configuración del dispositivo "identificador de hardware". El nombre simbólico del puerto se asigna en la ficha "Constantes del sistema" de la tabla de variables PLC. (Valor predeterminado: 0)
SIGNAL	IN	Byte	Selecciona las señales que deben activarse: (se permiten varias). Valor predeterminado: 0 <ul style="list-style-type: none"> • 01H = Activar RTS • 02H = Activar DTR • 04H = Activar DSR
RTS	IN	Bool	Petición de transmitir, módulo listo para transmitir el valor que debe ajustarse (True o False), valor predeterminado: False
DTR	IN	Bool	Terminal de datos disponible, módulo listo para transmitir el valor que debe ajustarse (True o False). Valor predeterminado: False
DSR	IN	Bool	Equipo de datos listo (aplicable sólo a las interfaces DCE) (no se utiliza).
DONE	OUT	Bool	TRUE durante una ejecución tras haberse finalizado la última petición sin error
ERROR	OUT	Bool	TRUE durante una ejecución tras haberse finalizado la última petición con un error
STATUS	OUT	Word	Código de condición de ejecución (valor predeterminado: 0)

Tabla 12- 33 Códigos de condición

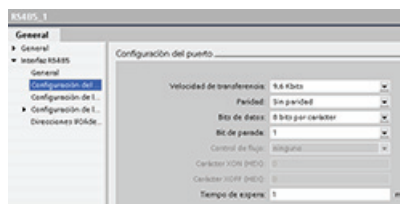
STATUS (W#16#...)	Descripción
80F0	El CM o la CB es un RS485; no pueden activarse señales
80F1	Las señales no pueden activarse debido al control de flujo por hardware
80F2	Imposible activar DSR porque el módulo es DTE
80F3	Imposible activar DTR porque el módulo es DCE

12.3.2 Configurar los puertos de comunicación

Las interfaces de comunicación pueden configurarse de dos maneras:

- Utilizando la configuración de dispositivos en STEP 7 para configurar los parámetros de puerto (velocidad de transferencia y paridad), así como los de transmisión y recepción. Los ajustes de la configuración de dispositivos se guardan en la CPU. Estos ajustes se aplican tras desconectar y conectar la alimentación y una transición de RUN a STOP.
- Utilice las instrucciones PORT_CFG (Página 594), SEND_CFG (Página 596) y RCV_CFG (Página 598) para configurar los parámetros. La configuración del puerto ajustada mediante las instrucciones es válida mientras la CPU esté en modo RUN. La configuración del puerto se inicializa con los valores predeterminados de la configuración del dispositivo tras cambiar a STOP o desconectar y volver a conectar la alimentación.

Tras configurar los dispositivos de hardware (Página 123) es preciso ajustar los parámetros de las interfaces de comunicación seleccionando uno de los CMs del rack o bien la CB, de estar configurada.



La ficha "Propiedades" de la ventana de inspección muestra los parámetros del CM o de la CB seleccionada. Seleccione "Configuración del puerto" para editar los parámetros siguientes:

- Velocidad de transferencia
- Paridad
- Bits de datos por carácter
- Número de bits de parada
- Control de flujo (sólo RS232)
- Tiempo de espera

Para el CM 1241 RS232 y RS485 y la CB RS485 (excepto el control de flujo, que sólo es soportado por el CM 1241 RS232), los parámetros de configuración del puerto son iguales, indistintamente de si se desea configurar un módulo de comunicación RS232 o RS485 o bien la placa de comunicación RS485. Los valores de los parámetros pueden diferir.

12.3 Comunicación punto a punto (PtP)

Para el CM 1241 RS422/485, la configuración del puerto se realiza de forma diferente, como se indica en la siguiente figura. El modo 422 del módulo CM 1241 RS422/485 también soporta el control de flujo por software.



Seleccione "Configuración del puerto" para editar los parámetros de RS422/485 siguientes:

- "Estado operativo":
 - Modo dúplex (RS422) de cuatro hilos (conexión punto a punto)
 - Modo dúplex (RS422) de cuatro hilos (maestro multipunto)
 - Modo dúplex (RS422) de cuatro hilos (esclavo multipunto)
 - Modo semidúplex (RS485) de dos hilos
- "Inicialización de la línea de recepción":
 - Ninguno
 - Polarización directa (señal R(A) 0V, señal R(B) 5V)

El programa de usuario de STEP 7 también puede configurar el puerto o modificar la configuración existente con la instrucción PORT_CFG (Página 594).

Nota

Los valores de parámetros ajustados mediante la instrucción PORT_CFG en el programa de usuario prevalecen sobre la configuración del puerto definida en la configuración de dispositivos. El S7-1200 no conserva los parámetros ajustados mediante la instrucción PORT_CFG si se desconecta la alimentación.

Parámetro	Definición
Velocidad de transferencia	El valor predeterminado para la velocidad de transferencia es 9,6 Kbits/s. Los ajustes posibles son: 300 baudios, 600 baudios, 1,2 Kbits, 2,4 Kbits, 4,8 Kbits, 9,6 Kbits, 19,2 Kbits, 38,4 Kbits, 57,6 Kbits, 76,8 Kbits y 115,2 Kbits.
Paridad	El valor predeterminado para la paridad es "sin paridad". Los ajustes posibles son: Sin paridad, par, impar, marca (bit de paridad siempre establecido a 1), espacio (bit de paridad siempre establecido a 0).
Bits de datos por carácter	Número de bits de datos de un carácter. Los valores válidos son 7 u 8.
Número de bits de parada	Es posible ajustar uno o dos bits de parada. El ajuste predeterminado es uno.
Control de flujo	Para el módulo de comunicación RS232 puede seleccionarse el control de flujo por hardware o software de la manera descrita en el apartado "Gestionar el control de flujo (Página 613)". Si se selecciona el control de flujo por hardware, es posible indicar si la señal RTS debe estar siempre ON o si debe conmutarse. Si se selecciona el control de flujo por software, es posible definir los caracteres XON y XOFF. Las interfaces de comunicación RS485 no soportan el control de flujo. El modo 422 del módulo CM 1241 RS422/485 soporta el control de flujo por software.

Parámetro	Definición
Tiempo de espera	El tiempo de espera especifica el periodo que el CM o la CB espera hasta recibir CTS tras confirmar RTS, o bien hasta recibir un XON tras recibir un XOFF, dependiendo del tipo de control de flujo. Si el tiempo de espera transcurre antes de que la interfaz de comunicación reciba un CTS o XON esperado, el CM o la CB cancelará la operación de transmisión y devolverá un error al programa de usuario. El tiempo de espera se indica en milisegundos. El rango válido está comprendido entre 0 y 65535 milisegundos.
Estado operativo	Selecciona el modo operativo de RS422 o RS485 y las configuraciones de red.
Inicialización de la línea de recepción	Selecciona las opciones de polarización. Los valores válidos son Ninguno, Polarización directa y Polarización inversa. La polarización inversa sirve para permitir la detección de rotura de cable.

12.3.2.1 Gestionar el control de flujo

El control de flujo es un mecanismo que permite regular el intercambio de datos entre un emisor y un receptor para evitar pérdidas de datos. El control de flujo garantiza que un emisor no envíe más información de la que el receptor es capaz de procesar. El control de flujo puede realizarse por hardware o software. El CM RS232 soporta el control de flujo tanto por hardware como por software. El CM y la CB RS485 no soportan el control de flujo. El modo 422 del módulo CM 1241 RS422/485 soporta el control de flujo por software. El tipo de control de flujo se especifica al configurar el puerto (Página 611) o con la instrucción PORT_CFG (Página 594).

El control de flujo por hardware funciona a través de las señales de comunicación RTS (Request To Send o petición de transmitir) y CTS (Clear To Send o listo para transmitir). En el caso del CM RS232, la señal RTS se emite desde el pin 7 y la señal CTS se recibe por el pin 8. El CM RS232 es un DTE (Data Terminal Equipment o equipo terminal de datos) que confirma RTS como salida y monitoriza CTS como entrada.

Control de flujo por hardware: RTS conmutado

Si se habilita el control de flujo por hardware con RTS conmutado en un CM RS232, el módulo activa la señal RTS para enviar datos. El módulo vigila la señal CTS para determinar si el receptor puede aceptar datos. Estando activa la señal CTS, el módulo puede transmitir datos mientras que la señal CTS permanezca activa. Si se desactiva la señal CTS, la transmisión debe detenerse.

La transmisión se reanuda cuando se reactiva la señal CTS. Si la señal CTS no se vuelve a activar dentro del tiempo de espera configurado, el módulo cancelará la transmisión y devolverá un error al programa de usuario. El tiempo de espera se especifica en la configuración del puerto (Página 611).

El control de flujo con RTS conmutado es útil para los dispositivos que requieren una señal de que la transmisión está activa. Un ejemplo sería un módem radio que utiliza RTS como señal "Key" para energizar el transmisor de radio. El control de flujo con RTS conmutado no funciona con módems telefónicos estándar. Seleccione la opción "RTS siempre on" para los módems telefónicos.

Control de flujo por hardware: RTS siempre ON

Si se utiliza la opción "RTS siempre on", el CM 1241 activará RTS de forma predeterminada. Un dispositivo (p. ej. un módem telefónico) vigila la señal RTS del CM y la utiliza como CTS (Clear To Send o listo para transmitir). El módem transmitirá al CM sólo si RTS está activo, es decir, cuando el módem telefónico detecte un CTS activo. Si RTS no está activo, el módem telefónico no transmitirá al CM.

Para permitir que el módem envíe datos al CM en cualquier momento, configure el control de flujo por hardware con "RTS siempre ON". De esta manera, el CM activa la señal RTS permanentemente. El CM no desactivará RTS incluso si el módulo no puede aceptar caracteres. El emisor debe garantizar que no se desborde el búfer de recepción del CM.

Uso de las señales DTR (Data Terminal Block Ready) y DSR (Data Set Ready)

El CM activa DTR para cualquier tipo de control de flujo por hardware. El módulo transmite datos sólo cuando se activa la señal DSR. El estado de DSR se evalúa únicamente al comienzo de la transmisión. Si DSR se desactiva tras haberse iniciado la transmisión, ésta no se detendrá.

Control de flujo por software

El control de flujo por software utiliza caracteres especiales en los mensajes para proporcionar el control de flujo. Se configuran caracteres hexadecimales que representan XON y XOFF.

XOFF indica que una transmisión debe detenerse. XON indica que una transmisión puede reanudarse. XOFF y XON no deben ser el mismo carácter.

Cuando el emisor recibe un carácter XOFF del receptor, deja de transmitir datos. La transmisión se reanuda cuando el emisor recibe un carácter XON. Si no recibe un carácter XON dentro del tiempo de espera indicado en la configuración del puerto (Página 611), el CM cancelará la transmisión y devolverá un error al programa de usuario.

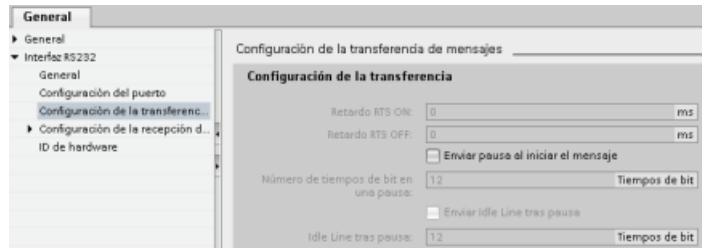
Para el control de flujo por software se requiere la comunicación dúplex, puesto que el receptor debe poder enviar XOFF al emisor durante una transmisión. El control de flujo por software sólo es posible en los mensajes que contengan únicamente caracteres ASCII. Los protocolos binarios no pueden utilizar el control de flujo por software.

12.3.3 Configurar los parámetros de transmisión y recepción

Para que la CPU pueda intervenir en la comunicación PtP es preciso configurar parámetros para transmitir y recibir mensajes. Estos parámetros determinan cómo deben funcionar las comunicaciones al transmitir o recibir mensajes a/de un dispositivo de destino.

12.3.3.1 Configurar los parámetros de transmisión

Desde la configuración de dispositivos se configura cómo debe transferir datos una interfaz de comunicación, definiendo las propiedades "Configuración de la transferencia de mensajes" para la interfaz seleccionada.



Los parámetros de transmisión de mensajes también se pueden configurar o modificar dinámicamente desde el programa de usuario utilizando la instrucción SEND_CFG (Página 596).

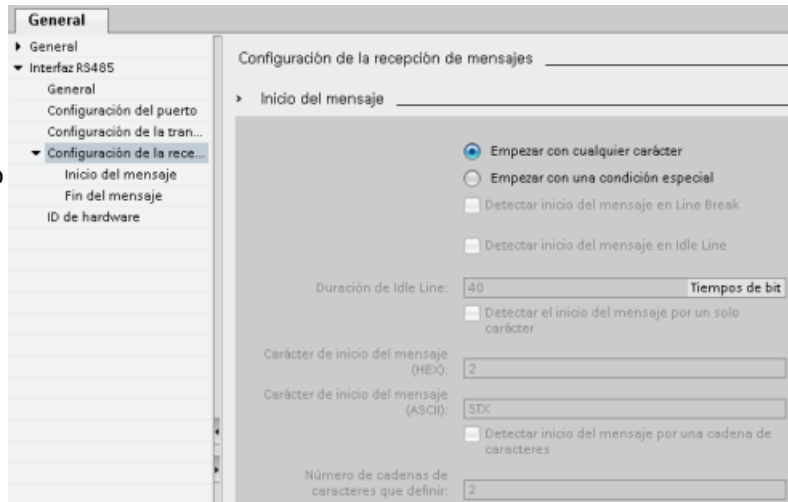
Nota

Los valores de parámetros ajustados mediante la instrucción SEND_CFG en el programa de usuario prevalecen sobre la configuración del puerto. La CPU no conserva los parámetros ajustados mediante la instrucción SEND_CFG si se desconecta la alimentación.

Parámetro	Definición
Retardo RTS ON	Determina el tiempo que debe esperarse tras activar RTS antes de iniciar la transmisión. El rango está comprendido entre 0 y 65535 ms (el valor predeterminado es 0). Este parámetro sólo es válido si en la configuración del puerto (Página 611) se ha definido el control de flujo por hardware. CTS se evalúa una vez transcurrido el retardo RTS ON. Este parámetro es aplicable únicamente a los módulos RS232.
Retardo RTS OFF	Determina el tiempo que debe esperarse antes de desactivar RTS tras finalizar la transmisión. El rango está comprendido entre 0 y 65535 ms (el valor predeterminado es 0). Este parámetro sólo es válido si en la configuración del puerto (Página 611) se ha definido el control de flujo por hardware. Este parámetro es aplicable únicamente a los módulos RS232.
Enviar pausa al inicio del mensaje Número de bit times en una pausa	Determina que, al inicio de cada mensaje, se enviará una pausa una vez transcurrido el tiempo de retardo RTS ON (si se ha configurado) y si CTS está activo. Es preciso indicar cuántos tiempos de bit ("bit times") constituyen una pausa cuando la línea se mantiene en una condición "Space". El ajuste predeterminado es 12 y el valor máximo es 65535, hasta un límite de ocho segundos.
Enviar Idle Line tras la pausa Idle Line tras pausa	Determina que se enviará una "idle line" antes del inicio del mensaje. Ésta se enviará tras una pausa, siempre y cuando esté configurada una pausa. El parámetro "Idle Line tras pausa" indica cuántos tiempos de bit constituyen una "idle line" cuando la línea se mantiene en una condición "Mark". El ajuste predeterminado es 12 y el valor máximo es 65535, hasta un límite de ocho segundos.

12.3.3.2 Configurar los parámetros de recepción

En la configuración de dispositivos se determina cómo debe recibir datos una interfaz de comunicación, así como detectar el comienzo y fin de un mensaje. Estos parámetros se definen en la configuración de la recepción de mensajes para la interfaz seleccionada.



Los parámetros de recepción de mensajes también se pueden configurar o modificar dinámicamente desde el programa de usuario utilizando la instrucción RCV_CFG (Página 598).

Nota

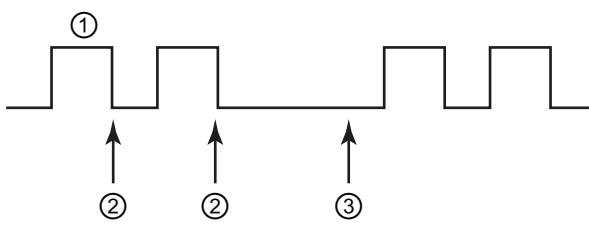
Los valores de parámetros ajustados mediante la instrucción RCV_CFG en el programa de usuario prevalecen sobre la configuración del puerto. La CPU no conserva los parámetros ajustados mediante la instrucción RCV_CFG si se desconecta la alimentación.

Condiciones de inicio del mensaje

Es posible determinar cómo la interfaz de comunicación debe detectar el inicio de un mensaje. Los caracteres iniciales y los caracteres que contiene el mensaje se depositan en el búfer de recepción hasta que se cumpla una condición de fin configurada.

Es posible definir varias condiciones de inicio. Si se especifica más de una condición de inicio, todas las condiciones de inicio deberán cumplirse antes de que el mensaje se considere como iniciado. Por ejemplo, si se configura un tiempo de "idle line" y un carácter de inicio específico, el CM o la CB determinará primero si se cumple el requisito de tiempo de "idle line" y buscará luego el carácter de inicio indicado. Si se recibe algún otro carácter (que no sea el carácter de inicio indicado), el CM o la CB reiniciará la búsqueda del inicio del mensaje buscando nuevamente el tiempo de "idle line".

Parámetro	Definición
Empezar con cualquier carácter	La condición "Carácter cualquiera" determina que cualquier carácter que se reciba correctamente indicará el inicio de un mensaje. Este carácter es el primer carácter del mensaje.
Line Break	La condición "Line Break" especifica que la recepción de un mensaje comienza cuando se reciba un carácter de salto de línea.

Parámetro	Definición
Idle Line	<p>La condición "Idle Line" especifica que la recepción de un mensaje comienza una vez que la línea de recepción haya estado inactiva o en reposo durante el número de tiempos de bit indicado. Cuando se cumple esta condición, se inicia el mensaje.</p>  <p>① Caracteres ② Reinicia el temporizador de "idle line" ③ Se ha detectado la "idle line" e iniciado la recepción de un mensaje</p>
Condición especial: Detectar el inicio del mensaje por un solo carácter	<p>Determina que un carácter en particular indique el principio de un mensaje. Este carácter es por consiguiente el primer carácter del mensaje. Cualquier carácter que se reciba antes de este carácter específico se descartará. El carácter predeterminado es STX.</p>
Condición especial: Detectar inicio del mensaje por una cadena de caracteres	<p>Determina que una secuencia de caracteres en particular de hasta cuatro secuencias configuradas indica el inicio de un mensaje. Es posible especificar cinco caracteres como máximo para una secuencia. Para cada posición de carácter se indica bien sea un carácter hexadecimal específico, o bien que el carácter se ignore en la comparación de secuencias (carácter comodín). El último carácter específico de una secuencia de caracteres finaliza esa secuencia de condiciones de inicio.</p> <p>Las secuencias entrantes se comparan con las condiciones de inicio configuradas hasta que se cumple una condición de inicio. Una vez que cumplida la secuencia de inicio, se iniciará la evaluación de las condiciones de fin.</p> <p>Es posible configurar hasta cuatro secuencias de caracteres específicos. Utilice una secuencia múltiple de condiciones de inicio cuando diferentes secuencias de caracteres pueden indicar el inicio de un mensaje. Si alguna de las secuencias de caracteres concuerda, se iniciará el mensaje.</p>

El orden de comprobación de las condiciones de inicio es el siguiente:

- Idle Line
- Line Break
- Caracteres o secuencias de caracteres

Si al comprobar varias condiciones de inicio no se cumple una de ellas, el CM o la CB reiniciará la comprobación con la primera condición requerida. Una vez el CM o la CB determina que se han cumplido las condiciones de inicio, procede a comparar las condiciones de fin.

Ejemplo de configuración - iniciar mensaje en una o dos secuencias de caracteres

Tenga en cuenta la siguiente configuración de las condiciones de inicio del mensaje:

Detectar inicio del mensaje por una cadena de caracteres

Número de cadenas de caracteres que definir:

Secuencia de 5 caracteres para iniciar el mensaje

Inicio del mensaje secuencia 1

Comprobar carácter 1

Valor de carácter (HEX):

Valor de carácter (ASCII):

Comprobar carácter 2

Valor de carácter (HEX):

Valor de carácter (ASCII):

Comprobar carácter 3

Valor de carácter (HEX):

Valor de carácter (ASCII):

Comprobar carácter 4

Valor de carácter (HEX):

Valor de carácter (ASCII):

Comprobar carácter 5

Valor de carácter (HEX):

Valor de carácter (ASCII):

Inicio del mensaje secuencia 2

Comprobar carácter 1

Valor de carácter (HEX):

Valor de carácter (ASCII):

Comprobar carácter 2

Valor de carácter (HEX):

Valor de carácter (ASCII):

Comprobar carácter 3

Valor de carácter (HEX):

Valor de carácter (ASCII):

Comprobar carácter 4

Valor de carácter (HEX):

Valor de carácter (ASCII):

Comprobar carácter 5

Valor de carácter (HEX):

Valor de carácter (ASCII):

En esta configuración, la condición de inicio se cumple cuando se presenta uno de los patrones siguientes:

- Cuando se recibe una secuencia de cinco caracteres en la que el primer carácter es 0x6A y, el quinto, 0x1C. Los caracteres de las posiciones 2, 3 y 4 pueden ser un carácter cualquiera en esta configuración. Tras recibirse el quinto carácter comienza la evaluación de las condiciones de fin.
- Cuando se reciben dos caracteres 0x6A consecutivos precedidos de un carácter cualquiera. En este caso, la evaluación de las condiciones de fin comienza tras recibirse el segundo 0x6A (3 caracteres). El carácter que precede el primer 0x6A se incluye en la condición de inicio.

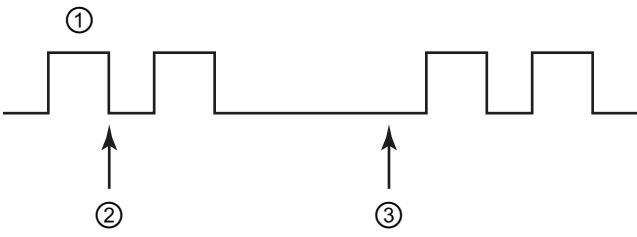
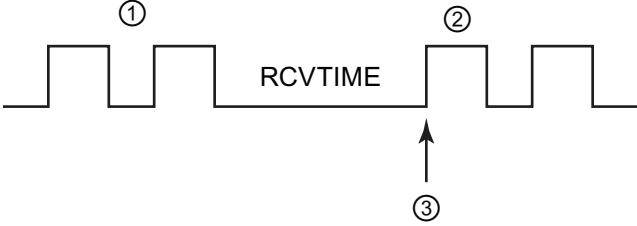
Secuencias de ejemplo que cumplirían esta condición de inicio:

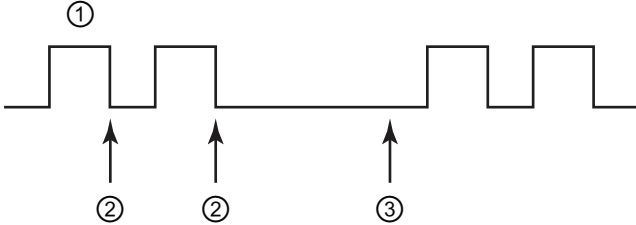
- <carácter cualquiera> 6A 6A
- 6A 12 14 18 1C
- 6A 44 A5 D2 1C

Condiciones de fin del mensaje

También es posible definir cómo la interfaz de comunicación debe detectar el fin de un mensaje. Es posible configurar varias condiciones de fin del mensaje. Si se cumple alguna de las condiciones configuradas, finalizará el mensaje.

Por ejemplo, es posible especificar una condición de fin con un timeout de fin de mensaje de 300 milisegundos, un timeout entre caracteres de 40 tiempos de bit y una longitud máxima de 50 bytes. El mensaje terminará si la recepción tarda más de 300 milisegundos o si la distancia temporal entre dos caracteres es superior a 40 tiempos de bit o si se reciben 50 bytes.

Parámetro	Definición
<p>Detectar fin del mensaje por tiempo de mensaje excedido</p>	<p>El mensaje finaliza cuando ha transcurrido el tiempo de espera configurado para el fin del mensaje. El periodo de timeout del mensaje comienza cuando se ha cumplido una condición de inicio. El valor predeterminado es 200 ms. El rango válido está comprendido entre 0 y 65535 ms.</p>  <p>① Caracteres recibidos ② Condición de inicio del mensaje cumplida: el temporizador del mensaje arranca ③ El temporizador del mensaje se detiene y finaliza el mensaje</p>
<p>Detectar fin del mensaje por tiempo de respuesta excedido</p>	<p>El mensaje finaliza cuando ha transcurrido el tiempo de espera configurado para una respuesta antes de que se reciba una secuencia de inicio válida. El periodo de timeout de respuesta comienza cuando una transmisión finaliza y el CM la CB inicia la recepción. El timeout de respuesta predeterminado es 200 ms y el rango está comprendido entre 0 y 65535 ms. Si un carácter no se recibe dentro del periodo de tiempo de respuesta RCVTIME, entonces se devuelve un error a la instrucción RCV_PTP correspondiente. El timeout de respuesta no define una condición de fin específica. Sólo determina que un carácter tiene que recibirse correctamente dentro del tiempo indicado. Hay que configurar otra condición de fin para indicar el fin real de un mensaje.</p>  <p>① Caracteres transmitidos ② Caracteres recibidos ③ En este momento el primer carácter debe haber sido recibido correctamente.</p>

Parámetro	Definición
<p>Detectar fin del mensaje por tiempo excedido entre caracteres</p>	<p>El mensaje finaliza cuando ha transcurrido el timeout máximo configurado entre cualquier par de caracteres consecutivos de un mensaje. El valor predeterminado del tiempo excedido entre caracteres es 12 tiempos de bit y el valor máximo es 65535 tiempos de bit, hasta un límite máximo de ocho segundos.</p>  <p>① Caracteres recibidos ② Reinicia el temporizador entre caracteres ③ El temporizador entre caracteres se detiene y finaliza el mensaje.</p>
<p>Detectar fin del mensaje por longitud máxima</p>	<p>El mensaje finaliza cuando se ha recibido el número de caracteres máximo configurado. El rango válido de la longitud máxima está comprendido entre 1 y 1023.</p> <p>Esta condición sirve para impedir un error de desbordamiento del búfer de mensajes. Si esta condición de fin se combina con condiciones de fin por tiempo excedido (timeout) y ocurre una condición de timeout, los caracteres válidos recibidos hasta entonces estarán disponibles, aunque no se haya alcanzado la longitud máxima. Esto permite soportar protocolos de longitud variable si sólo se conoce la longitud máxima.</p>
<p>Leer longitud del mensaje en el mensaje</p>	<p>El mensaje en sí especifica la longitud del mensaje. El mensaje finaliza cuando se recibe un mensaje con la longitud especificada. El método para especificar e interpretar la longitud del mensaje se describe más adelante.</p>
<p>Detectar fin del mensaje por un carácter</p>	<p>El mensaje finaliza cuando se recibe un carácter especificado.</p>
<p>Detectar fin del mensaje por una cadena de caracteres</p>	<p>El mensaje finaliza cuando se recibe una secuencia de caracteres especificada. Es posible especificar una secuencia de cinco caracteres como máximo. Para cada posición de carácter se indica bien sea un carácter hexadecimal específico, o bien que el carácter se ignore en la comparación de secuencias.</p> <p>Los caracteres iniciales ignorados no forman parte de la condición de fin. Los caracteres finales ignorados forman parte de la condición de fin.</p>

Ejemplo de configuración - finalizar mensaje con una secuencia de caracteres

Tenga en cuenta la siguiente configuración de las condiciones de fin del mensaje:

Detectar fin del mensaje por una cadena de caracteres

Secuencia de 5 caracteres para finalizar el mensaje

Comprobar carácter	Valor de carácter (HEX)	Valor de carácter (ASCII)
<input type="checkbox"/>	0	CUALQUIERA
<input checked="" type="checkbox"/>	6A	
<input checked="" type="checkbox"/>	6A	
<input type="checkbox"/>	0	CUALQUIERA
<input type="checkbox"/>	0	CUALQUIERA

En este caso, la condición de fin se cumple cuando se reciben dos caracteres 0x7A consecutivos, seguidos de dos caracteres cualquiera. El carácter que precede el patrón 0x7A 0x7A no forma parte de la secuencia de caracteres final. Los dos caracteres que le siguen al patrón 0x7A 0x7A se requieren para terminar la secuencia de caracteres final. Aunque los valores recibidos de las posiciones de carácter 4 y 5 son irrelevantes, deben recibirse para que se cumpla la condición de fin.

Indicación de la longitud del mensaje dentro del mensaje

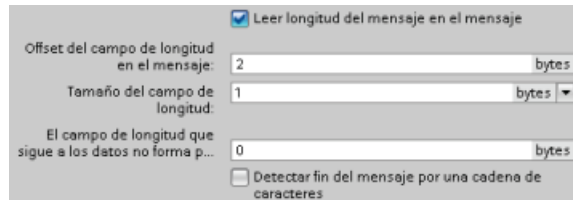
Si se selecciona la condición especial en la que la longitud del mensaje se incluye en el mensaje, es preciso indicar tres parámetros que definen la información acerca de la longitud del mensaje.

La estructura real del mensaje varía según el protocolo utilizado. Los tres parámetros son los siguientes:

- n: Posición de carácter (en base 1) dentro del mensaje que inicia el indicador de longitud
- Tamaño de longitud: Número de bytes (uno, dos o cuatro) del indicador de longitud
- Longitud m: Número de caracteres posteriores al indicador de longitud que no se incluyen en el conteo de longitud

Los caracteres de fin no tienen que ser contiguos. El valor "Longitud M" puede utilizarse para indicar la longitud de un campo de suma de verificación cuyo tamaño no se incluye en el campo de longitud.

Estos campos aparecen en la configuración de la recepción de mensajes de las propiedades del dispositivo:



Ejemplo 1: Considerar un mensaje estructurado según el protocolo siguiente:

STX	Len (n)	Caracteres 3 a 14 contados según la longitud											
		ADR	PKE		INDEX		PWD		STW		HSW		BCC
1	2	3	4	5	6	7	8	9	10	11	12	13	14
STX	0x0C	xx	xxxx		xxxx		xxxx		xxxx		xxxx		xx

Configure los parámetros de longitud de recepción de este mensaje como se indica a continuación:

- n = 2 (la longitud del mensaje comienza con el byte 2.)
- Tamaño de longitud = 1 (la longitud del mensaje se define en un byte.)
- Longitud m = 0 (no hay caracteres adicionales posteriores al indicador de longitud no incluidos en el conteo de longitud. Doce caracteres le siguen al indicador de longitud.)

En este ejemplo, los caracteres 3 a 14 (inclusive) son los caracteres que cuenta Len (n).

Ejemplo 2: Considerar otro mensaje estructurado según el protocolo siguiente:

SD1	Len (n)	Len (n)	SD2	Caracteres 5 a 10 contados según la longitud						FCS	ED
				DA	SA	FA	Unidad de datos=3 bytes				
1	2	3	4	5	6	7	8	9	10	11	12
xx	0x06	0x06	xx	xx	xx	xx	xx	xx	xx	xx	xx

Configure los parámetros de longitud de recepción de este mensaje como se indica a continuación:

- n = 3 (la longitud del mensaje comienza con el byte 3.)
- Tamaño de longitud = 1 (la longitud del mensaje se define en un byte.)
- Longitud m = 3 (tres caracteres posteriores al indicador de longitud no se cuentan en la longitud. En el protocolo de este ejemplo, los caracteres SD2, FCS y ED no se incluyen en el conteo de longitud. Los seis caracteres restantes se incluyen en el conteo de longitud. Por tanto, el número total de caracteres posteriores al indicador de longitud es nueve.)

En este ejemplo, los caracteres 5 a 10 (inclusive) son los caracteres que cuenta Len (n).

12.3.4 Programar la comunicación PtP

STEP 7 ofrece instrucciones avanzadas que permiten al programa de usuario establecer comunicaciones punto a punto utilizando un protocolo diseñado y definido en el programa de usuario. Estas instrucciones se dividen en dos categorías, a saber:

- Instrucciones de configuración
- Instrucciones de comunicación

Instrucciones de configuración

Para que el programa de usuario pueda intervenir en la comunicación PtP, es preciso configurar el puerto de comunicación y los parámetros para enviar y recibir datos.

El puerto y los mensajes pueden configurarse para todo CM o toda CB en la configuración de dispositivos o mediante estas instrucciones en el programa de usuario:

- PORT_CFG (Página 594)
- SEND_CFG (Página 596)
- RCV_CFG (Página 598)

Instrucciones de comunicación

Las instrucciones de comunicación PtP permiten al programa de usuario enviar y recibir mensajes a/de las interfaces de comunicación. Encontrará más información sobre cómo transferir datos con estas instrucciones en el apartado Coherencia de datos (Página 157).

Todas las funciones PtP funcionan de forma asíncrona. El programa de usuario puede utilizar una arquitectura de sondeo para determinar el estado de las transmisiones y recepciones. SEND_PTP y RCV_PTP pueden ejecutarse simultáneamente. Según sea necesario, los módulos de comunicación y la placa de comunicación depositan los mensajes de transmisión y recepción en un búfer cuyo tamaño máximo es 1024 bytes.

Los CMs y la CB envían y reciben mensajes a/de los dispositivos punto a punto reales. El protocolo de mensajes se encuentra en un búfer que se recibe de o se envía a un puerto de comunicación específico. El búfer y el puerto son parámetros de las instrucciones de envío y recepción.

- SEND_PTP (Página 602)
- RCV_PTP (Página 606)

Se dispone de instrucciones adicionales que permiten inicializar el búfer de recepción, así como consultar y activar determinadas señales RS232:

- RCV_RST (Página 607)
- SGN_GET (Página 608)
- SGN_SET (Página 609)

12.3.4.1 Arquitectura de sondeo

Las instrucciones punto a punto del S7-1200 deben llamarse cíclicamente/periódicamente para comprobar si se han recibido mensajes. Un sondeo de la transmisión notifica al programa de usuario el fin de la transmisión.

Arquitectura de sondeo: maestro

La secuencia típica de un maestro es la siguiente:

1. Una instrucción SEND_PTP inicia una transmisión al CM o la CB.
2. La instrucción SEND_PTP se ejecuta en los ciclos posteriores para determinar el progreso de la transmisión.
3. Cuando la instrucción SEND_PTP indica que ha finalizado la transmisión, el programa de usuario se prepara para recibir la respuesta.
4. La instrucción RCV_PTP se ejecuta repetidamente para comprobar si hay una respuesta. Una vez que el CM o la CB haya recibido un mensaje de respuesta, la instrucción RCV_PTP copia la respuesta en la CPU e indicará que se han recibido datos nuevos.
5. El programa de usuario puede procesar la respuesta.
6. Vaya al paso 1 y repita el ciclo.

Arquitectura de sondeo: esclavo

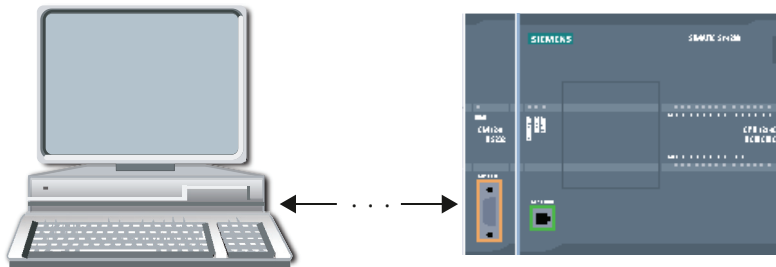
La secuencia típica de un esclavo es la siguiente:

1. El programa de usuario ejecuta la instrucción RCV_PTP en cada ciclo.
2. Una vez que el CM o la CB haya recibido una petición, la instrucción RCV_PTP indica que los nuevos datos están listos y la petición se copia en la CPU.
3. El programa de usuario procesa la petición y genera una respuesta.
4. Utilice una instrucción SEND_PTP para enviar la respuesta al maestro.
5. Ejecute repetidamente SEND_PTP para asegurarse de que se realiza la transmisión.
6. Vaya al paso 1 y repita el ciclo.

El esclavo debe llamar RCV_PTP con la suficiente frecuencia para recibir una transmisión del maestro antes de que se produzca un timeout del maestro a la espera de una respuesta. Para realizar esta tarea, el programa de usuario puede llamar RCV_PTP desde un OB de ciclo, previendo un tiempo de ciclo suficiente para recibir una transmisión del maestro antes de que transcurra el periodo de timeout. Si el tiempo de ciclo del OB se ajusta para que pueda ejecutarse dos veces dentro del periodo de timeout del maestro, el programa de usuario puede recibir todas las transmisiones sin pérdidas.

12.3.5 Ejemplo: Comunicación punto a punto

En este ejemplo, una CPU S7-1200 se comunica con un PC con emulador de terminal a través de un módulo CM 1241 RS232. La configuración punto a punto y el programa de STEP 7 muestran en este ejemplo cómo la CPU puede recibir un mensaje del PC y enviar el mensaje de vuelta al PC.



Hay que conectar la interfaz de comunicación del módulo CM 1241 RS232 a la interfaz RS232 del PC, que por lo general es COM1. Debido a que estos dos puertos son equipos terminales de datos (Data Terminal Equipment DTE), hay que intercambiar los pines de recepción y transmisión (2 y 3) al conectar ambos puertos. Esto se puede realizar aplicando uno de los métodos siguientes:

- Utilice un adaptador de módem NULO para intercambiar conjuntamente los pines 2 y 3 con un cable RS232 estándar.
- Utilice un cable de módem NULO que ya tenga intercambiados los pines 2 y 3. Generalmente, un cable de módem NULO se puede reconocer porque tiene en sus dos extremos un conector hembra sub-D de 9 pines.

12.3.5.1 Configurar los módulos de comunicación

El CM 1241 se puede configurar desde la configuración de dispositivos de STEP 7 o con las instrucciones del programa de usuario. Este ejemplo utiliza la variante de la configuración de dispositivos.

- Configuración del puerto: Haga clic en el puerto de comunicación del módulo CM desde la configuración de dispositivos y configure el puerto como se indica a continuación:

Configuración del puerto

Modo de operación

Dúplex (RS422) modo a 4 hilos (conexión punto a punto)

Dúplex (RS422) modo a 4 hilos (maestro multipunto)

Dúplex (RS422) modo a 4 hilos (esclavo multipunto)

Semidúplex (RS485) modo a 2 hilos

Iniciación de la línea de recepción

ninguno

Forward Bias (Señal R(A) 0V, señal R(B) 5V)

Velocidad de transferencia: 9,6 kbits

Paridad: Sin paridad

Bits de datos: 8 bits por carácter

Bit de parada: 1

Control de flujo: ninguno

Carácter XON (HEX): 0
(ASCII): NUL

Carácter XOFF (HEX): 0
(ASCII): NUL

Tiempo de espera: 1 ms

Nota

Los ajustes para la configuración del "Modo de operación" y "Iniciación de la línea de recepción" solo son aplicables al módulo CM 1241 (RS422/RS485). Los demás módulos CM 1241 no tienen estos ajustes para la configuración del puerto. Consulte Configurar RS422 y RS485 (Página 629)

- Configuración de la transferencia de mensajes: Acepte la configuración de la transferencia de mensajes predeterminada. No debe enviarse una pausa al inicio del programa.

12.3 Comunicación punto a punto (PtP)

- Configuración de inicio de recepción de mensajes: Configure el CM 1241 para que inicie la recepción de un mensaje cuando la línea de comunicación esté inactiva durante al menos 50 tiempos de bit (aprox. 5 milisegundos a 9600 baudios = $50 * 1/9600$):

Inicio del mensaje

Empezar con cualquier carácter

Empezar con una condición especial

Detectar inicio del mensaje en Line Break

Detectar inicio del mensaje en Idle Line

Duración de Idle Line: 50 Tiempos de bit

Detectar el inicio del mensaje por un solo carácter

Carácter de inicio del mensaje (HEX): 2

Carácter de inicio del mensaje (ASCII): STX

Detectar inicio del mensaje por una cadena de caracteres

Número de cadenas de caracteres que definir: 1

- Configuración de fin de recepción de mensajes: Configure el CM 1241 para que finalice un mensaje cuando haya recibido como máximo 100 bytes o un carácter de salto de línea (10 decimales o un hexadecimal). La secuencia de fin permite hasta cinco caracteres de fin en secuencia. El quinto carácter en la secuencia es el carácter de salto de línea. Los cuatro caracteres de fin precedentes son caracteres "irrelevantes" o no seleccionados. El CM 1241 no compara los caracteres "irrelevantes" pero busca un carácter de salto de línea precedido de cero o más caracteres "irrelevantes" para indicar el fin del mensaje.

Fin del mensaje

Definir condiciones para el fin del mensaje

Detectar fin del mensaje por tiempo de mensaje excedido

Tiempo de mensaje excedido: 200 ms

Detectar fin del mensaje por tiempo de respuesta excedido

Tiempo de respuesta excedido: 200 ms

Detectar fin del mensaje por tiempo excedido entre caracteres

Tiempo excedido entre caracteres: 12 Tiempos de bit

Detectar fin del mensaje por longitud máxima

Longitud máxima del mensaje: 100 bytes

Leer longitud del mensaje en el mensaje

Offset del campo de longitud en el mensaje: 1 bytes

Tamaño del campo de longitud: 1 bytes

El campo de longitud que sigue a los datos no forma p... 0 bytes

Detectar fin del mensaje por una cadena de caracteres

Secuencia de 5 caracteres para finalizar el mensaje

Comprobar carácter 1	<input type="checkbox"/>
Valor de carácter (HEX):	0
Valor de carácter (ASCII):	CUALQUIERA
Comprobar carácter 2	<input type="checkbox"/>
Valor de carácter (HEX):	0
Valor de carácter (ASCII):	CUALQUIERA
Comprobar carácter 3	<input type="checkbox"/>
Valor de carácter (HEX):	0
Valor de carácter (ASCII):	CUALQUIERA
Comprobar carácter 4	<input type="checkbox"/>
Valor de carácter (HEX):	0
Valor de carácter (ASCII):	CUALQUIERA
Comprobar carácter 5	<input checked="" type="checkbox"/>
Valor de carácter (HEX):	A
Valor de carácter (ASCII):	LF

12.3.5.2 Configurar RS422 y RS485

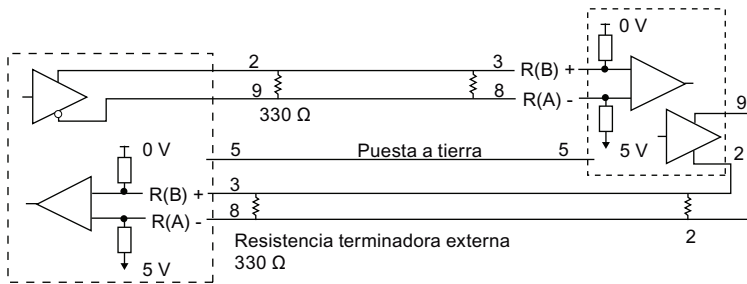
Configurar RS422

Para el modo RS422, hay tres modos de operación según la configuración de la red. Seleccione uno de estos modos en función de los dispositivos de la red. Según la selección que se realice en Inicialización de la línea de recepción, se podrían dar los casos que se indican a continuación.

- Modo dúplex (RS422) de cuatro hilos (conexión punto a punto): seleccione esta opción si hay dos dispositivos en la red. Inicialización de la línea de recepción:
 - Seleccione Ninguno si suministra la polarización y la terminación (caso 3).
 - Seleccione Polarización directa para utilizar la polarización y terminación internas (caso 2).
 - Seleccione Polarización inversa para utilizar la polarización y la terminación internas y habilitar la detección de rotura de cable para ambos dispositivos (caso 1).
- Modo dúplex (RS422) de cuatro hilos (maestro multipunto): Seleccione esta opción para el dispositivo maestro cuando tenga una red con un maestro y varios esclavos. Inicialización de la línea de recepción:
 - Seleccione Ninguno si suministra la polarización y la terminación (caso 3).
 - Seleccione Polarización directa para utilizar la polarización y terminación internas (caso 2).
 - La detección de rotura de cable no se puede realizar en este modo.
- Modo dúplex (RS422) de cuatro hilos (esclavo multipunto): Seleccione esta opción para todos los dispositivos esclavos cuando tenga una red con un maestro y varios esclavos. Inicialización de la línea de recepción:
 - Seleccione Ninguno si suministra la polarización y la terminación (caso 3).
 - Seleccione Polarización directa para utilizar la polarización y terminación internas (caso 2).
 - Seleccione Polarización inversa para utilizar la polarización y la terminación internas y habilitar la detección de rotura de cable para los esclavos (caso 1).

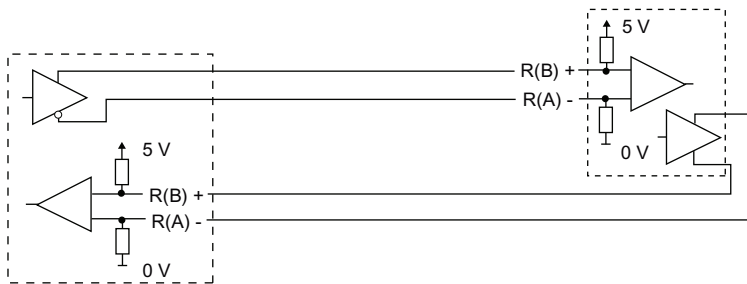
Caso 1: RS422 con detección de rotura de cable

- Modo de operación: RS422
- Inicialización de la línea de recepción: Polarización inversa (polarizada con $R(A) > R(B) > 0V$)
- Rotura de cable: Detección de rotura de cable habilitada (transmisor siempre activo)



Caso 2: RS422 sin detección de rotura de cable, polarización directa

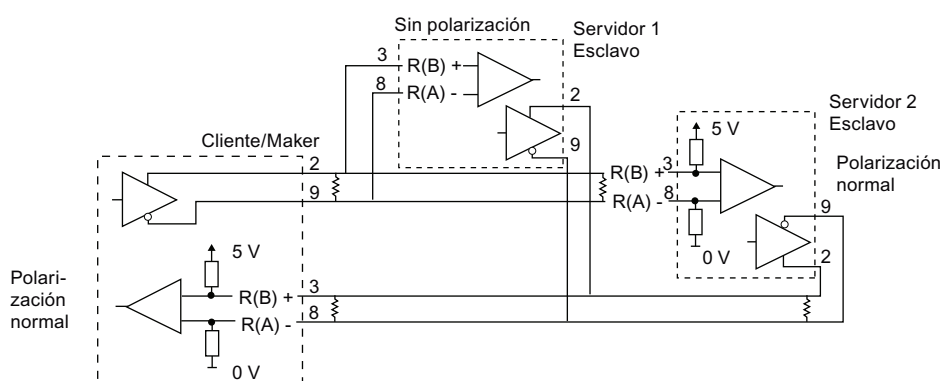
- Modo de operación: RS422
- Inicialización de la línea de recepción: Polarización directa (polarizada con $R(B) > R(A) > 0V$)
- Rotura de cable: Sin detección de rotura de cable (transmisor habilitado sólo durante la transmisión)



Caso 3: RS422: Sin detección de rotura de cable, sin polarización

- Modo de operación: RS422
- Inicialización de la línea de recepción: Sin polarización
- Rotura de cable: Sin detección de rotura de cable (transmisor habilitado sólo durante la transmisión)

La polarización y la terminación las añade el usuario en los nodos finales de la red.



Configurar RS485

Para el modo RS485, sólo hay un estado operativo. Según la selección que se realice en Inicialización de la línea de recepción, se podrían dar los casos que se indican a continuación.

- Modo semidúplex (RS485) de dos hilos. Inicialización de la línea de recepción:
 - Seleccione Ninguno si suministra la polarización y la terminación (caso 5).
 - Seleccione Polarización directa para utilizar la polarización y terminación internas (caso 4).

Caso 4: RS485: Polarización directa

- Modo de operación: RS485
- Inicialización de la línea de recepción: Polarización directa (polarizada con $R(B) > R(A) > 0V$)



Caso 5: RS485: Sin polarización (polarización externa)

- Modo de operación: RS485
- Inicialización de la línea de recepción: Sin polarización (polarización externa necesaria)

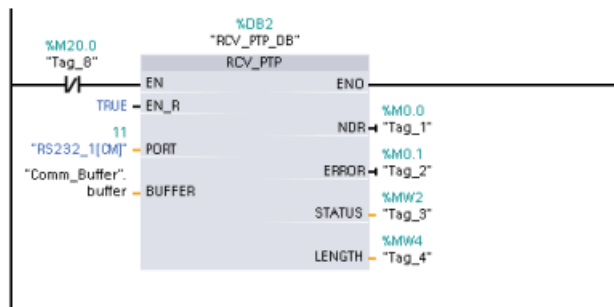


12.3.5.3 Programar el programa de STEP 7

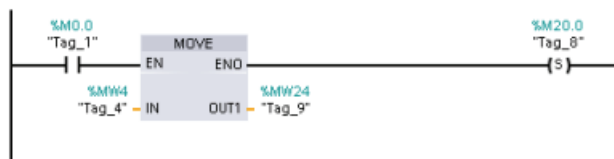
El programa de ejemplo utiliza un bloque de datos globales para el búfer de comunicación, una instrucción RCV_PTP (Página 606) para recibir datos del emulador de terminal y una instrucción SEND_PTP (Página 602) para enviar de regreso el búfer al emulador de terminal. Para programar el ejemplo, agregue la configuración del bloque de datos y programe OB1 como se indica a continuación.

Bloque de datos global "Comm_Buffer": cree un bloque de datos global (DB) y asígnele el nombre "Comm_Buffer". Cree un valor en el bloque de datos llamado "buffer" con un tipo de datos de "matriz [0 .. 99] de byte".

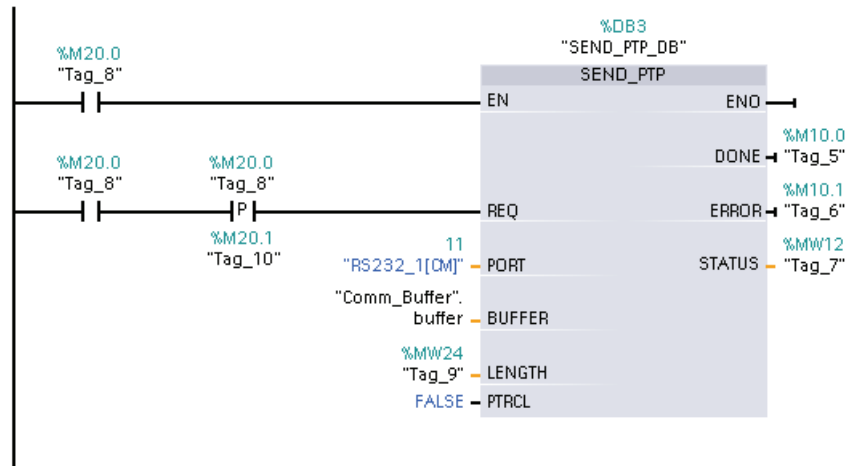
Segmento 1: habilite la instrucción RCV_PTP siempre que SEND_PTP no esté activo. Tag_8 en MW20.0 indica cuando el envío ha finalizado en la red 4 y cuando el módulo de comunicación está por lo tanto listo para recibir un mensaje.



Segmento 2: utilice el valor NDR (Tag_1 en M0.0) activado por la instrucción RCV_PTP para hacer una copia del número de bytes recibidos y para activar una marca (Tag_8 en M20.0) con el fin de lanzar la instrucción SEND_PTP.



Segmento 3: habilite la instrucción SEND_PTP cuando la marca M20.0 esté activada. Utilice también esta marca para poner la entrada REQ en TRUE durante un ciclo. La entrada REQ le notifica a la instrucción SEND_PTP que debe transmitirse una nueva petición. La entrada REQ únicamente debe ponerse a TRUE durante una ejecución de SEND_PTP. La instrucción SEND_PTP se ejecuta en cada ciclo hasta que finalice la transmisión. La transmisión finaliza cuando el último byte del mensaje ha sido transmitido desde el CM 1241. Cuando la transmisión ha finalizado, la salida DONE (Tag_5 en M10.0) se pone a TRUE durante una ejecución de SEND_PTP.



Segmento 4: vigile la salida DONE de SEND_PTP y desactive la marca de transmisión (Tag_8 en M20.0) cuando la transmisión haya finalizado. Cuando se desactiva la marca de transmisión, la instrucción RCV_PTP en el segmento 1 es habilitada para recibir el siguiente mensaje.



12.3.5.4 Configurar el emulador de terminal

Hay que configurar el emulador de terminal para soportar el programa de ejemplo. Se puede utilizar prácticamente cualquier tipo de emulador de terminal en su PC, p. ej. el HyperTerminal. Asegúrese de que el emulador de terminal esté en modo desconectado antes de editar los ajustes como se indica a continuación:

1. Ajuste el emulador de terminal para que utilice el puerto RS232 del PC (normalmente COM1).
2. Configure el puerto para 9600 baudios, 8 bits de datos, sin paridad (ninguna), 1 bit de parada y sin control de flujo.
3. Cambie los ajustes del emulador de terminal para emular un terminal ANSI.
4. Configure el emulador de terminal ASCII para enviar un salto de línea después de cada línea (después de que el usuario pulse la tecla Intro).
5. Active el eco de los caracteres escritos localmente para que el emulador de terminal visualice lo que se ha escrito.

12.3.5.5 Ejecutar el programa de ejemplo

Para poner en práctica el programa de ejemplo, proceda del siguiente modo:

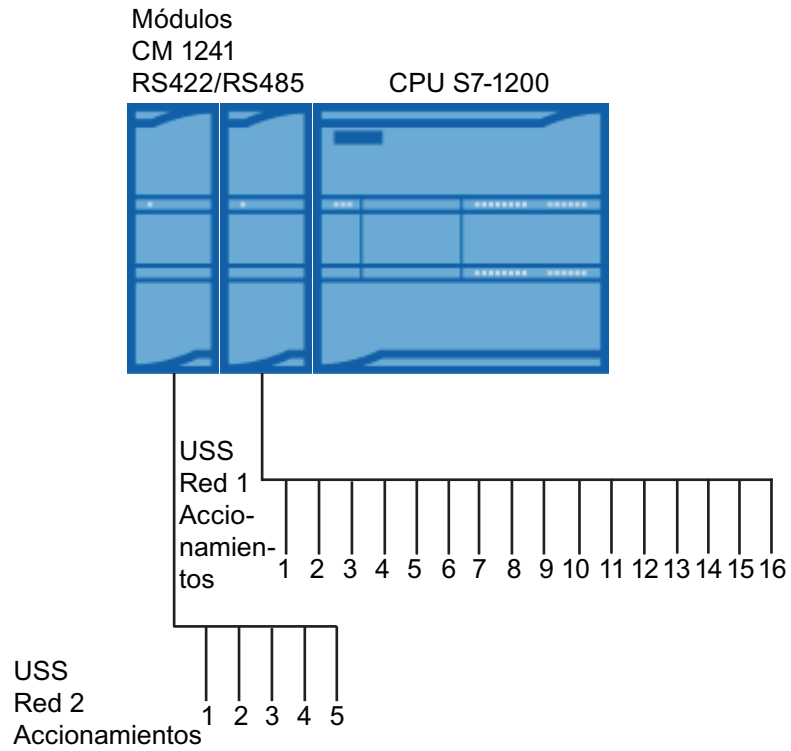
1. Cargue el programa STEP 7 en la CPU y asegúrese de que la misma esté en modo RUN.
2. Haga clic en el botón "conectar" en el emulador de terminal para transferir los cambios en la configuración y abrir una sesión de terminal en el CM 1241.
3. Escriba caracteres en el PC y pulse Intro.

El emulador de terminal envía los caracteres al CM 1241 y a la CPU. Luego, el programa de la CPU envía los caracteres de regreso al emulador de terminal.

12.4 Comunicación de interfaz serie universal (USS)

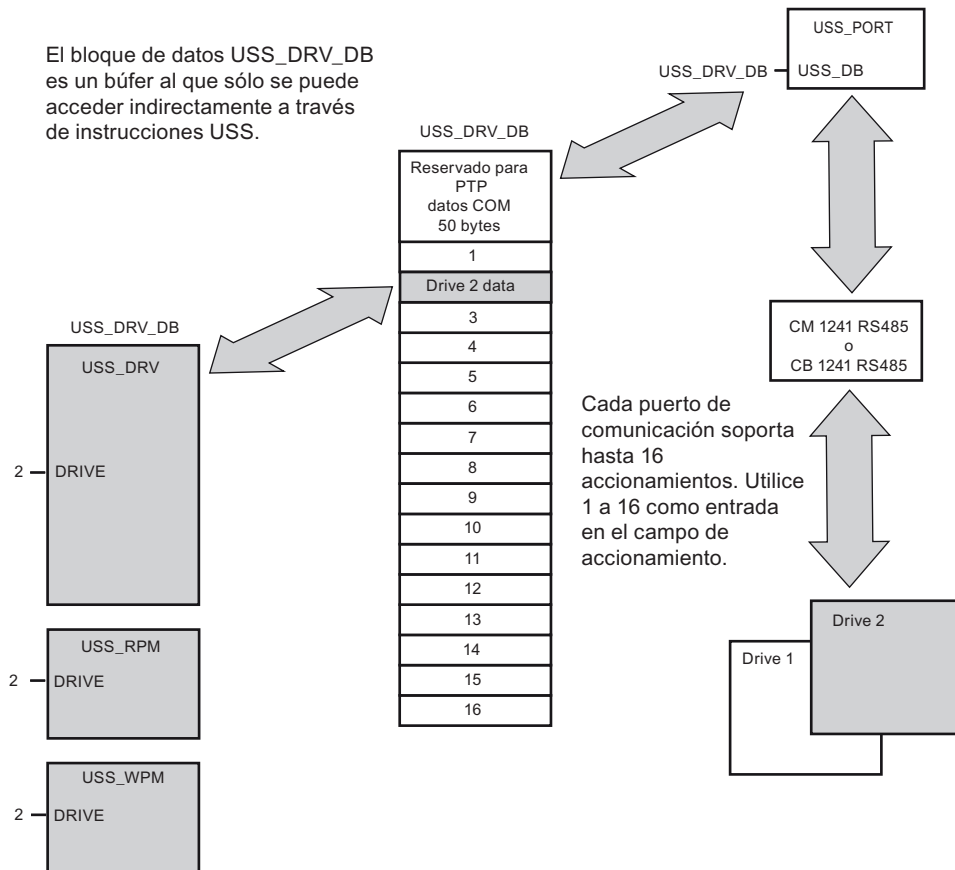
Las instrucciones USS controlan el funcionamiento de accionamientos motorizados que soportan el protocolo interfaz serie universal (USS). Las instrucciones USS se pueden utilizar para comunicarse con varios accionamientos a través de conexiones RS485 con módulos de comunicación CM 1241 RS485 o una placa de comunicación CB 1241 RS485. Pueden instalarse hasta tres módulos CM 1241 RS422/RS485 y una placa CB 1241 RS485 en una CPU S7-1200. Cada puerto RS485 puede operar hasta 16 accionamientos.

El protocolo USS utiliza una red maestro-esclavo para una comunicación vía bus serie. El maestro utiliza un parámetro de dirección para enviar un mensaje a un esclavo seleccionado. Un esclavo jamás puede transmitir sin primero haber recibido una petición correspondiente. No es posible una transferencia directa de mensajes entre los distintos esclavos. La comunicación USS funciona en modo semidúplex. La siguiente figura de USS muestra un diagrama de red para una aplicación con accionamientos a modo de ejemplo.



12.4.1 Requisitos para utilizar el protocolo USS

Las cuatro instrucciones USS utilizan 1 FB y 3 FCs que soportan el protocolo USS. Un bloque de datos instancia (DB) USS_PORT se utiliza para cada red USS. El bloque de datos instancia USS_PORT contiene un almacenamiento temporal y búfers para todos los accionamientos en la red USS. Las instrucciones USS comparten la información en este bloque de datos.



Todos los accionamientos (máx. 16) conectados a un solo puerto RS485 pertenecen a una misma red USS. Todos los accionamientos conectados a otro puerto RS485 pertenecen a diferentes redes USS. Cada red USS se gestiona utilizando un bloque de datos unívoco. Todas las instrucciones asociadas a una red USS deben compartir dicho bloque de datos. Esto incluye todas las instrucciones USS_DRV, USS_PORT, USS_RPM y USS_WPM utilizadas para controlar todos los accionamientos de una red USS.

La instrucción USS_DRV es un bloque de función (FB). Cuando la instrucción USS_DRV se coloca en el editor de programas, es preciso indicar qué DB debe asignarse a ese FB en el diálogo "Opciones de llamada". Si se trata de la primera instrucción USS_DRV de este programa para esta red USS, es posible aceptar la asignación de DB predeterminada (o cambiar el nombre a discreción). Entonces se crea el nuevo DB. No obstante, si no es la primera instrucción USS_DRV de este canal, es preciso utilizar la lista desplegable del diálogo "Opciones de llamada" para seleccionar el nombre de DB que fue asignado previamente a esta red USS.

Las instrucciones USS_PORT, USS_RPM y USS_WPM son funciones (FCs). A estas FCs no se les asigna ningún DB cuando se colocan en el editor. En cambio, hay que asignar la referencia de DB apropiada a la entrada "USS_DB" de estas instrucciones. Haga doble clic en el campo de parámetro. A continuación, haga clic en el símbolo de ayuda de parámetros para ver los nombres de DB disponibles.

La función USS_PORT gestiona la comunicación real entre la CPU y los accionamientos vía el puerto de comunicación punto a punto (PtP) RS485. Cada llamada a esta función gestiona una comunicación con un accionamiento. El programa debe llamar esta función lo suficientemente rápido para impedir un timeout de comunicación por parte de los módulos. Esta función puede llamarse desde un OB de ciclo de programa principal o desde cualquier OB de alarma.

Por lo general, la función USS_PORT debe llamarse desde un OB de alarma cíclica. El tiempo de ciclo del OB de alarma cíclica debe ajustarse a aprox. la mitad del intervalo mínimo de llamada (p. ej., una comunicación de 1200 baudios debe utilizar un tiempo de ciclo de 350 ms o menos).

El bloque de función USS_DRV permite al programa acceder a un accionamiento determinado en la red USS. Sus entradas y salidas representan el estado y los controles del accionamiento. Si la red comprende 16 accionamientos, el programa debe tener como mínimo 16 llamadas de USS_DRV, es decir, una para cada accionamiento. Estos bloques deben llamarse a la velocidad necesaria para controlar el funcionamiento del accionamiento.

El bloque de función USS_DRV sólo se puede llamar desde un OB de ciclo de programa principal.

 **PRECAUCIÓN**

USS_DRV, USS_RPM y USS_WPM deben llamarse únicamente desde un OB de ciclo de programa principal. USS_PORT puede llamarse desde un OB cualquiera (generalmente desde un OB de alarma cíclica).

No utilice las instrucciones USS_DRV, USS_RPM o USS_WPM en un OB de prioridad más alta que la instrucción USS_PORT correspondiente. Por ejemplo, no inserte USS_PORT en el OB principal ni USS_RPM en un OB de alarma cíclica. Si no se impide la interrupción de la ejecución de USS_PORT podrían producirse errores inesperados.

Las funciones USS_RPM y USS_WPM leen y escriben los parámetros operativos del accionamiento remoto. Estos parámetros controlan el funcionamiento interno del accionamiento. Estos parámetros se definen en el manual del accionamiento. El programa puede contener un número cualquiera de estas funciones. No obstante, sólo una petición de lectura o escritura puede estar activa en un accionamiento en un momento determinado. Las funciones USS_RPM y USS_WPM sólo pueden llamarse desde un OB de ciclo de programa principal.

Calcular el tiempo necesario para la comunicación con el accionamiento

La comunicación con el accionamiento es asíncrona al ciclo del S7-1200. Por lo general, pueden transcurrir varios ciclos del S7-1200 antes de que finalice una transacción de comunicación con un accionamiento.

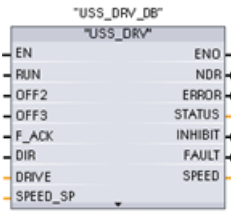

El intervalo de USS_PORT es el tiempo necesario para una transacción con un accionamiento. La tabla siguiente muestra el intervalo de USS_PORT mínimo para cada velocidad de transferencia de comunicación. Si la función USS_PORT se llama más frecuentemente que el intervalo de USS_PORT, no se incrementará el número de transacciones. El intervalo de timeout del accionamiento es el tiempo disponible para una transacción si, debido a errores de comunicación, se requieren 3 intentos para finalizar la transacción. De forma predeterminada, la librería del protocolo USS realiza automáticamente 2 reintentos por transacción.

Tabla 12- 34 Calcular los requisitos de tiempo

Velocidad de transferencia	Intervalo mínimo calculado para la llamada de USS_PORT (milisegundos)	Intervalo de timeout por accionamiento (milisegundos)
1200	790	2370
2400	405	1215
4800	212,5	638
9600	116,3	349
19200	68,2	205
38400	44,1	133
57600	36,1	109
115200	28,1	85

12.4.2 Instrucción USS_DRV

Tabla 12- 35 Instrucción USS_DRV

KOP / FUP	SCL	Descripción
<p>Vista predeterminada</p> 	<pre>"USS_DRV_DB" (RUN:=_bool_in_, OFF2:=_bool_in_, OFF3:=_bool_in_, F_ACK:=_bool_in_, DIR:=_bool_in_, DRIVE:=_usint_in_, PZD_LEN:=_usint_in_, SPEED_SP:=_real_in_, CTRL3:=_word_in_, CTRL4:=_word_in_, CTRL5:=_word_in_, CTRL6:=_word_in_, CTRL7:=_word_in_, CTRL8:=_word_in_, NDR=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_, RUN_EN=>_bool_out_, D_DIR=>_bool_out_, INHIBIT=>_bool_out_, FAULT=>_bool_out_, SPEED=>_real_out_, STATUS1=>_word_out_, STATUS3=>_word_out_, STATUS4=>_word_out_, STATUS5=>_word_out_, STATUS6=>_word_out_, STATUS7=>_word_out_, STATUS8=>_word_out_);</pre>	<p>La instrucción USS_DRV intercambia datos con el accionamiento creando peticiones e interpretando las respuestas del accionamiento. Para cada accionamiento debe utilizarse un bloque de función propio. No obstante, todas las funciones USS asociadas con una red USS y el puerto de comunicación PtP deben utilizar el mismo bloque de datos instancia. Es preciso crear el nombre del DB cuando se inserta la primera instrucción USS_DRV. Este DB creado al insertar la instrucción por primera vez se debe referenciar a continuación.</p> <p>STEP 7 crea el DB automáticamente al insertar la instrucción.</p>
<p>Vista ampliada</p> 		

¹ KOP y FUP: Haga clic en el lado inferior del cuadro para ampliarlo y ver todos los parámetros. Los parámetros que aparecen atenuados son opcionales y no se requiere una asignación de parámetros.

Tabla 12- 36 Tipos de datos para los parámetros

Parámetro y tipo	Tipo de datos	Descripción
RUN IN	Bool	Bit de arranque del accionamiento: Si es TRUE (verdadera), esta entrada habilita el accionamiento para que funcione a la velocidad predeterminada. Cuando RUN pasa a False mientras funciona un accionamiento, el motor se desacelerará hasta pararse. Este comportamiento difiere del de una caída de alimentación (OFF2) y del de un frenado del motor (OFF3).
OFF2 IN	Bool	Bit de parada eléctrica: Si es FALSE (falso), este bit hace que el accionamiento marche en inercia hasta parar sin frenar.

12.4 Comunicación de interfaz serie universal (USS)

Parámetro y tipo		Tipo de datos	Descripción
OFF3	IN	Bool	Bit de parada rápida: Si es FALSE (falso), este bit origina una parada rápida frenando el accionamiento, en vez de permitir que marche en inercia hasta parar.
F_ACK	IN	Bool	Bit de acuse de fallo: Este bit se activa para resetear el bit de fallo en un accionamiento. El bit se activa tras haberse solucionado el fallo para indicar al accionamiento que no tiene que seguir notificando el fallo anterior.
DIR	IN	Bool	Control de sentido del accionamiento: Este bit se activa para indicar que el sentido es hacia delante (SPEED_SP positiva).
DRIVE	IN	USInt	Dirección del accionamiento: Esta entrada es la dirección del accionamiento USS. El rango válido está comprendido entre el accionamiento 1 y el 16.
PZD_LEN	IN	USInt	Longitud de palabra: Este es el número de palabras de los datos PZD. Los valores válidos son 2, 4, 6 u 8 palabras. El valor predeterminado es 2.
SPEED_SP	IN	Real	Consigna de velocidad: Esta es la velocidad del accionamiento expresada como porcentaje de la frecuencia configurada. Un valor positivo indica el sentido hacia delante (si DIR es TRUE). El rango válido está comprendido entre 200,00 y -200,00.
CTRL3	IN	Word	Palabra de control 3: Valor escrito en un parámetro configurable por el usuario en el accionamiento. El usuario debe configurarlo en el accionamiento. (parámetro opcional)
CTRL4	IN	Word	Palabra de control 4: Valor escrito en un parámetro configurable por el usuario en el accionamiento. El usuario debe configurarlo en el accionamiento. (parámetro opcional)
CTRL5	IN	Word	Palabra de control 5: Valor escrito en un parámetro configurable por el usuario en el accionamiento. El usuario debe configurarlo en el accionamiento. (parámetro opcional)
CTRL6	IN	Word	Palabra de control 6: Valor escrito en un parámetro configurable por el usuario en el accionamiento. El usuario debe configurarlo en el accionamiento. (parámetro opcional)
CTRL7	IN	Word	Palabra de control 7: Valor escrito en un parámetro configurable por el usuario en el accionamiento. El usuario debe configurarlo en el accionamiento. (parámetro opcional)
CTRL8	IN	Word	Palabra de control 8: Valor escrito en un parámetro configurable por el usuario en el accionamiento. El usuario debe configurarlo en el accionamiento. (parámetro opcional)
NDR	OUT	Bool	Nuevos datos listos: Si es TRUE (verdadero), el bit indica que las salidas contienen datos de una petición de comunicación nueva.
ERROR	OUT	Bool	Ha ocurrido un error: Si es TRUE (verdadero), indica que ha ocurrido un error y la salida STATUS es válida. Todas las demás salidas se ponen a cero cuando ocurre un error. Los errores de comunicación se notifican sólo en las salidas ERROR y STATUS de la instrucción USS_PORT.
STATUS	OUT	Word	El valor de estado de la petición indica el resultado del ciclo. Esta no es una palabra de estado devuelta del accionamiento.
RUN_EN	OUT	Bool	Funcionamiento habilitado: Este bit indica si está funcionando el accionamiento.
D_DIR	OUT	Bool	Sentido del accionamiento: Este bit indica si el accionamiento está funcionando hacia delante.

Parámetro y tipo		Tipo de datos	Descripción
INHIBIT	OUT	Bool	Accionamiento inhibido: Este bit indica el estado del bit de inhibición del accionamiento.
FAULT	OUT	Bool	Fallo del accionamiento: Este bit indica que el accionamiento ha registrado un fallo. El usuario debe corregir el problema y activar el bit F_ACK para borrar este bit si está activado.
SPEED	OUT	Real	Velocidad actual del accionamiento (valor escalado de la palabra de estado 2 del accionamiento) Valor de velocidad del accionamiento expresado como porcentaje de la velocidad configurada.
STATUS1	OUT	Word	Palabra de estado 1 del accionamiento: Este valor contiene bits de estado fijos de un accionamiento.
STATUS3	OUT	Word	Palabra de estado 3 del accionamiento: Este valor contiene una palabra de estado configurable por el usuario en el accionamiento.
STATUS4	OUT	Word	Palabra de estado 4 del accionamiento: Este valor contiene una palabra de estado configurable por el usuario en el accionamiento.
STATUS5	OUT	Word	Palabra de estado 5 del accionamiento: Este valor contiene una palabra de estado configurable por el usuario en el accionamiento.
STATUS6	OUT	Word	Palabra de estado 6 del accionamiento: Este valor contiene una palabra de estado configurable por el usuario en el accionamiento.
STATUS7	OUT	Word	Palabra de estado 7 del accionamiento: Este valor contiene una palabra de estado configurable por el usuario en el accionamiento.
STATUS8	OUT	Word	Palabra de estado 8 del accionamiento: Este valor contiene una palabra de estado configurable por el usuario en el accionamiento.

Cuando la instrucción USS_DRV se ejecuta por primera vez, el accionamiento que indica la dirección USS (parámetro DRIVE) se inicializa en el DB de instancia. Después de esta inicialización, las ejecuciones siguientes de USS_PORT pueden iniciar la comunicación con el accionamiento en este número de accionamiento.

Si se modifica el número del accionamiento, la CPU debe cambiar de STOP a RUN con objeto de inicializar el DB de instancia. Los parámetros de entrada se configuran en el búfer de mensajes USS TX y las salidas se leen de un búfer de respuesta válido "anterior" (si existe). Durante la ejecución de USS_DRV no se transmiten datos. Los accionamientos se comunican cuando USS_PORT se ha ejecutado. USS_DRV configura únicamente los mensajes que deben enviarse e interpreta los datos que puedan haberse recibido de una petición anterior.

El sentido de rotación del accionamiento se puede controlar utilizando la entrada DIR (Bool) o el signo (positivo o negativo) con la entrada SPEED_SP (Real). La tabla siguiente indica cómo interactúan estas entradas para determinar el sentido del accionamiento, suponiendo que el motor está cableado para la rotación adelante.

Tabla 12- 37 Interacción de los parámetros SPEED_SP y DIR

SPEED_SP	DIR	Sentido de rotación del accionamiento
Valor > 0	0	Atrás
Valor > 0	1	Adelante
Valor < 0	0	Adelante
Valor < 0	1	Atrás

12.4.3 Instrucción USS_PORT

Tabla 12- 38 Instrucción USS_PORT

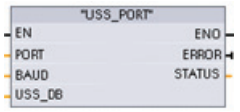
KOP / FUP	SCL	Descripción
	<pre>USS_PORT (PORT:= _uint_in_, BAUD:= _dint_in_, ERROR=> _bool_out_, STATUS=> _word_out_, USS DB:= fbtref inout);</pre>	<p>La instrucción USS_PORT gestiona la comunicación en una red USS.</p>

Tabla 12- 39 Tipos de datos para los parámetros

Parámetro y tipo		Tipo de datos	Descripción
PORT	IN	Port	Tras haber instalado y configurado un dispositivo de comunicación CM o CB, el identificador de puerto aparece en la lista desplegable de parámetros disponible en la conexión del cuadro PUERTO. El valor de puerto CM o CB asignado es la propiedad de configuración del dispositivo "identificador de hardware". El nombre simbólico del puerto se asigna en la ficha "Constantes del sistema" de la tabla de variables PLC.
BAUD	IN	DInt	Velocidad de transferencia utilizada para la comunicación USS.
USS_DB	INOUT	USS_BASE	El nombre del DB de instancia que se crea e inicializa cuando se inserta una instrucción USS_DRV en el programa.
ERROR	OUT	Bool	Si es TRUE (verdadero), esta salida indica que ha ocurrido un error y la salida STATUS es válida.
STATUS	OUT	Word	El valor de estado de la petición indica el resultado del ciclo o de la inicialización. Encontrará más información acerca de algunos códigos de estado en la variable "USS_Extended_Error".

Generalmente, el programa contiene sólo una instrucción USS_PORT por cada puerto de comunicación PtP. Cada llamada de esta función gestiona una transferencia hacia o desde un accionamiento. Todas las funciones USS asociadas a una red USS y a un puerto de comunicación PtP deben utilizar el mismo DB de instancia.

El programa debe ejecutar la instrucción USS_PORT con suficiente frecuencia para impedir timeouts del accionamiento. USS_PORT se llama generalmente desde un OB de alarma cíclica para impedir timeouts del accionamiento y para que las actualizaciones de datos USS más recientes estén disponibles para las llamadas de USS_DRV.

12.4.4 Instrucción USS_RPM

Tabla 12- 40 Instrucción USS_RPM

KOP / FUP	SCL	Descripción
	<pre>USS_RPM(REQ:=_bool_in_, DRIVE:=_usint_in_, PARAM:=_uint_in_, INDEX:=_uint_in_, DONE=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_, VALUE=>_variant_out_, USS_DB:=_fbtref_inout_);</pre>	<p>La instrucción USS_RPM lee un parámetro del accionamiento. Todas las funciones USS asociadas a una red USS y a un puerto de comunicación PtP deben utilizar el mismo bloque de datos. USS_RPM debe llamarse desde un OB de ciclo de programa principal.</p>

Tabla 12- 41 Tipos de datos para los parámetros

Tipo de parámetro		Tipo de datos	Descripción
REQ	IN	Bool	Enviar petición: Si es TRUE (verdadero), REQ indica que se desea una nueva petición de lectura. Esto se ignora si la petición para este parámetro ya está pendiente.
DRIVE	IN	USInt	Dirección del accionamiento: DRIVE es la dirección del accionamiento USS. El rango válido está comprendido entre el accionamiento 1 y el 16.
PARAM	IN	UInt	Número de parámetro: PARAM designa el parámetro del accionamiento que se escribe. El rango de este parámetro está comprendido entre 0 y 2047. En algunos accionamientos, el byte más significativo puede acceder a valores PARAM superiores a 2047. Encontrará más información sobre cómo acceder a un rango ampliado en el manual del accionamiento.
INDEX	IN	UInt	Índice de parámetro: INDEX designa el índice de parámetro del accionamiento que se escribirá. Valor de 16 bits en el que el byte menos significativo es el valor de índice real en un rango de 0 a 255. El accionamiento también puede utilizar el byte más significativo. Este byte es específico del accionamiento. Para más información, consulte el manual del accionamiento.
USS_DB	INOUT	USS_BASE	El nombre del DB de instancia que se crea e inicializa cuando se inserta una instrucción USS_DRV en el programa.
VALUE	IN	Word, Int, UInt, DWord, DInt, UDIInt, Real	Valor del parámetro que se ha leído y que es válido sólo si el bit DONE es TRUE (verdadero).
DONE ¹	OUT	Bool	Si es TRUE (verdadero), indica que la salida VALUE contiene el valor del parámetro de lectura solicitado anteriormente. Este bit se activa cuando USS_DRV detecta los datos de respuesta de lectura del accionamiento. Este bit se desactiva en los casos siguientes: los datos de respuesta se solicitan mediante otra consulta USS_RPM o bien en la segunda de las dos llamadas siguientes de USS_DRV.

12.4 Comunicación de interfaz serie universal (USS)

Tipo de parámetro		Tipo de datos	Descripción
ERROR	OUT	Bool	Ha ocurrido un error: Si es TRUE (verdadero), ERROR indica que ha ocurrido un error y la salida STATUS es válida. Todas las demás salidas se ponen a cero cuando ocurre un error. Los errores de comunicación se notifican sólo en las salidas ERROR y STATUS de la instrucción USS_PORT.
STATUS	OUT	Word	STATUS indica el resultado de la petición de lectura. Encontrará más información acerca de algunos códigos de estado en la variable "USS_Extended_Error".

¹ El bit DONE indica que se han leído datos válidos del accionamiento de motor referenciado y que se han transmitido a la CPU. No indica que la librería USS sea capaz de leer inmediatamente otros parámetros. Hay que enviar una petición PKW vacía al accionamiento del motor, la cual debe ser acusada por la instrucción antes de que el canal de parámetros para el accionamiento en cuestión vuelva a estar disponible. Si se llama inmediatamente una FC USS_RPM o USS_WPM para el accionamiento de motor especificado se producirá un error 0x818A.

12.4.5 Instrucción USS_WPM

Nota

Operaciones de escritura EEPROM (para la EEPROM dentro de un accionamiento USS)

No utilice excesivamente la operación de escritura permanente en EEPROM. Minimice el número de operaciones de escritura en EEPROM para prolongar la vida útil de la EEPROM.

Tabla 12- 42 Instrucción USS_WPM

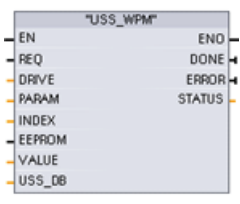
KOP / FUP	SCL	Descripción
	<pre> USS_WPM (REQ:=_bool_in_, DRIVE:=_usint_in_, PARAM:=_uint_in_, INDEX:=_uint_in_, EEPROM:=_bool_in_, VALUE:=_variant_in_, DONE=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_, USS_DB:=_fbtref_inout_); </pre>	<p>La instrucción USS_WPM modifica un parámetro en el accionamiento. Todas las funciones USS asociadas a una red USS y a un puerto de comunicación PTP deben utilizar el mismo bloque de datos.</p> <p>USS_WPM debe llamarse desde un OB de ciclo de programa principal.</p>

Tabla 12- 43 Tipos de datos para los parámetros

Parámetro y tipo		Tipo de datos	Descripción
REQ	IN	Bool	Enviar petición: Si es TRUE (verdadero), REQ indica que se desea una nueva petición de escritura. Esto se ignora si la petición para este parámetro ya está pendiente.
DRIVE	IN	USInt	Dirección del accionamiento: DRIVE es la dirección del accionamiento USS. El rango válido está comprendido entre el accionamiento 1 y el 16.
PARAM	IN	UInt	Número de parámetro: PARAM designa el parámetro del accionamiento que se escribe. El rango de este parámetro está comprendido entre 0 y 2047. En algunos accionamientos, el byte más significativo puede acceder a valores PARAM superiores a 2047. Encontrará más información sobre cómo acceder a un rango ampliado en el manual del accionamiento.
INDEX	IN	UInt	Índice de parámetro: INDEX designa el índice de parámetro del accionamiento que se escribirá. Valor de 16 bits en el que el byte menos significativo es el valor de índice real en un rango de 0 a 255. El accionamiento también puede utilizar el byte más significativo. Este byte es específico del accionamiento. Para más información, consulte el manual del accionamiento.
EEPROM	IN	Bool	Guardar en la EEPROM del accionamiento: Si es TRUE (verdadero), una operación de escritura en los parámetros del accionamiento se guardará en la EEPROM de éste. Si es FALSE (falso), la operación de escritura será temporal por lo que no se conservará tras desconectar y volver a conectar la alimentación del accionamiento.
VALUE	IN	Word, Int, UInt, DWord, DInt, UInt, Real	Valor del parámetro en el que se debe escribir. Debe ser válido en la transición de REQ.
USS_DB	INOUT	USS_BASE	El nombre del DB de instancia que se crea e inicializa cuando se inserta una instrucción USS_DRV en el programa.
DONE ¹	OUT	Bool	Si es TRUE (verdadero), DONE indica que la entrada VALUE se ha escrito en el accionamiento. Este bit se activa cuando USS_DRV detecta los datos de respuesta de escritura del accionamiento. El bit se desactiva cuando los datos de respuesta se solicitan mediante otra consulta USS_WPM o bien en la segunda de las dos llamadas siguientes de USS_DRV.
ERROR	OUT	Bool	Si es TRUE (verdadero), ERROR indica que ha ocurrido un error y la salida STATUS es válida. Todas las demás salidas se ponen a cero cuando ocurre un error. Los errores de comunicación se notifican sólo en las salidas ERROR y STATUS de la instrucción USS_PORT.
STATUS	OUT	Word	STATUS indica el resultado de la petición de escritura. Encontrará más información acerca de algunos códigos de estado en la variable "USS_Extended_Error".

¹ El bit DONE indica que se han leído datos válidos del accionamiento de motor referenciado y que se han transmitido a la CPU. No indica que la librería USS sea capaz de leer inmediatamente otros parámetros. Hay que enviar una petición PKW vacía al accionamiento del motor, la cual debe ser acusada por la instrucción antes de que el canal de parámetros para el accionamiento en cuestión vuelva a estar disponible. Si se llama inmediatamente una FC USS_RPM o USS_WPM para el accionamiento de motor especificado se producirá un error 0x818A.

12.4.6 Códigos de estado USS

La salida STATUS de las funciones USS devuelve los códigos de estado de estas funciones.

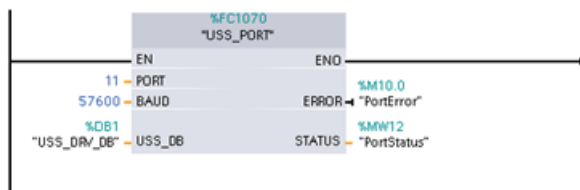
Tabla 12- 44 Códigos STATUS ¹

STATUS (W#16#....)	Descripción
0000	No hay error
8180	La longitud de la respuesta del accionamiento no concuerda con los caracteres recibidos del accionamiento. El número de accionamiento en el que ha ocurrido el error se devuelve en la variable "USS_Extended_Error". La descripción de los errores avanzados aparece a continuación de esta tabla.
8181	El parámetro VALUE no es un tipo de datos Word, Real o DWord.
8182	El usuario ha introducido un parámetro en formato Word y se ha recibido una respuesta en formato DWord o Real del accionamiento.
8183	El usuario ha introducido un parámetro en formato DWord o Real y se ha recibido una respuesta en formato Word del accionamiento.
8184	El telegrama de respuesta del accionamiento tiene una suma de verificación incorrecta. El número de accionamiento en el que ha ocurrido el error se devuelve en la variable "USS_Extended_Error". La descripción de los errores avanzados aparece a continuación de esta tabla.
8185	Dirección del accionamiento no válida (rango de direcciones válidas: de 1 a 16)
8186	La consigna de velocidad está fuera del rango válido (rango de SP de velocidad válido: -200% a 200%).
8187	Un número de accionamiento incorrecto ha respondido a la petición enviada. El número de accionamiento en el que ha ocurrido el error se devuelve en la variable "USS_Extended_Error". La descripción de los errores avanzados aparece a continuación de esta tabla.
8188	Se ha indicado una longitud de palabra PZD no permitida (rango válido = 2, 4, 6 u 8 palabras)
8189	Se ha indicado una velocidad de transferencia no permitida.
818A	Otra petición para este accionamiento está utilizando el canal de petición de parámetros.
818B	El accionamiento no ha respondido a las peticiones ni reintentos. El número de accionamiento en el que ha ocurrido el error se devuelve en la variable "USS_Extended_Error". La descripción de los errores avanzados aparece a continuación de esta tabla.
818C	El accionamiento ha devuelto un error avanzado relativo a la petición de parámetros. La descripción de los errores avanzados aparece a continuación de esta tabla.
818D	El accionamiento ha devuelto un error de acceso no permitido relativo a la petición de parámetros. Para más información sobre la limitación de acceso a los parámetros, consulte el manual del accionamiento.
818E	El accionamiento no se ha inicializado. Este código de error se devuelve a USS_RPM o USS_WPM si USS_DRV no se ha llamado por lo menos una vez para este accionamiento. De esta manera se impide que la inicialización de USS_DRV en el primer ciclo sobrescriba una petición pendiente de lectura o escritura de los parámetros, puesto que inicializa el accionamiento como entrada nueva. Para corregir este error, llame USS_DRV para este número de accionamiento.
80Ax-80Fx	Errores específicos que devuelven los FBs de comunicación PtP llamados por la librería USS - La librería USS no modifica estos códigos de error que se definen en las descripciones de la instrucción PtP.

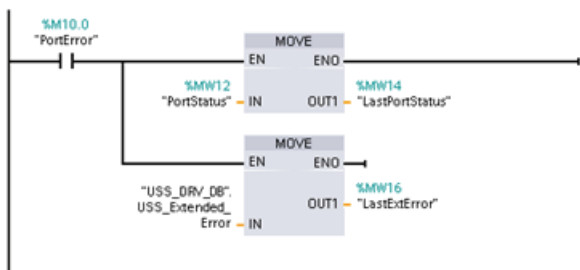
¹ Además de los errores de la instrucción USS indicados arriba, las instrucciones de comunicaciones PtP (Página 592) subyacentes pueden devolver errores.

Para muchos códigos STATUS se suministra información adicional en la variable "USS_Extended_Error" del DB instancia USS_DRV. Para los códigos STATUS hexadecimales 8180, 8184, 8187 y 818B, USS_Extended_Error contiene el número del accionamiento en que se ha presentado el error de comunicación. Para el código STATUS hexadecimal 818C, USS_Extended_Error contiene un código de error que devuelve el accionamiento cuando se utiliza una instrucción USS_RPM o USS_WPM.

Los errores de comunicación (STATUS = 16#818B) se notifican sólo en la instrucción USS_PORT y no en la instrucción USS_DRV. Si, por ejemplo, la red no ha sido finalizada correctamente, un accionamiento puede pasar a RUN, no obstante la instrucción USS_DRV mostrará el valor 0 para todos los parámetros de salida. En este caso, el error de comunicación sólo se puede detectar en la instrucción USS_PORT. Debido a que este error sólo es visible durante un ciclo, hay que agregar algunos elementos lógicos de detección, como se indica en el ejemplo a continuación. En este ejemplo, si el bit de error de la instrucción USS_PORT es TRUE, entonces los valores STATUS y USS_Extended_Error se guardan en el área de marcas. El número de accionamiento se deposita en la variable USS_Extended_Error si el código STATUS tiene el valor hexadecimal 8180, 8184, 8187 u 818B.



El estado del puerto de la **Red 1** "PortStatus" y los valores de los códigos de error avanzados de "USS_DRV_DB".USS_Extended_Error sólo son válidos durante un ciclo del programa. Los valores deben capturarse para un procesamiento posterior.



Red 2 El contacto "PortError" inicia el almacenamiento del valor "PortStatus" en "LastPortStatus" y del valor "USS_DRV_DB".USS_Extended_Error en "LastExtError".

Los accionamientos USS soportan accesos de lectura y escritura a los parámetros internos de los accionamientos. Esta función permite controlar y configurar el accionamiento de forma remota. Las operaciones de acceso a los parámetros del accionamiento pueden fallar debido a errores tales como valores fuera de rango o peticiones no permitidas del modo actual del accionamiento. El accionamiento genera un código de error cuyo valor se devuelve en la variable "USS_Extended_Error". Este valor del código de error es válido únicamente para la última ejecución de una instrucción USS_RPM o USS_WPM. El código de error del accionamiento se deposita en la variable USS_Extended_Error si el STATUS code tiene el valor hexadecimal 818C. El valor del código de error de "USS_Extended_Error" depende del modelo de accionamiento. Los códigos de error avanzados para las operaciones de lectura y escritura de parámetros se describen en el manual del accionamiento.

12.4.7 Información general sobre la configuración del accionamiento

Requisitos generales para la configuración del accionamiento

- Los accionamientos deben configurarse para utilizar 4 palabras PKW.
- Los accionamientos pueden configurarse para 2, 4, 6 u 8 palabras PZD.
- El número de palabras PZD en el accionamiento debe concordar con la entrada PZD_LEN de la instrucción USS_DRV para ese accionamiento.
- La velocidad de transferencia en todos los accionamientos debe concordar con la entrada BAUD de la instrucción USS_PORT.
- El accionamiento debe configurarse para control remoto.
- Para el accionamiento debe configurarse una consigna de frecuencia a USS en el puerto COM.
- La dirección del accionamiento debe configurarse en un rango de 1 a 16 y debe concordar con la entrada DRIVE del bloque USS_DRV para ese accionamiento.
- El control de sentido del accionamiento debe configurarse para que utilice la polaridad de la consigna del accionamiento.
- La red RS485 se debe finalizar correctamente.

Conectar un accionamiento MicroMaster

Esta información sobre accionamientos SIEMENS MicroMaster se suministra a modo de ejemplo. Para otros accionamientos, consulte el manual respectivo para obtener instrucciones de configuración.

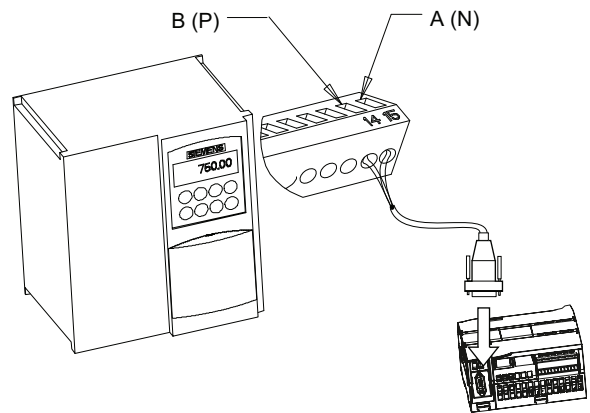
Para conectar un accionamiento MicroMaster 4 (MM4), inserte los extremos del cable RS485 en los dos terminales fijadores sin tornillos previstos para el protocolo USS. El cable PROFIBUS y los conectores estándar sirven para conectar el S7-1200.

PRECAUCIÓN

Si se interconectan equipos con potenciales de referencia diferentes, podrían circular corrientes indeseadas por el cable de conexión.

Estas corrientes pueden causar errores de comunicación o deteriorar los equipos. Para evitar corrientes indeseadas, vigile que todos los equipos conectados con un cable de comunicación compartan un circuito de referencia, o bien que estén aislados entre sí. El blindaje debe conectarse a tierra, o bien al pin 1 del conector de 9 pines. Es recomendable conectar a tierra el terminal 20V del accionamiento MicroMaster.

Los dos hilos del extremo opuesto del cable RS485 se deben insertar en los bloques de terminales del accionamiento MM4. Para conectar el cable a un accionamiento MM4, desmonte la(s) tapa(s) del accionamiento para acceder al bloque de terminales. Para más información sobre cómo desmontar la(s) tapa(s), consulte el manual del accionamiento MM4.

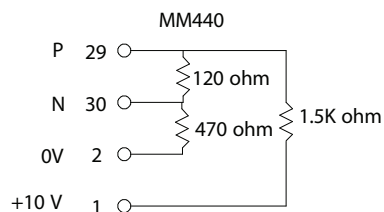
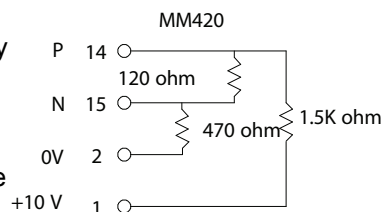


Las conexiones del bloque de terminales están etiquetadas de forma numérica. Utilizando un conector PROFIBUS en el lado del S7-1200, conecte el terminal A del cable al terminal 15 del accionamiento (si es un MM420), o bien al terminal 30 (si es un MM440). Conecte el terminal B del cable B (P) A (N) al terminal 14 (si es un MM420), o bien al terminal 29 (si es un MM440).

Si el S7-1200 está situado en un extremo de la red o si se trata de una conexión punto a punto, será preciso utilizar los terminales A1 y B1 (en vez de A2 y B2) del conector, puesto que éstos permiten ajustar el cierre (p. ej. utilizando el conector DP con el número de referencia 6ES7 972--0BA40--0X40).

	PRECAUCIÓN
<p>Monte correctamente de nuevo las tapas del accionamiento antes de conectar la alimentación.</p>	

Si el accionamiento está situado en un extremo de la red, los resistores de cierre y polarización se deberán conectar también a los terminales correctos. El diagrama muestra ejemplos de conexión del accionamiento MM4 necesarios para cierre y polarización.



Configurar accionamientos MicroMaster 4

Antes de conectar un accionamiento al S7-1200, vigile que tenga los siguientes parámetros de sistema. Utilice el teclado del accionamiento para ajustar los parámetros:

1. Restablezca los ajustes de fábrica del accionamiento (opcional).	P0010=30 P0970=1
Si omite el paso 1, entonces asegúrese de que estos parámetros tengan los valores indicados.	Longitud USS PZD = P2012 índice 0=(2, 4, 6 u 8) Longitud USS PKW = P2013 índice 0=4
2. Habilite el acceso de lectura/escritura a todos los parámetros (modo avanzado).	P0003=3
3. Compruebe los ajustes del motor. Los ajustes varían dependiendo del (de los) motor(es) utilizados. Para configurar los parámetros P304, P305, P307, P310 y P311, deberá ajustar primero el parámetro P010 a "1" (modo de puesta en servicio rápida). Tras concluir la configuración, ajuste el parámetro P010 a "0". Los parámetros P304, P305, P307, P310 y P311 sólo se pueden cambiar en el modo de puesta en servicio rápida.	P0304 = Tensión nominal del motor (V) P0305 = Intensidad nominal del motor (A) P0307 = Potencia nominal del motor (W) P03010 = Frecuencia nominal del motor (Hz) P0311 = Velocidad nominal del motor
4. Ajuste el modo de control (local o remoto).	P0700 índice 0=5
5. Seleccione la consigna de frecuencia a USS en el puerto COM.	P1000 índice 0=5
6. Tiempo de aceleración (opcional) Éste es el tiempo (indicado en segundos) que el motor necesita para acelerar hasta la frecuencia máxima.	P1120=(de 0 a 650,00)
7. Tiempo de deceleración (opcional) Éste es el tiempo (indicado en segundos) que el motor necesita para decelerar hasta una parada completa.	P1121=(de 0 a 650,00)
8. Ajuste la frecuencia de referencia del puerto serie:	P2000=(de 1 a 650 Hz)
9. Ajuste la normalización USS:	P2009 índice 0=0
10. Ajuste la velocidad de transferencia del puerto serie RS485:	P2010 índice 0= 4 (2400 bit/s) 5 (4800 bit/s) 6 (9600 bit/s) 7 (19200 bit/s) 8 (38400 bit/s) 9 (57600 bit/s) 12 (115200 bit/s)
11. Introduzca la dirección del esclavo. Todos los accionamientos (31 como máximo) se pueden controlar a través del bus.	P2011 índice 0=(de 0 a 31)
12. Ajuste el timeout para el puerto serie. Éste es el intervalo máximo admisible entre la recepción de dos telegramas de datos. Esta función se utiliza para desactivar el inversor en caso de un fallo de comunicación. La temporización comienza tras haberse recibido un telegrama de datos válido. Si no se recibe otro telegrama de datos dentro del período de tiempo indicado, el inversor se desactivará y se visualizará el código de error F0070. Ajustando el valor a cero se desconecta el control.	P2014 índice 0=(de 0 a 65.535 ms) 0=sin timeout
13. Transfiera los datos de la RAM a la EEPROM:	P0971=1 (iniciar la transferencia) Guardar los cambios de los parámetros en la EEPROM

12.5 Comunicación Modbus

12.5.1 Vista general de las comunicaciones Modbus RTU y TCP

Códigos de función Modbus

- Una CPU que actúe como maestro Modbus RTU (o cliente Modbus TCP) puede leer y escribir tanto datos como estados de E/S en un esclavo Modbus RTU (o servidor Modbus TCP) remoto. Los datos remotos se pueden leer y procesar en el programa de usuario.
- Una CPU que actúe como esclavo Modbus RTU (o servidor Modbus TCP) permite que un dispositivo de supervisión lea y escriba tanto datos como estados de E/S en una CPU remota. El dispositivo de supervisor puede escribir valores nuevos en memoria de la CPU remota que podrá procesar el programa de usuario.

Tabla 12- 45 Funciones de lectura de datos: Leer datos de programa y E/S remotos

Código de función Modbus	Funciones de lectura de esclavo (servidor), direccionamiento estándar
01	Leer bits de salida: De 1 a 2000 bits por petición
02	Leer bits de entrada: De 1 a 2000 bits por petición
03	Leer registros de retención: De 1 a 125 palabras por petición
04	Leer palabras de entrada: De 1 a 125 palabras por petición

Tabla 12- 46 Funciones de escritura de datos: Escribir en E/S remota y modificar datos de programa

Código de función Modbus	Funciones de escritura de esclavo (servidor), direccionamiento estándar
05	Escribir un bit de salida: 1 bit por petición
06	Escribir un registro de retención: 1 palabra por petición
15	Escribir uno o más bits de salida: De 1 a 1968 bits por petición
16	Escribir uno o más registros de retención: De 1 a 123 palabras por petición

- Los códigos de función Modbus 08 y 11 proporcionan información de diagnóstico de comunicaciones con dispositivos esclavos.
- El código de función Modbus 0 envía un mensaje Broadcast a todos los esclavos (sin respuesta de los esclavos). La función Broadcast no está disponible para Modbus TCP porque las comunicaciones se basan en conexión.

Tabla 12- 47 Direcciones de estación de red Modbus

Estación		Dirección
Estación RTU	Dirección de estación estándar	1 a 247
	Dirección de estación avanzada	1 a 65535
Estación TCP	Dirección de estación	Dirección IP y número de puerto

Direcciones de memoria Modbus

El número de direcciones de memoria Modbus disponible depende del modelo de CPU, la cantidad de memoria de trabajo y la cantidad de memoria de CPU usada por otros datos de programa. En esta tabla, se indican valores nominales de rangos de direcciones.

Tabla 12- 48 Direcciones de memoria Modbus

Estación		Rango de direcciones
Estación RTU	Dirección de memoria estándar	10K
	Dirección de memoria avanzada	64K
Estación TCP	Dirección de memoria estándar	10K

Comunicación Modbus RTU

Modbus RTU (Remote Terminal Unit) es un protocolo de comunicaciones en red estándar que utiliza conexiones eléctricas RS232 o RS485 para transferencia serie de datos entre dispositivos de red Modbus. Se pueden añadir puertos de red PtP (punto a punto) a una CPU con RS232, CM RS485 o CB RS485.

Modbus RTU utiliza una red maestro/esclavo en la que un solo dispositivo maestro inicia todas las comunicaciones y los esclavos sólo pueden responder a una petición del maestro. El maestro envía una petición a una dirección de esclavo y solo esa dirección de esclavo responde a la orden.

Comunicación Modbus TCP

Modbus TCP (Transmission Control Protocol) es un protocolo de comunicaciones en red estándar que usa el conector PROFINET de la CPU para comunicaciones TCP/IP. No se necesitan módulos hardware de comunicación adicionales.

Modbus TCP utiliza conexiones para Open User Communications (OUC) como vía de comunicaciones Modbus. Puede haber varias conexiones cliente-servidor, además de la conexión entre STEP 7 y la CPU. Se admiten conexiones de cliente y de servidor mezcladas hasta el número máximo de conexiones que admita el modelo de CPU (Página 442).

Cada conexión MB_SERVER debe usar un DB de instancia y número de puerto IP únicos. Sólo se admite 1 conexión por puerto IP. Cada MB_SERVER (con su DB de instancia y puerto IP únicos) se debe ejecutar individualmente para cada conexión.

Nota

Modbus TCP sólo funcionará correctamente con la versión V1.02 de firmware de CPU o posterior. Se producirá un error si se intentan ejecutar instrucciones Modbus en una versión de firmware anterior.

Un cliente Modbus TCP (maestro) debe controlar la conexión cliente-servidor con el parámetro DISCONNECT. Las acciones de cliente Modbus básicas se indican a continuación.

1. Iniciar una conexión a una dirección IP y número de puerto IP particulares de un servidor (esclavo)
2. Iniciar transmisión de cliente de un mensaje Modbus y recibir las respuestas del servidor
3. Cuando se desee, iniciar la desconexión de cliente y servidor para permitir la conexión con otro servidor.

Instrucciones Modbus RTU en el programa

- MB_COMM_LOAD: Una ejecución de MB_COMM_LOAD se utiliza para configurar parámetros de puerto PtP como velocidad de transferencia, paridad y control de flujo. Tras configurar un puerto de la CPU para el protocolo Modbus RTU, este sólo puede ser utilizado por la instrucción MB_MASTER o MB_SLAVE.
- MB_MASTER: La instrucción de maestro Modbus permite que la CPU actúe como maestro Modbus RTU y se comunique con uno o varios esclavos Modbus.
- MB_SLAVE: La instrucción de esclavo Modbus permite que la CPU actúe como esclavo Modbus RTU y se comunique con un maestro Modbus.

Instrucciones Modbus TCP en el programa

- MB_CLIENT: Establece la conexión TCP cliente-servidor, envía mensaje de orden, recibe respuesta y controla la desconexión desde el servidor
- MB_SERVER: Conecta a un cliente Modbus TCP bajo petición, recibe mensaje Modbus y envía respuesta

12.5.2 Modbus TCP

12.5.2.1 MB_CLIENT (Modbus TCP)

Tabla 12- 49 Instrucción MB_CLIENT

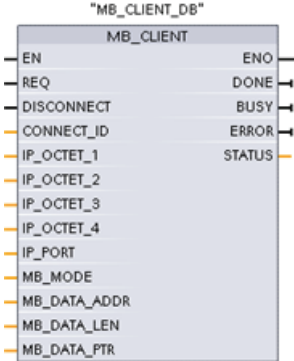
KOP / FUP	SCL	Descripción
	<pre>"MB_CLIENT_DB" (REQ:= bool_in_, DISCONNECT:= bool_in_, CONNECT_ID:= uint_in_, IP_OCTET_1:= byte_in_, IP_OCTET_2:= byte_in_, IP_OCTET_3:= byte_in_, IP_OCTET_4:= byte_in_, IP_PORT:= uint_in_, MB_MODE:= usint_in_, MB_DATA_ADDR:= udint_in_, MB_DATA_LEN:= uint_in_, DONE=> bool_out_, BUSY=> bool_out_, ERROR=> bool_out_, STATUS=> word_out_, MB_DATA_PTR:= variant inout);</pre>	<p>MB_CLIENT se comunica como cliente Modbus TCP a través del conector PROFINET de la CPU S7-1200. No se necesitan módulos hardware de comunicación adicionales.</p> <p>MB_CLIENT puede establecer una conexión cliente-servidor, enviar una petición de función Modbus, recibir una respuesta y controlar la desconexión de un servidor Modbus TCP.</p>

Tabla 12- 50 Tipos de datos para los parámetros

Parámetro y tipo	Tipo de datos	Descripción
REQ	In	Bool FALSE = Ninguna petición de comunicación Modbus TRUE = Petición de comunicación con un servidor Modbus TCP
DISCONNECT	IN	Bool El parámetro DISCONNECT permite que su programa controle la conexión y desconexión con un dispositivo servidor Modbus. Si DISCONNECT = 0 y no hay ninguna conexión, entonces MB_CLIENT intenta establecer una conexión con la dirección IP y el número de puerto asignados. Si DISCONNECT = 1 y existe una conexión, entonces se intenta una operación de desconexión. Siempre que esta entrada esté habilitada, no se intentará ninguna otra operación.
CONNECT_ID	IN	UInt El parámetro CONNECT_ID debe identificar unívocamente cada conexión dentro del PLC. Cada instancia única de la instrucción MB_CLIENT o MB_SERVER debe contener un parámetro CONNECT_ID único.
IP_OCTET_1	IN	USInt Dirección IP del servidor Modbus TCP: octeto 1 Parte de 8 bits de la dirección IP IPv4 de 32 bits del servidor Modbus TCP al que se conectará el cliente y con el que se comunicará mediante el protocolo Modbus TCP.
IP_OCTET_2	IN	USInt Dirección IP del servidor Modbus TCP: octeto 2
IP_OCTET_3	IN	USInt Dirección IP del servidor Modbus TCP: octeto 3

Parámetro y tipo		Tipo de datos	Descripción
IP_OCTET_4	IN	USInt	Dirección IP del servidor Modbus TCP: octeto 4
IP_PORT	IN	UInt	Valor predeterminado = 502: el número de puerto IP del servidor al que se intentará conectar el cliente y con el que se acabará comunicando mediante el protocolo TCP/IP.
MB_MODE	IN	USInt	Selección de modo: asigna el tipo de petición (lectura, escritura o diagnóstico). Consulte los detalles en la tabla de funciones Modbus que aparece más abajo.
MB_DATA_ADDR	IN	UDInt	Dirección inicial Modbus: asigna la dirección inicial de los datos a los que accederá MB_CLIENT. Las direcciones válidas se indican en la tabla de funciones Modbus que aparece más abajo.
MB_DATA_LEN	IN	UInt	Longitud de datos Modbus: asigna el número de bits o palabras a las que debe accederse en esta petición. Las longitudes válidas se indican en la tabla de funciones Modbus que aparece más abajo.
MB_DATA_PTR	IN_OUT	Variant	Puntero al registro de datos Modbus: el registro almacena temporalmente datos que van a un servidor Modbus o vienen de este. El puntero debe asignar una dirección de memoria M o un DB global estándar.
DONE	OUT	Bool	El bit DONE es TRUE durante un ciclo tras haberse finalizado la última petición sin errores.
BUSY	OUT	Bool	<ul style="list-style-type: none"> • 0 - Ninguna operación MB_CLIENT en curso • 1 - Operación MB_CLIENT en curso
ERROR	OUT	Bool	El bit ERROR es TRUE durante un ciclo tras haberse finalizado la ejecución de MB_CLIENT con un error. El valor del código de error en el parámetro STATUS es válido solo durante el ciclo en que ERROR = TRUE.
STATUS	OUT	Word	Código de condición de ejecución

Parámetro REQ

FALSE = Ninguna petición de comunicación Modbus

TRUE = Petición de comunicación con un servidor Modbus TCP

Si ninguna instancia de MB_CLIENT está activa y el parámetro DISCONNECT=0, se iniciará una nueva petición Modbus cuando REQ=1. Si la conexión no está todavía establecida, se establecerá una nueva.

Si la misma instancia de MB_CLIENT se ejecuta nuevamente con DISCONNECT=0 y REQ=1, antes de que finalice la petición actual no se realizarán más transmisiones Modbus. No obstante, en cuanto finalice la petición actual, se podrá procesar otra petición si MB_CLIENT se ejecuta con REQ=1.

Cuando finalice la petición de comunicación MB_CLIENT actual, el bit DONE será TRUE durante un ciclo. El bit DONE se puede usar como puerta temporal para secuenciar varias peticiones MB_CLIENT.

Nota

Coherencia de los datos de entrada durante el procesamiento de MB_CLIENT

Una vez que un cliente Modbus inicia una operación Modbus, todos los estados de entrada se guardan internamente y se comparan en cada llamada sucesiva. La comparación se utiliza para determinar si una llamada en particular originó la petición de cliente activa. Se puede realizar más de una llamada de MB_CLIENT con un DB de instancia común.

Por lo tanto, es importante que las entradas no se cambien durante el tiempo en el que se procesa activamente una operación MB_CLIENT . Si no se observa esta regla, un MB_CLIENT no puede determinar que es la instancia activa.

Los parámetros MB_MODE y MB_DATA_ADDR seleccionan la función de comunicación Modbus

MB_DATA_ADDR asigna la dirección Modbus inicial de los datos a los que se accederá. La instrucción MB_CLIENT utiliza una entrada MB_MODE en vez de una entrada de código de función.

La combinación de los valores MB_MODE y MB_DATA_ADDR determina el código de función utilizado en el mensaje Modbus real. La tabla siguiente muestra la correspondencia entre el parámetro MB_MODE, la función Modbus y el rango de direcciones Modbus.

Tabla 12- 51 Funciones Modbus

MB_MODE	Función Modbus	Longitud de datos	Operación y datos	MB_DATA_ADDR
0	01	De 1 a 2000	Leer bits de salida: de 1 a 2000 bits por petición	De 1 a 9999
0	02	De 1 a 2000	Leer bits de entrada: de 1 a 2000 bits por petición	De 10001 a 19999
0	03	De 1 a 125	Leer registros de retención: de 1 a 125 palabras por petición	De 40001 a 49999 o De 400001 a 465535
0	04	De 1 a 125	Leer palabras de entrada: de 1 a 125 palabras por petición	De 30001 a 39999
1	05	1	Escribir un bit de salida: un bit por petición	De 1 a 9999
1	06	1	Escribir un registro de retención: 1 palabra por petición	De 40001 a 49999 o De 400001 a 465535
1	15	De 2 a 1968	Escribir varios bits de salida: de 2 a 1968 bits por petición	De 1 a 9999
1	16	De 2 a 123	Escribir varios registros de retención: de 2 a 123 palabras por petición	De 40001 a 49999 o De 400001 a 465535
2	15	De 1 a 1968	Escribir uno o más bits de salida: de 1 a 1968 bits por petición	De 1 a 9999
2	16	De 1 a 123	Escribir uno o más registros de retención: de 1 a 123 palabras por petición	De 40001 a 49999 o De 400001 a 465535

MB_MODE	Función Modbus	Longitud de datos	Operación y datos	MB_DATA_ADDR
11	11	0	Leer la palabra de estado de la comunicación del servidor y el contador de eventos. Esta palabra de estado indica ocupado (0 – no ocupado, 0xFFFF - ocupado). El contador de eventos se incrementa cada vez que se finaliza un mensaje correctamente. En esta función se ignoran los parámetros MB_DATA_ADDR y MB_DATA_LEN de MB_CLIENT.	
80	08	1	Comprobar el estado del servidor utilizando el código de diagnóstico de datos 0x0000 (test de Loopback – el servidor envía de regreso la petición) 1 palabra por petición	
81	08	1	Inicializar el contador de eventos del servidor utilizando el código de diagnóstico de datos 0x000A 1 palabra por petición	
De 3 a 10, de 12 a 79, de 82 a 255			Reservado	

Nota

MB_DATA_PTR asigna un búfer para guardar datos leídos/escritos hasta/desde un servidor Modbus TCP

El búfer de datos puede estar en una dirección de memoria M o un DB global estándar.

Para un búfer en memoria M, utilice el formato de puntero ANY estándar. Este está en el formato P#"dirección de bit" "tipo de datos" "longitud"; por ejemplo: P#M1000.0 WORD 500.

MB_DATA_PTR asigna un búfer de comunicación

- Funciones de comunicación de MB_CLIENT:
 - Leer y escribir datos de 1 bit de las direcciones del servidor Modbus (00001 a 09999)
 - Leer datos de 1 bit de las direcciones del servidor Modbus (de 10001 a 19999)
 - Leer datos de palabras de 16 bits de las direcciones del servidor Modbus (de 30001 a 39999) y (de 40001 a 49999)
 - Escribir datos de palabras de 16 bit de las direcciones del servidor Modbus (40001 a 49999)
- Los datos de tamaño de palabra o de bit se transfieren a/desde el búfer de memoria M o el DB asignado por MB_DATA_PTR.

- Si MB_DATA_PTR asigna un DB como búfer, el usuario debe asignar tipos de datos a todos los elementos de datos del DB.
 - El tipo de datos de 1 bit Bool representa una dirección de bit Modbus
 - Los tipos de datos de una palabra de 16 bits como WORD, UInt e Int representan una dirección de palabra Modbus
 - Los tipos de datos de palabra doble de 32 bits como DWORD, DInt y Real representan dos direcciones de palabra Modbus
- MB_DATA_PTR puede asignar elementos de DB complejos como:
 - Matrices estándar
 - Estructuras con nombre en las que cada elemento es único
 - Estructuras complejas con nombre, en las que cada elemento tiene un nombre unívoco y un tipo de datos de 16 o 32 bits
- No es necesario que las áreas de datos de MB_DATA_PTR se encuentren en el mismo bloque de datos global (o área de memoria M). Se puede asignar un bloque de datos para lecturas Modbus, otro para escrituras Modbus o un bloque de datos para cada estación MB_CLIENT.

Varias conexiones de cliente

Un cliente Modbus TCP puede admitir conexiones concurrentes hasta el número máximo de conexiones para Open User Communications permitidas por el PLC. El número total de conexiones en un PLC dado, clientes y servidores Modbus TCP incluidos, no debe sobrepasar el número máximo de conexiones para Open User Communications admitidas (Página 442). Las conexiones Modbus TCP pueden compartirse entre conexiones de cliente o de servidor.

Cada conexión de cliente debe cumplir estas reglas:

- Cada conexión MB_CLIENT debe usar un DB de instancia distinto
- Cada conexión MB_CLIENT debe especificar una dirección IP de servidor única
- Cada conexión MB_CLIENT debe especificar una ID de conexión única
- Según la configuración del servidor pueden ser necesarios, o no, números únicos de puerto IP

La ID de conexión debe ser unívoca para cada conexión individual. Es decir, sólo debe usarse una ID de conexión única con cada DB de instancia individual. Resumiendo, el DB de instancia y la ID de conexión están emparejadas y deben ser únicas para cada conexión.

Tabla 12- 52 Variables estáticas del bloque de datos de instancia MB_CLIENT accesibles por el usuario

Variable	Tipo de datos	Valor predeterminado	Descripción
Blocked_Proc_Timeout	Real	3.0	Tiempo (en segundos) que hay que esperar a una instancia cliente Modbus bloqueada antes de eliminar esta instancia como ACTIVA. Esto puede ocurrir, por ejemplo, si se ha lanzado una petición de cliente y la aplicación deja de ejecutar la función de cliente antes de haber finalizado por completo la petición. El límite máximo de S7-1200 es 55 segundos.
MB_Unit_ID	Word	255	Identificador de unidad Modbus: Un servidor TCP Modbus TCP se direcciona a través de su dirección IP. Por lo tanto, el parámetro MB_UNIT_ID no se usa para el direccionamiento Modbus TCP. El parámetro MB_UNIT_ID corresponde a la dirección de esclavo en el protocolo Modbus RTU. Si un servidor Modbus TCP se usa como pasarela a un protocolo Modbus RTU, MB_UNIT_ID puede usarse para identificar el dispositivo esclavo conectado en la red serie. MB_UNIT_ID se usaría para reenviar la solicitud a la dirección del esclavo Modbus RTU correcta. Nótese que algunos dispositivos Modbus TCP pueden necesitar que el parámetro MB_UNIT_ID se inicialice dentro de un rango restringido de valores.
RCV_TIMEOUT	Real	2.0	Tiempo en segundos que el MB_CLIENT espera a que el servidor responda a una petición.
Conectado	Bool	0	Indica si la conexión al servidor asignado está conectada o desconectada: 1=conectada, 0=desconectada

Tabla 12- 53 Errores de protocolo MB_CLIENT

STATUS (W#16#)	Código de respuesta al cliente Modbus (B#16#)	Errores de protocolo Modbus
8381	01	Código de función no soportado
8382	03	Error de longitud de datos
8383	02	Error en la dirección de los datos o acceso fuera de los límites del área de direcciones de MB_HOLD_REG
8384	03	Error de valor de datos
8385	03	Valor de código de diagnóstico de datos no soportado (código de función 08)

Tabla 12- 54 Códigos de condición de ejecución de MB_CLIENT ¹

STATUS (W#16#)	Errores de parámetros de MB_CLIENT
7001	MB_CLIENT está esperando a que un servidor Modbus responda a una petición de conexión o de desconexión en el puerto TCP asignado. Esto solo se notifica en la primera ejecución de una operación de conexión o de desconexión.
7002	MB_CLIENT está esperando a que un servidor Modbus responda a una petición de conexión o de desconexión en el puerto TCP asignado. Esto se notificará en las ejecuciones sucesivas, mientras se espera a que finalice una operación de conexión o de desconexión.
7003	Una operación de desconexión ha finalizado correctamente (válido sólo para un ciclo del PLC).
80C8	El servidor no ha respondido en el tiempo asignado. MB_CLIENT debe recibir una respuesta mediante la ID de transacción que se transmitió originalmente dentro del tiempo asignado o se devuelve este error. Compruebe la conexión al dispositivo servidor Modbus. Este error solo se notifica una vez se han ejecutado los intentos configurados (si procede).
8188	El valor de modo no es válido
8189	Valor de dirección de datos no válido
818A	Valor de longitud de datos no válido
818B	Puntero al área DATA_PTR no válido. Puede ser la combinación de MB_DATA_ADDRESS + MB_DATA_LEN.
818C	Puntero a un área DATA_PTR optimizada (debe ser un área de memoria M o DB estándar)
8200	El puerto está ocupado porque está procesando una petición Modbus existente.
8380	La trama Modbus recibida está corrupta o no se han recibido suficientes bytes.
8387	El parámetro de ID de conexión asignado es diferente de la ID utilizada para peticiones anteriores. Solo se puede usar una ID de conexión en cada DB de instancia de MB_CLIENT. Esto también se usa como error interno si la ID de protocolo de Modbus TCP recibida de un servidor no es 0.
8388	Un servidor Modbus ha devuelto una cantidad de datos diferente de la solicitada. Sólo se aplica a las funciones Modbus 15 y 16.

¹ Además de los errores de MB_CLIENT indicados, las instrucciones de comunicación del bloque T subyacente (TCON, TDISCON, TSEND y TRCV) pueden devolver errores.

Consulte también

TCON, TDISCON, TSEND y TRCV (Página 457)

12.5.2.2 MB_SERVER (Modbus TCP)

Tabla 12- 55 Instrucción MB_SERVER

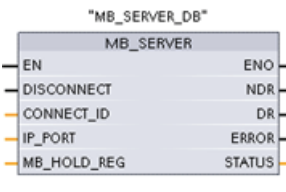
KOP / FUP	SCL	Descripción
	<pre>"MB_SERVER_DB" (DISCONNECT:=_bool_in_, CONNECT_ID:=_uint_in_, IP_PORT:=_uint_in_, NDR=>_bool_out_, DR=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_, MB_HOLD_REG:=_variant_inout_);</pre>	<p>MB_SERVER se comunica como servidor Modbus TCP a través del conector PROFINET de la CPU S7-1200. No se necesitan módulos hardware de comunicación adicionales.</p> <p>MB_SERVER puede aceptar una petición para conectarse a un cliente Modbus TCP, recibir una petición de función Modbus y enviar un mensaje de respuesta</p>

Tabla 12- 56 Tipos de datos para los parámetros

Parámetro y tipo	Tipo de datos	Descripción	
DISCONNECT	IN	Bool	MB_SERVER intenta establecer una conexión "pasiva" con un dispositivo interlocutor. Es decir, el servidor escucha pasivamente una petición de conexión TCP desde cualquier dirección IP que lo solicite. Si DISCONNECT = 0 y no existe una conexión, puede iniciarse una conexión pasiva. Si DISCONNECT = 1 y existe una conexión, se inicia una operación de desconexión. Esto permite que su programa controle cuándo se acepta una conexión. Siempre que esta entrada esté habilitada, no se intentará ninguna otra operación.
CONNECT_ID	IN	UInt	CONNECT_ID identifica unívocamente cada conexión dentro del PLC. Cada instancia única de la instrucción MB_CLIENT o MB_SERVER debe contener un parámetro CONNECT_ID único.
IP_PORT	IN	UInt	Valor predeterminado = 502: El número de puerto IP que identifica el puerto IP que se supervisará para peticiones de conexión de un cliente Modbus. Estos números de puerto TCP no se permiten en conexiones pasivas de MB_SERVER: 20, 21, 25, 80, 102, 123, 5001, 34962, 34963 y 34964.
MB_HOLD_REG	IN_OUT	Variant	Puntero al registro de retención Modbus de MB_SERVER: El registro de retención debe ser una dirección de memoria M o bien un DB global estándar. Esta área de memoria se utiliza para retener los valores a los que un cliente Modbus puede acceder con las funciones de registro Modbus 3 (lectura), 6 (escritura) y 16 (escritura).
NDR	OUT	Bool	Nuevos datos listos: 0 = No hay datos nuevos, 1= Indica que el cliente Modbus ha escrito datos nuevos
DR	OUT	Bool	Lectura de datos: 0 = No se han leído datos, 1= Indica que el cliente Modbus ha leído datos.

Parámetro y tipo		Tipo de datos	Descripción
ERROR	OUT	Bool	El bit ERROR es TRUE durante un ciclo tras haberse finalizado la ejecución de MB_SERVER con un error. El valor del código de error en el parámetro STATUS es válido solo durante el ciclo en que ERROR = TRUE.
STATUS	OUT	Word	Código de condición de ejecución

MB_SERVER permite que los códigos de funciones Modbus entrantes (1, 2, 4, 5 y 15) puedan leer y escribir bits y palabras directamente en la memoria imagen de proceso de las entradas y salidas de la CPU S7-1200. Para los códigos de función de transferencia de datos (3, 6 y 16), el parámetro MB_HOLD_REG debe definirse como un tipo de datos superior a un byte. La tabla siguiente muestra el mapeo de las direcciones Modbus en la memoria imagen de proceso de la CPU.

Tabla 12- 57 Mapeo de las direcciones Modbus en la memoria imagen de proceso

Funciones Modbus						S7-1200	
Códigos	Función	Área de datos	Rango de direcciones			Área de datos	Dirección de la CPU
01	Leer bits	Salida	1	a	8192	Memoria imagen de proceso de las salidas	Q0.0 a Q1023.7
02	Leer bits	Entrada	10001	a	18192	Memoria imagen de proceso de las entradas	I0.0 a I1023.7
04	Leer palabras	Entrada	30001	a	30512	Memoria imagen de proceso de las entradas	IW0 a IW1022
05	Escribir bit	Salida	1	a	8192	Memoria imagen de proceso de las salidas	Q0.0 a Q1023.7
15	Escribir bits	Salida	1	a	8192	Memoria imagen de proceso de las salidas	Q0.0 a Q1023.7

Los códigos de funciones de mensajes Modbus (3, 6 y 16) entrantes leen o escriben palabras en un registro de retención Modbus que puede ser un rango de direcciones de memoria M o un bloque de datos. El tipo de registro de retención se especifica con el parámetro MB_HOLD_REG.

Nota

Asignación del parámetro MB_HOLD_REG

El registro de retención Modbus puede estar en una dirección de memoria M o en un DB global estándar.

Para un registro de retención Modbus en memoria M, utilice el formato de puntero ANY estándar. Esto está en el formato P#"dirección de bit" "tipo de datos" "longitud". Por ejemplo: P#M1000.0 WORD 500.

La tabla siguiente muestra ejemplos del mapeo de las direcciones Modbus en el registro de retención usado para los códigos de función 03 (leer palabras), 06 (escribir palabras) y 16 (escribir palabras). El límite superior real de las direcciones de DB está determinado por los límites máximos de memoria de trabajo y de memoria M del respectivo modelo de CPU.

Tabla 12- 58 Ejemplos de mapeo de direcciones Modbus en direcciones de memoria de CPU

Dirección de ModBus	Ejemplos del parámetro MB_HOLD_REG		
	P#M100.0 Word 5	P#DB10.DBx0.0 Word 5	"Recipe".ingredient
40001	MW100	DB10.DBW0	"Recipe".ingredient[1]
40002	MW102	DB10.DBW2	"Recipe".ingredient[2]
40003	MW104	DB10.DBW4	"Recipe".ingredient[3]
40004	MW106	DB10.DBW6	"Recipe".ingredient[4]
40005	MW108	DB10.DBW8	"Recipe".ingredient[5]

Varias conexiones de servidor

Se pueden crear varias conexiones de servidor. Con eso, un solo PLC puede establecer conexiones concurrentes con varios clientes Modbus TCP.

Un servidor Modbus TCP puede admitir conexiones concurrentes hasta el número máximo de conexiones para Open User Communications permitidas por el PLC. El número total de conexiones en un PLC dado, clientes y servidores Modbus TCP incluidos, no debe sobrepasar el número máximo de conexiones para Open User Communications admitidas (Página 442). Las conexiones Modbus TCP pueden compartirse entre conexiones de cliente o de servidor.

Cada conexión de servidor debe cumplir estas reglas:

- Cada conexión MB_SERVER debe usar un DB de instancia distinto
- Cada conexión MB_SERVER debe establecerse con un número de puerto IP único. Sólo se admite 1 conexión por puerto.
- Cada conexión MB_SERVER debe usar una ID de conexión única.
- El MB_SERVER debe ser llamado individualmente para cada conexión (con su DB de instancia respectivo).

La ID de conexión debe ser unívoca para cada conexión individual. Es decir, sólo debe usarse una ID de conexión única con cada DB de instancia individual. Resumiendo, el DB de instancia y la ID de conexión están emparejadas y deben ser únicas para cada conexión.

Tabla 12- 59 Códigos de función de diagnóstico Modbus

Funciones de diagnóstico Modbus MB_SERVER		
Códigos	Subfunción	Descripción
08	0x0000	Devolver datos de consulta del test de eco: MB_SERVER reenvía a un cliente Modbus una palabra de datos que se ha recibido.

Funciones de diagnóstico Modbus MB_SERVER		
08	0x000A	Borrar contador de eventos de comunicación: La instrucción MB_SERVER borra el contador de eventos de comunicación utilizado para la función Modbus 11.
11		Consultar contador de eventos de comunicación: La instrucción MB_SERVER utiliza un contador de eventos de comunicación interno para registrar el número de peticiones de lectura y escritura Modbus correctas que se envían al servidor Modbus. El contador no se incrementa con ninguna de las peticiones de las funciones 8 ni 11. Tampoco se incrementa con las peticiones que producen un error de comunicación. La función Broadcast no está disponible para Modbus TCP porque sólo puede haber una conexión cliente-servidor a la vez.

Variables de MB_SERVER

Esta tabla muestra las variables estáticas públicas almacenadas en el bloque de datos instancia MB_SERVER que se pueden utilizar en el programa.

Tabla 12- 60 Variables estáticas públicas de MB_SERVER

Variable	Tipo de datos	Valor predeterminado	Descripción
HR_Start_Offset	Word	0	Asigna la dirección inicial del registro de retención Modbus.
Request_Count	Word	0	Número de todas las peticiones recibidas por este servidor.
Server_Message_Count	Word	0	Número de peticiones recibidas para este servidor específico.
Xmt_Rcv_Count	Word	0	Número de transmisiones o recepciones con error detectado. También se incrementa con cada mensaje recibido que sea un mensaje Modbus no válido.
Exception_Count	Word	0	Errores específicos Modbus que requieren una excepción devuelta
Success_Count	Word	0	Número de peticiones recibidas para este servidor específico que no tienen errores de protocolo.
Conectado	Bool	0	Indica si la conexión al cliente asignado está conectada o desconectada: 1=conectada, 0=desconectada

El programa puede escribir valores en las variables HR_Start_Offset y controlar operaciones del servidor Modbus. Las demás variables se pueden leer para vigilar el estado Modbus.

HR_Start_Offset

Las direcciones del registro de retención Modbus empiezan en 40001. Estas direcciones corresponden a la dirección de memoria del PLC inicial del registro de retención. No obstante, se puede configurar la variable "HR_Start_Offset" para que defina la dirección inicial del registro de retención Modbus en un valor diferente de 40001.

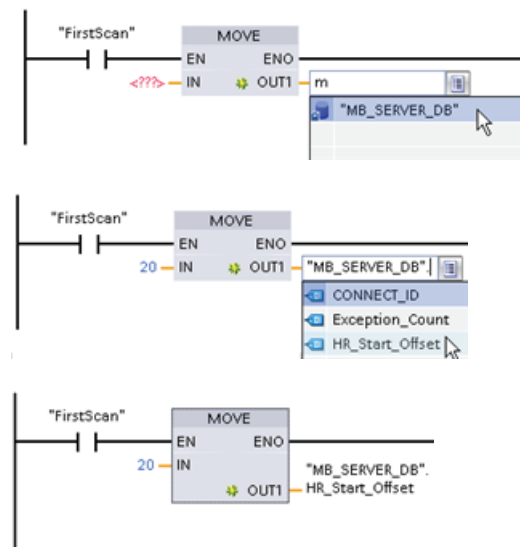
Por ejemplo, si ha configurado el registro de retención para que empiece en MW100 y su longitud es de 100 palabras. Un offset de 20 determina que la dirección inicial del registro de retención sea 40021 en vez de 40001. Cualquier dirección por debajo de 40021 y por encima de 40119 tendrá como resultado un error de direccionamiento.

Tabla 12- 61 Ejemplo de direccionamiento del registro de retención Modbus

HR_Start_Offset	Dirección	Mínimo	Máximo
0	Dirección Modbus (palabra)	40001	40099
	Dirección S7-1200	MW100	MW298
20	Dirección Modbus (palabra)	40021	40119
	Dirección S7-1200	MW100	MW298

HR_Start_Offset es un valor de palabra que especifica la dirección inicial del registro de retención Modbus y que se almacena en el bloque de datos instancia MB_SERVER. Puede ajustar el valor de esta variable estática pública utilizando la lista desplegable de asistencia de parámetros, una vez que haya insertado MB_SERVER en el programa.

Por ejemplo, una vez que haya insertado MB_SERVER en un segmento KOP, puede ir a un segmento anterior y asignar el valor HR_Start_Offset. El valor debe asignarse antes de ejecutar MB_SERVER.



Introducir una variable de servidor Modbus utilizando el nombre de DB predeterminado:

1. Posicione el cursor en el campo del parámetro y escriba un carácter m.
2. Seleccione "MB_SERVER_DB" en la lista desplegable de nombres de DB.
3. Seleccione "MB_SERVER_DB.HR_Start_Offset" en la lista desplegable de variables de DB.

Tabla 12- 62 Códigos de condición de ejecución de MB_SERVER ¹

STATUS (W#16#)	Código de respuesta al servidor Modbus (B#16#)	Errores de protocolo Modbus
7001		MB_SERVER está esperando a que un cliente Modbus se conecte al puerto TCP asignado. Se informa de este código en la primera ejecución de una operación de conexión o de desconexión.
7002		MB_SERVER está esperando a que un cliente Modbus se conecte al puerto TCP asignado. Se informará de este código en las ejecuciones sucesivas, mientras se espera la finalización de una operación de conexión o de desconexión.

STATUS (W#16#)	Código de respuesta al servidor Modbus (B#16#)	Errores de protocolo Modbus
7003		Una operación de desconexión ha finalizado correctamente (válido sólo para un ciclo del PLC).
8187		Puntero no válido a MB_HOLD_REG: Área demasiado pequeña
818C		Puntero a un área MB_HOLD_REG optimizada (debe ser un área de memoria M o DB estándar) o el temporizador de procesos bloqueados supera el límite de 55 segundos. (específico de S7-1200)
8381	01	Código de función no soportado
8382	03	Error de longitud de datos
8383	02	Error en la dirección de los datos o acceso fuera de los límites del área de direcciones de MB_HOLD_REG
8384	03	Error de valor de datos
8385	03	Valor de código de diagnóstico de datos no soportado (código de función 08)

¹ Además de los errores de MB_SERVER indicados, las instrucciones de comunicaciones del bloque T subyacente (TCON, TDISCON, TSEND y TRCV) pueden devolver errores.

Consulte también

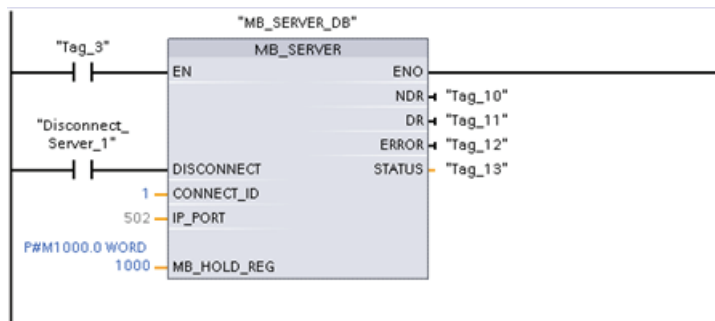
TCON, TDISCON, TSEND y TRCV (Página 457)

12.5.2.3 Ejemplo de MB_SERVER: Conexiones TCP múltiples

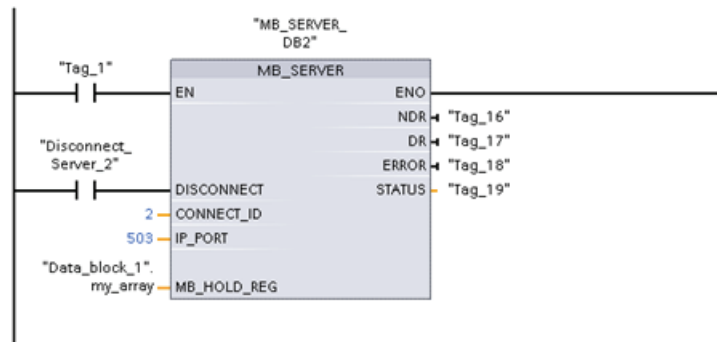
Puede haber múltiples conexiones de servidor Modbus TCP. Para conseguirlo, se debe ejecutar MB_SERVER independientemente para cada conexión. Cada conexión debe usar un DB instancia, una ID de conexión y un puerto IP independientes. El S7-1200 sólo permite una conexión por puerto IP.

Para el mejor rendimiento, se debe ejecutar MB_SERVER cada ciclo para cada conexión.

Red 1: Conexión n.º 1 con IP_PORT, ID de conexión y DB instancia independientes



Red 2: Conexión n.º 2 con IP_PORT, ID de conexión y DB instancia independientes



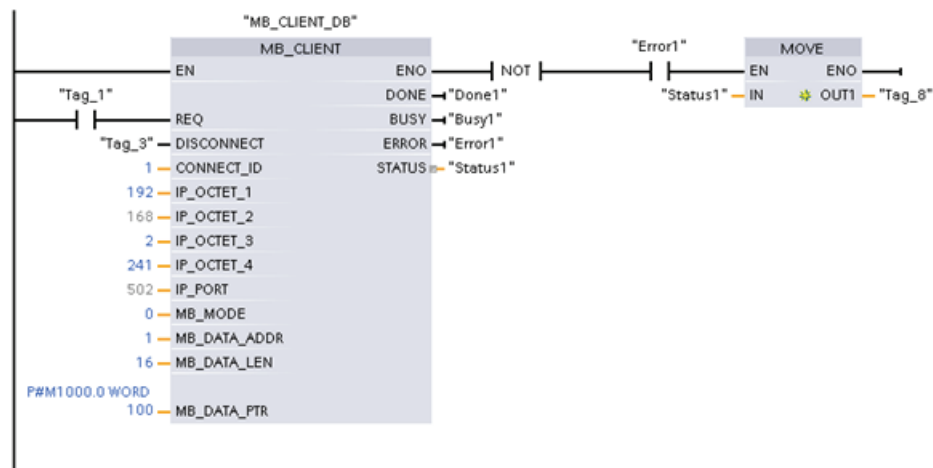
12.5.2.4 Ejemplo 1 de MB_CLIENT: Peticiones múltiples con conexión TCP común

Se pueden enviar varias peticiones de cliente Modbus por la misma conexión. Para ello, se deben utilizar los mismos DB de instancia, ID de conexión y número de puerto.

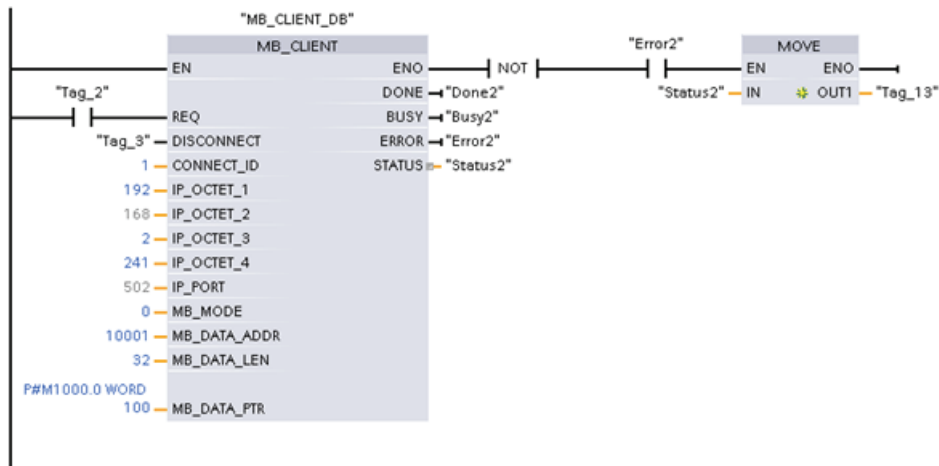
Sólo puede estar activo 1 cliente a la vez. Una vez que se termina la ejecución de un cliente, empieza la ejecución del siguiente. Su programa es responsable del orden de ejecución.

El ejemplo muestra ambos clientes escribiendo en la misma área de memoria. Además, de forma opcional, se captura un error devuelto.

Red 1: Función Modbus 1: leer 16 bits de imagen de salida



Red 2: Función Modbus 2: leer 32 bits de imagen de entrada



12.5.2.5 Ejemplo 2 de MB_CLIENT: Peticiones múltiples con varias conexiones TCP

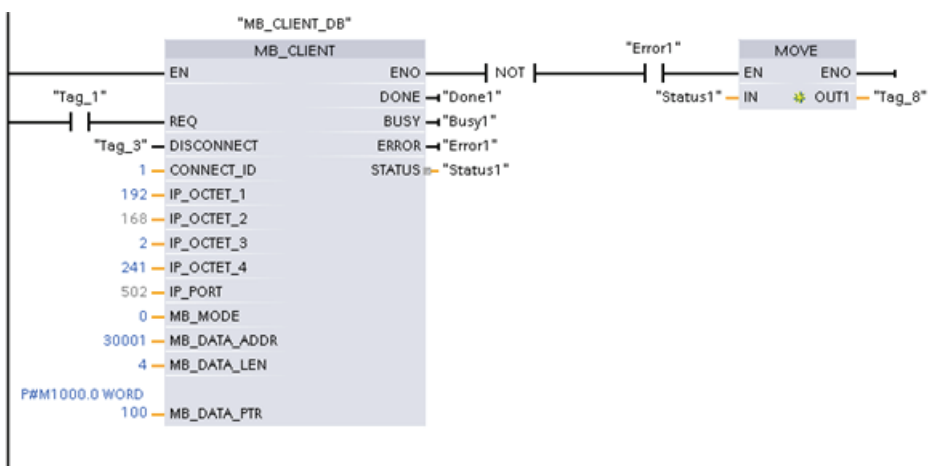
Las peticiones de cliente Modbus se pueden enviar por varias conexiones. Para ello, se deben utilizar diferentes DB de instancia, direcciones IP e ID de conexión.

El número de puerto debe ser diferente si las conexiones se establecen con el mismo servidor Modbus. Si las conexiones se realizan con servidores diferentes, no hay restricciones en número de puertos.

El ejemplo muestra ambos clientes escribiendo en la misma área de memoria. Además, de forma opcional, se captura un error devuelto.

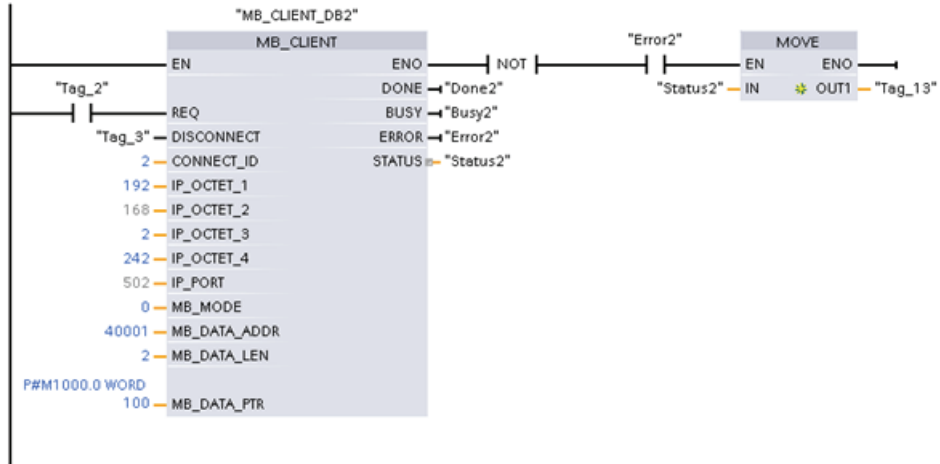
Red 1:

Función Modbus 4: leer palabras de entrada (en memoria de S7-1200)



Red 2:

Función Modbus 3: leer palabras de registro de retención (en memoria de S7-1200)

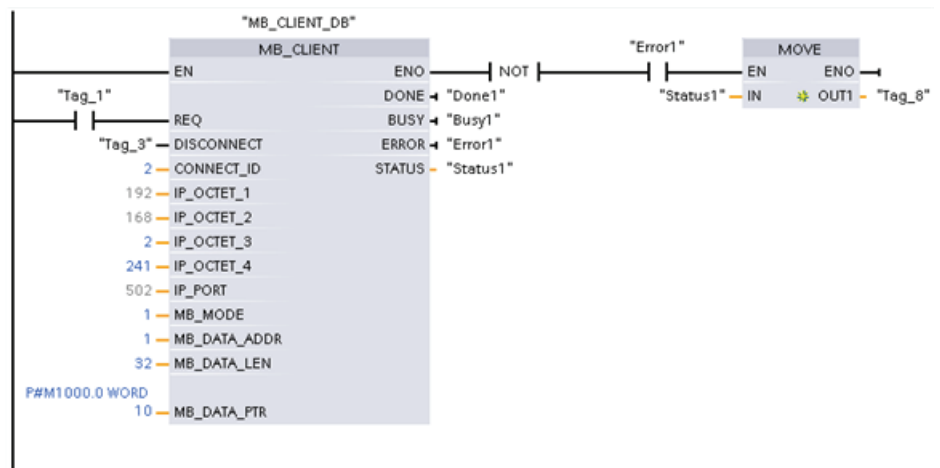


12.5.2.6

Ejemplo 3 de MB_CLIENT: Petición de escritura de imagen de salida

Este ejemplo muestra la petición de un cliente Modbus para escribir la imagen de salida de S7-1200.

Red 1: Función Modbus 15: escribir bits en la memoria imagen de las salidas del S7-1200

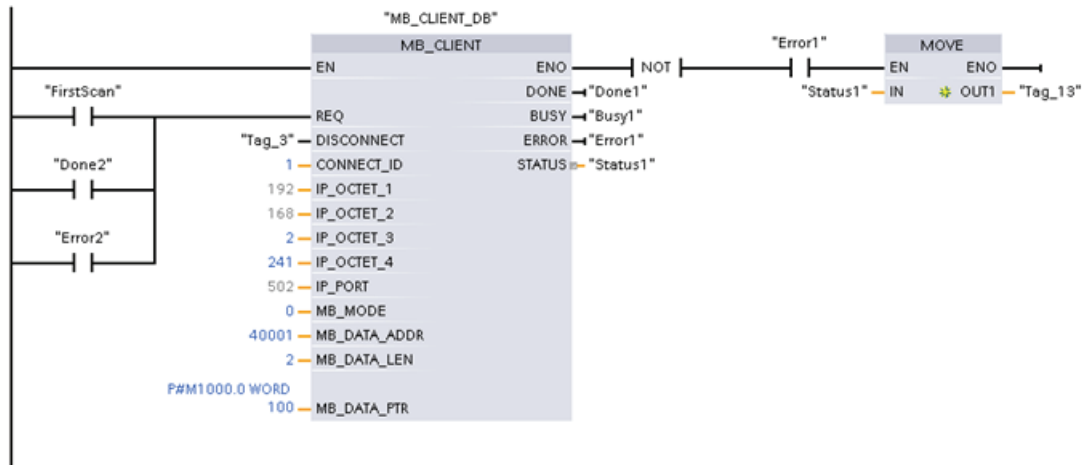


12.5.2.7 Ejemplo 4 de MB_CLIENT: Coordinación de varias peticiones

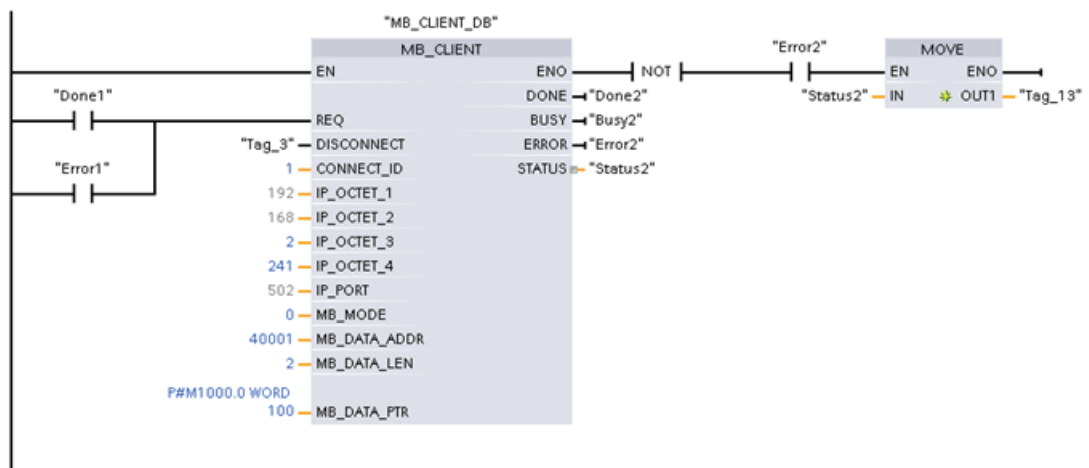
Se debe asegurar de que cada petición Modbus TCP acaba su ejecución. Su programa debe proporcionar esta coordinación. El ejemplo muestra cómo las salidas de las peticiones de cliente primera y segunda pueden usarse para coordinar la ejecución.

El ejemplo muestra ambos clientes escribiendo en la misma área de memoria. Además, de forma opcional, se captura un error devuelto.

Red 1: Función Modbus 3: leer palabras de registro de retención



Red 2: Función Modbus 3: leer palabras de registro de retención



12.5.3 Modbus RTU

En STEP 7, hay disponibles dos versiones de las instrucciones Modbus RTU:

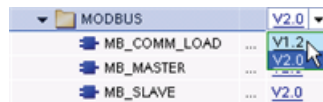
- La versión 1 estaba disponible inicialmente en STEP 7 Basic V10.5.
- La versión 2 está disponible en STEP 7 Basic/Professional V11. La configuración de la versión 2 agrega los parámetros REQ y DONE a MB_COMM_LOAD. Además, el parámetro MB_ADDR para MB_MASTER y MB_SLAVE admite ahora un valor UInt para el direccionamiento avanzado.

Para garantizar la compatibilidad y facilitar la migración, se puede elegir qué versión de la instrucción se debe insertar en el programa de usuario.

No utilice ambas versiones de la instrucción (1.x y 2.y) en el mismo programa de la CPU. Las instrucciones Modbus de su programa deben tener el mismo número de versión principal (1.x, 2.y o V.z). Las distintas instrucciones dentro de un grupo de versión principal pueden tener versiones secundarias diferentes (1.x).



Haga clic en el icono de la Task Card del árbol de instrucciones para activar los encabezados y columnas del árbol de instrucciones.



Para cambiar la versión de las instrucciones Modbus, seleccione la versión en la lista desplegable. Es posible seleccionar el grupo o bien instrucciones individuales.

Cuando se utiliza el árbol de instrucciones para insertar una instrucción Modbus en el programa, se crea una nueva instancia FB en el árbol de proyectos. El nuevo FB instancia se puede ver en el árbol de proyecto en PLC_x > Bloques de programa > Bloques de sistema > Recursos de programa.

Para comprobar la versión de la instrucción Modbus en un programa, es necesario examinar las propiedades del árbol de proyectos y no las propiedades de un cuadro que se visualiza en el editor de programas. Seleccione un FB instancia Modbus del árbol del proyecto, haga clic con el botón derecho del ratón y seleccione "Propiedades", luego seleccione la página de "información" para ver el número de versión de la instrucción Modbus.

12.5.3.1 MB_COMM_LOAD

Tabla 12- 63 Instrucción MB_COMM_LOAD

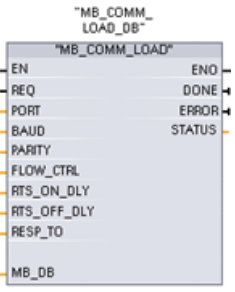
KOP / FUP	SCL	Descripción
	<pre>"MB_COMM_LOAD_DB" (REQ:=_bool_in_, PORT:=_uint_in_, BAUD:=_udint_in_, PARITY:=_uint_in_, FLOW_CTRL:=_uint_in_, RTS_ON_DLY:=_uint_in_, RTS_OFF_DLY:=_uint_in_, RESP_TO:=_uint_in_, DONE=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_, MB_DB:=_fbtref_inout_);</pre>	<p>La instrucción MB_COMM_LOAD configura un puerto PtP para la comunicación con el protocolo Modbus RTU. Opciones de hardware del puerto Modbus: Instale un máximo de tres CMs (RS485 o RS232) más una CB (R4845). Un bloque de datos instancia se asigna automáticamente cuando la instrucción MB_COMM_LOAD se inserta en el programa.</p>

Tabla 12- 64 Tipos de datos para los parámetros

Parámetro y tipo	Tipo de datos	Descripción	
REQ	IN	Bool	Un cambio de señal low a high (flanco ascendente) lanza la operación. (Sólo versión 2.0)
PORT	IN	Port	Tras haber instalado y configurado un dispositivo de comunicación CM o CB, el identificador de puerto aparece en la lista desplegable de parámetros disponible en la conexión del cuadro PUERTO. El valor de puerto CM o CB asignado es la propiedad de configuración del dispositivo "identificador de hardware". El nombre simbólico del puerto se asigna en la ficha "Constantes del sistema" de la tabla de variables PLC.
BAUD	IN	UDInt	Selección de la velocidad de transferencia: 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 76800, 115200, no se permite ningún otro valor
PARITY	IN	UInt	Selección de paridad: <ul style="list-style-type: none"> • 0 – Ninguna • 1 – Impar • 2 – Par
FLOW_CTRL	IN	UInt	Selección del control de flujo: <ul style="list-style-type: none"> • 0 – (ajuste predeterminado) sin control de flujo • 1 – Control de flujo por hardware con RTS siempre ON (no es aplicable a los puertos RS485) • 2 – Control de flujo por hardware con RTS conmutado

Parámetro y tipo		Tipo de datos	Descripción
RTS_ON_DLY	IN	UInt	Selección de retardo RTS ON: <ul style="list-style-type: none"> 0 – (ajuste predeterminado) Sin retardo desde RTS ON hasta que se transmite el primer carácter del mensaje 1 a 65535 – Retardo en milisegundos desde RTS ON hasta que se transmite el primer carácter del mensaje (no es aplicable a los puertos RS485). Los retardos RTS se aplican siempre independientemente de la selección de FLOW_CTRL.
RTS_OFF_DLY	IN	UInt	Selección de retardo RTS OFF: <ul style="list-style-type: none"> 0 – (ajuste predeterminado) Sin retardo desde el último carácter transmitido hasta que se desactiva RTS 1 a 65535 – Retardo en milisegundos desde el último carácter transmitido hasta que se desactiva RTS (no es aplicable a los puertos RS485). Los retardos RTS se aplican siempre independientemente de la selección de FLOW_CTRL.
RESP_TO	IN	UInt	Plazo de respuesta: Tiempo en milisegundos permitido por el MB_MASTER para la respuesta del esclavo. Si el esclavo no responde en este tiempo, MB_MASTER repetirá la petición o la finalizará con un error cuando se haya enviado el número de reintentos indicado. 5 ms a 65535 ms (valor predeterminado = 1000 ms).
MB_DB	IN	Variant	Referencia al bloque de datos instancia que utilizan las instrucciones MB_MASTER o MB_SLAVE. Una vez insertadas las instrucciones MB_SLAVE o MB_MASTER en el programa, el identificador de DB aparece en la lista desplegable de parámetros disponible en la conexión MB_DB del cuadro.
DONE	OUT	Bool	El bit DONE es TRUE durante un ciclo tras haberse finalizado la última petición sin errores. (sólo versión 2.0)
ERROR	OUT	Bool	El bit ERROR es TRUE durante un ciclo tras haberse finalizado la última petición con un error. El valor del código de error en el parámetro STATUS sólo es válido durante un ciclo en que ERROR = TRUE.
STATUS	OUT	Word	Código de condición de ejecución

MB_COMM_LOAD se ejecuta para configurar un puerto para el protocolo Modbus RTU. Al configurar un puerto para el protocolo Modbus RTU, el mismo sólo puede ser utilizado por la instrucción MB_MASTER o MB_SLAVE.

Es preciso utilizar una ejecución de MB_COMM_LOAD para configurar cada uno de los puertos de comunicación utilizado para la comunicación Modbus. Asígnele a cada puerto utilizado un DB de instancia MB_COMM_LOAD unívoco. En la CPU se pueden instalar hasta tres módulos de comunicación (RS232 o RS485) y una placa de comunicación (RS485). Llame MB_COMM_LOAD desde un OB de arranque y ejecute esta instrucción una vez, o bien utilice la marca de sistema del primer ciclo (Página 85) para iniciar la llamada y ejecutarla una vez. Ejecute nuevamente MB_COMM_LOAD sólo si es necesario modificar los parámetros de comunicación, p. ej. la velocidad de transferencia o la paridad.

Un bloque de datos instancia se asigna para MB_MASTER o MB_SLAVE cuando estas instrucciones se insertan en el programa. Este bloque de datos instancia se referencia cuando se especifica el parámetro MB_DB para la instrucción MB_COMM_LOAD.

Variables del bloque de datos MB_COMM_LOAD

La tabla siguiente muestra las variables estáticas públicas almacenadas en el DB de instancia MB_COMM_LOAD que se pueden utilizar en el programa.

Tabla 12- 65 Variables estáticas en el DB de instancia

Variable	Tipo de datos	Descripción
ICHAR_GAP	Word	Retardo del tiempo excedido entre caracteres. Este parámetro está especificado en milisegundos y se utiliza para aumentar el tiempo previsto entre caracteres recibidos. El número correspondiente de tiempos de bit para el parámetro se agrega al valor predeterminado Modbus de 35 tiempos de bit (3,5 tiempos de carácter).
RETRIES	Word	Número de reintentos del maestro antes de devolver el código de error "sin respuesta" 0x80C8.

Tabla 12- 66 Códigos de condición de ejecución de MB_COMM_LOAD ¹

STATUS (W#16#)	Descripción
0000	No hay error
8180	ID de puerto no válida (identificador de puerto/hardware incorrecto para el módulo de comunicación)
8181	Velocidad de transferencia no válida
8182	Paridad no válida
8183	Valor de control de flujo no válido
8184	Valor de timeout de respuesta no válido (timeout de respuesta por debajo del valor mínimo de 5 ms)
8185	El parámetro MB_DB no es un bloque de datos instancia de una instrucción MB_MASTER o MB_SLAVE.

¹ Además de los errores MB_COMM_LOAD indicados arriba, las instrucciones de comunicaciones PtP subyacentes pueden devolver errores.

Consulte también

Instrucciones de comunicación punto a punto (Página 592)

12.5.3.2 MB_MASTER

Tabla 12- 67 Instrucción MB_MASTER

KOP / FUP	SCL	Descripción
	<pre>"MB_MASTER_DB" (REQ:=_bool_in_, MB_ADDR:=_uint_in_, MODE:=_usint_in_, DATA_ADDR:=_udint_in_, DATA_LEN:=_uint_in_, DONE=>_bool_out_, BUSY=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_, DATA_PTR:=_variant_inout_);</pre>	<p>La instrucción MB_MASTER se comunica como un maestro Modbus utilizando un puerto configurado por una ejecución anterior de la instrucción MB_COMM_LOAD. Un bloque de datos instancia se asigna automáticamente cuando la instrucción MB_MASTER se inserta en el programa. El bloque de datos instancia MB_MASTER se utiliza cuando se especifica el parámetro MB_DB para la instrucción MB_COMM_LOAD.</p>

Tabla 12- 68 Tipos de datos para los parámetros

Parámetro y tipo	Tipo de datos	Descripción
REQ	IN	Bool 0 = No hay ninguna petición 1 = Petición de transmitir datos a esclavo Modbus
MB_ADDR	IN	V1.0: USInt V2.0: UInt Dirección de estación Modbus RTU: Rango de direccionamiento estándar (1 a 247) Rango de direccionamiento avanzado (1 a 65535) El valor 0 está reservado para enviar un mensaje Broadcast a todos los esclavos Modbus. Los códigos de función Modbus 05, 06, 15 y 16 son los únicos que se soportan para el Broadcast.
MODE	IN	USInt Selección de modo: Especifica el tipo de petición (lectura, escritura o diagnóstico). Consulte los detalles en la tabla de funciones Modbus que aparece más abajo.
DATA_ADDR	IN	UDInt Dirección inicial en el esclavo: Determina la dirección inicial de los datos a los que debe accederse en el esclavo Modbus. Las direcciones válidas se indican en la tabla de funciones Modbus que aparece más abajo.
DATA_LEN	IN	UInt Longitud de datos: Determina el número de bits o palabras a las que debe accederse en esta petición. Las longitudes válidas se indican en la tabla de funciones Modbus que aparece más abajo.
DATA_PTR	IN	Variant Puntero a los datos: Apunta a la dirección del M o del DB (tipo de DB estándar) de los datos que se están escribiendo o leyendo.
DONE	OUT	Bool El bit DONE es TRUE durante un ciclo tras haberse finalizado la última petición sin errores.
BUSY	OUT	Bool <ul style="list-style-type: none"> 0 – No hay ninguna operación MB_MASTER en curso 1 – Operación MB_MASTER en curso
ERROR	OUT	Bool El bit ERROR es TRUE durante un ciclo tras haberse finalizado la última petición con un error. El valor del código de error en el parámetro STATUS sólo es válido durante un ciclo en que ERROR = TRUE.
STATUS	OUT	Word Código de condición de ejecución

Reglas de comunicación del maestro Modbus

- MB_COMM_LOAD debe ejecutarse para configurar un puerto antes de que la instrucción MB_MASTER pueda comunicarse con ese puerto.
- Si un puerto debe utilizarse para iniciar peticiones de maestro Modbus, MB_SLAVE no debe utilizar este puerto. Una o más instancias de la ejecución de MB_MASTER pueden utilizarse en ese puerto, no obstante todas las ejecuciones de MB_MASTER deben utilizar el mismo DB de instancia MB_MASTER para ese puerto.
- Las instrucciones Modbus no utilizan eventos de alarma de comunicación para controlar el proceso de comunicación. El programa debe consultar la instrucción MB_MASTER para transmitir y recibir condiciones completas.
- Se recomienda llamar todas las ejecuciones de MB_MASTER para un puerto determinado desde un OB de ciclo de programa. Las instrucciones del maestro Modbus sólo se pueden ejecutar en un nivel de ejecución, ya sea en el de retardo/cíclico o bien en el de ciclo del programa. Éstas no se deben ejecutar en ambos niveles de prioridad. Si una instrucción de maestro Modbus de un nivel de prioridad de ejecución más alto inhibe la ejecución de otra instrucción de maestro Modbus, se puede obtener un funcionamiento incorrecto. Las instrucciones de maestro Modbus no deben ejecutarse en los niveles de prioridad para arranque, diagnóstico o error de tiempo.
- Cuando una instrucción de maestro inicia una transmisión, esta instancia debe ejecutarse continuamente con la entrada EN habilitada hasta que se devuelva un estado DONE=1 o un estado ERROR=1. Una instancia MB_MASTER en particular se considera activa hasta que ocurra uno de estos dos eventos. Mientras la instancia original esté activa, cualquier llamada a otra instancia con la salida REQ habilitada resultará en un error. Si la ejecución continua de la instancia original se detiene, el estado de la petición se mantiene activo por un periodo de tiempo especificado por la variable estática Blocked_Proc_Timeout. Una vez transcurre este periodo de tiempo, la siguiente instrucción de maestro llamada con una entrada REQ habilitada se convertirá en la instancia activa. Esto impide que una sola instancia de maestro Modbus monopolice o cierre el acceso a un puerto. Si la instancia original que está activa no se habilita dentro del periodo de tiempo especificado por la variable estática "Blocked_Proc_Timeout", entonces la siguiente ejecución realizada por esta instancia (con REQ sin activar) borrará el estado activo. Si REQ está activada, entonces esta ejecución inicia una nueva petición del maestro como si ninguna otra instancia estuviese activa.

Parámetro REQ

0 = No hay ninguna petición; 1 = Petición de transmitir datos a esclavo Modbus

Esta entrada se puede controlar con un contacto activado por nivel o por flanco. Siempre que esta entrada esté habilitada, se inicia un autómatas finito para garantizar que ningún otro MB_MASTER que use el mismo DB de instancia pueda emitir una petición mientras no se complete la petición actual. Todos los demás estados de entrada se capturan y se retienen internamente para la petición actual hasta que se recibe la respuesta o se detecta un error.

Si la misma instancia de MB_MASTER se ejecuta nuevamente con la entrada REQ = 1 antes de que se finalice la petición actual, no se realizarán más transmisiones. No obstante, al finalizar la petición se emite una nueva petición siempre que se vuelva a ejecutar MB_MASTER con entrada REQ = 1.

Los parámetros DATA_ADDR y MODE seleccionan el tipo de función Modbus

DATA_ADDR (dirección Modbus inicial en el esclavo): Determina la dirección inicial de los datos a los que debe accederse en el esclavo Modbus.

La instrucción MB_MASTER utiliza la entrada MODE en vez de una entrada de código de función. La combinación de MODE y la dirección Modbus determinan el código de función utilizado en el mensaje Modbus real. La tabla siguiente muestra la correlación entre el parámetro MODE, el código de función Modbus y el rango de direcciones Modbus.

Tabla 12- 69 Funciones Modbus

MODE	Función Modbus	Longitud de datos	Operación y datos	Dirección de Modbus
0	01	De 1 a 2000 De 1 a 1992 ¹	Leer bits de salida: De 1 a (1992 o 2000) bits por petición	De 1 a 9999
0	02	De 1 a 2000 De 1 a 1992 ¹	Leer bits de entrada: De 1 a (1992 o 2000) bits por petición	De 10001 a 19999
0	03	De 1 a 125 De 1 a 124 ¹	Leer registros de retención: 1 a (124 o 125) palabras por petición	De 40001 a 49999 o De 400001 a 465535
0	04	De 1 a 125 De 1 a 124 ¹	Leer palabras de entrada: De 1 a (124 o 125) palabras por petición	De 30001 a 39999
1	05	1	Escribir un bit de salida: Un bit por petición	De 1 a 9999
1	06	1	Escribir un registro de retención: 1 palabra por petición	De 40001 a 49999 o De 400001 a 465535
1	15	De 2 a 1968 De 2 a 1960 ¹	Escribir varios bits de salida: 2 a (1960 o 1968) bits por petición	De 1 a 9999
1	16	De 2 a 123 De 2 a 122 ¹	Escribir varios registros de retención: De 2 a (122 o 123) palabras por petición	De 40001 a 49999 o De 400001 a 465535
2	15	De 1 a 1968 De 2 a 1960 ¹	Escribir uno o más bits de salida: De 1 a (1960 o 1968) bits por petición	De 1 a 9999
2	16	De 1 a 123 De 1 a 122 ¹	Escribir uno o más registros de retención: De 1 a (122 o 123) palabras por petición	De 40001 a 49999 o De 400001 a 465535
11	11	0	Leer la palabra de estado de la comunicación del esclavo y el contador de eventos. Esta palabra de estado indica ocupado (0 – no ocupado, 0xFFFF - ocupado). El contador de eventos se incrementa cada vez que se finaliza un mensaje correctamente. En esta función se ignoran ambos operandos de MB_MASTER: DATA_ADDR y DATA_LEN.	
80	08	1	Compruebe el estado del esclavo utilizando el código de diagnóstico de datos 0x0000 (test de Loopback – el esclavo envía de regreso la petición) 1 palabra por petición	

MODE	Función Modbus	Longitud de datos	Operación y datos	Dirección de Modbus
81	08	1	Inicializar el contador de eventos del esclavo utilizando un código de diagnóstico de datos 0x000A 1 palabra por petición	
De 3 a 10, de 12 a 79, de 82 a 255			Reservado	

¹ Para el modo "Direccionamiento avanzado", las longitudes máximas de los datos se reduce en 1 byte o 1 palabra dependiendo del tipo de datos utilizados por la función.

Parámetro DATA_PTR

El parámetro DATA_PTR apunta a la dirección del M o del DB en que se escribe o bien del que se lee. Si se utiliza un bloque de datos, hay que crear un bloque de datos global que ofrezca un almacenamiento de datos para operaciones de lectura y escritura en esclavos Modbus.

Nota

El tipo de bloque de datos debe permitir un direccionamiento directo.

El bloque de datos debe permitir tanto el direccionamiento directo (absoluto) como el simbólico. Al crear el bloque de datos debe seleccionarse el atributo de acceso "estándar".

Estructuras del bloque de datos para el parámetro DATA_PTR

- Estos tipos de datos son válidos para la **lectura de palabras** de las direcciones Modbus de 30001 a 39999, de 40001 a 49999 y de 400001 a 465536, así como para la **escritura de palabras** en las direcciones Modbus 40001 a 49999 y 400001 a 465536.
 - Matriz estándar de tipos de datos WORD, UINT o INT
 - Estructura WORD, UINT o INT con nombres, en la que todo elemento tiene un nombre unívoco y un tipo de datos de 16 bits.
 - Estructura compleja con nombres, en la que todo elemento tiene un nombre unívoco y un tipo de datos de 16 ó 32 bits.
- Para la **lectura** y escritura de bits de las direcciones Modbus 00001 a 09999 y lecturas de bits de 10001 a 19999.
 - Matriz estándar de tipos de datos booleanos.
 - Estructura booleana con nombres que incluye variables booleanas con nombres unívocos.

- Aunque no es imprescindible, se recomienda que cada instrucción MB_MASTER tenga su propia área de memoria. El motivo de esta recomendación es que la posibilidad de que se corrompan los datos aumenta si varias instrucciones MB_MASTER están leyendo y escribiendo en la misma área de memoria.
- No es necesario que las áreas de datos de DATA_PTR se encuentren en el mismo bloque de datos global. Es posible crear un bloque de datos con varias áreas para lecturas Modbus, uno para escrituras Modbus, o bien uno para cada estación esclava.

Variables del bloque de datos del maestro Modbus

La tabla siguiente muestra las variables estáticas públicas almacenadas en el DB de instancia MB_MASTER que se pueden utilizar en el programa.

Tabla 12- 70 Variables estáticas en el DB de instancia

Variable	Tipo de datos	Valor inicial	Descripción
Blocked_Proc_Timeout	Real	3.0	Tiempo (en segundos) que hay que esperar a una instancia maestro Modbus boqueada antes de eliminar esta instancia por estar ACTIVA. Esto puede ocurrir, por ejemplo, si se ha lanzado una petición del maestro y luego el programa deja de llamar la función del maestro antes de que haya finalizado por completo la petición. El valor de tiempo debe ser mayor que 0 y menor que 55 segundos; de lo contrario se producirá un error. El valor predeterminado es 0,5 segundos.
Extended_Addressing	Bool	False	Configura el direccionamiento de esclavos de uno o dos bytes. El valor predeterminado es 0. (0=dirección de un byte, 1=dirección de dos bytes)

El programa puede escribir valores en las variables Blocked_Proc_Timeout y Extended_Addressing para controlar operaciones del maestro Modbus. Consulte en el apartado MB_SLAVE la descripción de HR_Start_Offset y Extended_Addressing para obtener un ejemplo de cómo utilizar estas variables en el editor de programas así como detalles relacionados con el direccionamiento avanzado Modbus. (Página 681)

Códigos de condición

Tabla 12- 71 Códigos de condición de ejecución MB_MASTER (errores de comunicación y configuración) ¹

STATUS (W#16#)	Descripción
0000	No hay error
80C8	Timeout del esclavo. Compruebe la velocidad de transferencia, la paridad y el cableado del esclavo.
80D1	El receptor ha lanzado una petición de control de flujo para suspender una transmisión activa y no ha habilitado nuevamente la transmisión en el tiempo de espera indicado. Este error también se genera durante el control de flujo por hardware cuando el receptor no confirma CTS en el tiempo de espera indicado.
80D2	La petición de transmisión se ha cancelado porque no se recibe ninguna señal DSR del DCE.
80E0	El mensaje se ha terminado porque el búfer de recepción está lleno.
80E1	El mensaje se ha terminado debido a un error de paridad.

STATUS (W#16#)	Descripción
80E2	El mensaje se ha terminado debido a un error de trama.
80E3	El mensaje se ha terminado debido a un error de desbordamiento.
80E4	El mensaje se ha terminado debido a que la longitud especificada excede el tamaño del búfer total.
8180	ID de puerto no válida o error en la instrucción MB_COMM_LOAD
8186	Dirección de estación Modbus no válida
8188	Modo no válido especificado para petición Broadcast
8189	Valor de dirección de datos no válido
818A	Valor de longitud de datos no válido
818B	Puntero no válido al origen/destino de datos local: tamaño incorrecto
818C	Puntero no válido para DATA_PTR o Blocked_Proc_Timeout no válido: El área de datos debe ser un DB (que permita acceso tanto simbólico como directo) o un área de marcas.
8200	El puerto está ocupado porque está procesando una petición de transmisión.

Tabla 12- 72 Códigos de condición de ejecución MB_MASTER (errores de protocolo Modbus) ¹

STATUS (W#16#)	Código de respuesta del esclavo	Errores de protocolo Modbus
8380	-	Error CRC
8381	01	Código de función no soportado
8382	03	Error de longitud de datos
8383	02	Error en la dirección de los datos o dirección fuera del rango válido del área DATA_PTR
8384	Más que 03	Error de valor de datos
8385	03	Valor de código de diagnóstico de datos no soportado (código de función 08)
8386	-	El código de función en la respuesta no coincide con el código de la petición.
8387	-	Ha respondido el esclavo incorrecto
8388	-	La respuesta del esclavo a una petición de escritura es incorrecta. La petición de escritura devuelta por el esclavo no coincide con lo que el maestro ha enviado realmente.

¹ Además de los errores MB_MASTER indicados arriba, las instrucciones de comunicaciones PtP subyacentes pueden devolver errores.

Consulte también

Instrucciones de comunicación punto a punto (Página 592)

12.5.3.3 MB_SLAVE

Tabla 12- 73 Instrucción MB_SLAVE

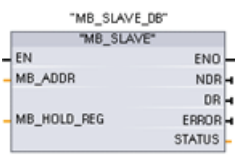
KOP / FUP	SCL	Descripción
	<pre>"MB_SLAVE_DB" (MB_ADDR:=_uint_in_, NDR=>_bool_out_, DR=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_, MB_HOLD_REG:=_variant_inout_);</pre>	<p>La instrucción MB_SLAVE permite al programa comunicarse como un esclavo Modbus a través de un puerto PtP en el CM (RS485 o RS232) y la CB (RS485). Cuando un maestro Modbus RTU remoto lanza una petición, el programa de usuario responde con la ejecución de MB_SLAVE. STEP 7 crea automáticamente un DB de instancia al introducir la instrucción. Utilice este nombre de MB_SLAVE_DB al especificar el parámetro MB_DB para la instrucción MB_COMM_LOAD.</p>

Tabla 12- 74 Tipos de datos para los parámetros

Parámetro y tipo	Tipo de datos	Descripción
MB_ADDR	IN	V1.0: USInt V2.0: UInt La dirección de estación del esclavo Modbus: Rango de direccionamiento estándar (1 a 247) Rango de direccionamiento avanzado (0 a 65535)
MB_HOLD_REG	IN	Variant Puntero hacia el DB del registro de retención Modbus: El registro de retención Modbus puede ser un área de marcas o un bloque de datos.
NDR	OUT	Bool Nuevos datos listos: <ul style="list-style-type: none"> • 0 – No hay datos nuevos • 1 – Indica que el maestro Modbus ha escrito datos nuevos
DR	OUT	Bool Lectura de datos: <ul style="list-style-type: none"> • 0 – No se han leído datos • 1 – Indica que el maestro Modbus ha leído datos
ERROR	OUT	Bool El bit ERROR es TRUE durante un ciclo tras haberse finalizado la última petición con un error. Si la ejecución ha finalizado con un error, el valor del código de error en el parámetro STATUS sólo es válido durante un ciclo en que ERROR = TRUE.
STATUS	OUT	Word Código de error de ejecución

Los códigos de las funciones de comunicación Modbus (1, 2, 4, 5 y 15) pueden leer y escribir bits y palabras directamente en la memoria imagen de proceso de las entradas y salidas de la CPU. Para estos códigos de función, el parámetro MB_HOLD_REG debe definirse como un tipo de datos superior a un byte. La tabla siguiente muestra un ejemplo del mapeo de las direcciones Modbus en la memoria imagen de proceso de la CPU.

Tabla 12- 75 Mapeo de las direcciones Modbus en la memoria imagen de proceso

Funciones Modbus						S7-1200	
Códigos	Función	Área de datos	Rango de direcciones			Área de datos	Dirección de la CPU
01	Leer bits	Salida	1	a	8192	Memoria imagen de proceso de las salidas	Q0.0 a Q1023.7
02	Leer bits	Entrada	10001	a	18192	Memoria imagen de proceso de las entradas	I0.0 a I1023.7
04	Leer palabras	Entrada	30001	a	30512	Memoria imagen de proceso de las entradas	IW0 a IW1022
05	Escribir bit	Salida	1	a	8192	Memoria imagen de proceso de las salidas	Q0.0 a Q1023.7
15	Escribir bits	Salida	1	a	8192	Memoria imagen de proceso de las salidas	Q0.0 a Q1023.7

Los códigos de las funciones de comunicación Modbus (3, 6, 16) utilizan un registro de retención Modbus que puede ser un rango de dirección de área de marcas o un bloque de datos. El tipo de registro de retención se especifica con el parámetro MB_HOLD_REG de la instrucción MB_SLAVE.

Nota

Tipo de bloque de datos MB_HOLD_REG

Un bloque de datos del registro de retención Modbus debe permitir tanto el direccionamiento directo (absoluto) como el simbólico. Al crear el bloque de datos debe seleccionarse el atributo de acceso "estándar".

La tabla siguiente muestra ejemplos del mapeo de las direcciones Modbus en el registro de retención para los códigos de función 03 (leer palabras), 06 (escribir palabras) y 16 (escribir palabras). El límite superior real de las direcciones de DB está determinado por los límites máximos de memoria de trabajo y de memoria M del respectivo modelo de CPU.

Tabla 12- 76 Mapeo de las direcciones Modbus en la memoria de la CPU

Dirección del maestro Modbus	Ejemplos del parámetro MB_HOLD_REG				
	MW100	DB10.DBW0	MW120	DB10.DBW50	"Recipe".ingredient
40001	MW100	DB10.DBW0	MW120	DB10.DBW50	"Recipe".ingredient[1]
40002	MW102	DB10.DBW2	MW122	DB10.DBW52	"Recipe".ingredient[2]
40003	MW104	DB10.DBW4	MW124	DB10.DBW54	"Recipe".ingredient[3]
40004	MW106	DB10.DBW6	MW126	DB10.DBW56	"Recipe".ingredient[4]
40005	MW108	DB10.DBW8	MW128	DB10.DBW58	"Recipe".ingredient[5]

Tabla 12- 77 Funciones de diagnóstico

Funciones de diagnóstico Modbus de MB_SLAVE en el S7-1200		
Códigos	Subfunción	Descripción
08	0000H	Devolver datos de consulta del test de eco: La instrucción MB_SLAVE responde al maestro Modbus con una palabra de datos que se están recibiendo datos.
08	000AH	Borrar contador de eventos de comunicación: La instrucción MB_SLAVE borra el contador de eventos de comunicación utilizado para la función Modbus 11.
11		Consultar contador de eventos de comunicación: La instrucción MB_SLAVE utiliza un contador de eventos de comunicación interno para registrar el número de peticiones de lectura y escritura Modbus correctas que se envían al esclavo Modbus. El contador no se incrementa con las funciones 8 ni 11, ni tampoco con peticiones Broadcast. Tampoco se incrementa con peticiones que resulten en un error de comunicación (p. ej. errores de paridad o CRC).

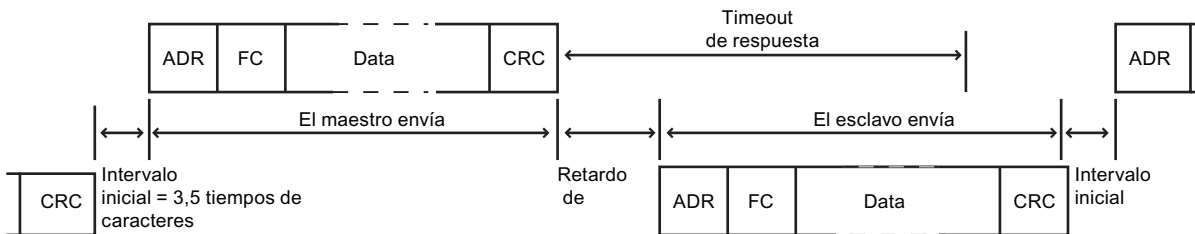
La instrucción MB_SLAVE soporta peticiones de escritura Broadcast de cualquier maestro Modbus, mientras que la petición sea para acceder a direcciones válidas. MB_SLAVE generará el código de error 0x8188 para códigos de función no soportados en Broadcast.

Reglas de comunicación del esclavo Modbus

- MB_COMM_LOAD debe ejecutarse para configurar un puerto antes de que la instrucción MB_SLAVE pueda comunicarse a través de ese puerto.
- Si un puerto debe responder como esclavo a un maestro Modbus, no programe este puerto con la instrucción MB_MASTER.
- Sólo se puede utilizar una instancia MB_SLAVE en un determinado puerto, de lo contrario puede presentarse un comportamiento erróneo.
- Las instrucciones Modbus no utilizan eventos de alarma de comunicación para controlar el proceso de comunicación. El programa debe controlar el proceso de comunicación consultando la instrucción MB_SLAVE para comprobar si se han finalizado las operaciones de transmisión y recepción.
- La instrucción MB_SLAVE debe ejecutarse periódicamente a una frecuencia que permita responder sin demora a las peticiones entrantes de un maestro Modbus. Se recomienda ejecutar MB_SLAVE en cada ciclo desde un OB de ciclo de programa. Es posible ejecutar MB_SLAVE desde un OB de alarma cíclica, pero no es recomendable debido a que el potencial de retardos excesivos en el subprograma bloquea temporalmente la ejecución de otros subprogramas.

Temporización de señales Modbus

MB_SLAVE debe ejecutarse periódicamente para recibir todas las peticiones del maestro Modbus y responder según sea necesario. La frecuencia de ejecución de MB_SLAVE depende del periodo de timeout de respuesta del maestro Modbus. Esto se ilustra en el diagrama siguiente.



El periodo de timeout de respuesta RESP_TO es el tiempo que un maestro Modbus espera hasta el inicio de la respuesta de un esclavo Modbus. Este periodo no está definido en el protocolo Modbus, sino que es un parámetro de todo maestro Modbus. La frecuencia de ejecución (es decir, el tiempo que transcurre entre dos ejecuciones) de MB_SLAVE debe basarse en los parámetros particulares del maestro Modbus. Como mínimo, MB_SLAVE debería ejecutarse dos veces en el periodo de timeout de respuesta del maestro Modbus.

Variables del esclavo Modbus

Esta tabla muestra las variables estáticas públicas almacenadas en el bloque de datos instancia MB_SLAVE que se pueden utilizar en el programa.

Tabla 12- 78 Variables del esclavo Modbus

Variable	Tipo de datos	Descripción
HR_Start_Offset	Word	Especifica la dirección inicial del registro de retención Modbus (valor predeterminado = 0)
Extended Addressing	Bool	Configura el direccionamiento de esclavos de uno o dos bytes (0=dirección de un byte, 1=dirección de dos bytes, valor predeterminado = 0)
Request_Count	Word	Número de todas las peticiones recibidas por este esclavo
Slave_Message_Count	Word	Número de peticiones recibidas para este esclavo específico
Bad_CRC_Count	Word	Número de peticiones recibidas que tienen un error CRC
Broadcast_Count	Word	Número de peticiones Broadcast recibidas
Exception_Count	Word	Errores específicos Modbus que requieren una excepción devuelta
Success_Count	Word	Número de peticiones recibidas para este esclavo específico que no tienen errores de protocolo

El programa puede escribir valores en las variables HR_Start_Offset y Extended Addressing y controlar operaciones del esclavo Modbus. Las demás variables se pueden leer para vigilar el estado Modbus.

HR_Start_Offset

Las direcciones del registro de retención Modbus comienzan en 40001 ó 400001. Estas direcciones se corresponden con la dirección inicial de memoria del PLC para el registro de retención. No obstante, es posible configurar la variable "HR_Start_Offset" para que defina la dirección inicial del registro de retención Modbus en otro valor diferente a 40001 ó 400001.

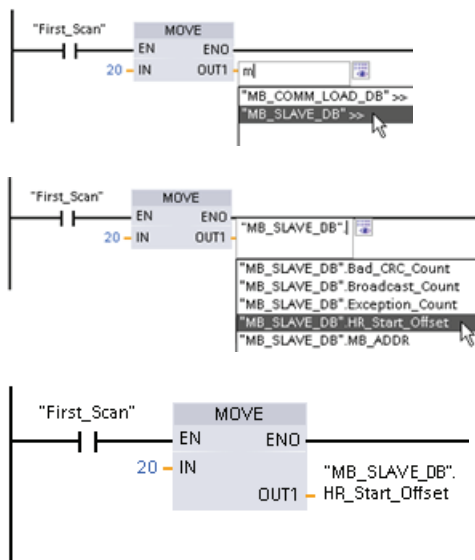
Por ejemplo, si ha configurado el registro de retención para que empiece en MW100 y su longitud es de 100 palabras. Un offset de 20 determina que la dirección inicial del registro de retención sea 40021 en vez de 40001. Cualquier dirección por debajo de 40021 y por encima de 400119 tendrá como resultado un error de direccionamiento.

Tabla 12- 79 Ejemplo de direccionamiento del registro de retención Modbus

HR_Start_Offset	Dirección	Mínimo	Máximo
0	Dirección Modbus (palabra)	40001	40099
	Dirección S7-1200	MW100	MW298
20	Dirección Modbus (palabra)	40021	40119
	Dirección S7-1200	MW100	MW298

HR_Start_Offset es un valor de palabra que especifica la dirección inicial del registro de retención Modbus y que se almacena en el bloque de datos instancia MB_SLAVE. Es posible ajustar el valor de esta variable estática pública utilizando la lista desplegable de parámetros, una vez haya insertado MB_SLAVE en el programa.

Por ejemplo, una vez haya insertado MB_SLAVE en un segmento KOP, es posible ir a un segmento anterior y asignar el valor HR_Start_Offset. El valor debe asignarse antes de ejecutar MB_SLAVE.



Introducir una variable de esclavo Modbus utilizando el nombre de DB predeterminado:

1. Posicione el cursor en el campo del parámetro y escriba un carácter m.
2. Seleccione "MB_SLAVE_DB" en la lista desplegable.
3. Posicione el cursor en el lado derecho del nombre del DB (detrás de las comillas) e introduzca un punto.
4. Seleccione "MB_SLAVE_DB.HR_Start_Offset" en la lista desplegable.

Extended Addressing

A la variable Extended Addressing se accede de una manera similar que a la variable HR_Start_Offset mencionada más arriba, excepto que la variable Extended Addressing es un valor booleano. El valor booleano se debe escribir a través de una bobina de salida y no de un cuadro de desplazamiento.

El direccionamiento del esclavo Modbus se puede configurar de modo que tenga un solo byte (ajuste predeterminado) o bien dos bytes. El direccionamiento avanzado se utiliza para direccionar más de 247 dispositivos dentro de una sola red. El direccionamiento avanzado permite direccionar como máximo 64000 direcciones. A continuación aparece una trama de la función 1 Modbus a modo de ejemplo.

Tabla 12- 80 Dirección de esclavo de un byte (byte 0)

Función 1	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	
Petición	Direc. esclava	Código F	Dirección inicial		Long. de bobinas		
Respuesta válida	Direc. esclava	Código F	Longitud	Datos bobina			
Respuesta errónea	Direc. esclava	0x81	Código E				

Tabla 12- 81 Dirección de esclavo de dos bytes (byte 0 y byte 1)

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Petición	Dirección esclava		Código F	Dirección inicial		Long. de bobinas	
Respuesta válida	Dirección esclava		Código F	Longitud	Datos bobina		
Respuesta errónea	Dirección esclava		0x81	Código E			

Códigos de condición

Tabla 12- 82 Códigos de condición de ejecución MB_SLAVE (errores de comunicación y configuración) ¹

STATUS (W#16#)	Descripción
80D1	El receptor ha lanzado una petición de control de flujo para suspender una transmisión activa y no ha habilitado nuevamente la transmisión en el tiempo de espera indicado. Este error también se genera durante el control de flujo por hardware cuando el receptor no confirma CTS en el tiempo de espera indicado.
80D2	La petición de transmisión se ha cancelado porque no se recibe ninguna señal DSR del DCE.
80E0	El mensaje se ha terminado porque el búfer de recepción está lleno.
80E1	El mensaje se ha terminado debido a un error de paridad.
80E2	El mensaje se ha terminado debido a un error de trama.
80E3	El mensaje se ha terminado debido a un error de desbordamiento.

STATUS (W#16#)	Descripción
80E4	El mensaje se ha terminado debido a que la longitud especificada excede el tamaño del búfer total.
8180	ID de puerto no válida o error en la instrucción MB_COMM_LOAD
8186	Dirección de estación Modbus no válida
8187	Puntero no válido a MB_HOLD_REG DB: Área demasiado pequeña
818C	Puntero MB_HOLD_REG no válido al área de marcas o al DB (el área DB debe permitir un direccionamiento tanto simbólico como directo)

Tabla 12- 83 Códigos de condición de ejecución MB_SLAVE (errores de protocolo Modbus) ¹

STATUS (W#16#)	Código de respuesta del esclavo	Errores de protocolo Modbus
8380	Sin respuesta	Error CRC
8381	01	Código de función no soportado o no soportado en Broadcast
8382	03	Error de longitud de datos
8383	02	Error en la dirección de los datos o dirección fuera del rango válido del área DATA_PTR
8384	03	Error de valor de datos
8385	03	Valor de código de diagnóstico de datos no soportado (código de función 08)

¹ Además de los errores MB_SLAVE indicados arriba, las instrucciones de comunicaciones PtP subyacentes pueden devolver errores.

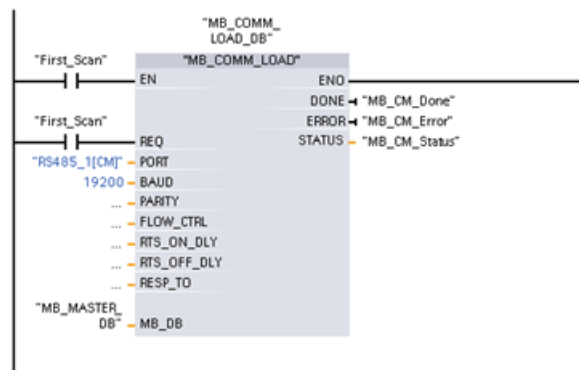
Consulte también

Instrucciones de comunicación punto a punto (Página 592)

12.5.3.4 Ejemplo de programa de maestro Modbus RTU

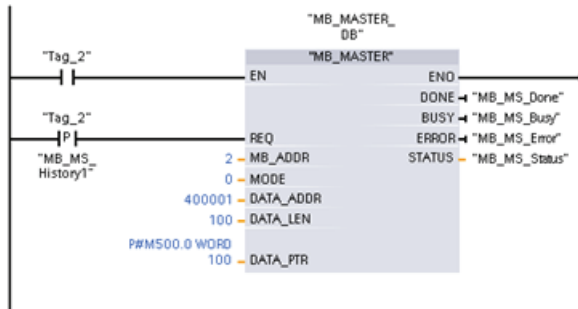
MB_COMM_LOAD se inicializa durante el arranque mediante la primera marca de exploración. La ejecución de MB_COMM_LOAD de esa forma sólo debe hacerse cuando la configuración de puerto serie no vaya a cambiar en tiempo de ejecución.

Red 1 Inicializa los parámetros del módulo RS485 una sola vez durante el primer ciclo.

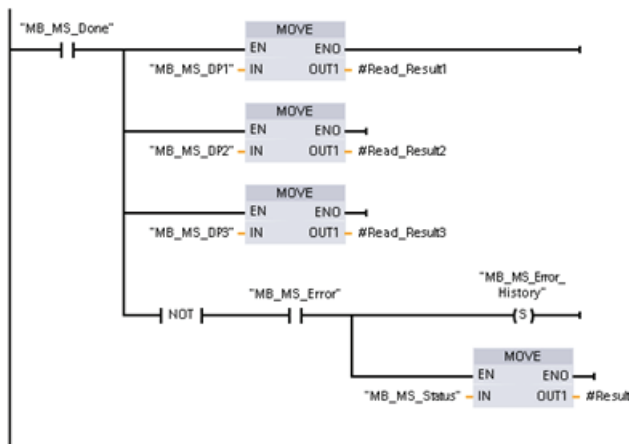


Una instrucción MB_MASTER se usa en el OB del ciclo de programa para comunicar con un solo esclavo. Se pueden usar más instrucciones MB_MASTER en el OB del ciclo de programa para comunicar con otros esclavos o se puede reutilizar un FB MB_MASTER para comunicar con más esclavos.

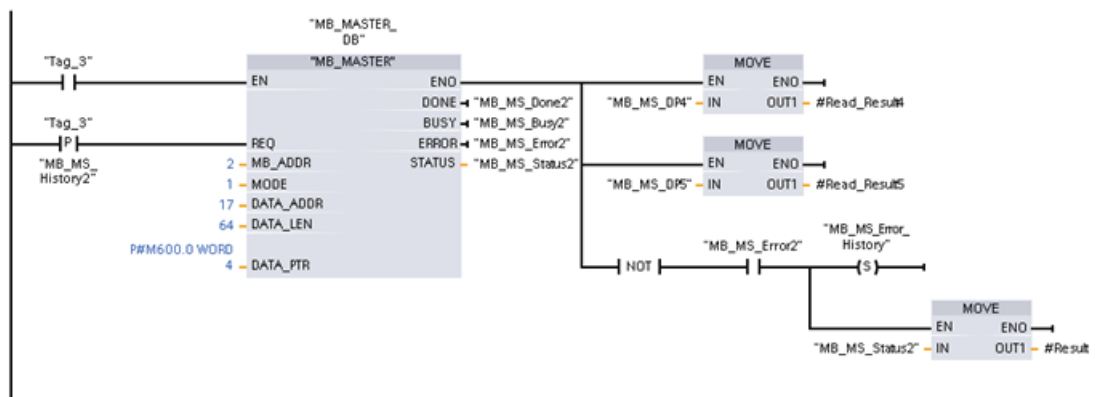
Red 2 Lee 100 palabras del registro de retención del esclavo.



Red 3 Esta es una red opcional que sólo muestra los valores de las primeras 3 palabras una vez ha finalizado la operación de lectura.



Red 4 Escribe 64 bits en el registro de imagen de proceso de las salidas comenzando por la dirección de esclavo Q2.0.

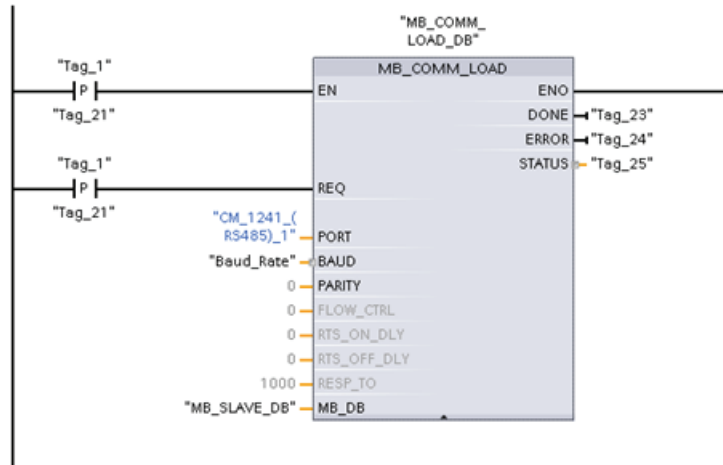


12.5.3.5 Ejemplo de programa de esclavo Modbus RTU

El MB_COMM_LOAD mostrado a continuación se inicializa cada vez que se habilita "Tag_1".

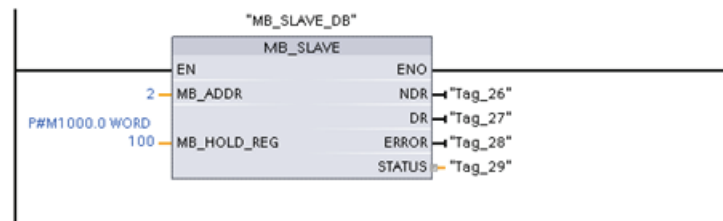
La ejecución de MB_COMM_LOAD de esa forma sólo debe hacerse cuando la configuración de puerto serie cambie en tiempo de ejecución, a consecuencia de la configuración de HMI.

Red 1 Inicializa los parámetros del módulo RS485 cada vez que un dispositivo HMI los cambia.



El MB_SLAVE mostrado a continuación se coloca en un OB cíclico que se ejecuta cada 10 ms. Con esto no se consigue que el esclavo proporcione la respuesta más rápida posible, pero proporciona buen rendimiento a 9600 baudios para mensajes cortos (20 bytes o menos en la petición).

Red 2 Comprueba las peticiones del maestro Modbus durante cada ciclo. El registro de retención Modbus está configurado para 100 palabras comenzando en MW1000.



12.6 Telecontrol y TeleService con el CP 1242-7

12.6.1 Conexión a una red GSM

Comunicación WAN basada en IP a través de GPRS

Con ayuda del procesador de comunicación CP 1242-7 se puede conectar la S7-1200 a redes GSM. El CP 1242-7 hace posible la comunicación vía WAN de estaciones remotas con una central así como la comunicación cruzada entre estaciones.

La comunicación cruzada entre estaciones sólo es posible a través de la red GSM. Para la comunicación de una estación remota con un puesto de control central se tiene que disponer de un PC con conexión a Internet en la central.

El CP 1242-7 da soporte a los siguientes servicios para la comunicación a través de la red GSM:

- GPRS (General Packet Radio Service)

El servicio de transmisión de datos orientado a paquetes "GPRS" se desarrolla a través de la red GSM.

- SMS (Short Message Service)

El CP 1242-7 puede recibir y enviar mensajes en forma de SMS. El interlocutor de comunicación puede ser un teléfono móvil o una S7-1200.

El CP 1242-7 es apropiado para el uso industrial en todo el mundo y es compatible con las siguientes bandas de frecuencia:

- 850 MHz
- 900 MHz
- 1 800 MHz
- 1 900 MHz

Requisitos:

El equipamiento de las estaciones o de la central depende de la respectiva aplicación.

- Para la comunicación con o a través de un puesto de control central se necesita en la central un PC con conexión a Internet.
- Para una estación remota S7-1200 con CP 1242-7 que deba utilizar la comunicación a través de la red GSM, además de disponer del equipamiento propio de la estación se necesita lo siguiente:

- Un contrato con un proveedor de red GSM apropiado

Si se debe trabajar con GPRS es necesario que el contrato permita el uso del servicio GPRS.

En caso de comunicación directa entre estaciones sólo a través de la red GSM, el proveedor de la red GSM debe asignar una dirección IP fija a los CPs. En tal caso, la comunicación entre estaciones no tiene lugar a través de la central.

- La tarjeta SIM perteneciente al contrato

La tarje SIM se inserta en el CP 1242-7.

- Disponibilidad local de una red GSM en el ámbito de la estación

12.6.2 Aplicaciones del CP 1242-7

Para el CP 1242-7 son posibles los siguientes casos de aplicación:

Aplicaciones de Telecontrol

- Envío de mensajes vía SMS

La CPU de una estación S7-1200 remota recibe mensajes SMS de la red GSM a través del CP 1242-7, o bien envía mensajes vía SMS a un teléfono móvil configurado o a un S7-1200.

- Comunicación con una central de supervisión

Las estaciones S7-1200 remotas se comunican con un servidor de Telecontrol de la central a través de la red GSM y de Internet. La aplicación "TELECONTROL SERVER BASIC" está instalada en el servidor de Telecontrol de la central para la transferencia de datos a través de GPRS. Este servidor de Telecontrol se comunica con un sistema central de nivel superior mediante la función de servidor OPC integrada.

- Comunicación entre las estaciones S7-1200 mediante una red GSM

La comunicación entre estaciones remotas con CP 1242-7 se puede desarrollar de dos modos distintos:

- Comunicación cruzada mediante una central

En esta configuración se establece una conexión segura y permanente entre las estaciones S7-1200 que se comunican entre sí y el servidor de Telecontrol de la central. La comunicación entre las estaciones tiene lugar siempre a través del servidor de Telecontrol. El CP 1242-7 trabaja en el modo "Telecontrol".

- Comunicación directa entre las estaciones

Para la comunicación directa entre las estaciones sin necesidad de pasar por una central se utilizan tarjetas SIM con dirección IP fija, que permiten direccionar las estaciones directamente. Los servicios de comunicación y las funciones de seguridad posibles (p. ej. VPN) dependen de la oferta del proveedor de la red. El CP 1242-7 funciona en el modo de operación "GPRS directo".

TeleService vía GPRS

Entre una estación de ingeniería con STEP 7 y una estación S7-1200 remota con un CP 1242-7 se puede establecer una conexión de TeleService a través de la red GSM y de Internet. La conexión tiene lugar desde la estación de ingeniería a través de un servidor de Telecontrol o una gateway de TeleService que reenvía los telegramas como intermediario y realiza la autorización. Estos PC utilizan las funciones de la aplicación "TELECONTROL SERVER BASIC".

La conexión de TeleService se puede utilizar para los siguientes fines:

- Carga de datos de configuración y de programa en la estación desde el proyecto STEP 7
- Consulta de datos de diagnóstico tomados de la estación

12.6.3 Otras propiedades del CP

Otros servicios y funciones del CP 1242-7

- Sincronización horaria del CP vía Internet

La hora del CP puede ajustarla de la siguiente forma:

- En el modo de operación "Telecontrol" la hora se transfiere desde el servidor de Telecontrol. El CP ajusta así su hora.
- En el modo "GPRS directo" el CP puede solicitar la hora a través de SNTP.

Para la sincronización de la hora de la CPU puede leer la hora actual del CP con ayuda de un bloque.

- Almacenamiento temporal de los telegramas a enviar en caso de problemas de conexión
- Mayor disponibilidad gracias a la posibilidad de conexión a un servidor sustitutivo de Telecontrol
- Volumen de datos optimizado (conexión temporal)

Como alternativa a una conexión permanente al servidor de Telecontrol, el CP se puede configurar en STEP 7 con una conexión temporal al servidor de Telecontrol. En este caso sólo se establece una conexión con el servidor de Telecontrol en caso de necesidad.

- Documentación del volumen de datos

Los volúmenes de datos transmitidos se documentan y se pueden evaluar con otros fines.

Configuración y sustitución de módulos

Para la configuración del módulo es preciso utilizar la siguiente herramienta de configuración:

STEP 7 versión V11.0 SP1 o superior

Para STEP 7 V11.0 SP1 se requiere además el Support Package "CP 1242-7" (HSP0003001).

Para la transmisión de datos de proceso vía GPRS, utilice las indicaciones de comunicación de Telecontrol en el programa de usuario de la estación.

Los datos de configuración del CP 1242-7 se almacenan en la respectiva CPU local. Gracias a esto, en caso de recambio se puede sustituir fácilmente el CP.

Por cada estación S7-1200 se pueden enchufar hasta tres módulos del tipo CP 1242-7. Ello permite establecer, por ejemplo, rutas de comunicación redundantes.

Conexiones eléctricas

- Alimentación eléctrica del CP 1242-7

El CP posee una conexión propia para la alimentación eléctrica externa con 24 V DC.

- Interfaz de radiofrecuencia para la red GSM

Para la comunicación vía GSM se necesita una antena externa. Ésta se conecta a través de la conexión hembra SMA del CP.

Información complementaria

El manual del CP 1242-7 contiene información detallada. Lo encontrará en Internet, en las páginas de Siemens Industrial Automation Customer Support, con el siguiente ID de referencia:

42330276 (<http://support.automation.siemens.com/WW/view/de/42330276>)

12.6.4 Accesorios

La antena de GSM/GPRS ANT794-4MR

Para el uso en redes GSM/GPRS están disponibles las siguientes antenas para el montaje en interiores y exteriores:

- Antena cuatribanda ANT794-4MR



Figura 12-1 Antena de GSM/GPRS ANT794-4MR

Denominación breve	Referencia	Explicación
ANT794-4MR	6NH9 860-1AA00	Antena cuatribanda (900, 1800/1900 MHz, UMTS); resistente a la intemperie, para interior y exterior; cable de conexión de 5 m unido fijo a la antena; conector SMA; incl. escuadra de montaje, tornillos y tacos

- Antena plana ANT794-3M

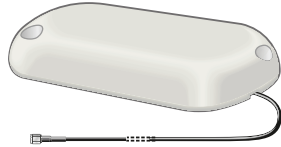


Figura 12-2 Antena plana ANT794-3M

Denominación breve	Referencia	Explicación
ANT794-3M	6NH9 870-1AA00	Antena plana (900, 1800/1900 MHz); resistente a la intemperie, para interior y exterior; cable de conexión de 1,2 m unido fijo a la antena; conector SMA; incl. almohadilla adhesiva, posible fijación con tornillos

Las antenas se tienen que pedir aparte.

Información complementaria

El manual del equipo contiene información detallada. Encontrará el manual en Internet, en las páginas de Siemens Industrial Automation Customer Support, con el siguiente ID de referencia:

23119005 (<http://support.automation.siemens.com/WW/view/es/23119005>)

12.6.5 Ejemplos de configuración para Telecontrol

A continuación encontrará algunos ejemplos de configuración para estaciones con CP 1242-7.

Envío de SMS

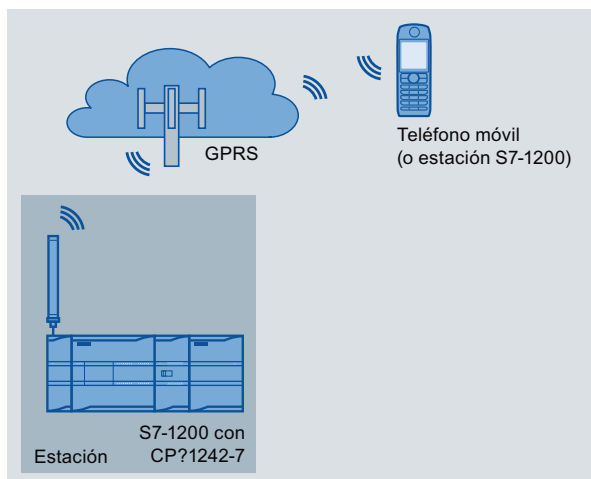


Figura 12-3 Envío de SMS de una estación S7-1200

Una estación SIMATIC S7-1200 con CP 1242-7 puede enviar mensajes vía SMS a un teléfono móvil o a una estación S7-1200 configurada.

Telecontrol a través de una central

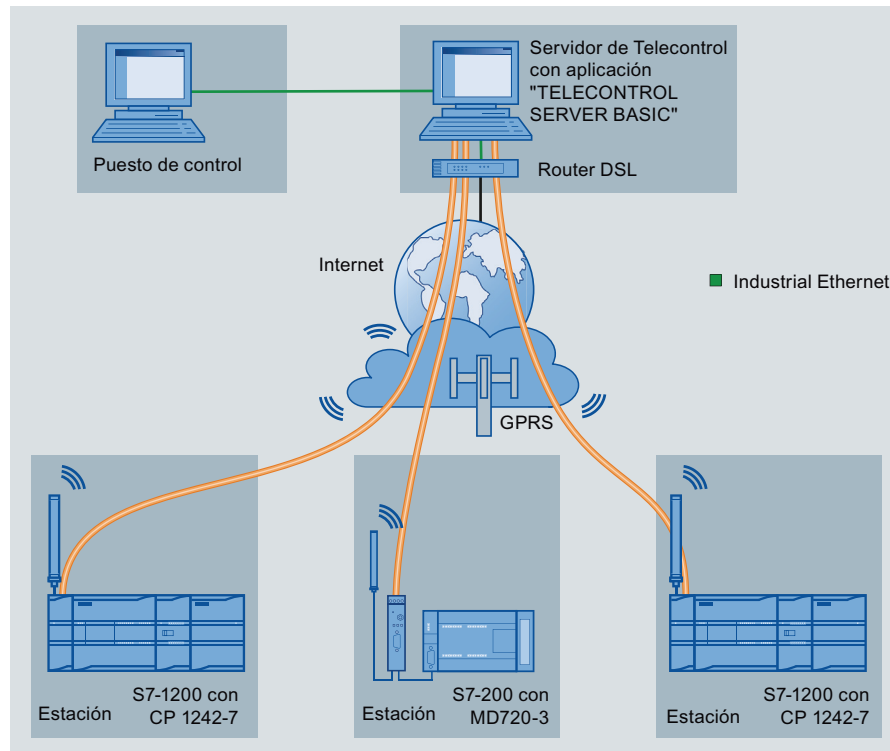


Figura 12-4 Comunicación de estaciones S7-1200 con una central

En el caso de las aplicaciones de Telecontrol, las estaciones SIMATIC S7-1200 con CP 1242-7 se comunican con una central a través de la red GSM y de Internet. El servidor de Telecontrol de la central tiene instalada la aplicación "TELECONTROL SERVER BASIC" (TCSB). De esto resultan los siguientes casos de aplicación:

- Comunicación de Telecontrol entre estación y central

En este caso, los datos se envían desde el campo de las estaciones al servidor de Telecontrol de la central a través de la red GSM y de Internet. El servidor de Telecontrol sirve para supervisar las estaciones remotas.

- Comunicación entre una estación y una central supervisora con cliente OPC

Como en el primer caso, las estaciones se comunican con el servidor de Telecontrol. Con ayuda del servidor OPC integrado, el servidor de Telecontrol intercambia datos con el cliente OPC de la central supervisora.

El cliente OPC y el servidor de Telecontrol pueden estar instalados en el mismo PC, p. ej. si TCSB se instala en el PC de un puesto de control con WinCC.

- Comunicación cruzada entre estaciones a través de una central

Es posible la comunicación cruzada con estaciones S7 que también están equipadas con un CP 1242-7.

Para la comunicación cruzada entre estaciones, el servidor de Telecontrol transmite los telegramas de la estación emisora a la estación receptora.

Comunicación directa entre estaciones

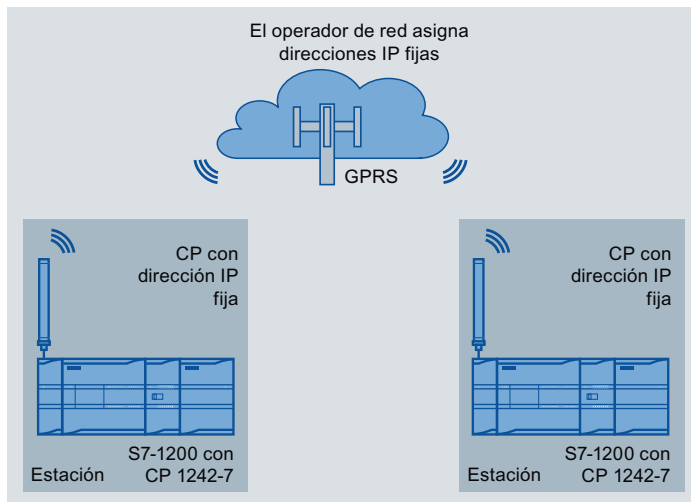


Figura 12-5 Comunicación directa de dos estaciones S7-1200

En esta configuración se comunican directamente entre sí dos estaciones SIMATIC S7-1200 a través de la red GSM con ayuda del CP 1242-7. Cada CP 1242-7 tiene una dirección IP fija. El servicio correspondiente del proveedor de red GSM debe permitirlo.

TeleService vía GPRS

En el caso de TeleService a través de GPRS, una estación de ingeniería, en la que está instalado STEP 7, se comunica con el CP 1242-7 de la estación S7-1200 a través de la red GSM y de Internet.

Dado que por norma general los cortafuegos están cerrados a solicitudes de conexión externas, es necesaria una estación intermediaria entre la estación remota y la de ingeniería. Esta estación intermediaria puede ser un servidor de Telecontrol o, si en la configuración no hay ningún servidor de Telecontrol, una gateway de TeleService.

TeleService con el servidor de Telecontrol

La conexión se desarrolla a través del servidor de Telecontrol.

- La estación de ingeniería y el servidor de Telecontrol están conectados a través de Intranet (LAN) o Internet.
- El servidor de Telecontrol y la estación remota están conectadas a través de Internet y de la red GSM.

La estación de ingeniería y el servidor de Telecontrol también pueden ser el mismo PC, de modo que STEP 7 y TCSB estarán instalados en el mismo equipo.

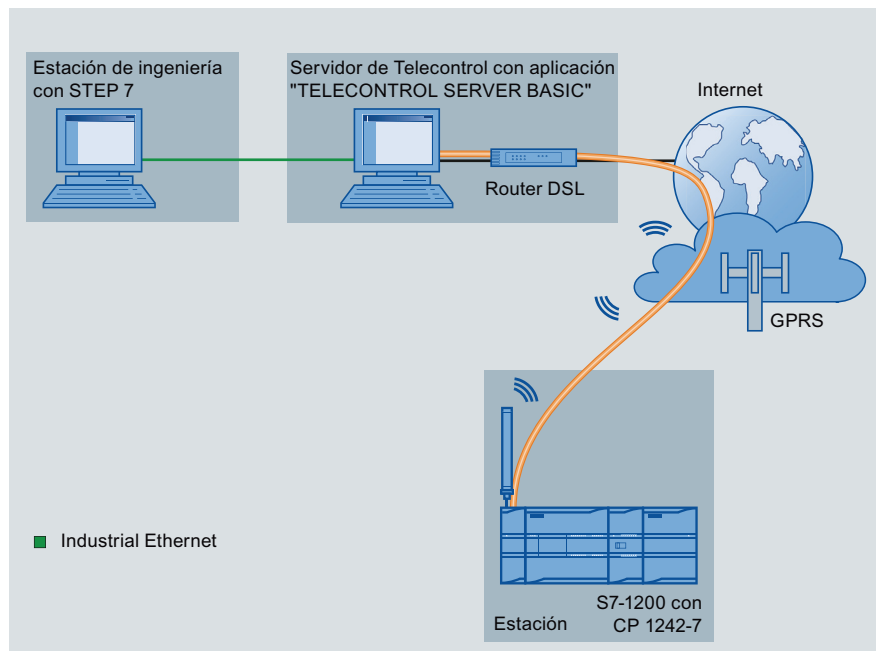


Figura 12-6 TeleService a través de GPRS en una configuración con servidor de Telecontrol

TeleService sin servidor de Telecontrol

La conexión se realiza a través de la gateway de TeleService.

La conexión entre la estación de Engineering y la gateway de TeleService puede desarrollarse de forma local a través de la LAN o a través de Internet.

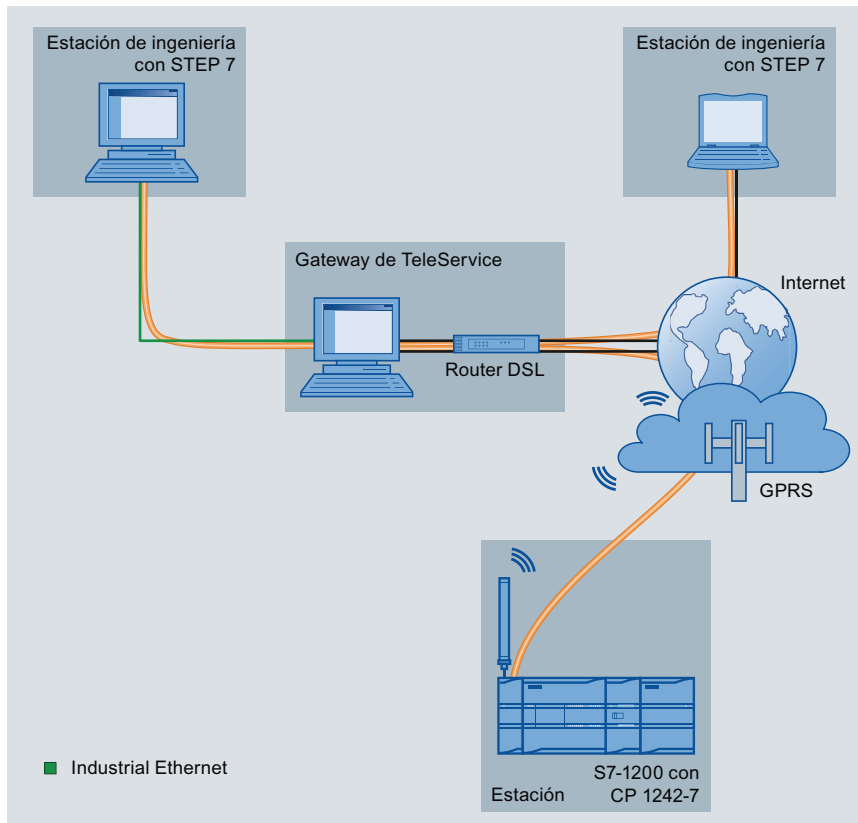


Figura 12-7 TeleService a través de GPRS en una configuración con gateway de TeleService

13.1 Instrucción de transferencia de correos electrónicos TM_Mail

Tabla 13- 1 Instrucción TM_MAIL

KOP / FUP	SCL	Descripción
<p>"TM_MAIL_DB"</p>	<pre>"TM_MAIL_DB" (REQ:=_bool_in_, ID:=_int_in_, TO_S:=_string_in_, CC:=_string_in_, SUBJECT:=_string_in_, TEXT:=_string_in_, ATTACHMENT:=_variant_in_, BUSY=>_bool_out_, DONE=>_bool_out_, ERROR=>_bool_out_, STATUS=>_word_out_,);</pre>	<p>La instrucción TM_MAIL envía un mensaje de correo electrónico mediante SMTP (Simple Mail Transfer Protocol) sobre TCP/IP a través de la conexión Industrial Ethernet de la CPU. Si no se dispone de conectividad Ethernet con Internet, se puede usar un adaptador Teleservice opcional para conectarse con la red telefónica conmutada. TM_MAIL se ejecuta asincrónicamente y la tarea abarca varias llamadas de TM_MAIL. Al llamar a TM_MAIL, se debe asignar un DB de instancia. No es necesario ajustar el atributo remanente del DB de instancia. Esto garantiza que el DB de instancia sea inicializado en la transición de la CPU de STOP a RUN y que se puede disparar una nueva operación TM_MAIL.</p>

1 STEP 7 crea el DB de instancia automáticamente al insertar la instrucción.

El envío de un correo electrónico se inicia con un flanco ascendente, de 0 a 1, en el parámetro de entrada REQ. La tabla siguiente muestra la relación entre los parámetros BUSY, DONE y ERROR. Es posible vigilar el progreso de ejecución de TM_MAIL y detectar que la misma haya finalizado evaluando estos parámetros en llamadas consecutivas.

Los parámetros de salida DONE, ERROR, STATUS, y SFC_STATUS sólo son válidos durante un ciclo, cuando el estado del parámetro de salida BUSY cambia de 1 a 0. La lógica del programa debe guardar temporalmente los valores de estado de la salida, de modo que sea posible detectar cambios de estado en posteriores ciclos de ejecución.

Tabla 13- 2 Interacción de los parámetros Done, Busy y Error

DONE	BUSY	ERROR	Descripción
Irrelevante	1	Irrelevante	Tarea en curso.
1	0	0	La tarea se ha ejecutado correctamente.
0	0	1	La tarea se ha finalizado con un error. Para averiguar la causa del error, consulte el parámetro STATUS.
0	0	0	Ninguna tarea en curso

13.1 Instrucción de transferencia de correos electrónicos TM_Mail

Si la CPU pasa a STOP mientras está activa la instrucción TM_MAIL, se cierra la comunicación con el servidor de correo electrónico. La comunicación con el servidor de correo electrónico también se pierde si se producen problemas de comunicación con la CPU en el bus Industrial Ethernet. En estos casos, se suspende el proceso de transmisión y el correo electrónico no llega al destinatario.

PRECAUCIÓN

Modificar programas de usuario

Modifique las partes del programa de usuario que afecten directamente las llamadas TM_MAIL únicamente en los siguientes casos:

- La CPU está en STOP
- No se envió ningún correo electrónico (REQ y BUSY = 0)

Esto rige especialmente en caso de borrar y reemplazar bloques de programa, las llamadas de TM_MAIL o llamadas de los DB de instancia de TM_MAIL.

Si no logra mantener conectados los bloques de programa, las funciones de comunicación TPC/IP pueden entrar en un estado indefinido. Después de transferir un bloque de programa modificado, se debe reiniciar la CPU (caliente) o ejecutar un arranque en frío.

Coherencia de datos

El parámetro de entrada ADDR_MAIL_SERVER se lee una vez iniciada la operación. Un valor nuevo no se hará efectivo hasta que la operación actual haya finalizado y se haya iniciado una nueva operación TM_MAIL.

Por el contrario, los parámetros WATCH_DOG_TIME, TO_S, CC, FROM, SUBJECT, TEXT, ATTACHMENT, USERNAME y PASSWORD se leen durante la ejecución de TM_MAIL y sólo se pueden modificar cuando la tarea ha finalizado (BUSY = 0).

Conexión telefónica: Configurar los parámetros de IE del adaptador de TS

Es necesario configurar los parámetros de IE del adaptador de TS para llamadas salientes con el fin de conectarse con el servidor de acceso telefónico de su proveedor de servicios de Internet. Si activa el atributo "sobre petición" para la llamada, la conexión sólo se establecerá cuando se vaya a transmitir un correo electrónico. Para una conexión analógica vía módem se requiere más tiempo para el proceso de conexión (aprox. un minuto más). Hay que incluir el tiempo extra en el valor de WATCH_DOG_TIME.

Tabla 13- 3 Tipos de datos para los parámetros

Parámetro y tipo		Tipos de datos	Descripción
REQ	IN	Bool	Un cambio de señal low a high (flanco ascendente) lanza la operación.
ID	IN	Int	Identificador de conexión: Véase el parámetro ID de las instrucciones TCON, TDISCON, TSEND y TRCV. Se debe usar un número que no se utiliza para instancias adicionales de esta instrucción en el programa de usuario.

13.1 Instrucción de transferencia de correos electrónicos TM_Mail

Parámetro y tipo		Tipos de datos	Descripción
TO_S	IN	String	Direcciones de destinatarios: datos STRING con una longitud máxima de 240 caracteres.
CC	IN	String	Copia CC para direcciones de destinatarios (opcional): datos STRING con una longitud máxima de 240 caracteres.
SUBJECT	IN	String	Asunto del correo electrónico: datos STRING con una longitud máxima de 240 caracteres.
TEXT	IN	String	Mensaje de texto del correo electrónico (opcional): datos STRING con una longitud máxima de 240 caracteres. Si este parámetro es una cadena vacía, el correo electrónico se enviará sin texto de mensaje.
ATTACHMENT	IN	Variant	Puntero hacia los datos adjuntos del correo electrónico: datos byte, palabra o palabra doble con una longitud máxima de 65534 bytes. Si no se asigna ningún valor, el correo electrónico se enviará sin datos adjuntos.
DONE	OUT	Bool	<ul style="list-style-type: none"> • 0 - Tarea no iniciada aún o en ejecución. • 1 - La tarea se ha ejecutado sin errores.
BUSY	OUT	Bool	<ul style="list-style-type: none"> • 0 - Ninguna operación en curso • 1- Operación en curso
ERROR	OUT	Bool	El bit ERROR es = 1 durante un ciclo tras haberse finalizado la última petición con un error. El valor del código de error en la salida STATUS sólo es válido durante un ciclo en que ERROR = 1.
STATUS	OUT	Word	Valor de retorno o información de error de la instrucción TM_MAIL.
ADDR_MAIL_SERVER	¹ Static	DWord	Dirección IP del servidor de correo electrónico: Se debe asignar cada fragmento de la dirección IP en forma de octeto de dos caracteres hexadecimales de 4 bits. Si el fragmento de la dirección IP = valor decimal 10, que equivale al valor hexadecimal A, se debe introducir "0A" para ese octeto. Ejemplo: Dirección IP = 192.168.0.10 ADDR_MAIL_SERVER = DW#16#C0A8000A, donde: <ul style="list-style-type: none"> • 192 = 16#C0, • 168 =16#A8 • 0 = 16#00 • 10 = 16#0A
WATCH_DOG_TIME	¹ Static	Time	Tiempo máximo permitido de que dispone TM_MAIL para establecer una conexión con un servidor. Si se excede este tiempo, la ejecución de TM_MAIL finalizará con un error. El retardo real hasta que finalice TM_MAIL y se emita el error puede exceder el WATCH_DOG_TIME debido a que la operación de desconexión requiere tiempo adicional. Inicialmente debe ajustarse un tiempo de 2 minutos. Este tiempo puede ser mucho más corto para una conexión telefónica ISDN.

13.1 Instrucción de transferencia de correos electrónicos TM_Mail

Parámetro y tipo		Tipos de datos	Descripción
USERNAME	¹ Static	String	Nombre de usuario de la cuenta de correo: datos STRING con una longitud máxima de 180 caracteres.
PASSWORD	¹ Static	String	Contraseña del servidor de correo: datos STRING con una longitud máxima de 180 caracteres.
FROM	¹ Static	String	Dirección del emisor: STRING con una longitud máxima de 240 caracteres.
SFC_STATUS	¹ Static	Word	Código de condición de ejecución de los bloques de comunicación invocados

¹ Los valores de esos parámetros no se modifican en cada llamada de TM_MAIL. Los valores se asignan en el bloque de datos de instancia de TM_MAIL y sólo se referencian una vez, en la primera llamada a TM_MAIL.

Autenticación SMTP

TM_MAIL soporta el método de autenticación SMTP AUTH LOGIN. Para más información sobre ese método de autenticación, consulte el manual del servidor de correo o la página web de su proveedor de servicios de Internet.

El método de autenticación AUTH LOGIN utiliza los parámetros de TM_MAIL USERNAME y PASSWORD para conectarse con el servidor de correo. El nombre de usuario y la contraseña deben estar previamente configurados en una cuenta de correo electrónico de un servidor de correo electrónico.

Si no se asigna ningún valor al parámetro USERNAME, el método de autenticación AUTH LOGIN no se usa y el correo electrónico se envía sin autenticación.

Parámetros TO_S, CC y FROM :

Los parámetros TO_S:, CC: y FROM: son cadenas, como se indica en los ejemplos siguientes:

TO_S: <wenna@mydomain.com>, <ruby@mydomain.com>,

CC: <admin@mydomain.com>, <judy@mydomain.com>,

FROM: <admin@mydomain.com>

Las siguientes reglas deben usarse al introducir esas cadenas de caracteres:

- Deben introducirse los caracteres "TO_S:", "CC:" y "FROM:", incluido el carácter de dos puntos.
- Antes de cada dirección debe introducirse un espacio y un corchete angular de apertura "<". Por ejemplo, debe haber un carácter de espacio entre "TO_S:" y <dirección de correo electrónico>.
- Después de cada dirección debe introducirse un corchete angular de cierre ">".
- Para las direcciones de TO_S: y CC: debe introducirse una coma "," después de cada dirección. Por ejemplo, "TO_S: <email address>," necesita una coma tras la única dirección de correo electrónico.
- Sólo se puede utilizar una dirección de correo electrónico para la entrada FROM:, sin coma al final.

Debido al modo run-time y al uso de memoria, no se ejecuta una prueba de sintaxis para los parámetros de TM_MAIL TO_S:, CC: y FROM:. Si no se observan estrictamente las reglas de formato anteriores, la transacción del servidor de correo electrónico SMTP fallará.

Parámetros STATUS y SFC_STATUS

Los códigos de condición de ejecución que devuelve TM_MAIL se pueden clasificar del siguiente modo:

- W#16#0000: operación de TM_MAIL finalizada correctamente
- W#16#7xxx: estado de la operación de TM_MAIL
- W#16#8xxx: error en una llamada interna de un dispositivo de comunicación o del servidor de correo electrónico

La tabla siguiente muestra los códigos de condición de ejecución de TM_MAIL con excepción de los códigos de error de módulos de comunicación llamados internamente.

Nota

Requisitos del servidor de correo electrónico

TM_MAIL sólo puede comunicarse con un servidor de correo electrónico utilizando SMTP mediante el puerto 25. El número de puerto asignado no se puede cambiar.

La mayoría de departamentos de IT y servidores de correo electrónico externos bloquean el puerto 25 para evitar que un PC infectado con un virus se convierta en un generador de correos electrónicos no autorizado.

Se puede conectar con un servidor de correo interno mediante SMTP y hacer que el servidor interno gestione las mejoras de seguridad actuales necesarias para reenviar correos electrónicos a través de Internet a un servidor de correo externo.

Ejemplo de configuración de servidor de correo electrónico interno

Si se usa Microsoft Exchange como servidor de correo interno, se puede configurar el servidor para que permita acceso SMTP desde la dirección IP asignada al PLC S7-1200. Configure la consola de administración de Exchange: Configuración del servidor > Transporte de concentradores. > Conectores de recepción > Relé. En la pestaña Red, hay un cuadro denominado "Recibir correo desde servidores remotos que tengan estas direcciones IP". Aquí se debe poner la dirección IP del PLC que esté ejecutando la instrucción TM_MAIL. No se requiere autenticación para este tipo de conexión con un servidor de Microsoft Exchange interno.

13.1 Instrucción de transferencia de correos electrónicos TM_Mail

Configuración del servidor de correo electrónico

TM_MAIL sólo puede usar un servidor de correo electrónico que permita la comunicación por el puerto 25, SMTP y autenticación AUTH LOGIN (opcional).

Configure una cuenta de servidor de correo electrónico compatible para que acepte el inicio de sesión SMTP remoto. Edite el DB de instancia para TM_MAIL para poner las cadenas de caracteres de TM_MAIL USERNAME y PASSWORD que se usan para autenticar la conexión con su cuenta de correo electrónico.

Tabla 13- 4 Códigos de condición

STATUS (W#16#...):	SFC_STATUS (W#16#...):	Descripción
0000	-	La operación de TM_MAIL ha finalizado sin errores. Este código cero de STATUS no garantiza que un correo electrónico realmente haya sido enviado (consulte el primer punto de la nota que sigue a esta tabla).
7001	-	TM_MAIL está activa (BUSY = 1).
7002	7002	TM_MAIL está activa (BUSY = 1).
8xxx	xxxx	La operación de TM_MAIL ha finalizado con un error en las llamadas internas de la instrucción de comunicación. Para más información sobre el parámetro SFC_STATUS, consulte la descripción del parámetro STATUS de las instrucciones de comunicación de usuario abierta PROFINET subyacente.
8010	xxxx	Fallo de conexión: para más información sobre el parámetro SFC_STATUS, consulte el parámetro STATUS de la instrucción TCON.
8011	xxxx	Error al transmitir datos: para más información sobre el parámetro SFC_STATUS, consulte el parámetro STATUS de la instrucción TSEND.
8012	xxxx	Error al recibir datos: para más información sobre el parámetro SFC_STATUS, consulte las descripciones del parámetro STATUS de la instrucción TRCV.
8013	xxxx	Fallo de conexión: para más información sobre cómo evaluar el parámetro SFC_STATUS, consulte las descripciones del parámetro STATUS de las instrucciones TCON y TDISCON.
8014	-	Fallo de conexión: es posible que haya introducido una dirección IP incorrecta para el servidor de correo electrónico (ADDR_MAIL_SERVER) o bien un tiempo muy corto (WATCH_DOG_TIME) para la conexión. Asimismo, es posible que la CPU no tenga conexión con la red o que su configuración sea incorrecta.
82xx, 84xx, 85xx	-	El mensaje de error viene del servidor de correo electrónico y corresponde al número de error "8" del protocolo SMTP. Consulte el segundo punto de la nota que sigue a esta tabla.
8450	-	La operación no se ejecuta: El buzón de correo no está disponible; inténtelo más tarde.
8451	-	Operación interrumpida: error local de procesamiento; inténtelo más tarde.
8500	-	Error de sintaxis en el comando: la causa puede ser que el servidor de correo electrónico no soporte el proceso de autenticación de LOGIN. Compruebe los parámetros de TM_MAIL. Intente enviar un correo electrónico sin autenticación. Intente reemplazar el parámetro USERNAME por una cadena vacía.
8501	-	Error de sintaxis: Parámetro o argumento incorrecto; es posible que haya tecleado una dirección incorrecta en los parámetros TO_S o CC.
8502	-	Comando desconocido o no implementado: verifique las entradas, especialmente el parámetro FROM. Posiblemente esté incompleto y usted haya omitido los caracteres "@" o ".".

STATUS (W#16#...):	SFC_STATUS (W#16#...):	Descripción
8535	-	La autenticación SMTP está incompleta. Es posible que haya introducido un nombre de usuario o una contraseña incorrecta.
8550	-	No es posible acceder al servidor de correo o bien el usuario no dispone de los derechos necesarios. Es posible que haya introducido un nombre de usuario o una contraseña incorrecta o su servidor de correo electrónico no soporta un acceso mediante inicio de sesión. Otra causa de este error puede ser una entrada errónea del nombre de dominio después del carácter "@" en los parámetros TO_S o CC.
8552	-	Operación interrumpida: se ha excedido el tamaño de la memoria asignada; inténtelo más tarde.
8554	-	Fallo de transmisión: inténtelo más tarde.

Nota**Posibles errores de transmisión de correo electrónico no reportados**

- Una entrada incorrecta de una dirección de destinatario no genera un error STATUS para TM_MAIL. En este caso, no hay garantía de que destinatarios adicionales (con direcciones de correo electrónico correctas) recibirán el correo electrónico.
- Encontrará más información sobre los códigos de error SMTP en Internet o en la documentación de errores del servidor de correo electrónico. También es posible leer el último mensaje de error del servidor de correo electrónico. El mensaje de error se almacena en el búfer1 del parámetro del DB de instancia para TM_MAIL.

Herramientas online y diagnóstico

14.1 LEDs de estado

La CPU y los módulos de E/S utilizan LEDs para indicar el estado operativo del módulo o de las E/S.

LEDs de estado en la CPU

La CPU incorpora los siguientes indicadores de estado:

- STOP/RUN
 - Luz amarilla permanente indica el estado operativo STOP
 - Luz verde permanente indica el estado operativo RUN
 - Si parpadea (alternando entre verde y amarillo), indica que la CPU está en estado operativo ARRANQUE
- ERROR
 - Luz roja intermitente indica un error, p. ej. un error interno de la CPU, de la Memory Card o un error de configuración (los módulos no se corresponden)
 - Luz roja permanente indica que hay un fallo de hardware
- El LED MAINT (mantenimiento) parpadea cuando se inserta una Memory Card. La CPU pasa entonces a estado operativo STOP. Tras cambiar la CPU a estado operativo STOP, realice una de las funciones siguientes para iniciar la evaluación de la Memory Card:
 - Cambiar la CPU a estado operativo RUN
 - Realizar un borrado total (MRES)
 - Desconectar y volver a conectar la alimentación de la CPU

También se puede utilizar la instrucción LED (Página 312) para determinar el estado de los LEDs.

Tabla 14- 1 LEDs de estado de la CPU

Descripción	STOP/RUN Amarillo/verde	ERROR Rojo	MAINT Amarillo
Alimentación desconectada	Off	Off	Off
Arranque, autotest o actualización de firmware	Parpadeo (alternando entre amarillo y verde)	-	Off
Estado operativo STOP	On (amarillo)	-	-
Estado operativo RUN	On (verde)	-	-
Extracción de la Memory Card	On (amarillo)	-	Parpadeo
Error	On (amarillo o verde)	Parpadeo	-
Mantenimiento solicitado	On (amarillo o verde)	-	On

14.1 LEDs de estado

Descripción	STOP/RUN Amarillo/verde	ERROR Rojo	MAINT Amarillo
Hardware averiado	On (amarillo)	On	Off
Test de LEDs o firmware de la CPU defectuoso	Parpadeo (alternando entre amarillo y verde)	Parpadeo	Parpadeo

La CPU incorpora asimismo dos LEDs que indican el estado de la comunicación PROFINET. Abra la tapa del bloque de terminales inferior para ver los LEDs PROFINET.

- Link (verde) se enciende para indicar una conexión correcta
- Rx/Tx (amarillo) se enciende para indicar la actividad de transmisión

La CPU y todos los módulos de señales (SM) digitales incorporan un LED I/O Channel para cada una de las entradas y salidas digitales. El LED I/O Channel (verde) se enciende o apaga para indicar el estado de la entrada o salida en cuestión.

LEDs de estado en el SM

Además, todo SM digital incorpora un LED DIAG que indica el estado del módulo:

- Verde indica que el módulo está operativo
- Rojo indica que el módulo está averiado o no operativo

Todo SM analógico incorpora un LED I/O Channel para cada una de las entradas y salidas analógicas.

- Verde indica que el canal se ha configurado y está activo
- Rojo indica una condición de error de la entrada o salida analógica en cuestión

Además, todo SM analógico incorpora un LED DIAG que indica el estado del módulo:

- Verde indica que el módulo está operativo
- Rojo indica que el módulo está averiado o no operativo

El SM detecta la presencia o ausencia de alimentación del módulo (alimentación de campo en caso necesario).

Tabla 14- 2 LEDs de estado de un módulo de señales (SM)

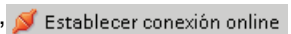
Descripción	DIAG (Rojo/verde)	I/O Channel (Rojo/verde)
Alimentación de campo desconectada	Rojo intermitente	Rojo intermitente
No se ha configurado o se está actualizando	Verde intermitente	Off
Módulo configurado sin errores	On (verde)	On (verde)
Condición de error	Rojo intermitente	-
Error de E/S (con diagnóstico habilitado)	-	Rojo intermitente
Error de E/S (con diagnóstico inhibido)	-	On (verde)

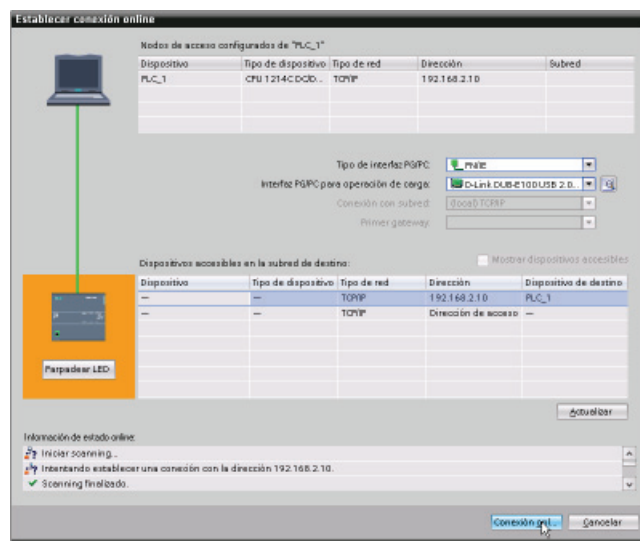
14.2 Establecer una conexión online con una CPU

Es necesaria una conexión online entre la programadora y la CPU para cargar programas y datos de ingeniería del proyecto, así como para las actividades siguientes:

- Comprobar programas de usuario
- Visualizar y cambiar el modo de operación de la CPU (Página 715)
- Visualizar y ajustar la fecha y hora de la CPU (Página 714)
- Visualizar la información del módulo
- Comparar y sincronizar (Página 718) bloques de programa de offline a online
- Cargar y descargar bloques de programa
- Mostrar diagnóstico y el búfer de diagnóstico (Página 717)
- Usar una tabla de observación (Página 722) para probar el programa de usuario vigilando y modificando valores
- Utilizar una tabla de forzado permanente para forzar valores en la CPU (Página 725)

Para establecer una conexión online en una CPU configurada, haga clic en la CPU en el árbol de navegación del proyecto y haga clic en el botón "Establecer conexión online" en la vista de proyectos:





Si es la primera vez que se realiza una conexión online con esta CPU, hay que seleccionar el tipo de interfaz PG/PC y la interfaz específica PG/PC en el cuadro de diálogo Establecer conexión online antes de establecer una conexión online a una CPU detectada en dicha interfaz.

La programadora se conecta a la CPU. Los marcos de color naranja indican una conexión online. Ahora, se pueden usar las herramientas online y de diagnóstico del árbol de proyectos, así como la Task Card de las herramientas online.

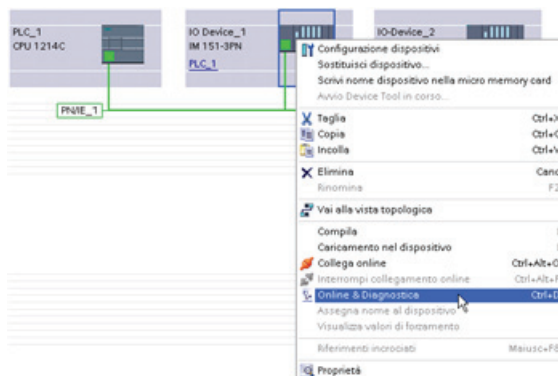
14.3 Asignar un nombre a un dispositivo PROFINET IO online

Los dispositivos de la red PROFINET deben tener asignados nombres antes de poder ser conectados a la CPU. Utilice el editor "Dispositivos y redes" para asignar nombres a los dispositivos PROFINET, en caso de que éstos no hayan sido asignados o bien si el nombre del dispositivo debe ser modificado.

A cada dispositivo PROFINET IO se le debe asignar el mismo nombre en el proyecto de STEP 7 y, utilizando la herramienta "Online y diagnóstico", en la memoria de configuración del dispositivo PROFINET IO (p. ej. en la memoria de configuración de un módulo de interfaz ET200 S). Si hace falta un nombre o éste no coincide en ninguna ubicación, el modo de intercambio de datos PROFINET IO no se ejecutará.

14.3 Asignar un nombre a un dispositivo PROFINET IO online

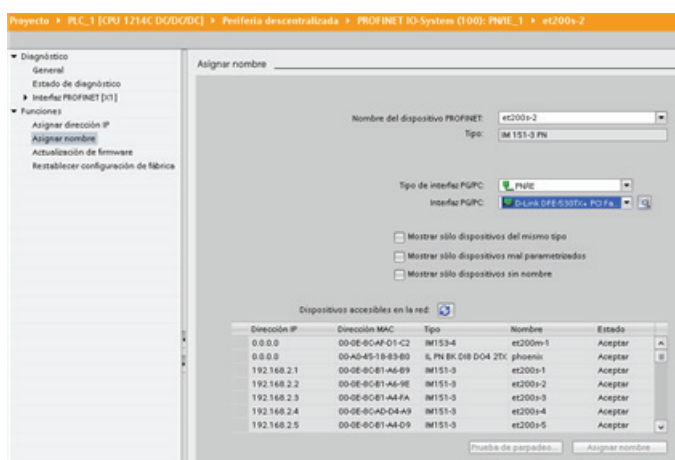
1. En el editor "Dispositivos y redes", haga clic con la tecla derecha del ratón en el dispositivo PROFINET IO deseado y seleccione "Online y diagnóstico".



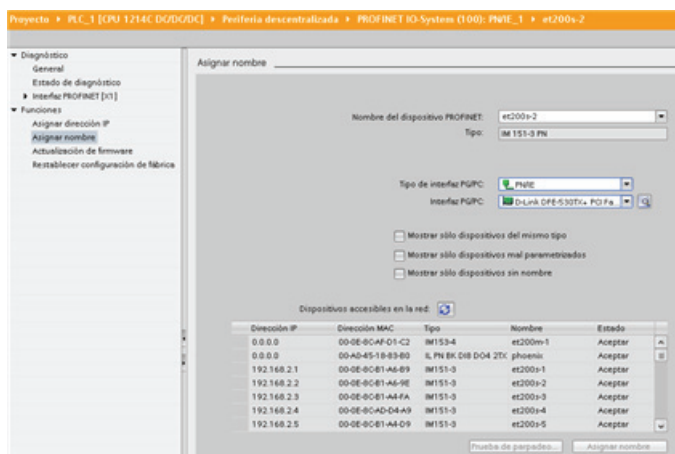
2. Seleccione los siguientes comandos de menú en el diálogo "Online y diagnóstico":

- "Funciones"
- "Asignar nombre"

Haga clic en el icono "Dispositivos accesibles en la red" para visualizar todos los dispositivos PROFINET IO en la red.



3. En la lista visualizada, haga clic en el dispositivo PROFINET IO deseado y haga clic el botón "Asignar nombre" para escribir el nombre en la memoria de configuración del dispositivo PROFINET IO.



14.4 Ajustar la dirección IP y la hora

Es posible ajustar la dirección IP (Página 139) y la hora en la CPU online. Tras acceder a "Online y diagnóstico" desde el árbol de proyectos para una CPU en línea, es posible visualizar o cambiar la dirección IP. También es posible visualizar o ajustar los parámetros de fecha y hora en la CPU online.



Nota

Esta función sólo está disponible para una CPU que sólo tenga una dirección MAC (que aún no se le haya asignado una dirección IP) o en la que se hayan restablecido los valores de fábrica.

14.5 Restablecimiento de los ajustes de fábrica

Se puede restablecer un S7-1200 a sus ajustes originales de fábrica bajo las siguientes condiciones:

- No se ha insertado una Memory Card en la CPU.
- La CPU tiene una conexión online.
- La CPU está en modo STOP.

Nota

Si la CPU está en modo RUN y se inicia la operación de restablecimiento, se puede poner en modo STOP después del acuse de recibo de una solicitud de confirmación.

Procedimiento

Para restablecer una CPU a sus ajustes de fábrica, proceda del siguiente modo:

1. Abra la vista Online y diagnóstico de la CPU.
2. Seleccione "Restablecer a ajustes de fábrica" en la carpeta "Funciones".
3. Seleccione la casilla de verificación "Conservar dirección IP" si desea conservar la dirección IP, o la casilla de verificación "Restablecer dirección IP" si desea eliminar la dirección IP.
4. Haga clic en el botón "Reset".
5. Confirme la solicitud de confirmación con "Aceptar".

Resultado

El módulo cambia a modo STOP si es necesario y se restablece a los ajustes de fábrica:

- Se borra la memoria de trabajo y la memoria de carga interna y todas las áreas de operandos.
- Todos los parámetros se restablecen a sus valores predeterminados.
- El búfer de diagnóstico se borra.
- Se restablece la hora del día.
- La dirección IP se conserva o se elimina en función del ajuste que haya seleccionado. (La dirección MAC es fija y nunca se cambia.)

14.6 Panel de control de la CPU online



El "Panel de control de la CPU" muestra el estado operativo (STOP o RUN) de la CPU online. También indica si la CPU tiene un error o si se están forzando valores.

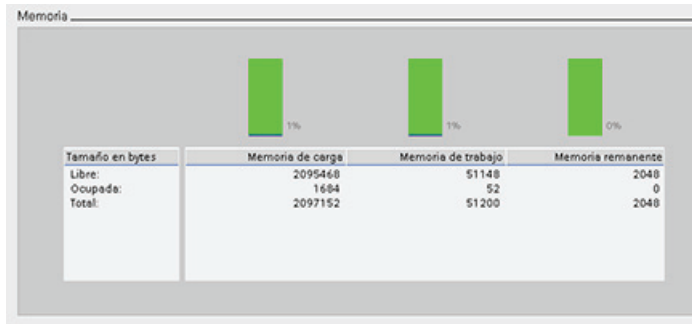
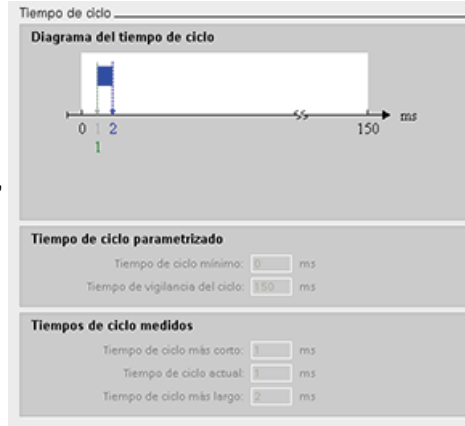
Utilice el panel de control de la CPU en la Task Card de las herramientas online para cambiar el estado operativo de una CPU online. La Task Card de las herramientas online es accesible siempre que la CPU esté online.

14.7 Vigilar el tiempo de ciclo y la carga de la memoria

Es posible vigilar el tiempo de ciclo y la carga de la memoria de una CPU online.

Tras establecer una conexión con la CPU online, abra la Task Card de las herramientas online para visualizar los siguientes valores medidos:

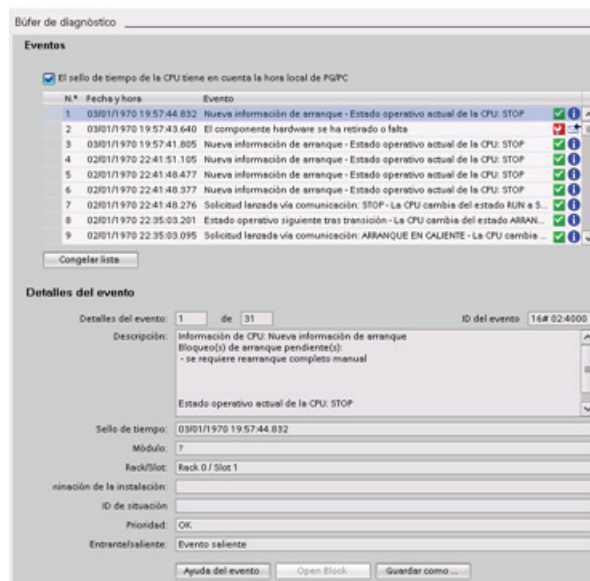
- Tiempo de ciclo
- Carga de la memoria



14.8 Visualizar los eventos de diagnóstico de la CPU

El búfer de diagnóstico permite consultar las actividades recientes de la CPU. El búfer de diagnóstico es accesible desde "Online y diagnóstico" para una CPU online en el árbol del proyecto. Contiene las entradas siguientes:

- Eventos de diagnóstico
- Cambios del estado operativo de la CPU (transiciones a STOP o RUN)



La primera entrada contiene el evento más reciente. Toda entrada del búfer de diagnóstico incluye la fecha y hora de registro del evento, así como una descripción.

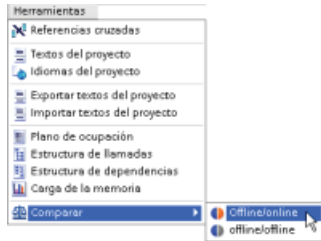
El número máximo de entradas depende de la CPU. Se soportan 50 entradas como máximo.

Sólo los 10 eventos más recientes del búfer de diagnóstico se almacenan de forma permanente. Si se restablece la configuración de fábrica de la CPU, se reinicializa el búfer de diagnóstico y se borran sus entradas.

También se puede utilizar la instrucción GET_DIAG (Página 316) para recopilar la información de diagnóstico.

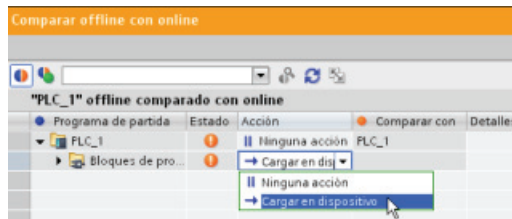
14.9 Comparar CPUs online y offline

Los bloques lógicos de una CPU online y los del proyecto se pueden comparar. Si los bloques lógicos del proyecto no coinciden con los de la CPU online, el editor de comparación permite sincronizar el proyecto con la CPU online descargando los bloques lógicos del proyecto en la CPU, o eliminando del proyecto los que no existan en la CPU online.



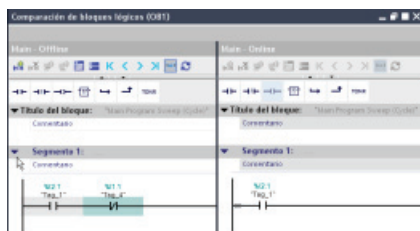
Seleccione la CPU del proyecto.

Use el comando "Comparar offline/online" para abrir el editor de comparación. (Acceda al comando desde el menú "Herramientas" o haciendo clic con el botón derecho del ratón en la CPU del proyecto.)



Haga clic en la columna "Acción" de un objeto para seleccionar si elimina el objeto, no realiza ninguna acción, o carga el objeto en el dispositivo.

Haga clic en el botón "Sincronizar" para cargar los bloques lógicos.



Haga clic con el botón derecho del ratón en un objeto en la columna "Comparar con" y seleccione el botón "Iniciar comparación detallada" para mostrar los bloques lógicos uno junto a otro.

La comparación detallada destaca las diferencias entre los bloques lógicos de la CPU online y los de la CPU del proyecto.

14.10 Vigilar y modificar valores en la CPU

STEP 7 ofrece herramientas online para vigilar la CPU:

- Es posible visualizar o vigilar los valores actuales de las variables. La función de vigilancia no modifica la secuencia del programa. Facilita información sobre la secuencia y los datos del programa en la CPU.
- También se pueden utilizar otras funciones para controlar la secuencia y los datos del programa de usuario:
 - Es posible modificar el valor de las variables en la CPU online para observar cómo reacciona el programa de usuario.
 - Se puede forzar de forma permanente una salida periférica (como Q0.1:P o "Start":P) a un valor específico.
 - Es posible desbloquear las salidas en modo STOP.

Nota

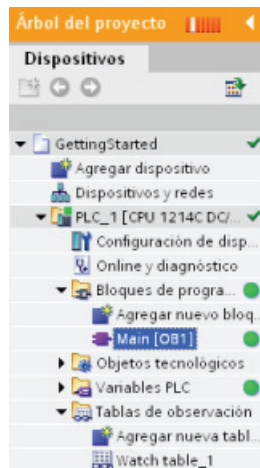
Hay que prestar siempre atención al utilizar las funciones de forzado. Estas funciones pueden influir seriamente en la ejecución del programa de usuario/de sistema.

Tabla 14- 3 Capacidades online de los editores de STEP 7

Editor	Vigilar	Modificar	Forzado permanente
Tabla de observación	Sí	Sí	No
Tabla de forzado permanente	Sí	No	Sí
Editor de programas	Sí	Sí	No
Tabla de variables	Sí	No	No
Editor de DB	Sí	No	No

14.10.1 Conexión online para observar los valores en la CPU

Para observar las variables debe existir una conexión online con la CPU. Haga clic en el botón "Establecer conexión online" de la barra de herramientas.



Una vez establecida la conexión con la CPU, los encabezados de las áreas de trabajo de STEP 7 aparecen en color naranja.

El árbol de proyectos muestra una comparación entre el proyecto offline y la CPU online. Un círculo verde significa que la CPU y el proyecto están sincronizados, es decir, que ambos tienen la misma configuración y programa de usuario.

Las tablas de variables muestran las variables. Las tablas de observación también pueden mostrar las variables, así como direcciones directas.

	Nombre	Dirección	Formato visualiza...	Valor de observac.	Valor de forzado
1	*On	%I 0.0	Bool		
2	*Off	%I 0.1	Bool		
3	*Run	%Q 0.0	Bool		

Para observar la ejecución del programa de usuario y visualizar los valores de las variables, haga clic en el botón "Observar todos" de la barra de herramientas.

	Nombre	Dirección	Formato visualiza...	Valor de observac.	Valor de forzado
1	*On	%I 0.0	Bool	<input type="checkbox"/> FALSE	
2	*Off	%I 0.1	Bool	<input type="checkbox"/> FALSE	
3	*Run	%Q 0.0	Bool	<input type="checkbox"/> FALSE	

El campo "Valor de observación" muestra el valor de cada variable.

14.10.2 Visualización del estado en el editor de programas

También es posible observar el estado de las variables en los editores de programas KOP y FUP. Utilice la barra de editores para visualizar el editor KOP. La barra de editores permite conmutar la vista entre los editores abiertos sin tener que abrirlos o cerrarlos.

En la barra de herramientas del editor de programas, haga clic en el botón "Activar/desactivar observación" para ver el estado del programa de usuario.



La red del editor de programas indica el flujo de corriente en color verde.

También se puede hacer clic con el botón derecho del ratón en la instrucción o parámetro para modificar el valor de la instrucción.

14.10.3 Obtener los valores online de un DB para restablecer los valores iniciales

Existe la posibilidad de obtener los valores actuales que se están vigilando en una CPU online con el fin de conseguir los valores iniciales de un DB global.

- Es necesario disponer de una conexión online con la CPU.
- La CPU debe estar en RUN.
- El DB debe estar abierto en STEP 7.



Utilice el botón "Muestra una instantánea de los valores de observación" para obtener los valores actuales de las variables seleccionadas en el DB. A continuación, puede copiar estos valores en la columna "Valor inicial" del DB.

1. En el editor de DB, haga clic en el botón "Observar todos". La columna "Valor de observación" muestra los valores de datos actuales.
2. Haga clic en el botón "Muestra una instantánea de los valores de observación" para visualizar los valores actuales en la columna "Instantánea".
3. Haga clic en el botón "Observar todos" para detener la vigilancia de datos en la CPU.
4. Copie un valor en la columna "Instantánea" de una variable.
 - Seleccione un valor que deba copiarse.
 - Haga clic con el botón derecho del ratón en el valor seleccionado para abrir el menú contextual.
 - Elija el comando "Copiar".
5. Pegue el valor copiado en la columna "Valor inicial" correspondiente a la variable. (Haga clic con el botón derecho del ratón en la celda y seleccione "Pegar" del menú contextual.)

14.10 Vigilar y modificar valores en la CPU

6. Guarde el proyecto para configurar los valores copiados como nuevos valores iniciales del DB.
7. Compile y cargue el DB en la CPU. El DB utiliza los nuevos valores iniciales una vez la CPU ha pasado a RUN.

Nota

Los valores que se muestran en la columna "Valor de observación" son siempre copiados de la CPU. STEP 7 no comprueba si todos los valores provienen del mismo ciclo de la CPU.

14.10.4 Utilizar una tabla de observación para observar y modificar valores en la CPU

Una tabla de observación permite observar y forzar datos a medida que la CPU ejecuta el programa. Estos datos pueden ser de la memoria imagen de proceso (I o Q), M, DB o entradas físicas (I_:P), dependiendo de la función de observación o forzado. No es posible observar con precisión las salidas físicas (Q_:P) debido a que la función de vigilancia sólo puede visualizar el último valor escrito desde la memoria Q y no lee los valores reales de las salidas físicas.

La función de vigilancia no modifica la secuencia del programa. Facilita información sobre la secuencia y los datos del programa en la CPU.

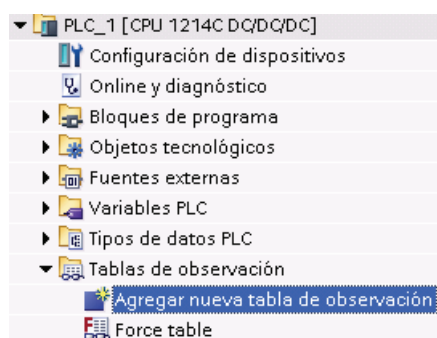
Las funciones de control permiten al usuario controlar la secuencia y los datos del programa. Hay que prestar atención al utilizar las funciones de forzado. Estas funciones pueden influir seriamente en la ejecución del programa de usuario/de sistema. Las tres funciones de control son: forzar, forzar permanentemente y desbloquear salidas en STOP.

La tabla de observación permite realizar las siguientes funciones online:

- Observar el estado de las variables
- Forzar los valores de las distintas variables

Es posible seleccionar cuándo se debe observar o forzar la variable:

- Inicio del ciclo: Lee o escribe el valor al inicio del ciclo
- Fin del ciclo: Lee o escribe el valor al final del ciclo
- Cambiar a STOP



Para crear una tabla de observación, proceda del siguiente modo:

1. Haga doble clic en "Agregar nueva tabla de observación" para abrir una tabla de observación nueva.
2. Introduzca el nombre de la variable o agregue una variable a la tabla de observación.

Las siguientes opciones están disponibles para observar variables:

- Observar todos: Este comando inicia la observación de las variables visibles en la tabla de observación activa.
- Observar inmediatamente: Este comando inicia la observación de las variables visibles en la tabla de observación activa. La tabla de observación observa las variables inmediatamente y una sola vez.

Las siguientes opciones están disponibles para forzar variables:

- "Forzar a 0" pone a "0" el valor de una dirección seleccionada.
- "Forzar a 1" pone a "1" el valor de una dirección seleccionada.
- "Forzar inmediatamente" cambia inmediatamente los valores de las direcciones seleccionadas durante un ciclo.
- "Forzar con disparador" cambia los valores de las direcciones seleccionadas.

Esta función no confirma si las direcciones seleccionadas se han forzado realmente. Si se requiere una confirmación del cambio, utilice la función "Forzar inmediatamente".

- "Habilitar salidas de periferia" desactiva el comando para inhibir las salidas y está disponible sólo cuando la CPU se encuentra en estado operativo STOP.

Para observar las variables debe existir una conexión online con la CPU.

	Nombre	Dirección	Formato visualización	Valor de observación	Observar con disparo	Forzar con disparador	Valor de forzado
1	"Start"	%I0.0	Bool		Permanente	Permanente	<input type="checkbox"/>
2	"Stop"	%I0.1	Bool		Permanente	Permanente	<input type="checkbox"/>
3	"Running"	%M0.0	Bool		Permanente	Permanente	<input type="checkbox"/>

Las distintas funciones pueden seleccionarse mediante los botones en el lado superior de la tabla de observación.

Introduzca el nombre de la variable que desea observar y seleccione un formato de visualización en la lista desplegable. Si existe una conexión online con la CPU y se hace clic en el botón "Observar", se visualizará el valor real del punto de datos en el campo "Valor de observación".

14.10.4.1 Utilizar un disparador para observar o forzar variables PLC

Si se utiliza un disparador es posible determinar en qué punto del ciclo debe observarse o forzarse la dirección seleccionada.

Tabla 14- 4 Tipos de disparos

Disparo	Descripción
Permanente	Recoge los datos continuamente
Al inicio del ciclo	Permanente: Recoge los datos continuamente al inicio del ciclo, después de que la CPU lee las entradas
	Único: Recoge los datos al inicio del ciclo, después de que la CPU lee las entradas
Al final del ciclo	Permanente: Recoge los datos continuamente al final del ciclo, antes de que la CPU escribe en las salidas
	Único: Recoge los datos una vez al final del ciclo, antes de que la CPU escribe en las salidas
Transición a STOP	Permanente: Recoge los datos continuamente cuando la CPU pasa a STOP
	Único: Recoge los datos una vez cuando la CPU pasa a STOP

Para modificar una variable PLC en un disparador determinado, seleccione el inicio o final del ciclo.

- Forzar una salida: El mejor evento de disparo para forzar una salida es al final del ciclo, inmediatamente antes de que la CPU escribe en las salidas.

Observe el valor de las salidas al inicio del ciclo para determinar qué valor se escribe en las salidas físicas. Asimismo, observe las salidas antes de que la CPU escriba los valores en las salidas físicas para comprobar la lógica del programa y compararla con la reacción real de las E/S.


- Forzar una entrada: El mejor evento de disparo para forzar una entrada es al inicio del ciclo, inmediatamente después de que la CPU lee las entradas y antes de que el programa utiliza los valores de entrada.

Si está forzando entradas al inicio del ciclo, deberá observar el valor de las entradas al final del ciclo para asegurarse de que no ha cambiado desde el principio.. Si los valores son diferentes, es posible que el programa de usuario esté escribiendo en una entrada por error.

Para diagnosticar la causa de una transición a STOP de la CPU, utilice el disparador "Transición a STOP" para capturar los últimos valores de proceso.

14.10.4.2 Desbloquear las salidas en modo STOP

La tabla de observación permite escribir en las salidas cuando la CPU está en modo STOP. Esta función sirve para comprobar el cableado de las salidas y verificar que el cable conectado a un borne de salida lanza una señal "high" o "low" al terminal del dispositivo de proceso al que está conectado.

 ADVERTENCIA
Aunque la CPU está en estado operativo STOP, la habilitación de una salida física puede activar el punto del proceso a la que está conectada.

El estado de las salidas se puede cambiar en estado operativo STOP si están habilitadas las salidas. Si las salidas están inhibidas, no es posible modificarlas en estado operativo STOP.

- Para habilitar la modificación de las salidas en STOP, seleccione la opción "Habilitar salidas de periferia" del comando "Forzar" del menú "Online", o bien haga clic con el botón derecho del ratón en la fila correspondiente de la tabla de observación.
No es posible habilitar salidas en modo STOP si se han configurado E/S descentralizadas. En caso de intentarlo, se devuelve un error.
- Cuando la CPU pasa a estado operativo RUN se inhibe la opción "Habilitar salidas de periferia".
- Si alguna entrada o salida se fuerza permanentemente, la CPU no podrá desbloquear las salidas en estado operativo STOP. Primero es preciso cancelar la función de forzado permanentemente.

14.10.5 Forzar valores permanentemente en la CPU

14.10.5.1 Utilizar la tabla de forzado permanente

Una tabla de forzado permanente incluye una función de "forzado permanente" que sobrescribe el valor de una entrada o salida con un valor específico para la dirección de entrada o salida de periferia. La CPU aplica el valor forzado permanentemente en la memoria imagen de proceso de las entradas antes de ejecutar el programa de usuario y en la memoria imagen de proceso de las salidas antes de escribir las salidas en los módulos.

Nota

Los valores de forzado permanente se guardan en la CPU y no en la tabla de forzado permanente.

No se puede forzar permanentemente una entrada (o dirección "I") ni una salida (o dirección "Q"). No obstante, sí que es posible forzar permanentemente una entrada o salida de periferia. La tabla de forzado permanente agrega automáticamente ":P" a la dirección (por ejemplo: "On":P o "Run":P).

	F.	F.				F
i	Nombre	Dirección	Formato visualización	Valor de observación	Valor de forzado permanente	
1	"On":P	%I0.0:P	Bool		TRUE	<input checked="" type="checkbox"/>
2	"Off":P	%I0.1:P	Bool			<input type="checkbox"/>
3	"Run":P	%Q0.1:P	Bool			<input type="checkbox"/>

En la celda "Valor de forzado permanente", introduzca el valor de la entrada o salida que debe forzarse. Puede emplearse la casilla de verificación de la columna "Forzado permanente" para permitir el forzado permanente de la entrada o salida.



Utilice el botón "Iniciar o reemplazar forzado permanente" para forzar permanentemente el valor de las variables en la tabla de forzado permanente. Haga clic en el botón "Finalizar forzado permanente" para inicializar el valor de las variables.

En la tabla de forzado permanente, es posible vigilar el estado del valor forzado para una entrada. Sin embargo, no es posible vigilar el valor forzado de una salida.

También es posible observar el estado del valor forzado permanentemente en el editor de programas.



ATENCIÓN

Quando una entrada o salida se fuerza permanentemente en la tabla de forzado permanente, las acciones de forzado permanente se convierten en parte de la configuración del proyecto. Si se cierra STEP 7, los elementos forzados permanentemente permanecen activados en el programa de la CPU hasta que se borran. Para borrar estos elementos forzados permanentemente es necesario utilizar STEP 7 para establecer una conexión con la CPU online y utilizar la tabla de forzado permanente para desactivar o detener la función de forzado permanente de estos elementos.

14.10.5.2 Funcionamiento de la función de forzado permanente

La CPU permite forzar permanentemente las entradas y salidas. Para ello, es preciso indicar la dirección de la entrada o salida física (I_:P o Q_:P) en la tabla de forzado permanente e iniciar la función de forzado permanente.

El valor forzado permanentemente sobrescribe las lecturas de las entradas físicas en el programa. El programa utiliza el valor forzado permanentemente para el procesamiento. Cuando el programa escribe en una salida física, el valor de forzado permanente sobrescribe el de la salida. El valor forzado permanentemente aparece en la salida física y es utilizado por el proceso.

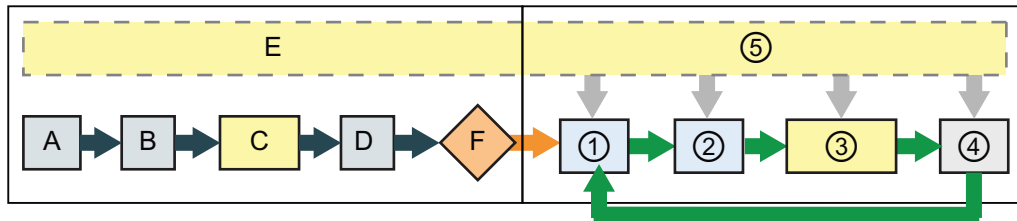
Cuando una entrada o salida se fuerza permanentemente en la tabla de forzado permanente, las acciones de forzado permanente se convierten en parte del programa de usuario. Aunque se haya cerrado el software de programación, las selecciones de forzado permanente permanecen activadas en el programa de la CPU hasta que son borradas al establecer una conexión online desde el software de programación y se para la función de forzado permanente. Los programas con entradas y/o salidas forzadas de forma permanente que se hayan cargado en una CPU diferente desde una Memory Card seguirán forzando permanentemente las E/S seleccionadas en el programa.

Si la CPU ejecuta el programa de usuario desde una Memory Card protegida contra escritura, el forzado permanente de una E/S no se puede iniciar ni cambiar desde una tabla de observación, ya que no es posible sobrescribir los valores en el programa de usuario protegido contra escritura. Todo intento de forzar permanentemente los valores protegidos contra escritura generará un error. Si se utiliza una Memory Card para transferir un programa de usuario, los elementos forzados permanentemente en esa Memory Card se transferirán a la CPU.

Nota

Las E/S digitales asignadas al HSC, PWM y PTO no se pueden forzar permanentemente

Las E/S digitales utilizadas por el contador rápido (HSC) y los dispositivos con modulación del ancho de pulso (PWM) y tren de impulsos (PTO) se asignan durante la configuración de dispositivos. Si se asignan direcciones de E/S digitales a dichos dispositivos, los valores de las direcciones de E/S asignadas no podrán modificarse utilizando la función de forzado permanente de la tabla de observación.



Arranque

- A La función de forzado permanente no afecta el borrado del área de memoria de las entradas (I).
- B La función de forzado permanente no afecta la inicialización de los valores de salida.
- C Durante la ejecución de los OBs de arranque, la CPU aplica el valor de forzado permanente cuando el programa de usuario accede a la entrada física.
- D El almacenamiento de los eventos de alarma en la cola de espera no se ve afectado.
- E La habilitación de escritura en las salidas no se ve afectada.

RUN

- ① Mientras escribe la memoria de las salidas (Q) en las salidas físicas, la CPU aplica el valor de forzado permanente a medida que se van actualizando las salidas.
- ② Al leer las entradas físicas, la CPU aplica los valores de forzado permanente inmediatamente antes de copiar las entradas en la memoria I.
- ③ Durante la ejecución del programa de usuario (OBs de ciclo), la CPU aplica el valor de forzado permanente cuando el programa de usuario accede a la entrada física o escribe en la salida física.
- ④ La función de forzado permanente no afecta el procesamiento de peticiones de comunicación ni los diagnósticos de autotest.
- ⑤ El procesamiento de las alarmas en cualquier parte del ciclo no se ve afectado.

14.11 Carga en estado operativo RUN

La CPU soporta la "carga en estado operativo RUN". Esta función permite efectuar cambios pequeños en el programa de usuario sin afectar demasiado a los equipos controlados. No obstante, también es posible realizar modificaciones considerables que podrían ser perjudiciales o incluso peligrosas.

⚠ ADVERTENCIA

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

Solo el personal cualificado que tenga pleno conocimiento de los efectos que esta función puede tener en el sistema debería efectuar la carga en el estado operativo RUN.

La función "Carga en estado operativo RUN" permite realizar cambios en un programa y luego cargarlos en la CPU sin necesidad de pasar al estado operativo STOP:

- Es posible realizar cambios menores en el proceso en curso sin tener que desconectar el equipo (p. ej. modificar el valor de un parámetro).
- Esta función permite comprobar un programa más rápidamente (p. ej. invertir la lógica de un contacto normalmente abierto o cerrado).

Se pueden realizar los siguientes cambios en bloques de programa y variables y cargarlos en RUN:

- Crear, sobrescribir y eliminar funciones (FC), bloques de función (FB) y tablas de variables.
- Crear y eliminar bloques de datos (DB). No obstante, los cambios de estructura de los DB no se pueden sobrescribir. Los valores iniciales de los DBs se pueden sobrescribir. No es posible descargar un DB de servidor web (control o fragmento) en RUN.
- Sobrescribir bloques de organización (OB); sin embargo, no se pueden crear ni eliminar OB.

En modo RUN se pueden descargar como máximo diez bloques a la vez. Si se descargan más de diez bloques, la CPU debe ponerse a STOP.

Si desea cargar cambios en un proceso real (a diferencia de un proceso simulado, lo cual podría ocurrir durante la comprobación de un programa), es de vital importancia analizar a fondo los posibles efectos en materia de seguridad para máquinas y operadores antes de realizar la carga.

Nota

Si la CPU está en RUN y se realizan cambios en el programa, STEP 7 intentará siempre descargar primero en RUN. Si no desea que esto suceda, deberá poner la CPU en estado operativo STOP.

Si los cambios realizados no se soportan en "Cargar en RUN", STEP 7 pedirá al usuario que ponga la CPU a STOP.

14.11.1 Requisitos para la "Carga en estado operativo RUN"

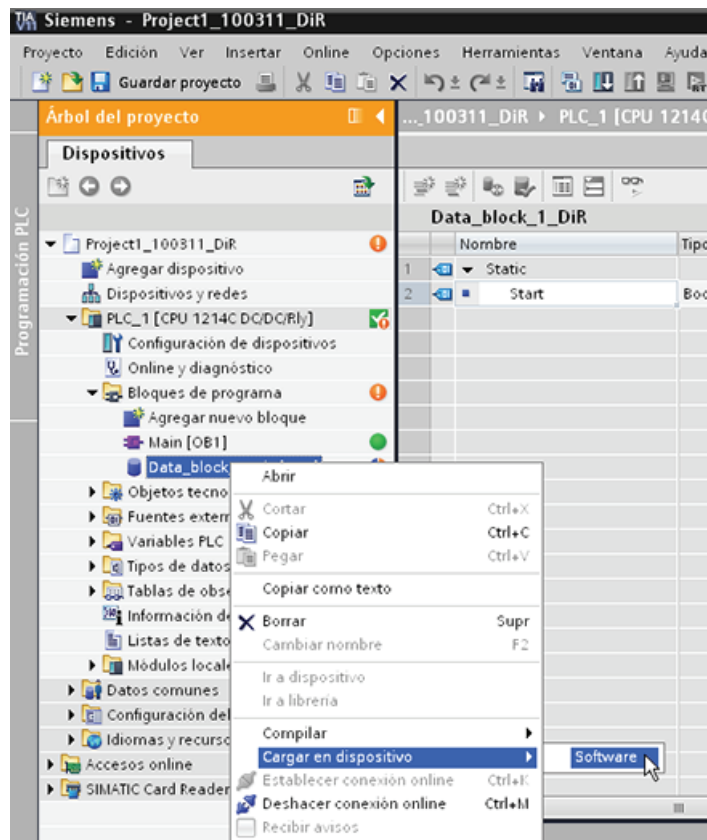
No se pueden descargar los cambios del programa en una CPU que esté en modo RUN, a menos que se cumplan estos requisitos:

- El programa debe compilarse correctamente.
- Se debe haber establecido correctamente la comunicación entre la programadora en la que se ejecuta STEP 7 y la CPU.
- En V3.0 y posteriores, el firmware de la CPU de destino debe soportar la función "Carga en estado operativo RUN".

14.11.2 Modificar el programa en el estado operativo RUN

Para modificar el programa en el estado operativo RUN, asegúrese primero de que la CPU soporte la función "Carga en estado operativo RUN" y que la CPU se encuentre efectivamente en el estado operativo RUN:

1. Para descargar el programa en RUN, seleccione uno de los métodos siguientes:
 - Comando "Cargar en dispositivo" del menú "Online"
 - Botón "Cargar en dispositivo" de la barra de herramientas
 - En el "Árbol del proyecto" haga clic con el botón derecho del ratón en "Bloques de programa" y seleccione el comando "Cargar en dispositivo > Software".



2. Si el programa se compila correctamente, STEP 7 carga el programa en la CPU.
3. STEP 7 le solicita al usuario que cargue el programa o que cancele la operación.
4. Si hace clic en "Cargar", STEP 7 carga el programa en la CPU.

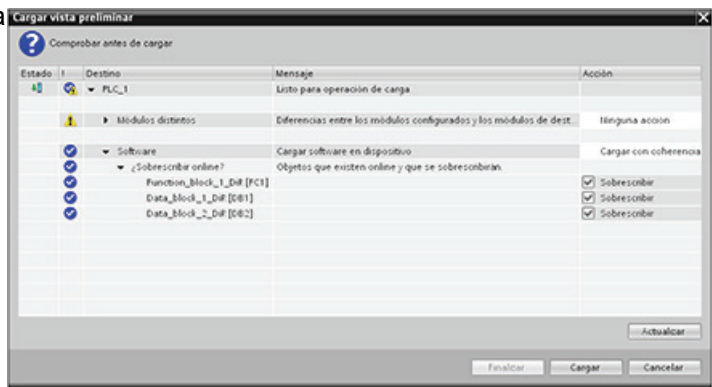
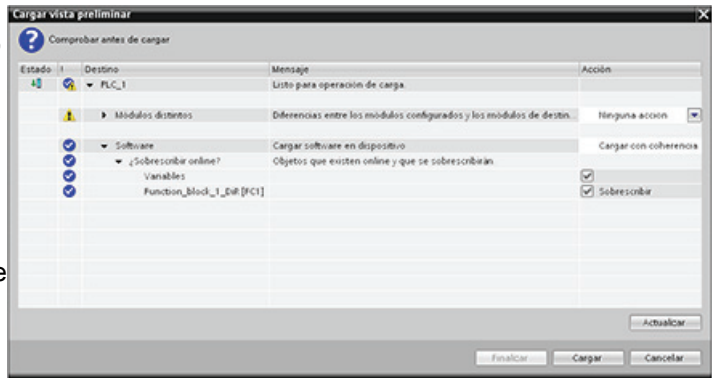
14.11.3 Descargar bloques seleccionados

Está marcada la carpeta Bloques de programa, la selección de bloques o un único bloque.

1. Si el usuario selecciona un único bloque para descargarlo desde el editor de bloques, entonces la única opción de la columna "Acción" es "Carga coherente".

El usuario puede ampliar la línea de categoría para asegurarse de los bloques que deben descargarse. En este ejemplo se ha realizado una pequeña modificación del bloque offline y no es necesario cargar otros bloques.

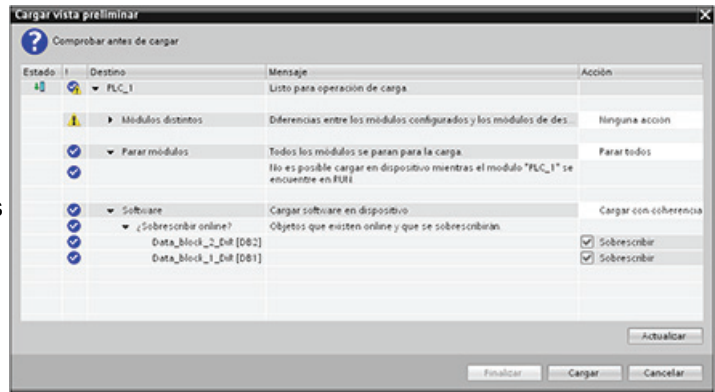
2. En este ejemplo se necesita más de un bloque para la descarga.



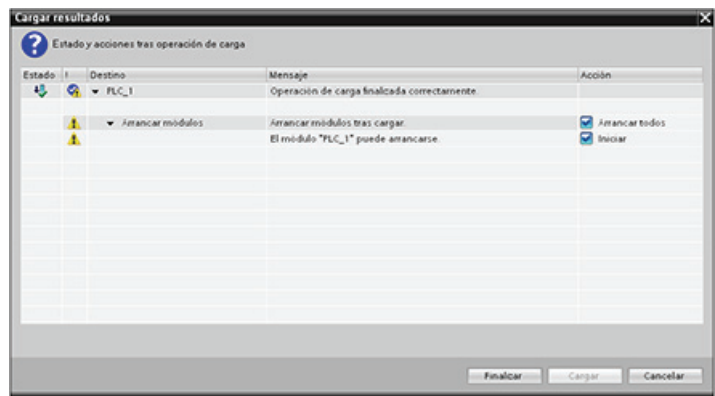
Nota

En el modo RUN se pueden descargar como máximo diez bloques a la vez. Si se descargan más de diez bloques, la CPU debe ponerse a STOP.

3. Si el usuario intenta descargar en RUN, pero el sistema detecta que esto no es posible antes de la descarga real, entonces en el cuadro de diálogo aparece la línea de categoría de módulos de parada.



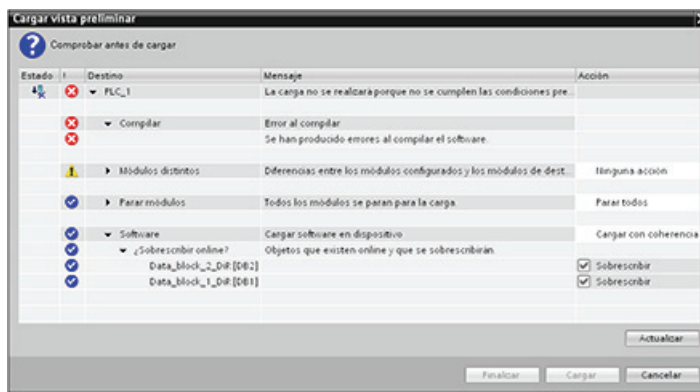
4. Haga clic en el botón "Cargar" y aparecerá el cuadro de diálogo "Cargar resultados". Haga clic en el botón "Finalizar" para finalizar la carga.



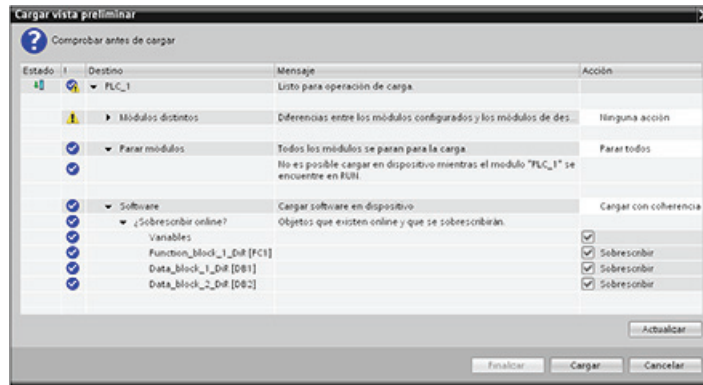
14.11.4

Descargar un único bloque seleccionado con un error de compilación en otro bloque

Si el usuario intenta realizar una descarga coherente con error de compilación en otro bloque, el cuadro de diálogo indicará un error y el botón de carga estará deshabilitado.

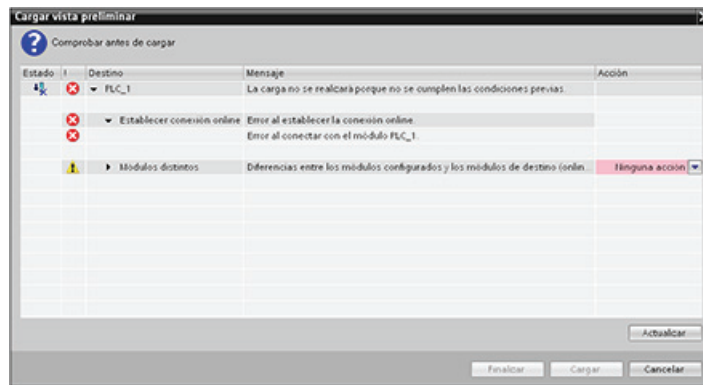


El usuario debe corregir el error de compilación en el otro bloque. En ese caso, el botón "Cargar" pasa a estar activo.



14.11.5 Reacción del sistema si falla la descarga

Si durante el proceso inicial de descarga en RUN falla una conexión de red, se mostrará el cuadro de diálogo "Cargar vista preliminar" tal como aparece en la figura inferior.



14.11.6 Descarga del programa en estado operativo RUN

Antes de descargar el programa en modo RUN, considere los efectos que puede tener una modificación en modo RUN en el funcionamiento de la CPU en las situaciones siguientes:

- Si ha borrado la lógica de control de una salida, la CPU conservará el último estado de la salida hasta la próxima desconexión y conexión, o bien, hasta la próxima transición al estado operativo STOP.
- Si ha borrado un contador rápido o funciones de salida de impulsos que se estaban ejecutando, el contador o la salida de impulsos se seguirá ejecutando hasta la próxima desconexión y posterior conexión, o bien, hasta que se cambie a modo STOP.
- Cualquier lógica que dependa del estado del bit de primera consulta no se ejecutará hasta la próxima desconexión y posterior conexión, o bien hasta que se cambie de STOP a RUN. El bit de primera consulta sólo se activa al cambiar a modo RUN y no se ve afectado por una descarga en modo RUN.
- Los valores actuales de los bloques de datos (DB) y/o variables no se sobrescriben.

Nota

Para poder descargar el programa en modo RUN, la CPU tiene que admitir la modificación en modo RUN, el programa debe compilarse correctamente y la comunicación entre STEP 7 y la CPU debe funcionar sin errores.

Se pueden realizar los cambios siguientes en variables y bloques de programa y descargarlos en modo RUN:

- Crear, sobrescribir y eliminar funciones (FC), bloques de función (FB) y tablas de variables.
- Crear y eliminar bloques de datos (DB). No obstante, los cambios de estructura de los DB no se pueden sobrescribir. Los valores iniciales de los DBs se pueden sobrescribir. No es posible descargar un DB de servidor web (control o fragmento) en RUN.
- Sobrescribir bloques de organización (OB); sin embargo, no se pueden crear ni eliminar OB.

En el modo RUN se pueden descargar como máximo diez bloques a la vez. Si se descargan más de diez bloques, la CPU debe ponerse a STOP.

Una vez iniciada una descarga, no se pueden realizar otras tareas en STEP 7 hasta que no se haya terminado la descarga.

Instrucciones que pueden fallar debido a una "Carga en estado operativo RUN"

Las instrucciones siguientes pueden sufrir un error temporal cuando se activan en la CPU cambios en la carga en RUN. El error se produce al iniciar la instrucción mientras la CPU se está preparando para activar los cambios descargados. Durante este tiempo, la CPU suspende la iniciación del acceso del programa de usuario a la memoria de carga mientras completa el acceso del programa de usuario en curso a la memoria de carga. Esto sucede para que los cambios descargados puedan activarse de forma coherente.

Instrucción	Respuesta con la activación pendiente
DataLogCreate	STATUS = W#16#80C0, ERROR = TRUE
DataLogOpen	STATUS = W#16#80C0, ERROR = TRUE
DataLogWrite	STATUS = W#16#80C0, ERROR = TRUE
DataLogClose	STATUS = W#16#80C0, ERROR = TRUE
DataLogNewFile	STATUS = W#16#80C0, ERROR = TRUE
READ_DBL	RET_VAL = W#16#82C0
WRIT_DBL	RET_VAL = W#16#82C0
RTM	RET_VAL = 0x80C0

En todos los casos, la salida RLO de la instrucción será "false" cuando se produzca el error. El error es temporal. Si esto ocurre, la instrucción debería repetirse más adelante.

Nota

No debe repetirse la operación durante la ejecución actual del OB.

Datos técnicos

A

A.1 Datos técnicos generales

Homologaciones

El diseño del sistema de automatización S7-1200 cumple las siguientes normas y especificaciones de ensayo. Los criterios de test del sistema de automatización S7-1200 se basan en estas normas y especificaciones de ensayo.

Nótese que no todos los modelos S7-1200 pueden certificarse según esas normas y el estado de certificación puede cambiar sin notificación. El usuario es responsable de determinar las certificaciones aplicables consultando las inscripciones marcadas en el producto. Contacte con el representante de Siemens más próximo para obtener una lista de las homologaciones actuales con las referencias respectivas.

Homologación CE



El sistema de automatización S7-1200 satisface los requisitos y objetivos relacionados con la seguridad según las directivas CE indicadas a continuación y cumple las normas europeas (EN) armonizadas para controladores programables publicadas en los Diarios Oficiales de la Unión Europea.

- Directiva CE 2006/95/CE (Directiva de baja tensión) "Material eléctrico destinado a utilizarse con determinados límites de tensión"
 - EN 61131-2:2007 Autómatas programables - Requisitos y ensayos de los equipos
- Directiva CE 2004/108/CE (Directiva CEM) "Compatibilidad electromagnética"
 - Norma de emisión
EN 61000-6-4:2007: Entornos industriales
 - Norma de inmunidad
EN 61000-6-2:2005: Entornos industriales
- Directiva CE 94/9/CE (ATEX) "Equipos y sistemas de protección para uso en atmósferas potencialmente explosivas"
 - EN 60079-15:2005: Tipo de protección 'n'

La Declaración de conformidad CE se encuentra a disposición de las autoridades competentes en:

Siemens AG
IA AS RD ST PLC Amberg
Werner-von-Siemens-Str. 50
D92224 Amberg
Germany

Homologación cULus



Underwriters Laboratories Inc. cumple:

- Underwriters Laboratories, Inc.: UL 508 Listed (Industrial Control Equipment)
- Canadian Standards Association: CSA C22.2 Number 142 (Process Control Equipment)

ATENCIÓN
La gama SIMATIC S7-1200 cumple la norma CSA.
El logotipo cULus indica que Underwriters Laboratories (UL) ha examinado y certificado el S7-1200 según las normas UL 508 y CSA 22.2 No. 142.

Homologación FM



Factory Mutual Research (FM)

Números de clase 3600 y 3611 de la norma de aprobación


Aprobado para ser utilizado en:

Class I, Division 2, Gas Group A, B, C, D, Temperature Class T3C Ta = 60 °C

Class I, Zone 2, IIC, Temperature Class T3 Ta = 60 °C

Canadian Class I, Zone 2 instalación según CEC 18-150

EXCEPCIÓN IMPORTANTE: Consulte en las especificaciones técnicas el número de entradas y salidas simultáneamente en ON permitido. Algunos modelos tienen características reducidas para Ta = 60 °C.

 ADVERTENCIA
Si se sustituyen componentes, podría perderse la idoneidad para la clase I, categoría 2 o zona 2.
Sólo un centro de asistencia Siemens autorizado puede reparar estas unidades.

Homologación ATEX



La homologación ATEX sólo es válida para los modelos DC. La homologación ATEX no es válida para los modelos AC y de relé.

EN 60079-0:2006: Atmósferas explosivas - Requisitos generales

EN 60079-15:2005: Material eléctrico para atmósferas de gas explosivas;

Tipo de protección 'nA'

II 3 G Ex nA II T3

EXCEPCIÓN IMPORTANTE: Consulte en las especificaciones técnicas el número de entradas y salidas simultáneamente en ON permitido. Algunos modelos tienen características reducidas para Ta = 60 °C.

Los módulos deben montarse en una carcasa apropiada con un grado de protección mínimo de IP54 según EN 60529, considerando las condiciones ambientales en las que se utilizarán los equipos.

Si, en condiciones nominales, la temperatura excede 70 °C en el punto de entrada del cable, o bien 80 °C en el punto de derivación de los conductores, la temperatura realmente medida deberá estar comprendida en el rango de temperatura admisible del cable seleccionado.

Se deberán tomar las medidas necesarias para impedir que se exceda la tensión nominal en más de un 40% a causa de perturbaciones transitorias.

Aprobación C-Tick



El S7-1200 cumple los requisitos de las normas según AS/NZS 2064 (clase A).

Homologación coreana



El sistema de automatización S7-1200 cumple los requisitos de la homologación coreana (KC Mark). Este sistema ha sido definido como equipo clase A para aplicaciones industriales y no está previsto para uso doméstico.

Aprobación marina

Los productos S7-1200 se someten con regularidad a pruebas para obtener homologaciones especiales para aplicaciones y mercados específicos. Contacte con el representante de Siemens más próximo para obtener una lista de las homologaciones actuales con las referencias respectivas.

Sociedades de clasificación:

- ABS (American Bureau of Shipping)
- BV (Bureau Veritas)
- DNV (Det Norske Veritas)
- GL (Germanischer Lloyd)
- LRS (Lloyds Register of Shipping)
- Class NK (Nippon Kaiji Kyokai)

Entornos industriales

El sistema de automatización S7-1200 está diseñado para ser utilizado en entornos industriales.

Tabla A- 1 Entornos industriales

Campo de aplicación	Requisitos respecto a la emisión de interferencias	Requisitos respecto a la inmunidad a interferencias
Industrial	EN 61000-6-4:2007	EN 61000-6-2:2005

Compatibilidad electromagnética

La compatibilidad electromagnética (también conocida por sus siglas CEM o EMC) es la capacidad de un dispositivo eléctrico para funcionar de forma satisfactoria en un entorno electromagnético sin causar interferencias electromagnéticas (EMI) sobre otros dispositivos eléctricos de ese entorno.

Tabla A- 2 Inmunidad según EN 61000-6-2

Compatibilidad electromagnética - Inmunidad según EN 61000-6-2	
EN 61000-4-2 Descargas electrostáticas	Descarga en el aire de 8 kV en todas las superficies Descarga en contactos de 6 kV en las superficies conductoras expuestas
EN 61000-4-3 Prueba de inmunidad de campos electromagnéticos de radiofrecuencia radiada	80 a 1000 MHz, 10 V/m, 80% AM a 1 kHz 1,4 a 2,0 GHz, 3 V/m, 80% AM a 1 kHz 2,0 a 2,7 GHz, 1 V/m, 80% AM a 1 kHz
EN 61000-4-4 Transitorios eléctricos rápidos	2 kV, 5 kHz con red de conexión a la alimentación AC y DC 2 kV, 5 kHz con borne de conexión a las E/S
EN 6100-4-5 Inmunidad a ondas de choque	Sistemas AC - 2 kV en modo común, 1kV en modo diferencial Sistemas DC - 2 kV en modo común, 1kV en modo diferencial Para los sistemas DC (señales E/S, sistemas de alimentación DC) se requiere protección externa.
EN 61000-4-6 Perturbaciones conducidas	150 kHz a 80 MHz, 10 V RMS, 80% AM a 1kHz
EN 61000-4-11 Inmunidad a cortes e interrupciones breves	Sistemas AC 0% durante 1 ciclo, 40% durante 12 ciclos y 70% durante 30 ciclos a 60 Hz

Tabla A- 3 Emisiones conducidas y radiadas según EN 61000-6-4

Compatibilidad electromagnética - Emisiones conducidas y radiadas según EN 61000-6-4		
Emisiones conducidas EN 55011, clase A, grupo 1	De 0,15 MHz a 0,5 MHz	<79dB (µV) casi cresta; <66 dB (µV) valor medio
	De 0,5 MHz a 5 MHz	<73dB (µV) casi cresta; <60 dB (µV) valor medio
	De 5 MHz a 30 MHz	<73dB (µV) casi cresta; <60 dB (µV) valor medio
Emisiones radiadas EN 55011, clase A, grupo 1	De 30 MHz a 230 MHz	<40dB (µV/m) casi cresta; medido a 10 m
	De 230 MHz a 1 GHz	<47dB (µV/m) casi cresta; medido a 10 m

Condiciones ambientales

Tabla A- 4 Transporte y almacenamiento

Condiciones ambientales - Transporte y almacenamiento	
EN 60068-2-2, ensayo Bb, calor seco y EN 60068-2-1, ensayo Ab, frío	-40 °C a +70 °C
EN 60068230, ensayo Dd, calor húmedo	25° C a 55° C, 95% de humedad
EN 60068-2-14, ensayo Na, choque de temperatura	40 °C a +70 °C, tiempo de permanencia 3 horas, 5 ciclos

Condiciones ambientales - Transporte y almacenamiento	
EN 60068232, caída libre	0,3 m, 5 veces, embalado para embarque
Presión atmosférica	1080 a 660h Pa (equivale a una altitud de -1000 a 3500m)

Tabla A- 5 Condiciones de manejo

Condiciones ambientales - Funcionamiento	
Rango de temperatura ambiente (aire de entrada 25 mm bajo la unidad)	-20 °C a 60 °C en montaje horizontal 20 °C a 50 °C en montaje vertical 95% de humedad sin condensación A no ser que se especifique otra cosa
Presión atmosférica	1080 a 795 hPa (equivale a una altitud de -1000 a 2000m)
Concentración de contaminantes	SO ₂ : < 0,5 ppm; H ₂ S: < 0,1 ppm; RH < 60% sin condensación
EN 60068214, ensayo Nb, cambio de temperatura	5° C a 55° C, 3° C/minuto
EN 60068227, choque mecánico	15 G, 11 ms impulso, 6 choques en c/u de 3 ejes
EN 6006826, vibración sinusoidal	Montaje en perfil DIN: 3,5 mm de 5 a 9 Hz, 1G de 9 a 150 Hz Montaje en panel: 7,0 mm de 5 a 9 Hz, 2G de 9 a 150 Hz 10 barridos por eje, 1 octava por minuto

ATENCIÓN
En sistemas que deben arrancar entre -20 °C y 0 °C, el programa de usuario debe retrasar la energización de las salidas durante 10 segundos tras el encendido.

Tabla A- 6 Prueba de aislamiento a muy alta tensión

Prueba de aislamiento a muy alta tensión	
Circuitos nominales de 24 V/5 V	520 V DC (ensayo de tipo de límites de aislamiento óptico)
Circuitos de 115/230 V a tierra	1500 V AC
Circuitos de 115/230 V a circuitos de 115/230 V	1500 V AC
Circuitos de 115 V/230V a circuitos de 24 V/5 V	1500 V AC (ensayo de tipo 3000 V AC / 4242 V DC)
Puerto Ethernet a circuitos de 24 V/5 V y masa ¹	1500 V AC (solo ensayo de tipo)

¹ El aislamiento del puerto Ethernet está diseñado para limitar el riesgo por tensiones peligrosas durante defectos de red de corta duración. No es conforme con los requisitos de seguridad usuales para el aislamiento a tensión de línea a frecuencia industrial.

Clase de protección

- Clase de protección II según EN 61131-2 (el conductor protector no se requiere)

Grado de protección

- Protección mecánica IP20, EN 60529
- Protege los dedos contra el contacto con alta tensión, según ensayos realizados con sondas estándar. Se requiere protección externa contra polvo, impurezas, agua y objetos extraños de < 12,5mm de diámetro.

Tensiones nominales

Tabla A- 7 Tensiones nominales

Tensión nominal	Tolerancia
24 V DC	20,4 V DC a 28,8 V DC 22,0 V DC a 28,8 V DC para temperatura ambiente entre 20 °C y 0 °C
120/230 V AC	85 V AC a 264 V AC, 47 a 63 Hz

ATENCIÓN

Cuando un contacto mecánico aplica tensión a una CPU S7-1200, o bien a un módulo de señales digitales, envía una señal "1" a las salidas digitales durante aprox. 50 microsegundos. Ello podría causar un funcionamiento inesperado de los equipos o del proceso, lo que podría ocasionar la muerte o lesiones graves al personal y/o daños al equipo. Eso debe considerarse, especialmente si desea utilizar dispositivos que reaccionen a impulsos de breve duración.

Protección contra inversión de polaridad

Se suministra circuitería de protección contra inversión de polaridad en cada par de bornes de alimentación de +24 V DC o de alimentación de entrada de usuario para CPUs, módulos de señales (SM) y Signal Boards (SB). No obstante, el sistema puede sufrir daños si se cablean pares de bornes distintos en polaridades opuestas.

Algunos bornes de entrada de la alimentación de 24 V DC del sistema S7-1200 están interconectados, teniendo un circuito lógico común que conecta varios bornes M. Por ejemplo, los circuitos siguientes están interconectados si no tienen aislamiento galvánico según las hojas de datos técnicos: la fuente de alimentación de 24 V DC de la CPU, la entrada de alimentación de la bobina de relé de un SM, o bien la fuente de alimentación de una entrada analógica no aislada. Todos los bornes M sin aislamiento galvánico deben conectarse al mismo potencial de referencia externo.

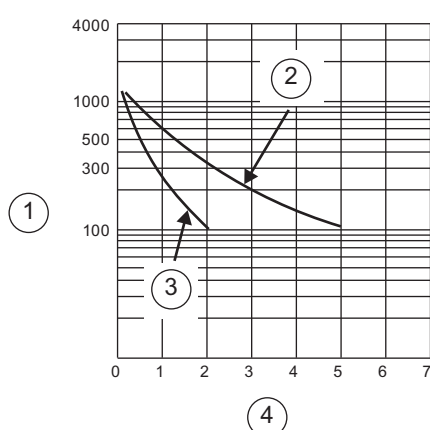
<p>! ADVERTENCIA</p> <p>Si los bornes M sin aislamiento galvánico se conectan a diferentes potenciales de referencia, circularán corrientes indeseadas que podrían averiar o causar reacciones inesperadas en el PLC y los equipos conectados.</p> <p>Si no se cumplen estas directrices, es posible que se produzcan averías o reacciones inesperadas que podrían causar la muerte, lesiones corporales graves y/o daños materiales.</p> <p>Asegúrese que todos los bornes M sin aislamiento galvánico de un sistema S7-1200 están conectados al mismo potencial de referencia.</p>

Salidas DC

No se suministra circuitería de protección contra cortocircuitos en las salidas DC de las CPUs, módulos de señales (SM) y Signal Boards (SB).

Vida útil de los relés

La figura siguiente muestra los datos típicos de rendimiento de los relés suministrados por el comercio especializado. El rendimiento real puede variar dependiendo de la aplicación. Un circuito de protección externo adaptado a la carga permite prolongar la vida útil de los contactos.



- ① Vida útil (x 10³ operaciones)
- ② 250 V AC carga resistiva
30 V DC carga resistiva
- ③ 250 V AC de carga inductiva (p.f.=0,4)
30 V DC de carga inductiva (L/R=7 ms)
- ④ Intensidad normal de servicio (A)

A.2 CPU 1211C

A.2.1 Especificaciones generales y propiedades

Tabla A- 8 Especificaciones generales

Datos técnicos	CPU 1211C AC/DC/relé	CPU 1211C DC/DC/relé	CPU 1211C DC/DC/DC
Referencia	6ES7 211-1BE31-0XB0	6ES7 211-1HE31-0XB0	6ES7 211-1AE31-0XB0
Dimensiones A x A x P (mm)	90 x 100 x 75	90 x 100 x 75	90 x 100 x 75
Peso de envío	420 gramos	380 gramos	370 gramos
Disipación de potencia	10 W	8 W	8 W
Intensidad disponible (bus CM)	750 mA máx. (5 V DC)	750 mA máx. (5 V DC)	750 mA máx. (5 V DC)
Intensidad disponible (24 V DC)	300 mA máx. (alimentación de sensores)	300 mA máx. (alimentación de sensores)	300 mA máx. (alimentación de sensores)
Consumo de corriente de las entradas digitales (24 V DC)	4 mA/entrada utilizada	4 mA/entrada utilizada	4 mA/entrada utilizada

Tabla A- 9 Propiedades de la CPU

Datos técnicos	Descripción	
Memoria de usuario ¹	Trabajo	30 KB
	Carga	1 MB, interna, ampliable hasta tamaño de tarjeta SD
	Remanente	10 KB
E/S digitales integradas	6 entradas/4 salidas	
E/S analógicas integradas	2 entradas	
Tamaño de la memoria imagen de proceso	1024 bytes de entradas (I)/1024 bytes de salidas (Q)	
Área de marcas (M)	4096 bytes	
Memoria temporal (local)	<ul style="list-style-type: none"> • 16 KB para arranque y ciclo (incluyendo los FB y FC asociados) • 4 KB para eventos de alarma estándar, incluyendo FBs y FCs • 4 KB para eventos de alarma de error, incluyendo FBs y FCs 	
Ampliación con módulos de señales	Ninguno	
Ampliación con SB, CB o BB	1 máx.	
Ampliación con módulos de comunicación	3 CMs máx.	
Contadores rápidos	3 E/S incorporadas, 5 con Signal Board; consulte la tabla Asignaciones de entradas de HSC para la CPU 1211C (Página 355) <ul style="list-style-type: none"> • Fase simple: 3 a 100 kHz, SB: 2 a 30 kHz • Fase en cuadratura: 3 a 80 kHz, SB: 2 a 20 kHz 	
Generadores de impulsos ²	4	
Entradas de captura de impulsos	6	

Datos técnicos	Descripción
Alarmas de retardo/cíclicas	4 en total con resolución de 1 ms
Alarmas de flanco	6 ascendentes y 6 descendentes (10 y 10 con Signal Board opcional)
Memory Card	SIMATIC Memory Card (opcional)
Precisión del reloj en tiempo real	+/- 60 segundos/mes
Tiempo de respaldo del reloj de tiempo real	20 días típ./12 días mín. a 40 °C (condensador de alto rendimiento sin mantenimiento)

- ¹ El tamaño del programa de usuario, los datos y la configuración están limitados por la memoria de carga y memoria de trabajo disponibles de la CPU. No hay un límite determinado para el número de bloques OB, FC, FB y DB soportados o en lo referente al tamaño de un bloque específico. El único límite está sujeto al tamaño total de la memoria.
- ² Para modelos de CPU con salidas de relé se debe instalar una Signal Board (SB) digital para emplear las salidas de impulsos.

Tabla A- 10 Rendimiento

Tipo de instrucción	Velocidad de ejecución
Booleano	0,08 µs/instrucción
Transferir palabra	1,7 µs/instrucción
Funciones matemáticas con números reales	2,3 µs/instrucción

A.2.2 Temporizadores, contadores y bloques lógicos soportados por la CPU 1211C

Tabla A- 11 Bloques, temporizadores y contadores soportados por la CPU 1211C

Elemento	Descripción	
Bloques	Tipo	OB, FB, FC, DB
	Tamaño	30 KB
	Cantidad	Un total de hasta 1024 bloques (OB + FB + FC + DB)
	Rango de direcciones para FB, FC y DB	De 1 a 65535 (p. ej. del FB 1 al FB 65535)
	Profundidad de anidamiento	16 del OB de arranque o de ciclo; 4 del OB de alarma de retardo, alarma horaria, alarma cíclica, alarma de proceso, alarma de error de tiempo o alarma de diagnóstico
	Observar	Se puede observar a la vez el estado de 2 bloques lógicos
OB	Ciclo del programa	Múltiple: OB 1, de OB 200 a OB 65535
	Arranque	Múltiple: OB 100, de OB 200 a OB 65535
	Alarmas de retardo y alarmas cíclicas	4 ¹ (1 por evento): De OB 200 a OB 65535
	Alarmas de proceso (flancos y HSC)	50 (1 por evento): De OB 200 a OB 65535
	Alarmas de error de tiempo	1: OB 80
	Alarmas de error de diagnóstico	1: OB 82

Elemento		Descripción
Temporizadores	Tipo	CEI
	Cantidad	Sólo limitada por el tamaño de la memoria
	Almacenamiento	Estructura en DB, 16 bytes por temporizador
Contadores	Tipo	CEI
	Cantidad	Sólo limitada por el tamaño de la memoria
	Almacenamiento	Estructura en DB, tamaño dependiente del tipo de conteo <ul style="list-style-type: none"> • SInt, USInt: 3 bytes • Int, UInt: 6 bytes • DInt, UDInt: 12 bytes

¹ Las alarmas de retardo y cíclicas usan los mismos recursos en la CPU. Sólo se puede contar con un total máximo de 4 de estas alarmas (suma de alarmas de retardo y cíclicas). No puede haber 4 alarmas de retardo y 4 alarmas cíclicas.

Tabla A- 12 Comunicación

Datos técnicos	Descripción
Número de puertos	1
Tipo	Ethernet
Dispositivo HMI ¹	3
Programadora (PG)	1
Conexiones	<ul style="list-style-type: none"> • 8 para Open User Communication (activa o pasiva): TSEND_C, TRCV_C, TCON, TDISCON, TSEND y TRCV • 3 para comunicaciones S7 GET/PUT (CPU a CPU) de servidor • 8 para comunicaciones S7 GET/PUT (CPU a CPU) de cliente
Transferencia de datos	10/100 Mb/s
Aislamiento (señal externa a lógica del PLC)	Aislado por transformador, 1500 V AC, sólo para seguridad frente a defectos breves
Tipo de cable	CAT5e apantallado

¹ La CPU proporciona conexiones HMI dedicadas que admiten un máximo de 3 dispositivos HMI. Se pueden tener hasta 2 SIMATIC Comfort Panel. El número total de HMI depende del tipo de paneles HMI indicados en la configuración. Así, por ejemplo, pueden conectarse a la CPU hasta tres SIMATIC Basic Panels o bien hasta dos SIMATIC Comfort Panel con un Basic Panel.

Tabla A- 13 Alimentación eléctrica

Datos técnicos	CPU 1211C AC/DC/relé	CPU 1211C DC/DC/relé	CPU 1211C DC/DC/DC
Rango de tensión	85 a 264 V AC	20,4 V DC a 28,8 V DC 22,0 V DC a 28,8 V DC para temperatura ambiente de -20 °C a 0 °C	
Frecuencia de línea	47 a 63 Hz	--	--
Intensidad de entrada	CPU sólo a carga máx. 60 mA a 120 V AC 30 mA a 240 V AC	300 mA a 24 V DC	300 mA a 24 V DC

Datos técnicos	CPU 1211C AC/DC/relé	CPU 1211C DC/DC/relé	CPU 1211C DC/DC/DC
CPU con todos los accesorios de ampliación a carga máx.	180 mA a 120 V AC 90 mA a 240 V AC	900 mA a 24 V DC	900 mA a 24 V DC
Corriente de irrupción (máx.)	20 A a 264 V AC	12 A a 28,8 V DC	12 A a 28,8 V DC
Aislamiento (alimentación de entrada a lógica)	1500 V AC	Sin aislamiento	Sin aislamiento
Corriente de fuga a tierra, línea AC a tierra funcional	0,5 mA máx.	--	--
Tiempo de mantenimiento (pérdida de alimentación)	20 ms a 120 V AC 80 ms a 240 V AC	10 ms a 24 V DC	10 ms a 24 V DC
Fusible interno, no reemplazable por el usuario	3 A, 250 V, de acción lenta	3 A, 250 V, de acción lenta	3 A, 250 V, de acción lenta

Tabla A- 14 Alimentación de sensores

Datos técnicos	CPU 1211C AC/DC/relé	CPU 1211C DC/DC/relé	CPU 1211C DC/DC/DC
Rango de tensión	20,4 a 28,8 V DC	L+ menos 4 V DC mín. L+ menos 5 V DC mín. para temperatura ambiente de -20 °C a 0 °C	
Intensidad de salida nominal (máx.)	300 mA (protegido contra cortocircuito)	300 mA (protegido contra cortocircuito)	300 mA (protegido contra cortocircuito)
Ruido de rizado máx. (<10 MHz)	< 1 V de pico a pico	Igual a la línea de entrada	Igual a la línea de entrada
Aislamiento (lógica de la CPU a alimentación de sensores)	Sin aislamiento	Sin aislamiento	Sin aislamiento

A.2.3 Entradas y salidas digitales

Tabla A- 15 Entradas digitales

Datos técnicos	CPU 1211C AC/DC/relé, CPU 1211C DC/DC/relé y CPU 1211C DC/DC/DC
Número de entradas	6
Tipo	Sumidero/fuente (tipo 1 IEC sumidero)
Tensión nominal	24 V DC a 4 mA, nominal
Tensión continua admisible	30 V DC, máx.
Sobretensión transitoria	35 V DC durante 0,5 s
Señal 1 lógica (mín.)	15 V DC a 2,5 mA
Señal 0 lógica (máx.)	5 V DC a 1 mA
Aislamiento (campo a lógica)	500 V AC durante 1 minuto
Grupos de aislamiento	1

Datos técnicos

A.2 CPU 1211C

Datos técnicos	CPU 1211C AC/DC/relé, CPU 1211C DC/DC/relé y CPU 1211C DC/DC/DC
Tiempos de filtro	0,2, 0,4, 0,8, 1,6, 3,2, 6,4 y 12,8 ms (seleccionable en grupos de 4)
Frecuencias de entrada de reloj HSC (máx.) (señal 1 lógica = 15 a 26 V DC)	Fase simple: 100 kHz Fase en cuadratura: 80 kHz
Número de entradas ON simultáneamente	6 a 60 °C en horizontal o 50 °C en vertical
Longitud de cable (metros)	500 m apantallado, 300 m no apantallado, 50 m apantallado para entradas HSC

Tabla A- 16 Salidas digitales

Datos técnicos	CPU 1211C AC/DC/relé y CPU 1211C DC/DC/relé	CPU 1211C DC/DC/DC
Número de salidas	4	4
Tipo	Relé, contacto seco	Estado sólido MOSFET (fuente)
Rango de tensión	5 a 30 V DC ó 5 a 250 V AC	20,4 a 28,8 V DC
Señal 1 lógica a intensidad máx.	--	20 V DC mín.
Señal 0 lógica con carga de 10 K Ω	--	0,1 V DC máx.
Intensidad (máx.)	2,0 A	0,5 A
Carga de lámparas	30 W DC / 200 W AC	5 W
Resistencia en estado ON	Máx. 0,2 Ω (si son nuevas)	0,6 Ω máx.
Corriente de fuga por salida	--	10 μ A máx.
Sobrecorriente momentánea	7 A si están cerrados los contactos	8 A durante máx. 100 ms
Protección contra sobrecargas	No	No
Aislamiento (campo a lógica)	1500 V AC durante 1 minuto (entre bobina y contacto) Ninguno (entre bobina y circuito lógico)	500 V AC durante 1 minuto
Resistencia de aislamiento	100 M Ω mín. si son nuevas	--
Aislamiento entre contactos abiertos	750 V AC durante 1 minuto	--
Grupos de aislamiento	1	1
Tensión de bloqueo inductiva	--	L+ menos 48 V DC, disipación de 1 W
Frecuencia máxima de conmutación de relé	1 Hz	--
Retardo de conmutación (Qa.0 a Qa.3)	10 ms máx.	1,0 μ s máx., OFF a ON 3,0 μ s máx., ON a OFF
Frecuencia de salida de tren de impulsos (Qa.0 y Qa.2)	No recomendada ¹	100 KHz máx., 2 Hz mín. ²
Vida útil mecánica (sin carga)	10.000.000 ciclos apertura/cierre	--
Vida útil de los contactos bajo carga nominal	100.000 ciclos apertura/cierre	--
Reacción al cambiar de RUN a STOP	Último valor o valor sustitutivo (valor predeterminado: 0)	Último valor o valor sustitutivo (valor predeterminado: 0)

Datos técnicos	CPU 1211C AC/DC/relé y CPU 1211C DC/DC/relé	CPU 1211C DC/DC/DC
Número de salidas ON simultáneamente	4 a 60 °C en horizontal o 50 °C en vertical	
Longitud de cable (metros)	500 m apantallado, 150 m no apantallado	500 m apantallado, 150 m no apantallado

- ¹ Para modelos de CPU con salidas de relé, se debe instalar una Signal Board (SB) digital para emplear las salidas de impulsos.
- ² En función del receptor de impulsos y del cable, una resistencia de carga adicional (al menos 10% de la intensidad nominal) puede mejorar la calidad de señal de los impulsos y la inmunidad a interferencias.

A.2.4 Entradas analógicas

Tabla A- 17 Entradas analógicas

Datos técnicos	Descripción
Número de entradas	2
Tipo	Tensión (asimétrica)
Rango total	De 0 a 10 V
Rango total (palabra de datos)	0 a 27.648
Rango de sobreimpulso	10,001 a 11,759 V
Rango de sobreimpulso (palabra de datos)	27.649 a 32.511
Rango de desbordamiento	11,760 a 11,852 V
Rango de desbordamiento (palabra de datos)	32.512 a 32.767
Resolución	10 bits
Tensión soportada máxima	35 V DC
Filtrado	Ninguno, débil, medio o fuerte Consulte la tabla de respuesta a un escalón (ms) para las entradas analógicas de la CPU (Página 750).
Supresión de ruido	10, 50 ó 60 Hz
Impedancia	≥100 KΩ
Aislamiento (campo a lógica)	Ninguno
Precisión (25 °C/-20 a 60 °C)	3,0%/3,5% de rango máximo
Longitud de cable (metros)	100 m, par trenzado apantallado

A.2.4.1 Respuesta a un escalón de las entradas analógicas integradas en la CPU

Tabla A- 18 Respuesta a un escalón (ms), 0V a 10V medido a 95%

Selección de filtrado (valor medio de muestreo)	Supresión de frecuencias (tiempo de integración)		
	60 Hz	50 Hz	10 Hz
Ninguno (1 ciclo): Sin media	50 ms	50 ms	100 ms
Débil (4 ciclos): 4 muestreos	60 ms	70 ms	200 ms
Medio (16 ciclos): 16 muestreos	200 ms	240 ms	1150 ms
Fuerte (32 ciclos): 32 muestreos	400 ms	480 ms	2300 ms
Tiempo de muestreo	4,17 ms	5 ms	25 ms

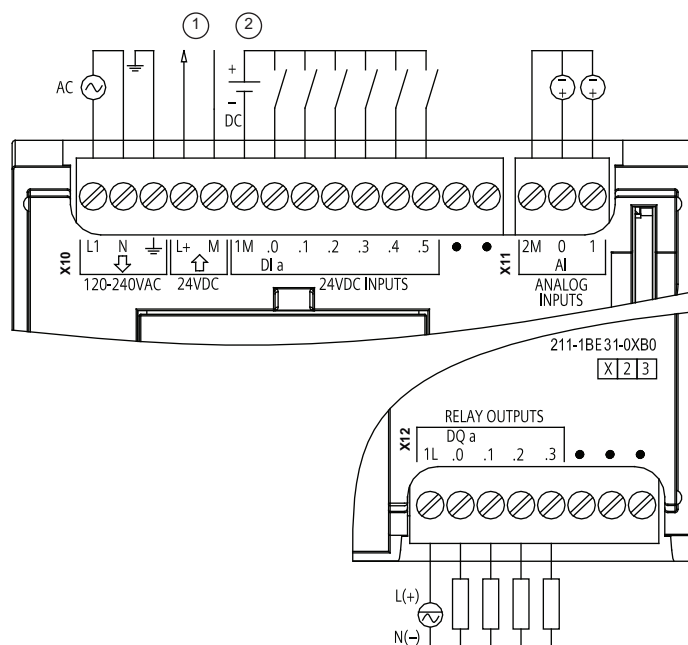
A.2.4.2 Tiempo de muestreo para los puertos analógicos integrados en la CPU

Tabla A- 19 Tiempo de muestreo para las entradas analógicas integradas en la CPU

Supresión de frecuencias (selección del tiempo de integración)	Tiempo de muestreo
60 Hz (16,6 ms)	4,17 ms
50 Hz (20 ms)	5 ms
10 Hz (100 ms)	25 ms

A.2.5 Diagramas de cableado de la CPU 1211C

Tabla A- 20 CPU 1211C AC/DC/relé (6ES7 211-1BE31-0XB0)



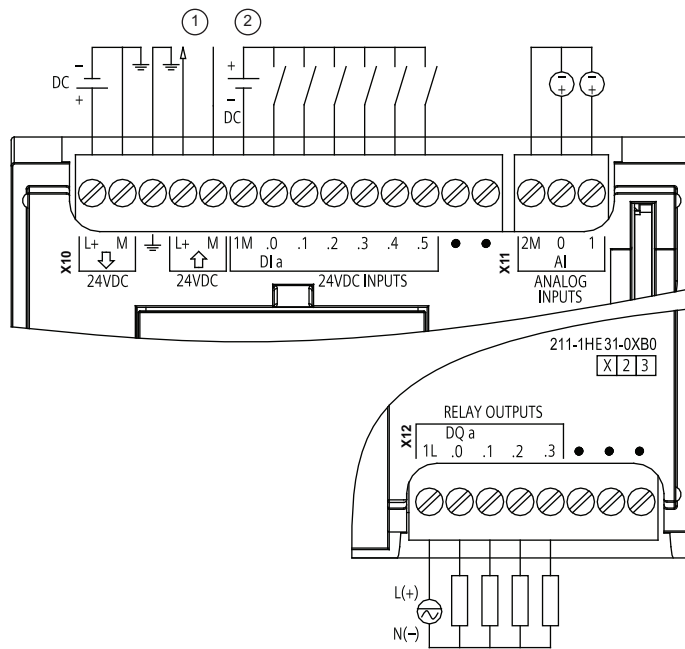
- ① Alimentación de sensores 24 V DC
Para una inmunidad a interferencias adicional, conecte "M" a masa incluso si no se utiliza la alimentación del sensor.
- ② Para entradas en sumidero, conecte "-" a "M" (como se indica).
Para entradas en fuente, conecte "+" a "M".

Nota: Los conectores X11 deben ser de oro. Consulte el anexo C, Piezas de repuesto, para ver la referencia.

Tabla A- 21 Asignación de pines de conectores para CPU 1211C AC/DC/relé (6ES7 211-1BE31-0XB0)

Pin	X10	X11 (oro)	X12
1	L1/120-240 V AC	2 M	1L
2	N/120-240 V AC	AI 0	DQ a.0
3	Tierra funcional	AI 1	DQ a.1
4	Salida sensor L+/24 V DC	--	DQ a.2
5	Salida sensor M/24 V DC	--	DQ a.3
6	1M	--	Sin conexión
7	DI a.0	--	Sin conexión
8	DI a.1	--	Sin conexión
9	DI a.2	--	--
10	DI a.3	--	--
11	DI a.4	--	--
12	DI a.5	--	--
13	Sin conexión	--	--
14	Sin conexión	--	--

Tabla A- 22 CPU 1211C DC/DC/relé (6ES7 211-1HE31-0XB0)



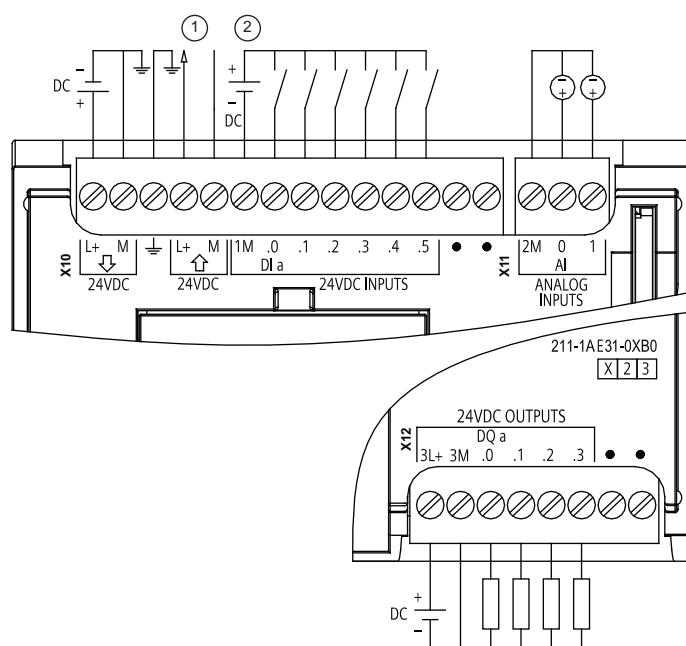
- ① Alimentación de sensores 24 V DC
Para una inmunidad a interferencias adicional, conecte "M" a masa incluso si no se utiliza la alimentación de sensores.
- ② Para entradas en sumidero, conecte "-" a "M" (como se indica).
Para entradas en fuente, conecte "+" a "M".

Nota: Los conectores X11 deben ser de oro. Consulte el anexo C, Piezas de repuesto, para ver la referencia.

Tabla A- 23 Asignación de pines de conectores para CPU 1211C DC/DC/relé (6ES7 211-1HE31-0XB0)

Pin	X10	X11 (oro)	X12
1	L+/24 V DC	2 M	1L
2	M/24 V DC	AI 0	DQ a.0
3	Tierra funcional	AI 1	DQ a.1
4	Salida sensor L+/24 V DC	--	DQ a.2
5	Salida sensor M/24 V DC	--	DQ a.3
6	1M	--	Sin conexión
7	DI a.0	--	Sin conexión
8	DI a.1	--	Sin conexión
9	DI a.2	--	--
10	DI a.3	--	--
11	DI a.4	--	--
12	DI a.5	--	--
13	Sin conexión	--	--
14	Sin conexión	--	--

Tabla A- 24 CPU 1211C DC/DC/DC (6ES7 211-1AE31-0XB0)



- ① Alimentación de sensores 24 V DC
Para una inmunidad a interferencias adicional, conecte "M" a masa incluso si no se utiliza la alimentación de sensores.
- ② Para entradas en sumidero, conecte "-" a "M" (como se indica).
Para entradas en fuente, conecte "+" a "M".

Nota: Los conectores X11 deben ser de oro. Consulte el anexo C, Piezas de repuesto, para ver la referencia.

Tabla A- 25 Asignación de pines de conectores para CPU 1211C DC/DC/DC (6ES7 211-1AE31-0XB0)

Pin	X10	X11 (oro)	X12
1	L+/24 V DC	2 M	3L+
2	M/24 V DC	AI 0	3M
3	Tierra funcional	AI 1	DQ a.0
4	Salida sensor L+/24 V DC	--	DQ a.1
5	Salida sensor M/24 V DC	--	DQ a.2
6	1M	--	DQ a.3
7	DI a.0	--	Sin conexión
8	DI a.1	--	Sin conexión
9	DI a.2	--	--
10	DI a.3	--	--
11	DI a.4	--	--
12	DI a.5	--	--
13	Sin conexión	--	--
14	Sin conexión	--	--

Nota

Las entradas analógicas que no se utilicen deben cortocircuitarse.

A.3 CPU 1212C

A.3.1 Especificaciones generales y propiedades

Tabla A- 26 General

Datos técnicos	CPU 1212C AC/DC/relé	CPU 1212C DC/DC/relé	CPU 1212C DC/DC/DC
Referencia	6ES7 212-1BE31-0XB0	6ES7 212-1HE31-0XB0	6ES7 212-1AE31-0XB0
Dimensiones A x A x P (mm)	90 x 100 x 75	90 x 100 x 75	90 x 100 x 75
Peso de envío	425 gramos	385 gramos	370 gramos
Disipación de potencia	11 W	9 W	9 W
Intensidad disponible (SM y bus CM)	1000 mA máx. (5 V DC)	1000 mA máx. (5 V DC)	1000 mA máx. (5 V DC)
Intensidad disponible (24 V DC)	300 mA máx. (alimentación de sensores)	300 mA máx. (alimentación de sensores)	300 mA máx. (alimentación de sensores)
Consumo de corriente de las entradas digitales (24 V DC)	4 mA/entrada utilizada	4 mA/entrada utilizada	4 mA/entrada utilizada

Tabla A- 27 Propiedades de la CPU

Datos técnicos	Descripción	
Memoria de usuario ¹	Trabajo	50 KB
	Carga	1 MB, interna, ampliable hasta tamaño de tarjeta SD
	Remanente	10 KB
E/S digitales integradas	8 entradas/6 salidas	
E/S analógicas integradas	2 entradas	
Tamaño de la memoria imagen de proceso	1024 bytes de entradas (I)/1024 bytes de salidas (Q)	
Área de marcas (M)	4096 bytes	
Memoria temporal (local)	<ul style="list-style-type: none"> • 16 KB para arranque y ciclo (incluyendo los FB y FC asociados) • 4 KB para eventos de alarma estándar, incluyendo FBs y FCs • 4 KB para eventos de alarma de error, incluyendo FBs y FCs 	
Ampliación con módulos de señales	2 SM máx.	

Datos técnicos	Descripción
Ampliación con SB, CB o BB	1 máx.
Ampliación con módulos de comunicación	3 CMs máx.
Contadores rápidos	5 E/S incorporadas, 6 con Signal Board; consulte la tabla Asignaciones de entradas de HSC para la CPU 1212C (Página 355) <ul style="list-style-type: none"> • Fase simple: 3 a 100 kHz y 1 a 30 kHz de frecuencia de reloj, SB: 2 a 30 kHz • Fase en cuadratura: 3 a 80 kHz y 1 a 20 kHz de frecuencia de reloj, SB: 2 a 20 kHz
Generadores de impulsos ²	4
Entradas de captura de impulsos	8
Alarmas de retardo/cíclicas	4 en total con resolución de 1 ms
Alarmas de flanco	8 ascendentes y 8 descendentes (12 y 12 con Signal Board opcional)
Memory Card	SIMATIC Memory Card (opcional)
Precisión del reloj en tiempo real	+/- 60 segundos/mes
Tiempo de respaldo del reloj de tiempo real	20 días típ./12 días mín. a 40 °C (condensador de alto rendimiento sin mantenimiento)

¹ El tamaño del programa de usuario, los datos y la configuración están limitados por la memoria de carga y memoria de trabajo disponibles de la CPU. No hay un límite determinado para el número de bloques OB, FC, FB y DB soportados o en lo referente al tamaño de un bloque específico. El único límite está sujeto al tamaño total de la memoria.

² Para modelos de CPU con salidas de relé se debe instalar una Signal Board (SB) digital para emplear las salidas de impulsos.

Tabla A- 28 Rendimiento

Tipo de instrucción	Velocidad de ejecución
Booleano	0,08 µs/instrucción
Transferir palabra	1,7 µs/instrucción
Funciones matemáticas con números reales	2,3 µs/instrucción

A.3.2 Temporizadores, contadores y bloques lógicos soportados por la CPU 1212C

Tabla A- 29 Bloques, temporizadores y contadores soportados por la CPU 1212C

Elemento	Descripción	
Bloques	Tipo	OB, FB, FC, DB
	Tamaño	50 KB
	Cantidad	Un total de hasta 1024 bloques (OB + FB + FC + DB)
	Rango de direcciones para FB, FC y DB	De 1 a 65535 (p. ej. del FB 1 al FB 65535)

Elemento	Descripción	
OB	Profundidad de anidamiento	16 del OB de arranque o de ciclo; 4 del OB de alarma de retardo, alarma horaria, alarma cíclica, alarma de proceso, alarma de error de tiempo o alarma de diagnóstico
	Observar	Se puede observar a la vez el estado de 2 bloques lógicos
	Ciclo del programa	Múltiple: OB 1, de OB 200 a OB 65535
	Arranque	Múltiple: OB 100, de OB 200 a OB 65535
	Alarmas de retardo y alarmas cíclicas	4 ¹ (1 por evento): De OB 200 a OB 65535
	Alarmas de proceso (flancos y HSC)	50 (1 por evento): De OB 200 a OB 65535
	Alarmas de error de tiempo	1: OB 80
Temporizadores	Alarmas de error de diagnóstico	1: OB 82
	Tipo	CEI
	Cantidad	Sólo limitada por el tamaño de la memoria
Contadores	Almacenamiento	Estructura en DB, 16 bytes por temporizador
	Tipo	CEI
	Cantidad	Sólo limitada por el tamaño de la memoria
	Almacenamiento	Estructura en DB, tamaño dependiente del tipo de contaje <ul style="list-style-type: none"> • SInt, USInt: 3 bytes • Int, UInt: 6 bytes • DInt, UDInt: 12 bytes

¹ Las alarmas de retardo y cíclicas usan los mismos recursos en la CPU. Sólo se puede contar con un total máximo de 4 de estas alarmas (suma de alarmas de retardo y cíclicas). No puede haber 4 alarmas de retardo y 4 alarmas cíclicas.

Tabla A- 30 Comunicación

Datos técnicos	Descripción
Número de puertos	1
Tipo	Ethernet
Dispositivo HMI ¹	3
Programadora (PG)	1
Conexiones	<ul style="list-style-type: none"> • 8 para Open User Communication (activa o pasiva): TSEND_C, TRCV_C, TCON, TDISCON, TSEND y TRCV • 3 para comunicaciones S7 GET/PUT (CPU a CPU) de servidor • 8 para comunicaciones S7 GET/PUT (CPU a CPU) de cliente
Transferencia de datos	10/100 Mb/s

Datos técnicos	Descripción
Aislamiento (señal externa a lógica del PLC)	Aislado por transformador, 1500 V AC, sólo para seguridad frente a defectos breves
Tipo de cable	CAT5e apantallado

¹ La CPU proporciona conexiones HMI dedicadas que admiten un máximo de 3 dispositivos HMI. Se pueden tener hasta 2 SIMATIC Comfort Panel. El número total de HMI depende del tipo de paneles HMI indicados en la configuración. Así, por ejemplo, pueden conectarse a la CPU hasta tres SIMATIC Basic Panels o bien hasta dos SIMATIC Comfort Panel con un Basic Panel.

Tabla A- 31 Alimentación eléctrica

Datos técnicos	CPU 1212C AC/DC/relé	CPU 1212C DC/DC/relé	CPU 1212C DC/DC/DC
Rango de tensión	85 a 264 V AC	20,4 V DC a 28,8 V DC 22,0 V DC a 28,8 V DC para temperatura ambiente de -20 °C a 0 °C	
Frecuencia de línea	47 a 63 Hz	--	--
Intensidad de entrada (carga máx.)	sólo CPU	80 mA a 120 V AC 40 mA a 240 V AC	400 mA a 24 V DC 400 mA a 24 V DC
	CPU con todos los accesorios de ampliación	240 mA a 120 V AC 120 mA a 240 V AC	1200 mA a 24 V DC 1200 mA a 24 V DC
Corriente de irrupción (máx.)	20 A a 264 V AC	12 A a 28,8 V DC	12 A a 28,8 V DC
Aislamiento (alimentación de entrada a lógica)	1500 V AC	Sin aislamiento	Sin aislamiento
Corriente de fuga a tierra, línea AC a tierra funcional	0,5 mA máx.	--	--
Tiempo de mantenimiento (pérdida de alimentación)	20 ms a 120 V AC 80 ms a 240 V AC	10 ms a 24 V DC	10 ms a 24 V DC
Fusible interno, no reemplazable por el usuario	3 A, 250 V, de acción lenta	3 A, 250 V, de acción lenta	3 A, 250 V, de acción lenta

Tabla A- 32 Alimentación de sensores

Datos técnicos	CPU 1212C AC/DC/relé	CPU 1212C DC/DC/relé	CPU 1212C DC/DC/DC
Rango de tensión	20,4 a 28,8 V DC	L+ menos 4 V DC mín. L+ menos 5 V DC mín. para temperatura ambiente de -20 °C a 0 °C	
Intensidad de salida nominal (máx.)	300 mA (protegido contra cortocircuito)	300 mA (protegido contra cortocircuito)	300 mA (protegido contra cortocircuito)
Ruido de rizado máx. (<10 MHz)	< 1 V de pico a pico	Igual a la línea de entrada	Igual a la línea de entrada
Aislamiento (lógica de la CPU a alimentación de sensores)	Sin aislamiento	Sin aislamiento	Sin aislamiento

A.3.3 Entradas y salidas digitales

Tabla A- 33 Entradas digitales

Datos técnicos	CPU 1212C AC/DC/relé, DC/DC/relé y DC/DC/DC
Número de entradas	8
Tipo	Sumidero/fuente (tipo 1 IEC sumidero)
Tensión nominal	24 V DC a 4 mA, nominal
Tensión continua admisible	30 V DC, máx.
Sobretensión transitoria	35 V DC durante 0,5 s
Señal 1 lógica (mín.)	15 V DC a 2,5 mA
Señal 0 lógica (máx.)	5 V DC a 1 mA
Aislamiento (campo a lógica)	500 V AC durante 1 minuto
Grupos de aislamiento	1
Tiempos de filtro	0,2, 0,4, 0,8, 1,6, 3,2, 6,4 y 12,8 ms (seleccionable en grupos de 4)
Frecuencias de entrada de reloj HSC (máx.) (señal 1 lógica = 15 a 26 V DC)	Fase simple: 100 KHz (la.0 a la.5) y 30 KHz (la.6 a la.7) Fase en cuadratura: 80 KHz (la.0 a la.5) y 20 KHz (la.6 a la.7)
Número de entradas ON simultáneamente	8 a 60 °C en horizontal o 50 °C en vertical
Longitud de cable (metros)	500 m apantallado, 300 m no apantallado, 50 m apantallado para entradas HSC

Tabla A- 34 Salidas digitales

Datos técnicos	CPU 1212C AC/DC/relé y DC/DC/relé	CPU 1212C DC/DC/DC
Número de salidas	6	6
Tipo	Relé, contacto seco	Estado sólido MOSFET (fuente)
Rango de tensión	5 a 30 V DC ó 5 a 250 V AC	20,4 a 28,8 V DC
Señal 1 lógica a intensidad máx.	--	20 V DC mín.
Señal 0 lógica con carga de 10 K Ω	--	0,1 V DC máx.
Intensidad (máx.)	2,0 A	0,5 A
Carga de lámparas	30 W DC / 200 W AC	5 W
Resistencia en estado ON	Máx. 0,2 Ω (si son nuevas)	0,6 Ω máx.
Corriente de fuga por salida	--	10 μ A máx.
Sobrecorriente momentánea	7 A si están cerrados los contactos	8 A durante máx. 100 ms
Protección contra sobrecargas	No	No
Aislamiento (campo a lógica)	1500 V AC durante 1 minuto (entre bobina y contacto) Ninguno (entre bobina y circuito lógico)	500 V AC durante 1 minuto
Resistencia de aislamiento	100 M Ω mín. si son nuevas	--
Aislamiento entre contactos abiertos	750 V AC durante 1 minuto	--
Grupos de aislamiento	2	1

Datos técnicos	CPU 1212C AC/DC/relé y DC/DC/relé	CPU 1212C DC/DC/DC
Tensión de bloqueo inductiva	--	L+ menos 48 V DC, disipación de 1 W
Retardo de conmutación (Qa.0 a Qa.3)	10 ms máx.	1,0 µs máx., OFF a ON 3,0 µs máx., ON a OFF
Retardo de conmutación (Qa.4 a Qa.5)	10 ms máx.	50 µs máx., OFF a ON 200 µs máx., ON a OFF
Frecuencia máxima de conmutación de relé	1 Hz	--
Frecuencia de salida de tren de impulsos (Qa.0 y Qa.2)	No recomendada ¹	100 KHz máx., 2 Hz mín. ²
Vida útil mecánica (sin carga)	10 000 000 ciclos apertura/cierre	--
Vida útil de los contactos bajo carga nominal	100 000 ciclos apertura/cierre	--
Reacción al cambiar de RUN a STOP	Último valor o valor sustitutivo (valor predeterminado: 0)	Último valor o valor sustitutivo (valor predeterminado: 0)
Número de salidas ON simultáneamente	6 a 60 °C en horizontal o 50 °C en vertical	
Longitud de cable (metros)	500 m apantallado, 150 m no apantallado	500 m apantallado, 150 m no apantallado

¹ Para modelos de CPU con salidas de relé, se debe instalar una Signal Board (SB) digital para emplear las salidas de impulsos.

² En función del receptor de impulsos y del cable, una resistencia de carga adicional (al menos 10% de la intensidad nominal) puede mejorar la calidad de señal de los impulsos y la inmunidad a interferencias.

A.3.4 Entradas analógicas

Tabla A- 35 Entradas analógicas

Datos técnicos	Descripción
Número de entradas	2
Tipo	Tensión (asimétrica)
Rango total	De 0 a 10 V
Rango total (palabra de datos)	0 a 27.648
Rango de sobreimpulso	10,001 a 11,759 V
Rango de sobreimpulso (palabra de datos)	27.649 a 32.511
Rango de desbordamiento	11,760 a 11,852 V
Rango de desbordamiento (palabra de datos)	32.512 a 32.767
Resolución	10 bits
Tensión soportada máxima	35 V DC
Filtrado	Ninguno, débil, medio o fuerte Consulte la tabla de respuesta a un escalón (ms) para las entradas analógicas de la CPU (Página 760).

Datos técnicos	Descripción
Supresión de ruido	10, 50 ó 60 Hz
Impedancia	≥100 KΩ
Aislamiento (campo a lógica)	Ninguno
Precisión (25 °C/-20 a 60 °C)	3,0%/3,5% de rango máximo
Longitud de cable (metros)	100 m, par trenzado apantallado

A.3.4.1 Respuesta a un escalón de las entradas analógicas integradas en la CPU

Tabla A- 36 Respuesta a un escalón (ms), 0V a 10V medido a 95%

Selección de filtrado (valor medio de muestreo)	Supresión de frecuencias (tiempo de integración)		
	60 Hz	50 Hz	10 Hz
Ninguno (1 ciclo): Sin media	50 ms	50 ms	100 ms
Débil (4 ciclos): 4 muestreos	60 ms	70 ms	200 ms
Medio (16 ciclos): 16 muestreos	200 ms	240 ms	1150 ms
Fuerte (32 ciclos): 32 muestreos	400 ms	480 ms	2300 ms
Tiempo de muestreo	4,17 ms	5 ms	25 ms

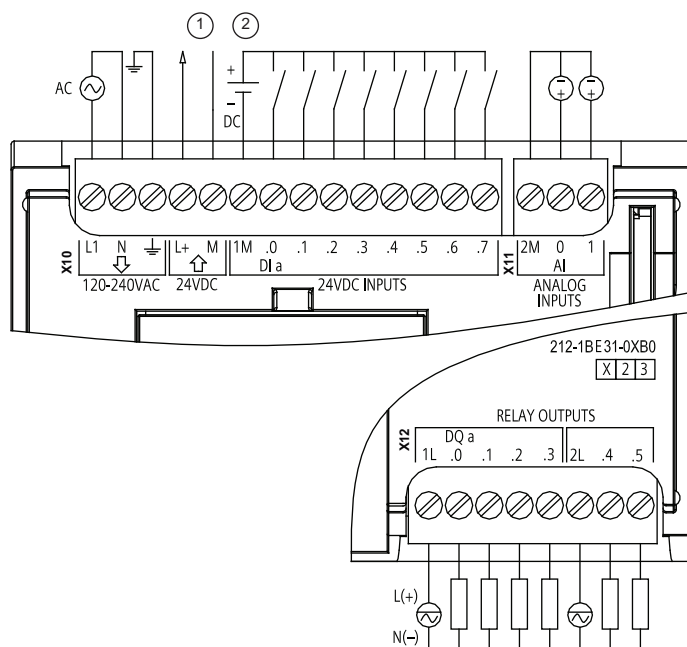
A.3.4.2 Tiempo de muestreo para los puertos analógicos integrados en la CPU

Tabla A- 37 Tiempo de muestreo para las entradas analógicas integradas en la CPU

Supresión de frecuencias (selección del tiempo de integración)	Tiempo de muestreo
60 Hz (16,6 ms)	4,17 ms
50 Hz (20 ms)	5 ms
10 Hz (100 ms)	25 ms

A.3.5 Diagramas de cableado de la CPU 1212C

Tabla A- 38 CPU 1212C AC/DC/relé (6ES7 212-1BE31-0XB0)



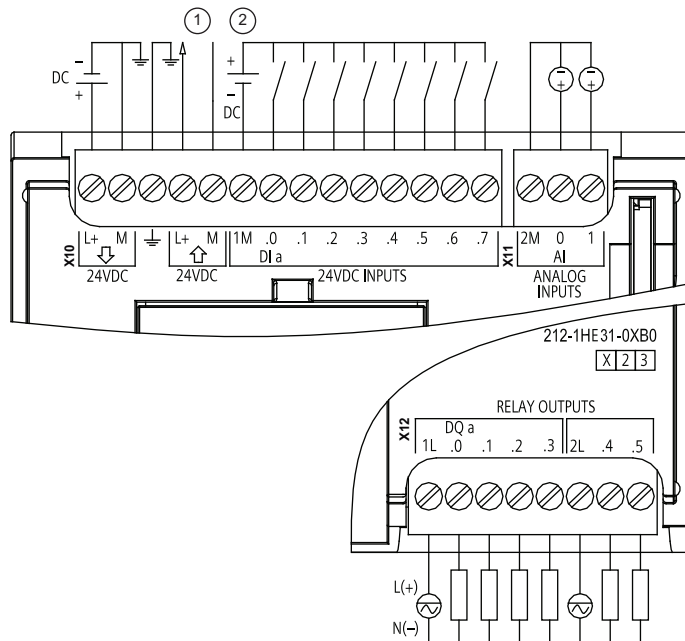
- ① Alimentación de sensores 24 V DC
Para una inmunidad a interferencias adicional, conecte "M" a masa incluso si no se utiliza la alimentación de sensores.
- ② Para entradas en sumidero, conecte "-" a "M" (como se indica).
Para entradas en fuente, conecte "+" a "M".

Nota: Los conectores X11 deben ser de oro. Consulte el anexo C, Piezas de repuesto, para ver la referencia.

Tabla A- 39 Asignación de pines de conectores para CPU 1212C AC/DC/relé (6ES7 212-1BE31-0XB0)

Pin	X10	X11 (oro)	X12
1	L1/120-240 V AC	2 M	1L
2	N/120-240 V AC	AI 0	DQ a.0
3	Tierra funcional	AI 1	DQ a.1
4	Salida sensor L+/24 V DC	--	DQ a.2
5	Salida sensor M/24 V DC	--	DQ a.3
6	1M	--	2L
7	DI a.0	--	DQ a.4
8	DI a.1	--	DQ a.5
9	DI a.2	--	--
10	DI a.3	--	--
11	DI a.4	--	--
12	DI a.5	--	--
13	DI a.6	--	--
14	DI a.7	--	--

Tabla A- 40 CPU 1212C DC/DC/relé (6ES7 212-1HE31-0XB0)



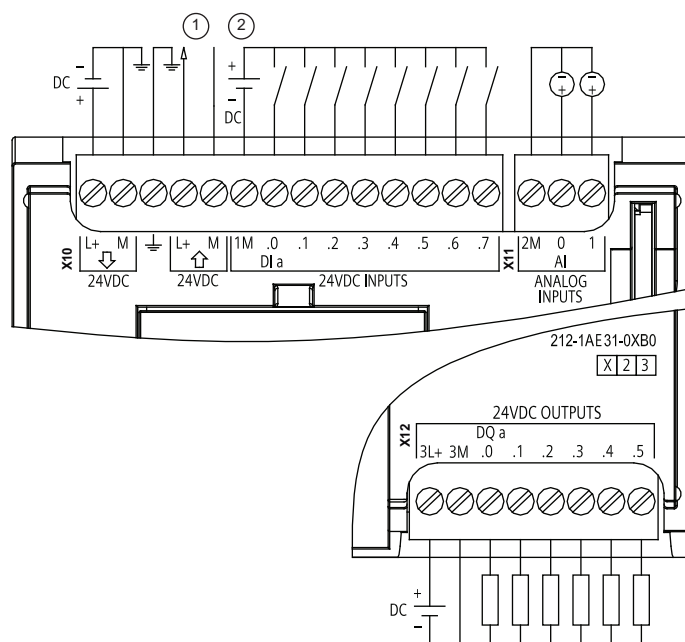
- ① Alimentación de sensores 24 V DC
Para una inmunidad a interferencias adicional, conecte "M" a masa incluso si no se utiliza la alimentación de sensores.
- ② Para entradas en sumidero, conecte "-" a "M" (como se indica).
Para entradas en fuente, conecte "+" a "M".

Nota: Los conectores X11 deben ser de oro. Consulte el anexo C, Piezas de repuesto, para ver la referencia.

Tabla A- 41 Asignación de pines de conectores para CPU 1212C DC/DC/relé (6ES7 212-1HE31-0XB0)

Pin	X10	X11 (oro)	X12
1	L+/24 V DC	2 M	1L
2	M/24 V DC	AI 0	DQ a.0
3	Tierra funcional	AI 1	DQ a.1
4	Salida sensor L+/24 V DC	--	DQ a.2
5	Salida sensor M/24 V DC	--	DQ a.3
6	1M	--	2L
7	DI a.0	--	DQ a.4
8	DI a.1	--	DQ a.5
9	DI a.2	--	--
10	DI a.3	--	--
11	DI a.4	--	--
12	DI a.5	--	--
13	DI a.6	--	--
14	DI a.7	--	--

Tabla A- 42 CPU 1212C DC/DC/DC (6ES7-212-1AE31-0XB0)



- ① Alimentación de sensores 24 V DC
Para una inmunidad a interferencias adicional, conecte "M" a masa incluso si no se utiliza la alimentación de sensores.
- ② Para entradas en sumidero, conecte "-" a "M" (como se indica).
Para entradas en fuente, conecte "+" a "M".

Nota: Los conectores X11 deben ser de oro. Consulte el anexo C, Piezas de repuesto, para ver la referencia.

Tabla A- 43 Asignación de pines de conectores para CPU 1212C DC/DC/DC (6ES7 212-1AE31-0XB0)

Pin	X10	X11 (oro)	X12
1	L+/24 V DC	2 M	3L+
2	M/24 V DC	AI 0	3M
3	Tierra funcional	AI 1	DQ a.0
4	Salida sensor L+/24 V DC	--	DQ a.1
5	Salida sensor M/24 V DC	--	DQ a.2
6	1M	--	DQ a.3
7	DI a.0	--	DQ a.4
8	DI a.1	--	DQ a.5
9	DI a.2	--	--
10	DI a.3	--	--
11	DI a.4	--	--
12	DI a.5	--	--
13	DI a.6	--	--
14	DI a.7	--	--

Nota

Las entradas analógicas que no se utilicen deben cortocircuitarse.

A.4 CPU 1214C

A.4.1 Especificaciones generales y propiedades

Tabla A- 44 General

Datos técnicos	CPU 1214C AC/DC/relé	CPU 1214C DC/DC/relé	CPU 1214C DC/DC/DC
Referencia	6ES7 214-1BG31-0XB0	6ES7 214-1HG31-0XB0	6ES7 214-1AG31-0XB0
Dimensiones A x A x P (mm)	110 x 100 x 75	110 x 100 x 75	110 x 100 x 75
Peso de envío	475 gramos	435 gramos	415 gramos
Disipación de potencia	14 W	12 W	12 W
Intensidad disponible (SM y bus CM)	1600 mA máx. (5 V DC)	1600 mA máx. (5 V DC)	1600 mA máx. (5 V DC)
Intensidad disponible (24 V DC)	400 mA máx. (alimentación de sensores)	400 mA máx. (alimentación de sensores)	400 mA máx. (alimentación de sensores)
Consumo de corriente de las entradas digitales (24 V DC)	4 mA/entrada utilizada	4 mA/entrada utilizada	4 mA/entrada utilizada

Tabla A- 45 Propiedades de la CPU

Datos técnicos	Descripción	
Memoria de usuario ¹	Trabajo	75 KB
	Carga	4 MB, interna, ampliable hasta tamaño de tarjeta SD
	Remanente	10 KB
E/S digitales integradas	14 entradas/10 salidas	
E/S analógicas integradas	2 entradas	
Tamaño de la memoria imagen de proceso	1024 bytes de entradas (I)/1024 bytes de salidas (Q)	
Área de marcas (M)	8192 bytes	
Memoria temporal (local)	<ul style="list-style-type: none"> • 16 KB para arranque y ciclo (incluyendo los FB y FC asociados) • 4 KB para eventos de alarma estándar, incluyendo FBs y FCs • 4 KB para eventos de alarma de error, incluyendo FBs y FCs 	

Datos técnicos	Descripción
Ampliación con módulos de señales	8 SM máx.
Ampliación con SB, CB o BB	1 máx.
Ampliación con módulos de comunicación	3 CMs máx.
Contadores rápidos	6 en total, ver tabla Funcionamiento del contador rápido (Página 355) <ul style="list-style-type: none"> • Fase simple: 3 a 100 kHz y 3 a 30 kHz de frecuencia de reloj • Fase en cuadratura: 3 a 80 kHz y 3 a 20 kHz de frecuencia de reloj
Generadores de impulsos ²	4
Entradas de captura de impulsos	14
Alarmas de retardo/cíclicas	4 en total con resolución de 1 ms
Alarmas de flanco	12 ascendentes y 12 descendentes (14 y 14 con Signal Board opcional)
Memory Card	SIMATIC Memory Card (opcional)
Precisión del reloj en tiempo real	+/- 60 segundos/mes
Tiempo de respaldo del reloj de tiempo real	20 días típ./12 días mín. a 40 °C (condensador de alto rendimiento sin mantenimiento)

¹ El tamaño del programa de usuario, los datos y la configuración están limitados por la memoria de carga y memoria de trabajo disponibles de la CPU. No hay un límite determinado para el número de bloques OB, FC, FB y DB soportados o en lo referente al tamaño de un bloque específico. El único límite está sujeto al tamaño total de la memoria.

² Para modelos de CPU con salidas de relé se debe instalar una Signal Board (SB) digital para emplear las salidas de impulsos.

Tabla A- 46 Rendimiento

Tipo de instrucción	Velocidad de ejecución
Booleano	0,08 µs/instrucción
Transferir palabra	1,7 µs/instrucción
Funciones matemáticas con números reales	2,3 µs/instrucción

A.4.2 Temporizadores, contadores y bloques lógicos soportados por la CPU 1214C

Tabla A- 47 Bloques, temporizadores y contadores soportados por la CPU 1214C

Elemento	Descripción	
Bloques	Tipo	OB, FB, FC, DB
	Tamaño	64 KB
	Cantidad	Un total de hasta 1024 bloques (OB + FB + FC + DB)
	Rango de direcciones para FB, FC y DB	de 1 a 65535 (p. ej. del FB 1 al FB 65535)
	Profundidad de anidamiento	16 del OB de arranque o de ciclo; 4 del OB de alarma de retardo, alarma horaria, alarma cíclica, alarma de proceso, alarma de error de tiempo o alarma de diagnóstico

Elemento	Descripción	
	Observar	Se puede observar a la vez el estado de 2 bloques lógicos
OB	Ciclo del programa	Múltiple: OB 1, de OB 200 a OB 65535
	Arranque	Múltiple: OB 100, de OB 200 a OB 65535
	Alarmas de retardo y alarmas cíclicas	4 ¹ (1 por evento): de OB 200 a OB 65535
	Alarmas de proceso (flancos y HSC)	50 (1 por evento): De OB 200 a OB 65535
	Alarmas de error de tiempo	1: OB 80
	Alarmas de error de diagnóstico	1: OB 82
Temporizadores	Tipo	CEI
	Cantidad	Sólo limitada por el tamaño de la memoria
	Almacenamiento	Estructura en DB, 16 bytes por temporizador
Contadores	Tipo	CEI
	Cantidad	Sólo limitada por el tamaño de la memoria
	Almacenamiento	Estructura en DB, tamaño dependiente del tipo de contaje <ul style="list-style-type: none"> • SInt, USInt: 3 bytes • Int, UInt: 6 bytes • DInt, UDInt: 12 bytes

¹ Las alarmas de retardo y cíclicas usan los mismos recursos en la CPU. Sólo se puede contar con un total máximo de 4 de estas alarmas (suma de alarmas de retardo y cíclicas). No puede haber 4 alarmas de retardo y 4 alarmas cíclicas.

Tabla A- 48 Comunicación

Datos técnicos	Descripción
Número de puertos	1
Tipo	Ethernet
Dispositivo HMI ¹	3
Programadora (PG)	1
Conexiones	<ul style="list-style-type: none"> • 8 para Open User Communication (activa o pasiva): TSEND_C, TRCV_C, TCON, TDISCON, TSEND y TRCV • 3 para comunicaciones S7 GET/PUT (CPU a CPU) de servidor • 8 para comunicaciones S7 GET/PUT (CPU a CPU) de cliente
Transferencia de datos	10/100 Mb/s
Aislamiento (señal externa a lógica del PLC)	Aislado por transformador, 1500 V AC, sólo para seguridad frente a defectos breves
Tipo de cable	CAT5e apantallado

¹ La CPU proporciona conexiones HMI dedicadas que admiten un máximo de 3 dispositivos HMI. Se pueden tener hasta 2 SIMATIC Comfort Panel. El número total de HMI depende del tipo de paneles HMI indicados en la configuración. Así, por ejemplo, pueden conectarse a la CPU hasta tres SIMATIC Basic Panels o bien hasta dos SIMATIC Comfort Panel con un Basic Panel.

Tabla A- 49 Alimentación eléctrica

Datos técnicos	CPU 1214C AC/DC/relé	CPU 1214C DC/DC/relé	CPU 1214C DC/DC/DC
Rango de tensión	85 a 264 V AC	20,4 V DC a 28,8 V DC 22,0 V DC a 28,8 V DC para temperatura ambiente de -20 °C a 0 °C	
Frecuencia de línea	47 a 63 Hz	--	
Intensidad de entrada (carga máx.)	sólo CPU 100 mA a 120 V AC 50 mA a 240 V AC	500 mA a 24 V DC	
	CPU con todos los accesorios de ampliación 300 mA a 120 V AC 150 mA a 240 V AC	1500 mA a 24 V DC	
Corriente de irrupción (máx.)	20 A a 264 V AC	12 A a 28,8 V DC	
Aislamiento (alimentación de entrada a lógica)	1500 V AC	Sin aislamiento	
Corriente de fuga a tierra, línea AC a tierra funcional	0,5 mA máx.	-	
Tiempo de mantenimiento (pérdida de alimentación)	20 ms a 120 V AC 80 ms a 240 V AC	10 ms a 24 V DC	
Fusible interno, no reemplazable por el usuario	3 A, 250 V, de acción lenta		

Tabla A- 50 Alimentación de sensores

Datos técnicos	CPU 1214C AC/DC/relé	CPU 1214C DC/DC/relé	CPU 1214C DC/DC/DC
Rango de tensión	20,4 a 28,8 V DC	L+ menos 4 V DC mín. L+ menos 5 V DC mín. para temperatura ambiente de -20 °C a 0 °C	
Intensidad de salida nominal (máx.)	400 mA (protegido contra cortocircuito)		
Ruido de rizado máx. (<10 MHz)	< 1 V de pico a pico	Igual a la línea de entrada	
Aislamiento (lógica de la CPU a alimentación de sensores)	Sin aislamiento		

A.4.3 Entradas y salidas digitales

Tabla A- 51 Entradas digitales

Datos técnicos	CPU 1214C AC/DC/relé	CPU 1214C DC/DC/relé	CPU 1214C DC/DC/DC
Número de entradas	14		
Tipo	Sumidero/fuente (tipo 1 IEC sumidero)		
Tensión nominal	24 V DC a 4 mA, nominal		
Tensión continua admisible	30 V DC, máx.		

Datos técnicos

A.4 CPU 1214C

Datos técnicos	CPU 1214C AC/DC/relé	CPU 1214C DC/DC/relé	CPU 1214C DC/DC/DC
Sobretensión transitoria	35 V DC durante 0,5 s		
Señal 1 lógica (mín.)	15 V DC a 2,5 mA		
Señal 0 lógica (máx.)	5 V DC a 1 mA		
Aislamiento (campo a lógica)	500 V AC durante 1 minuto		
Grupos de aislamiento	1		
Tiempos de filtro	0,2, 0,4, 0,8, 1,6, 3,2, 6,4 y 12,8 ms (seleccionable en grupos de 4)		
Frecuencias de entrada de reloj HSC (máx.) (señal 1 lógica = 15 a 26 V DC)	Fase simple: 100 KHz (Ia.0 a Ia.5) y 30 KHz (Ia.6 a Ib.5) Fase en cuadratura: 80 KHz (Ia.0 a Ia.5) y 20 KHz (Ia.6 a Ib.5)		
Número de entradas ON simultáneamente	<ul style="list-style-type: none"> • 7 (no adyacentes) a 60 °C en horizontal o 50 °C en vertical • 14 a 55 °C en horizontal o 45 °C en vertical 		
Longitud de cable (metros)	500 m apantallado, 300 m no apantallado, 50 m apantallado para entradas HSC		

Tabla A- 52 Salidas digitales

Datos técnicos	CPU 1214C AC/DC/relé y DC/DC/relé	CPU 1214C DC/DC/DC
Número de salidas	10	10
Tipo	Relé, contacto seco	Estado sólido MOSFET (fuente)
Rango de tensión	5 a 30 V DC ó 5 a 250 V AC	20,4 a 28,8 V DC
Señal 1 lógica a intensidad máx.	--	20 V DC mín.
Señal 0 lógica con carga de 10 K Ω	--	0,1 V DC máx.
Intensidad (máx.)	2,0 A	0,5 A
Carga de lámparas	30 W DC / 200 W AC	5 W
Resistencia en estado ON	Máx. 0,2 Ω (si son nuevas)	0,6 Ω máx.
Corriente de fuga por salida	--	10 μ A máx.
Sobrecorriente momentánea	7 A si están cerrados los contactos	8 A durante máx. 100 ms
Protección contra sobrecargas	No	No
Aislamiento (campo a lógica)	1500 V AC durante 1 minuto (entre bobina y contacto) Ninguno (entre bobina y circuito lógico)	500 V AC durante 1 minuto
Resistencia de aislamiento	100 M Ω mín. si son nuevas	--
Aislamiento entre contactos abiertos	750 V AC durante 1 minuto	--
Grupos de aislamiento	2	1
Tensión de bloqueo inductiva	--	L+ menos 48 V DC, disipación de 1 W
Retardo de conmutación (Qa.0 a Qa.3)	10 ms máx.	1,0 μ s máx., OFF a ON 3,0 μ s máx., ON a OFF
Retardo de conmutación (Qa.4 a Qb.1)	10 ms máx.	50 μ s máx., OFF a ON 200 μ s máx., ON a OFF

Datos técnicos	CPU 1214C AC/DC/relé y DC/DC/relé	CPU 1214C DC/DC/DC
Frecuencia máxima de conmutación de relé	1 Hz	--
Frecuencia de salida de tren de impulsos (Qa.0 y Qa.2)	No recomendada ¹	100 KHz máx., 2 Hz mín. ²
Vida útil mecánica (sin carga)	10 000 000 ciclos apertura/cierre	--
Vida útil de los contactos bajo carga nominal	100 000 ciclos apertura/cierre	--
Reacción al cambiar de RUN a STOP	Último valor o valor sustitutivo (valor predeterminado: 0)	
Número de salidas ON simultáneamente	<ul style="list-style-type: none"> • 5 (no adyacentes) a 60 °C en horizontal o 50 °C en vertical • 10 a 55 °C en horizontal o 45 °C en vertical 	
Longitud de cable (metros)	500 m apantallado, 150 m no apantallado	

¹ Para modelos de CPU con salidas de relé, se debe instalar una Signal Board (SB) digital para emplear las salidas de impulsos.

² En función del receptor de impulsos y del cable, una resistencia de carga adicional (al menos 10% de la intensidad nominal) puede mejorar la calidad de señal de los impulsos y la inmunidad a interferencias.

A.4.4 Entradas analógicas

Tabla A- 53 Entradas analógicas

Datos técnicos	Descripción
Número de entradas	2
Tipo	Tensión (asimétrica)
Rango total	De 0 a 10 V
Rango total (palabra de datos)	0 a 27.648
Rango de sobreimpulso	10,001 a 11,759 V
Rango de sobreimpulso (palabra de datos)	27.649 a 32.511
Rango de desbordamiento	11,760 a 11,852 V
Rango de desbordamiento (palabra de datos)	32.512 a 32.767
Resolución	10 bits
Tensión soportada máxima	35 V DC
Filtrado	Ninguno, débil, medio o fuerte Consulte la tabla de respuesta a un escalón (ms) para las entradas analógicas de la CPU (Página 770).
Supresión de ruido	10, 50 ó 60 Hz
Impedancia	≥100 KΩ
Aislamiento (campo a lógica)	Ninguno

Datos técnicos	Descripción
Precisión (25 °C/-20 a 60 °C)	3,0%/3,5% de rango máximo
Longitud de cable (metros)	100 m, par trenzado apantallado

A.4.4.1 Respuesta a un escalón de las entradas analógicas integradas en la CPU

Tabla A- 54 Respuesta a un escalón (ms), 0V a 10V medido a 95%

Selección de filtrado (valor medio de muestreo)	Supresión de frecuencias (tiempo de integración)		
	60 Hz	50 Hz	10 Hz
Ninguno (1 ciclo): Sin media	50 ms	50 ms	100 ms
Débil (4 ciclos): 4 muestreos	60 ms	70 ms	200 ms
Medio (16 ciclos): 16 muestreos	200 ms	240 ms	1150 ms
Fuerte (32 ciclos): 32 muestreos	400 ms	480 ms	2300 ms
Tiempo de muestreo	4,17 ms	5 ms	25 ms

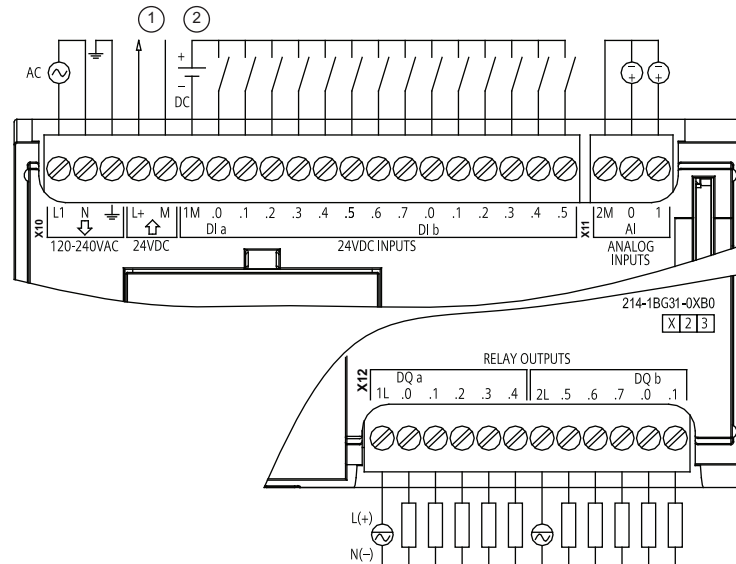
A.4.4.2 Tiempo de muestreo para los puertos analógicos integrados en la CPU

Tabla A- 55 Tiempo de muestreo para las entradas analógicas integradas en la CPU

Supresión de frecuencias (selección del tiempo de integración)	Tiempo de muestreo
60 Hz (16,6 ms)	4,17 ms
50 Hz (20 ms)	5 ms
10 Hz (100 ms)	25 ms

A.4.5 Diagramas de cableado de la CPU 1214C

Tabla A- 56 CPU 1214C AC/DC/relé (6ES7 214-1BG31-0XB0)



- ① Alimentación de sensores 24 V DC
Para una inmunidad a interferencias adicional, conecte "M" a masa incluso si no se utiliza la alimentación de sensores.
- ② Para entradas en sumidero, conecte "-" a "M" (como se indica).
Para entradas en fuente, conecte "+" a "M".

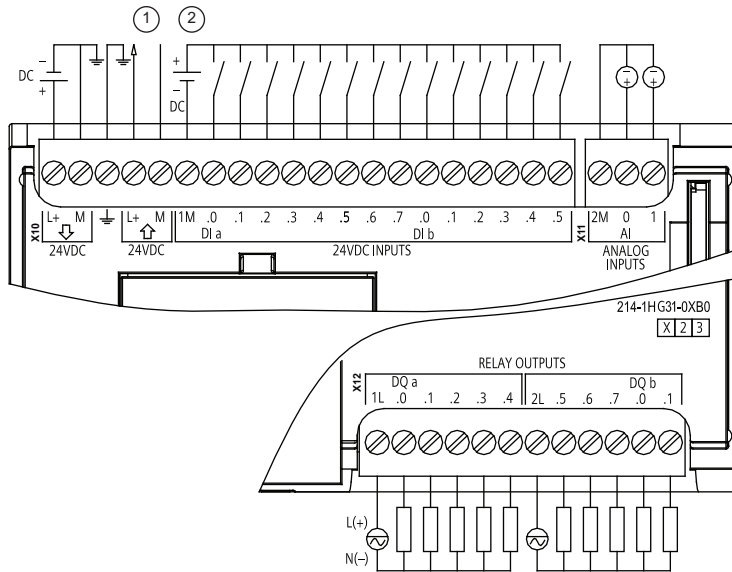
Nota: Los conectores X11 deben ser de oro. Consulte el anexo C, Piezas de repuesto, para ver la referencia.

Tabla A- 57 Asignación de pines de conectores para CPU 1214C AC/DC/relé (6ES7 214-1BG31-0XB0)

Pin	X10	X11 (oro)	X12
1	L1/120-240 V AC	2 M	1L
2	N/120-240 V AC	AI 0	DQ a.0
3	Tierra funcional	AI 1	DQ a.1
4	Salida sensor L+/24 V DC	--	DQ a.2
5	Salida sensor M/24 V DC	--	DQ a.3
6	1M	--	DQ a.4
7	DI a.0	--	2L
8	DI a.1	--	DQ a.5
9	DI a.2	--	DQ a.6
10	DI a.3	--	DQ a.7
11	DI a.4	--	DQ b.0
12	DI a.5	--	DQ b.1
13	DI a.6	--	--
14	DI a.7	--	--
15	DI b.0	--	--
16	DI b.1	--	--
17	DI b.2	--	--

Pin	X10	X11 (oro)	X12
18	DI b.3	--	--
19	DI b.4	--	--
20	DI b.5	--	--

Tabla A- 58 CPU 1214C DC/DC/relé (6ES7 214-1HG31-0XB0)



- ① Alimentación de sensores 24 V DC
Para una inmunidad a interferencias adicional, conecte "M" a masa incluso si no se utiliza la alimentación de sensores.
- ② Para entradas en sumidero, conecte "-" a "M" (como se indica).
Para entradas en fuente, conecte "+" a "M".

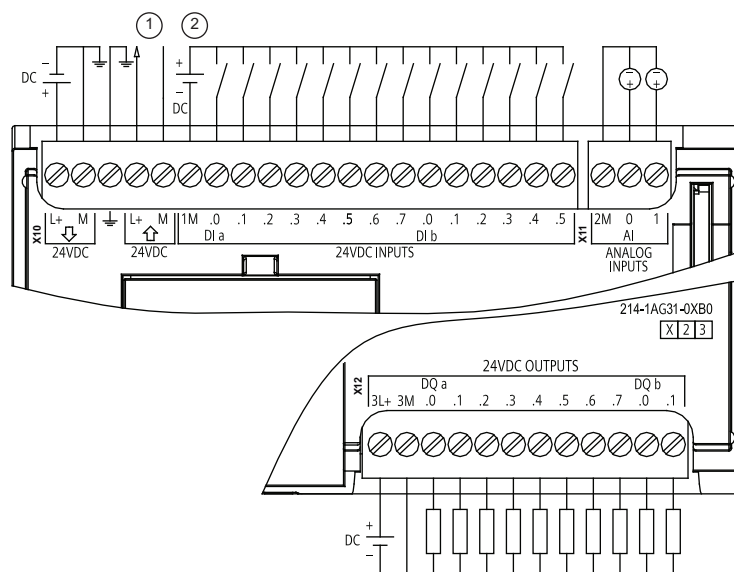
Nota: Los conectores X11 deben ser de oro. Consulte el anexo C, Piezas de repuesto, para ver la referencia.

Tabla A- 59 Asignación de pines de conectores para CPU 1214C DC/DC/relé (6ES7 214-1HG31-0XB0)

Pin	X10	X11 (oro)	X12
1	L+/24 V DC	2 M	1L
2	M/24 V DC	AI 0	DQ a.0
3	Tierra funcional	AI 1	DQ a.1
4	Salida sensor L+/24 V DC	--	DQ a.2
5	Salida sensor M/24 V DC	--	DQ a.3
6	1M	--	DQ a.4
7	DI a.0	--	2L
8	DI a.1	--	DQ a.5
9	DI a.2	--	DQ a.6
10	DI a.3	--	DQ a.7
11	DI a.4	--	DQ b.0
12	DI a.5	--	DQ b.1
13	DI a.6	--	--

Pin	X10	X11 (oro)	X12
14	DI a.7	--	--
15	DI b.0	--	--
16	DI b.1	--	--
17	DI b.2	--	--
18	DI b.3	--	--
19	DI b.4	--	--
20	DI b.5	--	--

Tabla A- 60 CPU 1214C DC/DC/DC (6ES7 214-1AG31-0XB0)



- ① Alimentación de sensores 24 V DC
Para una inmunidad a interferencias adicional, conecte "M" a masa incluso si no se utiliza la alimentación de sensores.
- ② Para entradas en sumidero, conecte "-" a "M" (como se indica).
Para entradas en fuente, conecte "+" a "M".

Nota: Los conectores X11 deben ser de oro. Consulte el anexo C, Piezas de repuesto, para ver la referencia.

Tabla A- 61 Asignación de pines de conectores para CPU 1214C DC/DC/DC (6ES7 214-1AG31-0XB0)

Pin	X10	X11 (oro)	X12
1	L+/24 V DC	2 M	3L+
2	M/24 V DC	AI 0	3M
3	Tierra funcional	AI 1	DQ a.0
4	Salida sensor L+/24 V DC	--	DQ a.1
5	Salida sensor M/24 V DC	--	DQ a.2
6	1M	--	DQ a.3
7	DI a.0	--	DQ a.4
8	DI a.1	--	DQ a.5
9	DI a.2	--	DQ a.6

Pin	X10	X11 (oro)	X12
10	DI a.3	--	DQ a.7
11	DI a.4	--	DQ b.0
12	DI a.5	--	DQ b.1
13	DI a.6	--	--
14	DI a.7	--	-
15	DI b.0	--	--
16	DI b.1	--	--
17	DI b.2	--	--
18	DI b.3	--	--
19	DI b.4	--	--
20	DI b.5	--	--

Nota

Las entradas analógicas que no se utilicen deben cortocircuitarse.

A.5 CPU 1215C

A.5.1 Especificaciones generales y propiedades

Tabla A- 62 General

Datos técnicos	CPU 1215C AC/DC/relé	CPU 1215C DC/DC/relé	CPU 1215C DC/DC/DC
Referencia	6ES7 215-1BG31-0XB0	6ES7 215-1HG31-0XB0	6ES7 215-1AG31-0XB0
Dimensiones A x A x P (mm)	130 x 100 x 75	130 x 100 x 75	130 x 100 x 75
Peso de envío	550 gramos	585 gramos	520 gramos
Disipación de potencia	14 W	12 W	12 W
Intensidad disponible (SM y bus CM)	1600 mA máx. (5 V DC)	1600 mA máx. (5 V DC)	1600 mA máx. (5 V DC)
Intensidad disponible (24 V DC)	400 mA máx. (alimentación de sensores)	400 mA máx. (alimentación de sensores)	400 mA máx. (alimentación de sensores)
Consumo de corriente de las entradas digitales (24 V DC)	4 mA/entrada utilizada	4 mA/entrada utilizada	4 mA/entrada utilizada

Tabla A- 63 Propiedades de la CPU

Datos técnicos		Descripción
Memoria de usuario ¹	Trabajo	100 KB
	Carga	4 MB, interna, ampliable hasta tamaño de tarjeta SD
	Remanente	10 KB
E/S digitales integradas		14 entradas/10 salidas
E/S analógicas integradas		2 entradas/2 salidas
Tamaño de la memoria imagen de proceso		1024 bytes de entradas (I)/1024 bytes de salidas (Q)
Área de marcas (M)		8192 bytes
Memoria temporal (local)		<ul style="list-style-type: none"> • 16 KB para arranque y ciclo (incluyendo los FB y FC asociados) • 4 KB para eventos de alarma estándar, incluyendo FBs y FCs • 4 KB para eventos de alarma de error, incluyendo FBs y FCs
Ampliación con módulos de señales		8 SM máx.
Ampliación con SB, CB o BB		1 máx.
Ampliación con módulos de comunicación		3 CMs máx.
Contadores rápidos		6 en total, ver tabla Asignaciones de entradas de HSC para la CPU 1215C <ul style="list-style-type: none"> • Fase simple: 3 a 100 kHz y 3 a 30 kHz de frecuencia de reloj • Fase en cuadratura: 3 a 80 kHz y 3 a 20 kHz de frecuencia de reloj
Generadores de impulsos ²		4
Entradas de captura de impulsos		14
Alarmas de retardo/cíclicas		4 en total con resolución de 1 ms
Alarmas de flanco		12 ascendentes y 12 descendentes (14 y 14 con Signal Board opcional)
Memory Card		SIMATIC Memory Card (opcional)
Precisión del reloj en tiempo real		+/- 60 segundos/mes
Tiempo de respaldo del reloj de tiempo real		20 días típ./12 días mín. a 40 °C (condensador de alto rendimiento sin mantenimiento)

¹ El tamaño del programa de usuario, los datos y la configuración están limitados por la memoria de carga y memoria de trabajo disponibles de la CPU. No hay un límite determinado para el número de bloques OB, FC, FB y DB soportados o en lo referente al tamaño de un bloque específico. El único límite está sujeto al tamaño total de la memoria.

² Para modelos de CPU con salidas de relé se debe instalar una Signal Board (SB) digital para emplear las salidas de impulsos.

Tabla A- 64 Rendimiento

Tipo de instrucción	Velocidad de ejecución
Booleano	0,08 µs/instrucción
Transferir palabra	1,7 µs/instrucción
Funciones matemáticas con números reales	2,3 µs/instrucción

A.5.2 Temporizadores, contadores y bloques lógicos soportados por la CPU 1215C

Tabla A- 65 Bloques, temporizadores y contadores soportados por la CPU 1215C

Elemento	Descripción	
Bloques	Tipo	OB, FB, FC, DB
	Tamaño	64 KB
	Cantidad	Un total de hasta 1024 bloques (OB + FB + FC + DB)
	Rango de direcciones para FB, FC y DB	De 1 a 65535 (p. ej. del FB 1 al FB 65535)
	Profundidad de anidamiento	16 del OB de arranque o de ciclo; 4 del OB de alarma de retardo, alarma horaria, alarma cíclica, alarma de proceso, alarma de error de tiempo o alarma de diagnóstico
	Observar	Se puede observar a la vez el estado de 2 bloques lógicos
OB	Ciclo del programa	Múltiple: OB 1, de OB 200 a OB 65535
	Arranque	Múltiple: OB 100, de OB 200 a OB 65535
	Alarmas de retardo y alarmas cíclicas	4 ¹ (1 por evento): De OB 200 a OB 65535
	Alarmas de proceso (flancos y HSC)	50 (1 por evento): De OB 200 a OB 65535
	Alarmas de error de tiempo	1: OB 80
	Alarmas de error de diagnóstico	1: OB 82
Temporizadores	Tipo	CEI
	Cantidad	Sólo limitada por el tamaño de la memoria
	Almacenamiento	Estructura en DB, 16 bytes por temporizador
Contadores	Tipo	CEI
	Cantidad	Sólo limitada por el tamaño de la memoria
	Almacenamiento	Estructura en DB, tamaño dependiente del tipo de contaje <ul style="list-style-type: none"> • SInt, USInt: 3 bytes • Int, UInt: 6 bytes • DInt, UDIInt: 12 bytes

¹ Las alarmas de retardo y cíclicas usan los mismos recursos en la CPU. Sólo se puede contar con un total máximo de 4 de estas alarmas (suma de alarmas de retardo y cíclicas). No puede haber 4 alarmas de retardo y 4 alarmas cíclicas.

Tabla A- 66 Comunicación

Datos técnicos	Descripción
Número de puertos	2
Tipo	Ethernet
Dispositivo HMI ¹	3
Programadora (PG)	1

Datos técnicos	Descripción
Conexiones	<ul style="list-style-type: none"> • 8 para Open User Communication (activa o pasiva): TSEND_C, TRCV_C, TCON, TDISCON, TSEND y TRCV • 3 para comunicaciones S7 GET/PUT (CPU a CPU) de servidor • 8 para comunicaciones S7 GET/PUT (CPU a CPU) de cliente
Transferencia de datos	10/100 Mb/s
Aislamiento (señal externa a lógica del PLC)	Aislado por transformador, 1500 V AC, sólo para seguridad frente a defectos breves
Tipo de cable	CAT5e apantallado

¹ La CPU proporciona conexiones HMI dedicadas que admiten un máximo de 3 dispositivos HMI. Se pueden tener hasta 2 SIMATIC Comfort Panel. El número total de HMI depende del tipo de paneles HMI indicados en la configuración. Así, por ejemplo, pueden conectarse a la CPU hasta tres SIMATIC Basic Panels o bien hasta dos SIMATIC Comfort Panel con un Basic Panel.

Tabla A- 67 Alimentación eléctrica

Datos técnicos	CPU 1215C AC/DC/relé	CPU 1215C DC/DC/relé	CPU 1215C DC/DC/DC
Rango de tensión	85 a 264 V AC	20,4 V DC a 28,8 V DC 22,0 V DC a 28,8 V DC para temperatura ambiente de -20 °C a 0 °C	
Frecuencia de línea	47 a 63 Hz	--	
Intensidad de entrada (carga máx.)	sólo CPU	100 mA a 120 V AC 50 mA a 240 V AC	500 mA a 24 V DC
	CPU con todos los accesorios de ampliación	300 mA a 120 V AC 150 mA a 240 V AC	1500 mA a 24 V DC
Corriente de irrupción (máx.)	20 A a 264 V AC	12 A a 28,8 V DC	
Aislamiento (alimentación de entrada a lógica)	1500 V AC	Sin aislamiento	
Corriente de fuga a tierra, línea AC a tierra funcional	0,5 mA máx.	-	
Tiempo de mantenimiento (pérdida de alimentación)	20 ms a 120 V AC 80 ms a 240 V AC	10 ms a 24 V DC	
Fusible interno, no reemplazable por el usuario	3 A, 250 V, de acción lenta		

Tabla A- 68 Alimentación de sensores

Datos técnicos	CPU 1215C AC/DC/relé	CPU 1215C DC/DC/relé	CPU 1215C DC/DC/DC
Rango de tensión	20,4 a 28,8 V DC	L+ menos 4 V DC mín. L+ menos 5 V DC mín. para temperatura ambiente de -20 °C a 0 °C	
Intensidad de salida nominal (máx.)	400 mA (protegido contra cortocircuito)		
Ruido de rizado máx. (<10 MHz)	< 1 V de pico a pico	Igual a la línea de entrada	
Aislamiento (lógica de la CPU a alimentación de sensores)	Sin aislamiento		

A.5.3 Entradas y salidas digitales

Tabla A- 69 Entradas digitales

Datos técnicos	CPU 1215C AC/DC/relé	CPU 1215C DC/DC/relé	CPU 1215C DC/DC/DC
Número de entradas	14		
Tipo	Sumidero/fuente (tipo 1 IEC sumidero)		
Tensión nominal	24 V DC a 4 mA, nominal		
Tensión continua admisible	30 V DC, máx.		
Sobretensión transitoria	35 V DC durante 0,5 s		
Señal 1 lógica (mín.)	15 V DC a 2,5 mA		
Señal 0 lógica (máx.)	5 V DC a 1 mA		
Aislamiento (campo a lógica)	500 V AC durante 1 minuto		
Grupos de aislamiento	1		
Tiempos de filtro	0,2, 0,4, 0,8, 1,6, 3,2, 6,4 y 12,8 ms (seleccionable en grupos de 4)		
Frecuencias de entrada de reloj HSC (máx.) (señal 1 lógica = 15 a 26 V DC)	Fase simple: 100 kHz (Ia.0 a Ia.5) y 30 kHz (Ia.6 a Ib.5) Fase en cuadratura: 80 kHz (Ia.0 a Ia.5) y 20 kHz (Ia.6 a Ib.5)		
Número de entradas ON simultáneamente	<ul style="list-style-type: none"> • 7 (no adyacentes) a 60 °C en horizontal o 50 °C en vertical • 14 a 55 °C en horizontal o 45 °C en vertical 		
Longitud de cable (metros)	500 m apantallado, 300 m no apantallado, 50 m apantallado para entradas HSC		

Tabla A- 70 Salidas digitales

Datos técnicos	CPU 1215C AC/DC/relé y CPU 1215C DC/DC/relé	CPU 1215C DC/DC/DC
Número de salidas	10	10
Tipo	Relé, contacto seco	Estado sólido MOSFET (fuente)
Rango de tensión	5 a 30 V DC ó 5 a 250 V AC	20,4 a 28,8 V DC

Datos técnicos	CPU 1215C AC/DC/relé y CPU 1215C DC/DC/relé	CPU 1215C DC/DC/DC
Señal 1 lógica a intensidad máx.	--	20 V DC mín.
Señal 0 lógica con carga de 10 k Ω	--	0,1 V DC máx.
Intensidad (máx.)	2,0 A	0,5 A
Carga de lámparas	30 W DC/200 W AC	5 W
Resistencia en estado ON	Máx. 0,2 Ω (si son nuevas)	0,6 Ω máx.
Corriente de fuga por salida	--	10 μ A máx.
Sobrecorriente momentánea	7 A si están cerrados los contactos	8 A durante máx. 100 ms
Protección contra sobrecargas	No	No
Aislamiento (campo a lógica)	1500 V AC durante 1 minuto (entre bobina y contacto) Ninguno (entre bobina y circuito lógico)	500 V AC durante 1 minuto
Resistencia de aislamiento	100 M Ω mín. si son nuevas	--
Aislamiento entre contactos abiertos	750 V AC durante 1 minuto	--
Grupos de aislamiento	2	1
Tensión de bloqueo inductiva	--	L+ menos 48 V DC, disipación de 1 W
Retardo de conmutación (Qa.0 a Qa.3)	10 ms máx.	1,0 μ s máx., OFF a ON 3,0 μ s máx., ON a OFF
Retardo de conmutación (Qa.4 a Qb.1)	10 ms máx.	50 μ s máx., OFF a ON 200 μ s máx., ON a OFF
Frecuencia máxima de conmutación de relé	1 Hz	--
Frecuencia de salida de tren de impulsos (Qa.0 y Qa.2)	No recomendada ¹	100 kHz máx., 2 Hz mín. ²
Vida útil mecánica (sin carga)	10 000 000 ciclos apertura/cierre	--
Vida útil de los contactos bajo carga nominal	100 000 ciclos apertura/cierre	--
Reacción al cambiar de RUN a STOP	Último valor o valor sustitutivo (valor predeterminado: 0)	
Número de salidas ON simultáneamente	<ul style="list-style-type: none"> • 5 (no adyacentes) a 60 °C en horizontal o 50 °C en vertical • 10 a 55 °C en horizontal o 45 °C en vertical 	
Longitud de cable (metros)	500 m apantallado, 150 m no apantallado	

¹ Para modelos de CPU con salidas de relé, se debe instalar una Signal Board (SB) digital para emplear las salidas de impulsos.

² En función del receptor de impulsos y del cable, una resistencia de carga adicional (al menos 10% de la intensidad nominal) puede mejorar la calidad de señal de los impulsos y la inmunidad a interferencias.

A.5.4 Entradas y salidas analógicas

A.5.4.1 Datos técnicos de la entrada analógica

Tabla A- 71 Entradas analógicas

Datos técnicos	Descripción
Número de entradas	2
Tipo	Tensión (asimétrica)
Rango total	De 0 a 10 V
Rango total (palabra de datos)	0 a 27.648
Rango de sobreimpulso	10,001 a 11,759 V
Rango de sobreimpulso (palabra de datos)	27.649 a 32.511
Rango de desbordamiento	11,760 a 11,852 V
Rango de desbordamiento (palabra de datos)	32.512 a 32.767
Resolución	10 bits
Tensión soportada máxima	35 V DC
Filtrado	Ninguno, débil, medio o fuerte Consulte la tabla de respuesta a un escalón (ms) para las entradas analógicas de la CPU.
Supresión de ruido	10, 50 ó 60 Hz
Impedancia	≥100 KΩ
Aislamiento (campo a lógica)	Ninguno
Precisión (25 °C/-20 a 60 °C)	3,0%/3,5% de rango máximo
Longitud de cable (metros)	100 m, par trenzado apantallado

A.5.4.2 Respuesta a escalón de las entradas analógicas integradas en la CPU

Tabla A- 72 Respuesta a un escalón (ms), 0V a 10V medido a 95%

Selección de filtrado (valor medio de muestreo)	Supresión de frecuencias (tiempo de integración)		
	60 Hz	50 Hz	10 Hz
Ninguno (1 ciclo): Sin media	50 ms	50 ms	100 ms
Débil (4 ciclos): 4 muestreos	60 ms	70 ms	200 ms
Medio (16 ciclos): 16 muestreos	200 ms	240 ms	1150 ms
Fuerte (32 ciclos): 32 muestreos	400 ms	480 ms	2300 ms
Tiempo de muestreo	4,17 ms	5 ms	25 ms

A.5.4.3 Tiempo de muestreo para los puertos analógicos integrados en la CPU

Tabla A- 73 Tiempo de muestreo para las entradas analógicas integradas en la CPU

Supresión de frecuencias (selección del tiempo de integración)	Tiempo de muestreo
60 Hz (16,6 ms)	4,17 ms
50 Hz (20 ms)	5 ms
10 Hz (100 ms)	25 ms

A.5.4.4 Especificaciones de salidas analógicas

Salidas analógicas

Tabla A- 74 Salidas analógicas

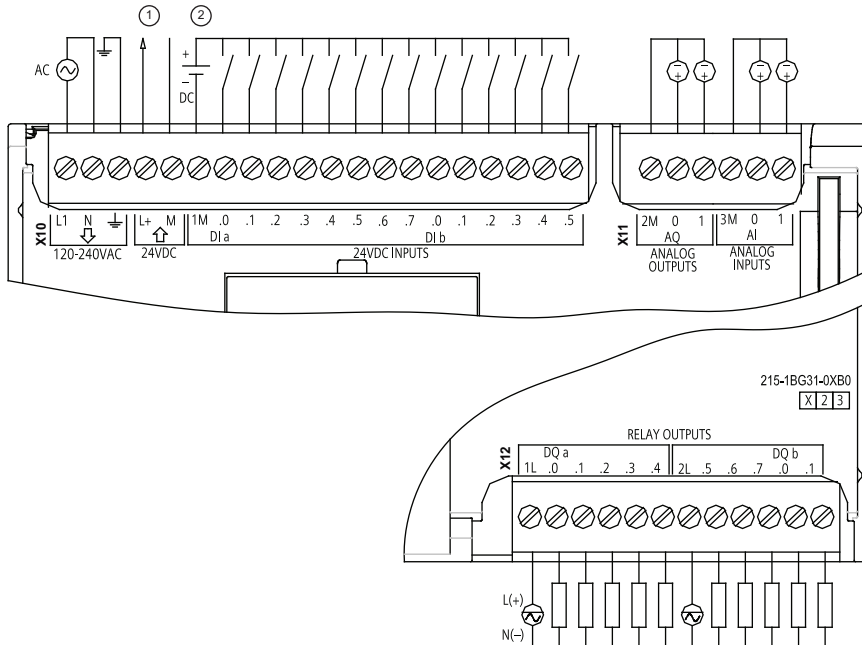
Datos técnicos	Descripción
Número de salidas	2
Tipo	Intensidad
Rango total	0 a 20 mA
Rango total (palabra de datos)	0 a 27.648
Rango de sobreimpulso	20,01 a 23,52 mA ¹
Rango de sobreimpulso (palabra de datos)	27.649 a 32.511
Rango de desbordamiento	Ver nota al pie ²
Rango de desbordamiento (palabra de datos)	32.512 a 32.767
Resolución	10 bits
Impedancia de salida	≤500 Ω máx.
Aislamiento (campo a lógica)	Ninguno
Precisión (25 °C/-20 a 60 °C)	3,0%/3,5% de rango máximo
Tiempo de estabilización	2 ms
Longitud de cable (metros)	100 m, par trenzado apantallado

¹ Para la CPU 1215C con fuente de alimentación DC: a una tensión de red de 20,4 V DC, se soportan hasta 400 Ω de impedancia de salida en el rango de sobreimpulso.

² En una situación de desbordamiento, las salidas analógicas se comportarán según los valores de las propiedades de configuración del dispositivo. En el parámetro "Reacción a STOP de la CPU", seleccione: "Aplicar valor sustitutivo" o "Mantener último valor".

A.5.5 Diagramas de cableado de la CPU 1215C

Tabla A- 75 CPU 1215C AC/DC/relé (6ES7 215-1BG31-0XB0)



- ① Alimentación de sensores 24 V DC
Para una inmunidad a interferencias adicional, conecte "M" a masa incluso si no se utiliza la alimentación del sensor.
- ② Para entradas en sumidero, conecte "-" a "M" (como se indica).
Para entradas en fuente, conecte "+" a "M".

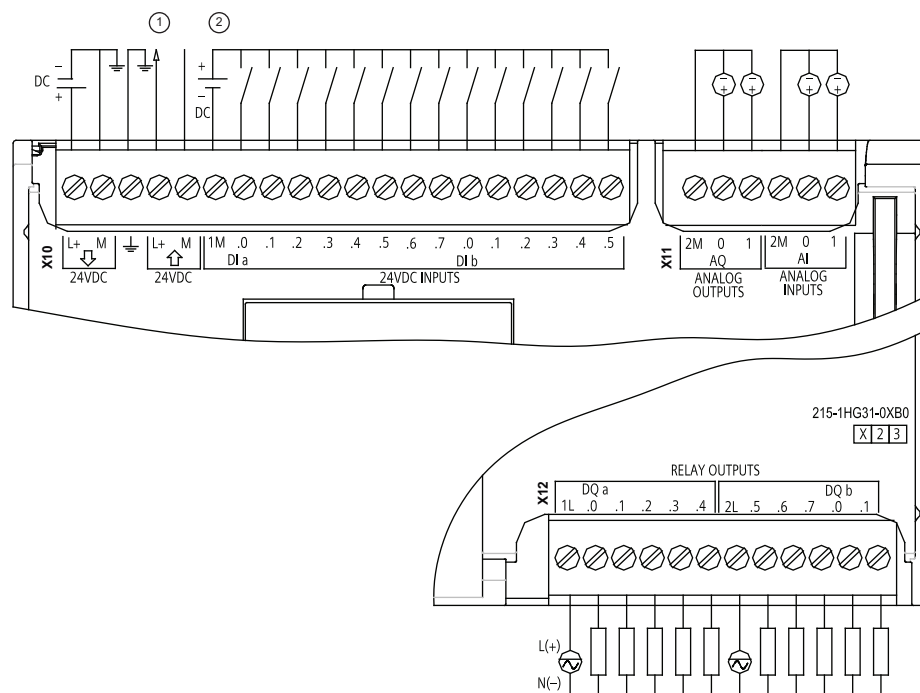
Nota: Los conectores X11 deben ser de oro. Consulte el anexo C, Piezas de repuesto, para ver la referencia.

Tabla A- 76 Asignación de pines de conectores para CPU 1215C AC/DC/relé (6ES7 215-1BG31-0XB0)

Pin	X10	X11 (oro)	X12
1	L1/120-240 V AC	2 M	1L
2	N/120 - 240 V AC	AQ 0	DQ a.0
3	Tierra funcional	AQ 1	DQ a.1
4	Salida sensor L+/24 V DC	3M	DQ a.2
5	Salida sensor M/24 V DC	AI 0	DQ a.3
6	1M	AI 1	DQ a.4
7	DI a.0	--	2L
8	DI a.1	--	DQ a.5
9	DI a.2	--	DQ a.6
10	DI a.3	--	DQ a.7

Pin	X10	X11 (oro)	X12
11	DI a.4	--	DQ b.0
12	DI a.5	--	DQ b.1
13	DI a.6	--	--
14	DI a.7	--	--
15	DI b.0	--	--
16	DI b.1	--	--
17	DI b.2	--	--
18	DI b.3	--	--
19	DI b.4	--	--
20	DI b.5	--	--

Tabla A- 77 CPU 1215C DC/DC/relé (6ES7 215-1HG31-0XB0)



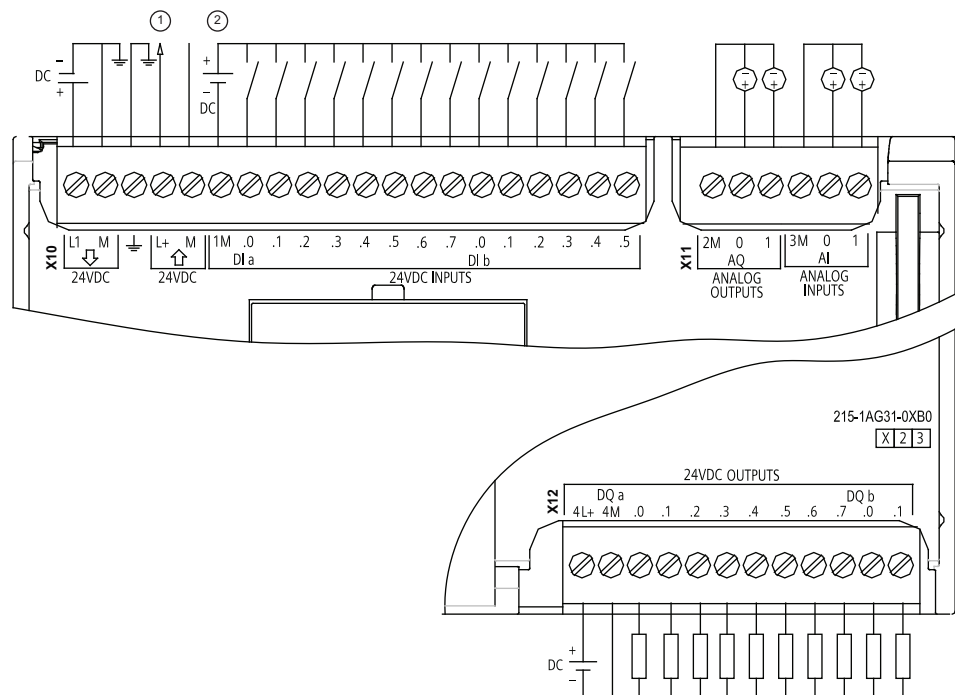
- ① Alimentación de sensores 24 V DC
Para una inmunidad a interferencias adicional, conecte "M" a masa incluso si no se utiliza la alimentación del sensor.
- ② Para entradas en sumidero, conecte "-" a "M" (como se indica). Para entradas en fuente, conecte "+" a "M".

Nota: Los conectores X11 deben ser de oro. Consulte el anexo C, Piezas de repuesto, para ver la referencia.

Tabla A- 78 Asignación de pines de conectores para CPU 1215C DC/DC/relé (6ES7 215-1HG31-0XB0)

Pin	X10	X11 (oro)	X12
1	L+/24 V DC	2 M	1L
2	M/24 V DC	AQ 0	DQ a.0
3	Tierra funcional	AQ 1	DQ a.1
4	Salida sensor L+/24 V DC	2M	DQ a.2
5	Salida sensor M/24 V DC	AI 0	DQ a.3
6	1M	AI 1	DQ a.4
7	DI a.0	--	2L
8	DI a.1	--	DQ a.5
9	DI a.2	--	DQ a.6
10	DI a.3	--	DQ a.7
11	DI a.4	--	DQ b.0
12	DI a.5	--	DQ b.1
13	DI a.6	--	--
14	DI a.7	--	--
15	DI b.0	--	--
16	DI b.1	--	--
17	DI b.2	--	--
18	DI b.3	--	--
19	DI b.4	--	--
20	DI b.5	--	--

Tabla A- 79 CPU 1215C DC/DC/DC (6ES7 215-1AG31-0XB0)



- ① Alimentación de sensores 24 V DC
Para una inmunidad a interferencias adicional, conecte "M" a masa incluso si no se utiliza la alimentación del sensor.
- ② Para entradas en sumidero, conecte "-" a "M" (como se indica). Para entradas en fuente, conecte "+" a "M".

Nota: Los conectores X11 deben ser de oro. Consulte el anexo C, Piezas de repuesto, para ver la referencia.

Tabla A- 80 Asignación de pines de conectores para CPU 1215C DC/DC/DC (6ES7 215-1AG31-0XB0)

Pin	X10	X11 (oro)	X12
1	L1/24 V DC	2 M	4L+
2	M/24 V DC	AQ 0	4M
3	Tierra funcional	AQ 1	DQ a.0
4	Salida sensor L+/24 V DC	3M	DQ a.1
5	Salida sensor M/24 V DC	AI 0	DQ a.2
6	1M	AI 1	DQ a.3
7	DI a.0	--	DQ a.4
8	DI a.1	--	DQ a.5
9	DI a.2	--	DQ a.6
10	DI a.3	--	DQ a.7
11	DI a.4	--	DQ b.0

Pin	X10	X11 (oro)	X12
12	DI a.5	--	DQ b.1
13	DI a.6	--	--
14	DI a.7	--	--
15	DI b.0	--	--
16	DI b.1	--	--
17	DI b.2	--	--
18	DI b.3	--	--
19	DI b.4	--	--
20	DI b.5	--	--

Nota

Las entradas analógicas que no se utilicen deben cortocircuitarse.

A.6 Módulos de señales digitales (SMs)

A.6.1 Datos técnicos del módulo de entradas digitales SM 1221

Tabla A- 81 Especificaciones generales

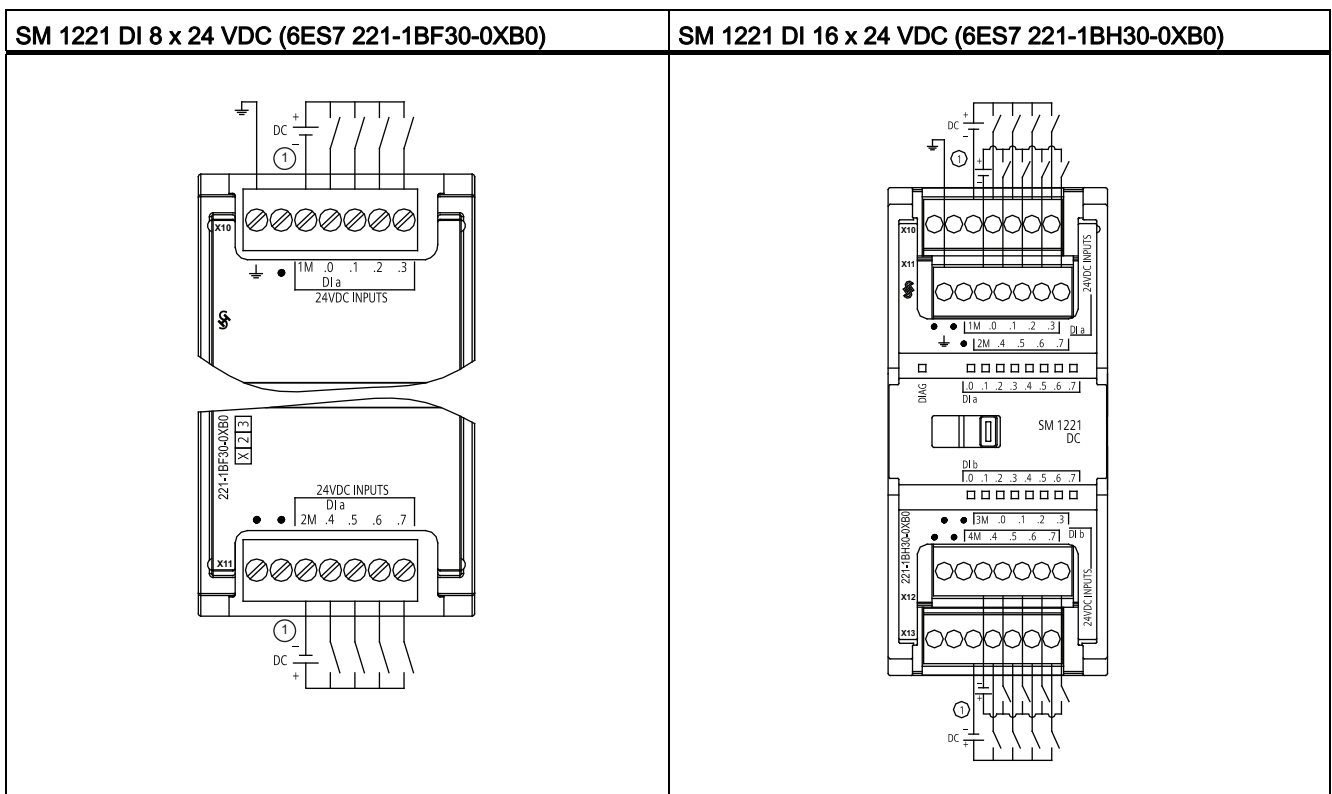
Modelo	SM 1221 DI 8 x 24 V DC	SM 1221 DI 16 x 24 V DC
Referencia	6ES7 221-1BF30-0XB0	6ES7 221-1BH30-0XB0
Dimensiones A x A x P (mm)	45 x 100 x 75	45 x 100 x 75
Peso	170 gramos	210 gramos
Disipación de potencia	1,5 W	2,5 W
Consumo de corriente (bus SM)	105 mA	130 mA
Consumo de corriente (24 V DC)	4 mA/entrada utilizada	4 mA/entrada utilizada

Tabla A- 82 Entradas digitales

Modelo	SM 1221 DI 8 x 24 V DC	SM 1221 DI 16 x 24 V DC
Número de entradas	8	16
Tipo	Sumidero/fuente (tipo 1 IEC sumidero)	Sumidero/fuente (tipo 1 IEC sumidero)
Tensión nominal	24 V DC a 4 mA, nominal	24 V DC a 4 mA, nominal
Tensión continua admisible	30 V DC, máx.	30 V DC, máx.
Sobretensión transitoria	35 V DC durante 0,5 seg.	35 V DC durante 0,5 seg.
Señal 1 lógica (mín.)	15 V DC a 2,5 mA	15 V DC a 2,5 mA

Modelo	SM 1221 DI 8 x 24 V DC	SM 1221 DI 16 x 24 V DC
Señal 0 lógica (máx.)	5 V DC a 1 mA	5 V DC a 1 mA
Aislamiento (campo a lógica)	500 V AC durante 1 minuto	500 V AC durante 1 minuto
Grupos de aislamiento	2	4
Tiempos de filtro	0,2, 0,4, 0,8, 1,6, 3,2, 6,4 y 12,8 ms (seleccionable en grupos de 4)	0,2, 0,4, 0,8, 1,6, 3,2, 6,4 y 12,8 ms (seleccionable en grupos de 4)
Número de entradas ON simultáneamente	8	16
Longitud de cable (metros)	500 m apantallado, 300 no apantallado	500 m apantallado, 300 no apantallado

Tabla A- 83 Diagramas de cableado de los SM de entradas digitales



① Para entradas NPN, conecte "-" a "M" (como se indica). Para entradas en fuente, conecte "+" a "M".

A.6 Módulos de señales digitales (SMs)

Tabla A- 84 Asignación de pines de conectores para SM 1221 DI 8 x 24 V DC (6ES7 221-1BF30-0XB0)

Pin	X10	X11
1	GND	Sin conexión
2	Sin conexión	Sin conexión
3	1M	2M
4	DI a.0	DI a.4
5	DI a.1	DI a.5
6	DI a.2	DI a.6
7	DI a.3	DI a.7

Tabla A- 85 Asignación de pines de conectores para SM 1221 DI 16 x 24 V DC (6ES7 221-1BH30-0XB0)

Pin	X10	X11	X12	X13
1	GND	Sin conexión	Sin conexión	Sin conexión
2	Sin conexión	Sin conexión	Sin conexión	Sin conexión
3	1M	2M	3 M	4 M
4	DI a.0	DI a.4	DI b.0	DI b.4
5	DI a.1	DI a.5	DI b.1	DI b.5
6	DI a.2	DI a.6	DI b.2	DI b.6
7	DI a.3	DI a.7	DI b.3	DI b.7

A.6.2 Datos técnicos del módulo de salidas digitales SM 1222 de 8 salidas

Tabla A- 86 Especificaciones generales

Modelo	SM 1222 DQ 8 x relé	SM 1222 DQ8 RLY inversor	SM 1222 DQ 8 x 24 V DC
Referencia	6ES7 222-1HF30-0XB0	6ES7 222-1XF30-0XB0	6ES7 222-1BF30-0XB0
Dimensiones A x A x P (mm)	45 x 100 x 75	70 x 100 x 75	45 x 100 x 75
Peso	190 gramos	310 gramos	180 gramos
Disipación de potencia	4,5 W	5 W	1,5 W
Consumo de corriente (bus SM)	120 mA	140 mA	120 mA
Consumo de corriente (24 V DC)	11 mA/bobina de relé utilizada	16,7 mA/bobina de relé utilizada	11 mA/bobina de relé utilizada

Tabla A- 87 Salidas digitales

Modelo	SM 1222 DQ 8 x relé	SM 1222 DQ8 RLY inversor	SM 1222 DQ 8 x 24 V DC
Número de salidas	8	8	8
Tipo	Relé, contacto seco	Contacto inversor de relé	Estado sólido MOSFET (fuente)
Rango de tensión	5 a 30 V DC ó 5 a 250 V AC	5 a 30 V DC ó 5 a 250 V AC	20,4 a 28,8 V DC
Señal 1 lógica a intensidad máx.	--	--	20 V DC mín.
Señal 0 lógica con carga de 10K Ω	--	--	0,1 V DC máx.
Intensidad (máx.)	2,0 A	2,0 A	0,5 A
Carga de lámparas	30 W DC / 200 W AC	30 W DC / 200 W AC	5W
Resistencia en estado ON (contactos)	Máx. 0,2 Ω (si son nuevas)	Máx. 0,2 Ω (si son nuevas)	0,6 Ω máx.
Corriente de fuga por salida	--	--	10 μ A máx.
Sobrecorriente momentánea	7 A si están cerrados los contactos	7 A si están cerrados los contactos	8 A durante máx. 100 ms
Protección contra sobrecargas	No	No	No
Aislamiento (campo a lógica)	1500 V AC durante 1 minuto (bobina a contacto) Ninguno (bobina a lógica)	1500 V AC durante 1 minuto (entre bobina y contacto)	500 V AC durante 1 minuto
Resistencia de aislamiento	100 M Ω mín. si son nuevas	100 M Ω mín. si son nuevas	--
Aislamiento entre contactos abiertos	750 V AC durante 1 minuto	750 V AC durante 1 minuto	--
Grupos de aislamiento	2	8	1
Intensidad por neutro (máx.)	10 A	2 A	4 A
Tensión de bloqueo inductiva	--	--	L+ menos 48 V, disipación de 1 W
Retardo de conmutación	10 ms máx.	10 ms máx.	50 μ s máx. OFF a ON 200 μ s máx. ON a OFF
Frecuencia máxima de conmutación de relé	1 Hz	1 Hz	--
Vida útil mecánica (sin carga)	10 000 000 ciclos apertura/cierre	10 000 000 ciclos apertura/cierre	--
Vida útil de los contactos bajo carga nominal	100 000 ciclos apertura/cierre	100 000 ciclos apertura/cierre	--
Reacción al cambiar de RUN a STOP	Último valor o valor sustitutivo (valor predeterminado: 0)	Último valor o valor sustitutivo (valor predeterminado: 0)	Último valor o valor sustitutivo (valor predeterminado: 0)

Datos técnicos

A.6 Módulos de señales digitales (SMs)

Modelo	SM 1222 DQ 8 x relé	SM 1222 DQ8 RLY inversor	SM 1222 DQ 8 x 24 V DC
Número de salidas ON simultáneamente	8	<ul style="list-style-type: none"> 4 (no adyacentes) a 60 °C en horizontal o 50 °C en vertical 8 a 55 °C en horizontal o 45 °C en vertical 	8
Longitud de cable (metros)	500 m apantallado, 150 m no apantallado	500 m apantallado, 150 m no apantallado	500 m apantallado, 150 m no apantallado

A.6.3 Datos técnicos del módulo de salidas digitales SM 1222 de 16 salidas

Tabla A- 88 Especificaciones generales

Modelo	SM 1222 DQ 16 x relé	SM 1222 DQ 16 x 24 V DC
Referencia	6ES7 222-1HH30-0XB0	6ES7 222-1BH30-0XB0
Dimensiones A x A x P (mm)	45 x 100 x 75	45 x 100 x 75
Peso	260 gramos	220 gramos
Disipación de potencia	8,5 W	2,5 W
Consumo de corriente (bus SM)	135 mA	140 mA
Consumo de corriente (24 V DC)	11 mA/bobina de relé utilizada	-

Tabla A- 89 Salidas digitales

Modelo	SM 1222 DQ 16 x relé	SM 1222 DQ 16 x 24 V DC
Número de salidas	16	16
Tipo	Relé, contacto seco	Estado sólido MOSFET (fuente)
Rango de tensión	5 a 30 V DC ó 5 a 250 V AC	20,4 a 28,8 V DC
Señal 1 lógica a intensidad máx.	-	20 V DC mín.
Señal 0 lógica con carga de 10K Ω	-	0,1 V DC máx.
Intensidad (máx.)	2,0 A	0,5 A
Carga de lámparas	30 W DC / 200 W AC	5W
Resistencia en estado ON (contactos)	Máx. 0,2 Ω (si son nuevas)	0,6 Ω máx.
Corriente de fuga por salida	--	10 μ A máx.
Sobrecorriente momentánea	7 A si están cerrados los contactos	8 A durante máx. 100 ms
Protección contra sobrecargas	No	No

Modelo	SM 1222 DQ 16 x relé	SM 1222 DQ 16 x 24 V DC
Aislamiento (campo a lógica)	1500 V AC durante 1 minuto (bobina a contacto) Ninguno (bobina a lógica)	500 V AC durante 1 minuto
Resistencia de aislamiento	100 MΩ mín. si son nuevas	-
Aislamiento entre contactos abiertos	750 V AC durante 1 minuto	-
Grupos de aislamiento	4	1
Intensidad por neutro (máx.)	10 A	8 A
Tensión de bloqueo inductiva	-	L+ menos 48 V, disipación de 1 W
Retardo de conmutación	10 ms máx.	50 μs máx. OFF a ON 200 μs máx. ON a OFF
Frecuencia máxima de conmutación de relé	1 Hz	-
Vida útil mecánica (sin carga)	10 000 000 ciclos apertura/cierre	-
Vida útil de los contactos bajo carga nominal	100 000 ciclos apertura/cierre	-
Reacción al cambiar de RUN a STOP	Último valor o valor sustitutivo (valor predeterminado: 0)	Último valor o valor sustitutivo (valor predeterminado: 0)
Número de salidas ON simultáneamente	<ul style="list-style-type: none"> • 8 (no adyacentes) a 60 °C en horizontal o 50 °C en vertical • 16 a 55 °C en horizontal o 45 °C en vertical 	16
Longitud de cable (metros)	500 m apantallado, 150 m no apantallado	500 m apantallado, 150 m no apantallado

Tabla A- 90 Diagramas de cableado de los SM de 8 salidas digitales

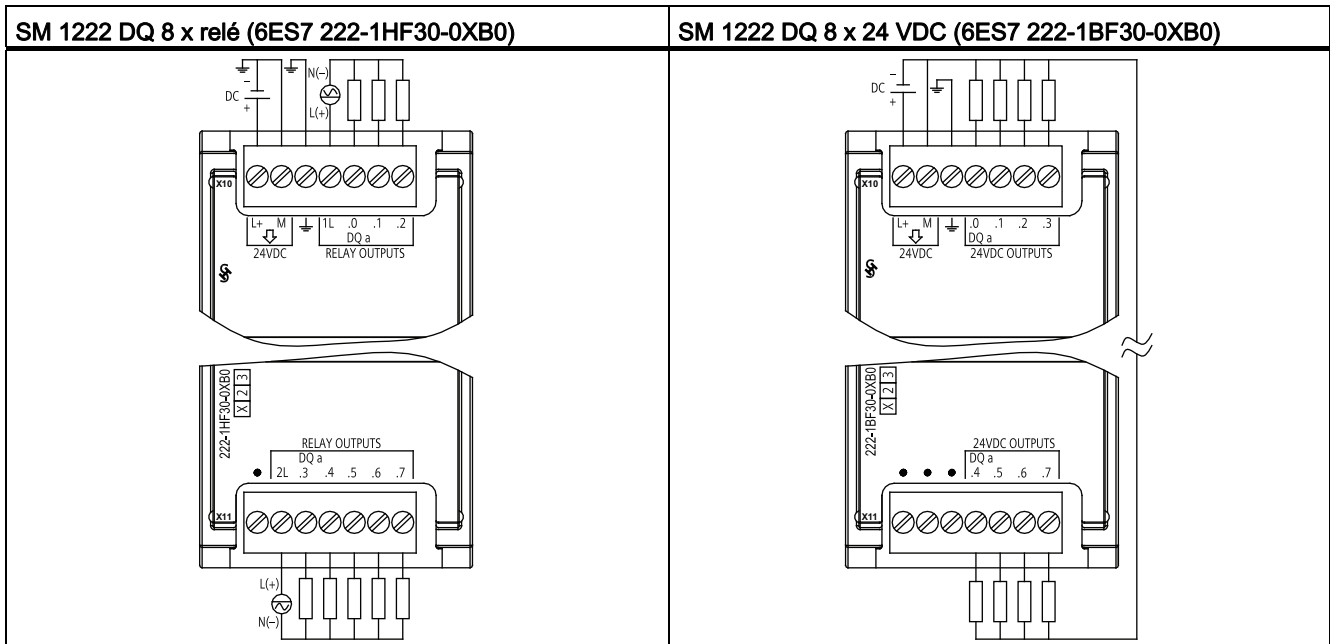


Tabla A- 91 Asignación de pines de conectores para SM 1222 DQ 8 x relé (6ES7 222-1HF30-0XB0)

Pin	X10	X11
1	L+/24 V DC	Sin conexión
2	M/24 V DC	2L
3	Tierra funcional	DQ a.3
4	1L	DQ a.4
5	DQ a.0	DQ a.5
6	DQ a.1	DQ a.6
7	DQ a.2	DQ a.7

Tabla A- 92 Asignación de pines de conectores para SM 1222 DQ 8 x 24 V DC (6ES7 222-1BF30-0XB0)

Pin	X10	X11
1	L+/24 V DC	Sin conexión
2	M/24 V DC	Sin conexión
3	Tierra funcional	Sin conexión
4	DQ a.0	DQ a.4
5	DQ a.1	DQ a.5
6	DQ a.2	DQ a.6
7	DQ a.2	DQ a.7

Tabla A- 93 Diagrama de cableado del SM inversor de relé de 8 salidas digitales

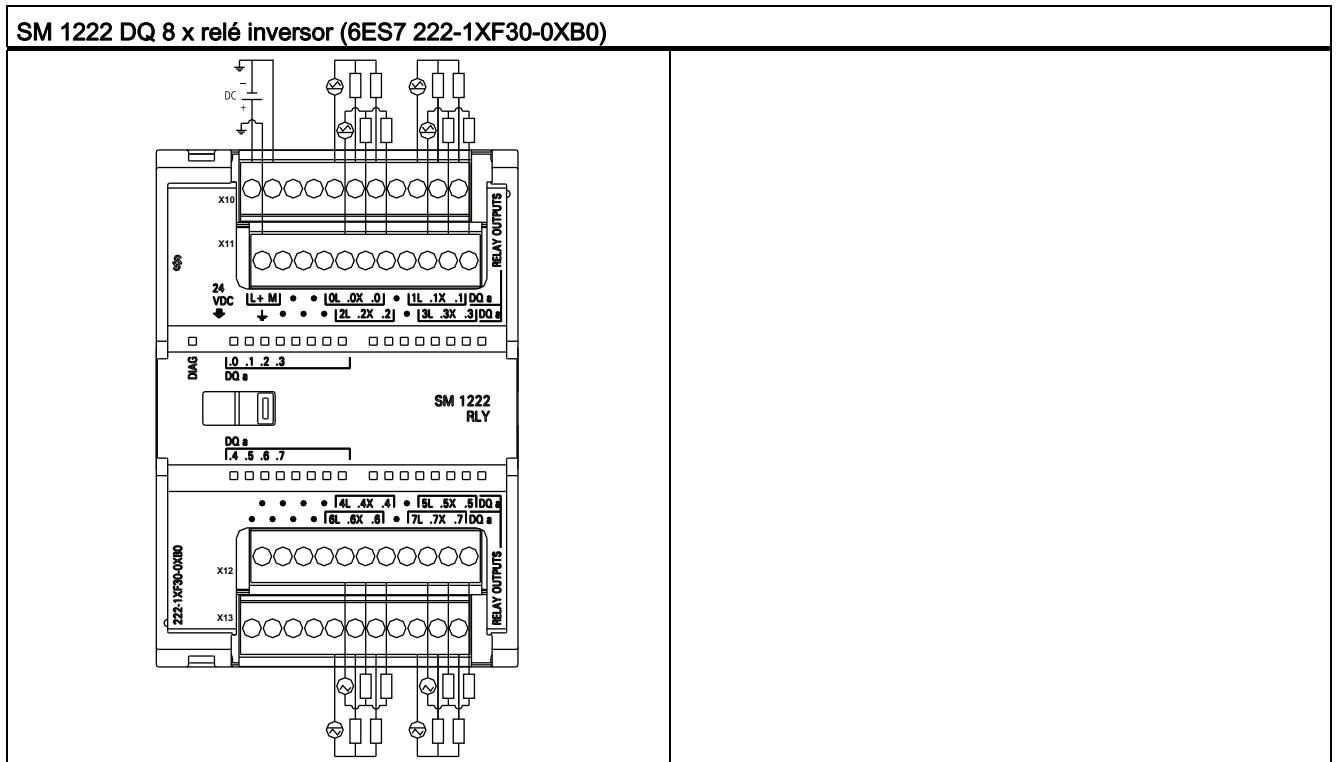


Tabla A- 94 Asignación de pines de conectores para SM 1222 DQ 8 x relé inversor (6ES7 222-1XF30-0XB0)

Pin	X10	X11	X12	X13
1	L+/24 V DC	Tierra funcional	Sin conexión	Sin conexión
2	M/24 V DC	Sin conexión	Sin conexión	Sin conexión
3	Sin conexión	Sin conexión	Sin conexión	Sin conexión
4	Sin conexión	Sin conexión	Sin conexión	Sin conexión
5	0L	2L	4L	6L
6	DQ a.0X	DQ a.2X	DQ a.4X	DQ a.6X
7	DQ a.0	DQ a.2	DQ a.4	DQ a.6
8	Sin conexión	Sin conexión	Sin conexión	Sin conexión
9	1L	3L	5L	7L
10	DQ a.1X	DQ a.3X	DQ a.5X	DQ a.7X
11	DQ a.1	DQ a.3	DQ a.5	DQ a.7

Tabla A- 95 Diagramas de cableado de los SM de 16 salidas digitales

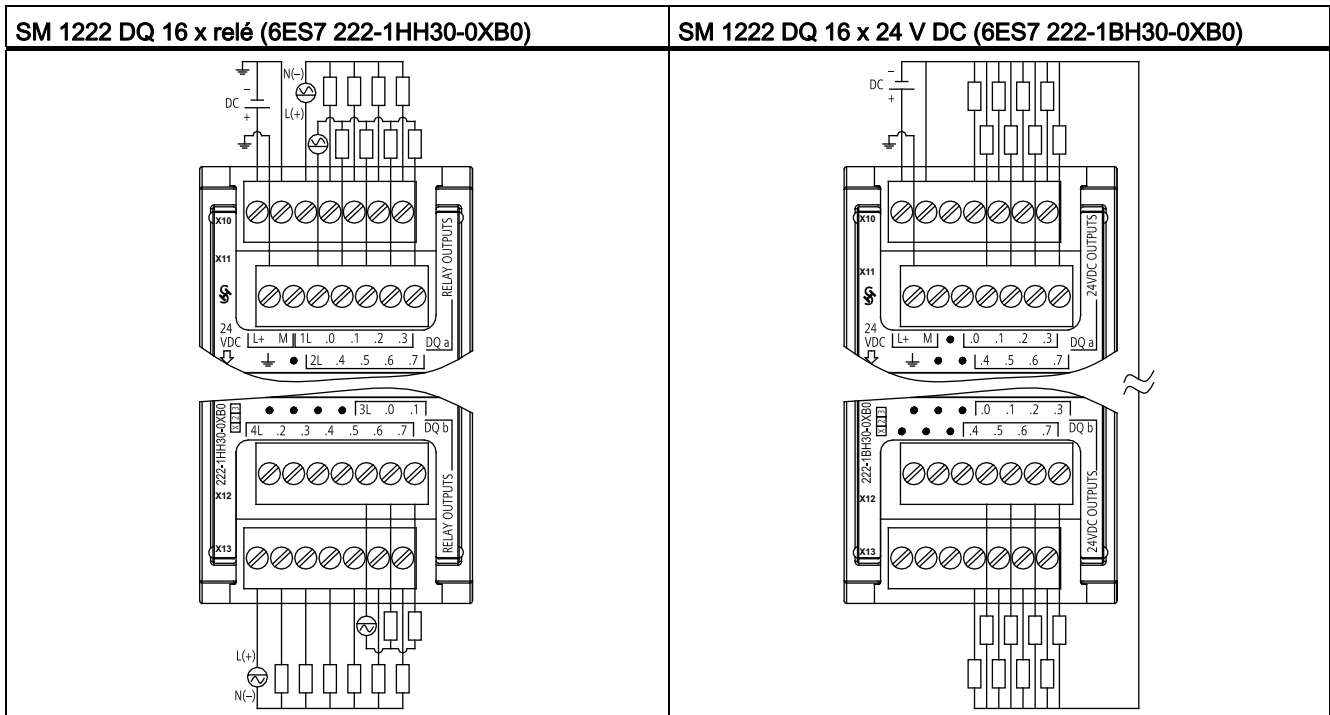


Tabla A- 96 Asignación de pines de conectores para SM 1222 DQ 16 x relé (6ES7 222-1HH30-0XB0)

Pin	X10	X11	X12	X13
1	L+/24 V DC	Tierra funcional	Sin conexión	4L
2	M/24 V DC	Sin conexión	Sin conexión	DQ b.2
3	1L	2L	Sin conexión	DQ b.3
4	DQ a.0	DQ a.4	Sin conexión	DQ b.4
5	DQ a.1	DQ a.5	3L	DQ b.5
6	DQ a.2	DQ a.6	DQ b.0	DQ b.6
7	DQ a.3	DQ a.7	DQ b.1	DQ b.7

Tabla A- 97 Asignación de pines de conectores para SM 1222 DQ 16 x 24 V DC (6ES7 222-1BH30-0XB0)

Pin	X10	X11	X12	X13
1	L+/24 V DC	Tierra funcional	Sin conexión	Sin conexión
2	M/24 V DC	Sin conexión	Sin conexión	Sin conexión
3	Sin conexión	Sin conexión	Sin conexión	Sin conexión
4	DQ a.0	DQ a.4	DQ b.0	DQ b.4
5	DQ a.1	DQ a.5	DQ b.1	DQ b.5

Pin	X10	X11	X12	X13
6	DQ a.2	DQ a.6	DQ b.2	DQ b.6
7	DQ a.3	DQ a.7	DQ b.3	DQ b.7

A.6.4 Datos técnicos del módulo de entradas/salidas digitales V DC SM 1223

Tabla A- 98 Especificaciones generales

Modelo	SM 1223 DI 8 x 24 V DC, DQ 8 x relé	SM 1223 DI 16 x 24 V DC, DQ 16 x relé	SM 1223 DI 8 x 24 V DC, DQ 8 x 24 V DC	SM 1223 DI 16 x 24 V DC, DQ 16 x 24 V DC
Referencia	6ES7 223-1PH30-0XB0	6ES7 223-1PL30-0XB0	6ES7 223-1BH30-0XB0	6ES7 223-1BL30-0XB0
Dimensiones A x A x P (mm)	45 x 100 x 75	70 x 100 x 75	45 x 100 x 75	70 x 100 x 75
Peso	230 gramos	350 gramos	210 gramos	310 gramos
Disipación de potencia	5,5 W	10 W	2,5 W	4,5 W
Consumo de corriente (bus SM)	145 mA	180 mA	145 mA	185 mA
Consumo de corriente (24 V DC)	4 mA/entrada utilizada 11 mA/bobina de relé utilizada		4 mA/entrada utilizada	

Tabla A- 99 Entradas digitales

Modelo	SM 1223 DI 8 x 24 V DC, DQ 8 x relé	SM 1223 DI 16 x 24 V DC, DQ 16 x relé	SM 1223 DI 8 x 24 V DC, DQ 8 x 24 V DC	SM 1223 DI 16 x 24 V DC, DQ 16 x 24 V DC
Número de entradas	8	16	8	16
Tipo	Sumidero/fuente (tipo 1 IEC sumidero)			
Tensión nominal	24 V DC a 4 mA, nominal			
Tensión continua admisible	30 V DC máx.			
Sobretensión transitoria	35 V DC durante 0,5 s			
Señal 1 lógica (mín.)	15 V DC a 2,5 mA			
Señal 0 lógica (máx.)	5 V DC a 1 mA			
Aislamiento (campo a lógica)	500 V AC durante 1 minuto			
Grupos de aislamiento	2	2	2	2
Tiempos de filtro	0,2, 0,4, 0,8, 1,6, 3,2, 6,4 y 12,8 ms, seleccionable en grupos de 4			

Datos técnicos

A.6 Módulos de señales digitales (SMs)

Modelo	SM 1223 DI 8 x 24 V DC, DQ 8 x relé	SM 1223 DI 16 x 24 V DC, DQ 16 x relé	SM 1223 DI 8 x 24 V DC, DQ 8 x 24 V DC	SM 1223 DI 16 x 24 V DC, DQ 16 x 24 V DC
Número de entradas ON simultáneamente	8	<ul style="list-style-type: none"> 8 (no adyacentes) a 60 °C en horizontal o 50 °C en vertical 16 a 55 °C en horizontal o 45 °C en vertical 	8	16
Longitud de cable (metros)	500 m apantallado, 300 m no apantallado			

Tabla A- 100 Salidas digitales

Modelo	SM 1223 DI 8 x 24 V DC, DQ 8 x relé	SM 1223 DI 16 x 24 V DC, DQ 16 x relé	SM 1223 DI 8 x 24 V DC, DQ 8 x 24 V DC	SM 1223 DI 16 x 24 V DC, DQ 16 x 24 V DC
Número de salidas	8	16	8	16
Tipo	Relé, contacto seco		Estado sólido MOSFET (fuente)	
Rango de tensión	5 a 30 V DC ó 5 a 250 V AC		20,4 a 28,8 V DC	
Señal 1 lógica a intensidad máx.	--		20 V DC mín.	
Señal 0 lógica con carga de 10 K Ω	--		0,1 V DC, máx.	
Intensidad (máx.)	2,0 A		0,5 A	
Carga de lámparas	30 W DC / 200 W AC		5 W	
Resistencia en estado ON (contactos)	Máx. 0,2 Ω (si son nuevas)		0,6 Ω máx.	
Corriente de fuga por salida	--		10 μ A máx.	
Sobrecorriente momentánea	7 A si están cerrados los contactos		8 A durante máx. 100 ms	
Protección contra sobrecargas	No			
Aislamiento (campo a lógica)	1500 V AC durante 1 minuto (bobina a contacto) Ninguno (bobina a lógica)		500 V AC durante 1 minuto	
Resistencia de aislamiento	100 M Ω mín. si son nuevas		--	
Aislamiento entre contactos abiertos	750 V AC durante 1 minuto		--	
Grupos de aislamiento	2	4	1	1
Intensidad por neutro	10A	8 A	4 A	8 A
Tensión de bloqueo inductiva	--		L+ menos 48 V, disipación de 1 W	
Retardo de conmutación	10 ms máx.		50 μ s máx. OFF a ON 200 μ s máx. ON a OFF	
Frecuencia máxima de conmutación de relé	1 Hz		--	
Vida útil mecánica (sin carga)	10 000 000 ciclos apertura/cierre		--	

Modelo	SM 1223 DI 8 x 24 V DC, DQ 8 x relé	SM 1223 DI 16 x 24 V DC, DQ 16 x relé	SM 1223 DI 8 x 24 V DC, DQ 8 x 24 V DC	SM 1223 DI 16 x 24 V DC, DQ 16 x 24 V DC
Vida útil de los contactos bajo carga nominal	100 000 ciclos apertura/cierre		--	
Reacción al cambiar de RUN a STOP	Último valor o valor sustitutivo (valor predeterminado: 0)			
Número de salidas ON simultáneamente	8	<ul style="list-style-type: none"> 8 (no adyacentes) a 60 °C en horizontal o 50 °C en vertical 16 a 55 °C en horizontal o 45 °C en vertical 	8	16
Longitud de cable (metros)	500 m apantallado, 150 m no apantallado			

Tabla A- 101 Diagramas de cableado de los SM de entradas digitales V DC/salidas digitales de relé

SM 1223 DI 8 x 24 V DC, DQ 8 x relé (6ES7 223-1PH30-0XB0)	SM 1223 DI 16 x 24 V DC, DQ 16 x relé (6ES7 223-1PL30-0XB0)	Notas
		<p>① Para entradas NPN, conecte "-" a "M" (como se indica).</p> <p>Para entradas en fuente, conecte "+" a "M".</p>

Tabla A- 102 Asignación de pines de conectores para SM 1223 DI 8 x 24 V DC, DQ 8 x relé (6ES7 223-1PH30-0XB0)

Pin	X10	X11	X12	X13
1	L+/24 V DC	GND	Sin conexión	Sin conexión
2	M/24 V DC	Sin conexión	Sin conexión	Sin conexión
3	1M	2M	1L	2L
4	DI a.0	DI a.4	DQ a.0	DQ a.4
5	DI a.1	DI a.5	DQ a.1	DQ a.5
6	DI a.2	DI a.6	DQ a.2	DQ a.6
7	DI a.3	DI a.7	DQ a.3	DQ a.7

Tabla A- 103 Asignación de pines de conectores para SM 1223 DI 16 x 24 V DC, DQ 16 x relé (6ES7 223-1PL30-0XB0)

Pin	X10	X11	X12	X13
1	L+/24 V DC	GND	1L	3L
2	M/24 V DC	Sin conexión	DQ a.0	DQ b.0
3	1M	2M	DQ a.1	DQ b.1
4	DI a.0	DI b.0	DQ a.2	DQ b.2
5	DI a.1	DI b.1	DQ a.3	DQ b.3
6	DI a.2	DI b.2	Sin conexión	Sin conexión
7	DI a.3	DI b.3	2L	4L
8	DI a.4	DI b.4	DQ a.4	DQ b.4
9	DI a.5	DI b.5	DQ a.5	DQ b.5
10	DI a.6	DI b.6	DQ a.6	DQ b.6
11	DI a.7	DI b.7	DQ a.7	DQ b.7

Tabla A- 104 Diagramas de cableado de los SM de entradas digitales V DC/salidas digitales

SM 1223 DI 8 x 24 V DC, DQ 8 x 24 V DC (6ES7 223-1BH30-0XB0)	SM 1223 DI 16 x 24 V DC, DQ 16 x 24 V DC (6ES7 223-1BL30-0XB0)	Notas
		<p>① Para entradas NPN, conecte "-" a "M" (como se indica). Para entradas en fuente, conecte "+" a "M".</p>

Tabla A- 105 Asignación de pines de conectores para SM 1223 DI 8 x 24 V DC, DQ 8 x 24 V DC (6ES7 223-1BH30-0XB0)

Pin	X10	X11	X12	X13
1	L+/24 V DC	GND	Sin conexión	Sin conexión
2	M/24 V DC	Sin conexión	Sin conexión	Sin conexión
3	1M	2M	Sin conexión	Sin conexión
4	DI a.0	DI a.4	DQ a.0	DQ a.4
5	DI a.1	DI a.5	DQ a.1	DQ a.5
6	DI a.2	DI a.6	DQ a.2	DQ a.6
7	DI a.3	DI a.7	DQ a.3	DQ a.7

Tabla A- 106 Asignación de pines de conectores para SM 1223 DI 16 x 24 V DC, DQ 16 x 24 V DC (6ES7 223-1BL30-0XB0)

Pin	X10	X11	X12	X13
1	L+/24 V DC	GND	Sin conexión	Sin conexión
2	M/24 V DC	Sin conexión	Sin conexión	Sin conexión
3	1M	2M	Sin conexión	Sin conexión
4	DI a.0	DI b.0	DQ a.0	DQ b.0
5	DI a.1	DI b.1	DQ a.1	DQ b.1

Pin	X10	X11	X12	X13
6	DI a.2	DI b.2	DQ a.2	DQ b.2
7	DI a.3	DI b.3	DQ a.3	DQ b.3
8	DI a.4	DI b.4	DQ a.4	DQ b.4
9	DI a.5	DI b.5	DQ a.5	DQ b.5
10	DI a.6	DI b.6	DQ a.6	DQ b.6
11	DI a.7	DI b.7	DQ a.7	DQ b.7

A.6.5 Datos técnicos del módulo de entradas/salidas digitales AC SM 1223

Tabla A- 107 Especificaciones generales

Modelo	SM 1223 DI 8 x120/230 V AC / DQ 8 x relé
Referencia	6ES7 223-1QH30-0XB0
Dimensiones A x A x P (mm)	45 x 100 x 75 mm
Peso	190 gramos
Disipación de potencia	7,5 W
Consumo de corriente (bus SM)	120 mA
Consumo de corriente (24 V DC)	11 mA por salida en estado ON

Tabla A- 108 Entradas digitales

Modelo	SM 1223 DI 8 x 120/230 V AC / DQ 8 x relé
Número de entradas	8
Tipo	Tipo 1 IEC
Tensión nominal	120 V AC a 6 mA, 230 V AC a 9 mA
Tensión continua admisible	264 V AC
Sobretensión transitoria	--
Señal 1 lógica (mín.)	79 VAC a 2,5 mA
Señal 0 lógica (máx.)	20 VAC a 1 mA
Corriente de fuga (máx.)	1 mA
Aislamiento (campo a lógica)	1.500 V AC durante 1 minuto
Grupos de aislamiento ¹	4
Tiempos de retardo de las entradas	Típ. 0,2 a 12,8 ms, seleccionable por el usuario Máximo: -
Conexión de sensor de proximidad de 2 hilos (Bero) (máx.)	1 mA
Longitud del cable	No apantallado: 300 metros Apantallado: 500 metros
Número de entradas ON simultáneamente	8

¹ Los canales dentro de un grupo deben ser de la misma fase.

Tabla A- 109 Salidas digitales

Modelo	SM 1223 DI 8 x 120/230 V AC / DQ 8 x relé
Número de salidas	8
Tipo	Relé, contacto seco
Rango de tensión	5 a 30 V DC ó 5 a 250 V AC
Señal 1 lógica a intensidad máx.	--
Señal 0 lógica con carga de 10K Ω	--
Intensidad (máx.)	2,0 A
Carga de lámparas	30 W DC / 200 W AC
Resistencia en estado ON (contactos)	Máx. 0,2 Ω (si son nuevas)
Corriente de fuga por salida	--
Sobrecorriente momentánea	7 A si están cerrados los contactos
Protección contra sobrecargas	No
Aislamiento (campo a lógica)	1500 V AC durante 1 minuto (bobina a contacto) Ninguno (bobina a lógica)
Resistencia de aislamiento	100 M Ω mín. si son nuevas
Aislamiento entre contactos abiertos	750 V AC durante 1 minuto
Grupos de aislamiento	2
Intensidad por neutro (máx.)	10 A
Tensión de bloqueo inductiva	--
Retardo de conmutación (máx.)	10 ms
Frecuencia máxima de conmutación de relé	1 Hz
Vida útil mecánica (sin carga)	10.000.000 ciclos apertura/cierre
Vida útil de los contactos bajo carga nominal	1000.000 ciclos abierto/cerrado
Reacción al cambiar de RUN a STOP	Último valor o valor sustitutivo (valor predeterminado: 0)
Número de salidas ON simultáneamente	<ul style="list-style-type: none"> • 4 (no adyacentes) a 60 °C en horizontal o 50 °C en vertical • 8 a 55 °C en horizontal o 45 °C en vertical
Longitud de cable (metros)	500 m apantallado, 150 m no apantallado

A.6 Módulos de señales digitales (SMs)

Tabla A- 110 SM 1223 DI 8 x 120/230 V AC, DQ 8 x relé (6ES7 223-1QH30-0XB0)

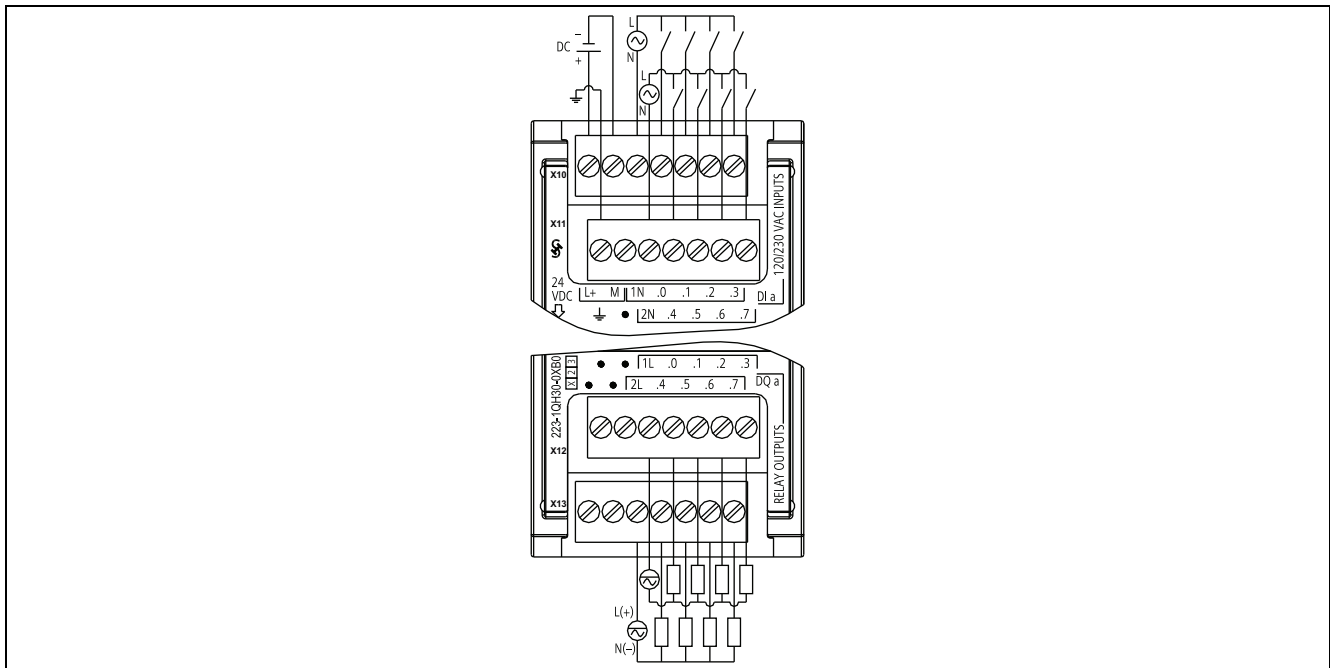


Tabla A- 111 Asignación de pines de conectores para SM 1223 DI 8 x 120/240 V AC, DQ 8 x relé (6ES7 223-1QH30-0XB0)

Pin	X10	X11	X12	X13
1	L+/24 V DC	GND	Sin conexión	Sin conexión
2	M/24 V DC	Sin conexión	Sin conexión	Sin conexión
3	1N	2N	1L	2L
4	DI a.0	DI a.4	DQ a.0	DQ a.4
5	DI a.1	DI a.5	DQ a.1	DQ a.5
6	DI a.2	DI a.6	DQ a.2	DQ a.6
7	DI a.3	DI a.7	DQ a.3	DQ a.7

A.7 Módulos de señales analógicas (SMs)

A.7.1 Datos técnicos del módulo de entradas analógicas SM 1231

Tabla A- 112 Especificaciones generales

Modelo	SM 1231 AI 4 x 13 bits	SM 1231 AI 8 x 13 bits	SM 1231 AI 4 x 16 bits
Referencia	6ES7 231-4HD30-0XB0	6ES7 231-4HF30-0XB0	6ES7 231-5ND30-0XB0
Dimensiones A x A x P (mm)	45 x 100 x 75	45 x 100 x 75	45 x 100 x 75
Peso	180 gramos	180 gramos	180 gramos
Disipación de potencia	2,2 W	2,3 W	2,0 W
Consumo de corriente (bus SM)	80 mA	90 mA	80 mA
Consumo de corriente (24 V DC)	45 mA	45 mA	65 mA

Tabla A- 113 Entradas analógicas

Modelo	SM 1231 AI 4 x 13 bits	SM 1231 AI 8 x 13 bits	SM 1231 AI 4 x 16 bits
Número de entradas	4	8	4
Tipo	Tensión o intensidad (diferencial): Seleccionable en grupos de 2		Tensión o intensidad (diferencial)
Rango	±10 V, ±5 V, ±2,5 V ó 0 a 20 mA		±10 V, ±5 V, ±2,5 V, ±1,25 V, 0 a 20 mA o 4 mA a 20 mA
Rango total (palabra de datos)	-27.648 a 27.648 tensión/0 a 27.648 intensidad		
Rango de sobreimpulso/subimpulso (palabra de datos) Consulte el apartado en que aparecen los rangos de entrada analógica de tensión e intensidad (Página 814).	Tensión: 32.511 a 27.649 / -27.649 a -32.512 Intensidad: 32.511 a 27.649 / 0 a -4864		Tensión: 32.511 a 27.649/ -27.649 a -32.512 Intensidad: (0-20 mA): a 32.511 a 27.649/ 0 a -4864; 4-20 mA: 32.511 a 27.649// -1 a -4864
Rebase por exceso/defecto (palabra de datos) Consulte el apartado en que aparecen los rangos de entrada de tensión e intensidad (Página 814).	Tensión: 32.767 a 32.512 / -32.513 a -32.768 Intensidad: 32.767 a 32.512 / -4865 a -32.768		Tensión: 32.767 a 32.512 / -32.513 a -32.768 Intensidad: 0 a 20 mA 32.767 a 32.512 / -4865 a -32.768 4-20 mA 32.767 a 32.512/-4865 a -32.768
Resolución	12 bits + bit de signo		15 bits + bit de signo
Tensión/intensidad soportada máxima	±35 V / ±40 mA		
Filtrado	Ninguno, débil, medio o fuerte Consulte el apartado en que aparecen los tiempos de respuesta a un escalón (Página 813).		

Datos técnicos

A.7 Módulos de señales analógicas (SMs)

Modelo	SM 1231 AI 4 x 13 bits	SM 1231 AI 8 x 13 bits	SM 1231 AI 4 x 16 bits
Supresión de ruido	400, 60, 50 ó 10 Hz Consulte el apartado en que aparecen las frecuencias de muestreo (Página 813).		
Impedancia de entrada	≥ 9 MΩ (tensión) / 280 Ω (intensidad)		≥ 1 MΩ (tensión)/ <315 Ω, >280 Ω (intensidad)
Aislamiento	Ninguno		
De campo a lógica	500 V AC		
Campo a 24 V DC	500 V AC		
24 V DC a lógica	500 V AC		
Entre canales	ninguno		
Precisión (25 °C/-20 a 60 °C)	±0,1% / ±0,2% de rango máximo		±0,1%/±0,3% de rango máximo
Principio de medición	Conversión de valor real		
Rechazo en modo común	40 dB, DC a 60 Hz		
Rango de señales operativo ¹	La tensión de señal más la tensión en modo común debe ser menor que +12 V y mayor que -12 V		
Longitud de cable (metros)	100 m, trenzado y apantallado		

¹ La aplicación de tensiones que estén fuera del rango operativo a un canal puede causar interferencias en los demás canales.

Tabla A- 114 Diagnóstico

Modelo	SM 1231 AI 4 x 13 bits	SM 1231 AI 8 x 13 bits	SM 1231 AI 4 x 16 bits
Rebase por exceso/defecto	Sí ¹	Sí	Sí
24 V DC, baja tensión	Sí	Sí	Sí
Rotura de hilo	--	--	Sólo en el rango 4 a 20 mA (si la entrada es inferior a -4164; 1,0 mA)

¹ Para SM 1231 AI 4 x 13 bits: Si se aplica una tensión superior a +30 V DC o inferior a -15 V DC a la entrada, el valor resultante se desconocerá y es posible que no se active el rebase por exceso o por defecto correspondiente.

Tabla A- 115 Diagramas de cableado de los SM de entradas analógicas

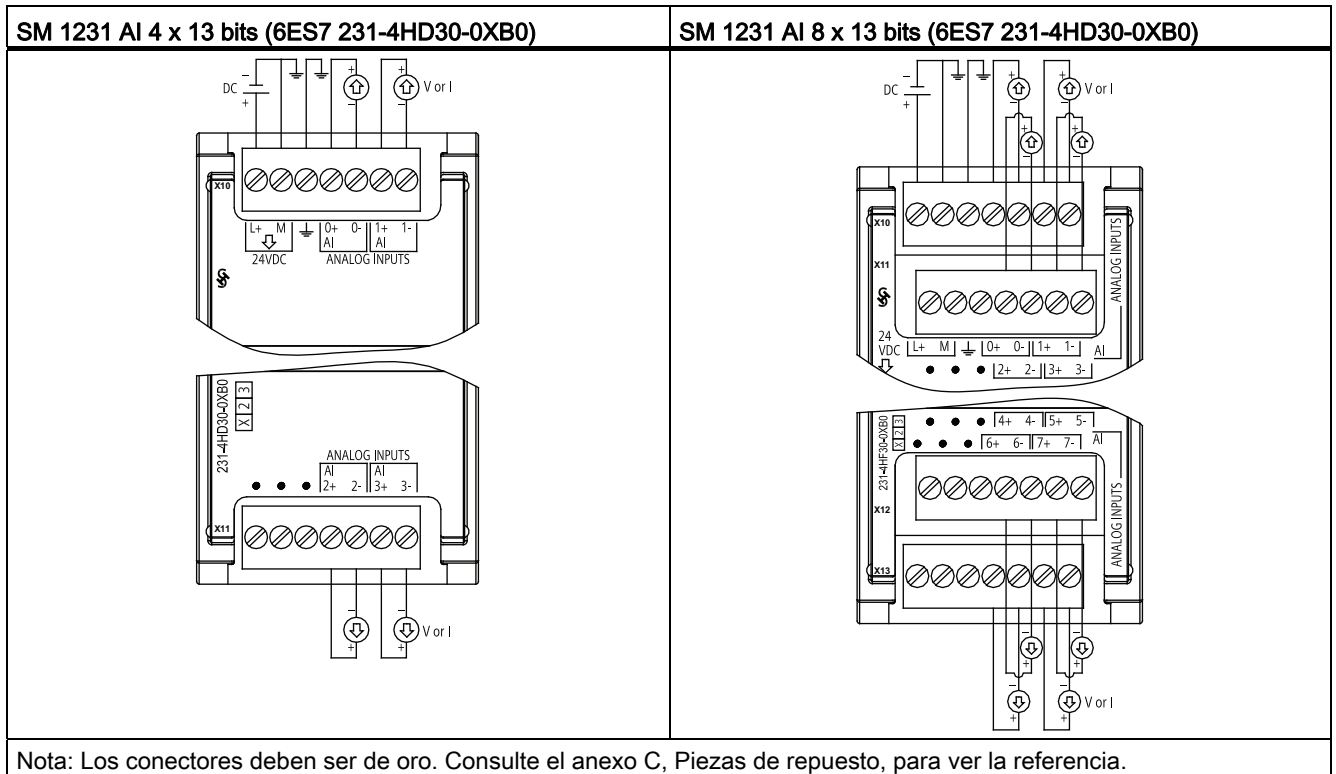


Tabla A- 116 Asignación de pines de conectores para SM 1231 AI 4 x 13 bits (6ES7 231-4HD30-0XB0)

Pin	X10 (oro)	X11 (oro)
1	L+/24 V DC	Sin conexión
2	M/24 V DC	Sin conexión
3	GND	Sin conexión
4	AI 0+	AI 2+
5	AI 0-	AI 2-
6	AI 1+	AI 3+
7	AI 1-	AI 3-

Tabla A- 117 Asignación de pines de conectores para SM 1231 AI 8 x 13 bits (6ES7 231-4HF30-0XB0)

Pin	X10 (oro)	X11 (oro)	X12 (oro)	X13 (oro)
1	L+/24 V DC	Sin conexión	Sin conexión	Sin conexión
2	M/24 V DC	Sin conexión	Sin conexión	Sin conexión
3	GND	Sin conexión	Sin conexión	Sin conexión
4	AI 0+	AI 2+	AI 4+	AI 6+
5	AI 0-	AI 2-	AI 4-	AI 6-
6	AI 1+	AI 3+	AI 5+	AI 7+
7	AI 1-	AI 3-	AI 5-	AI 7-

Tabla A- 118 Diagrama de cableado del SM de entrada analógica

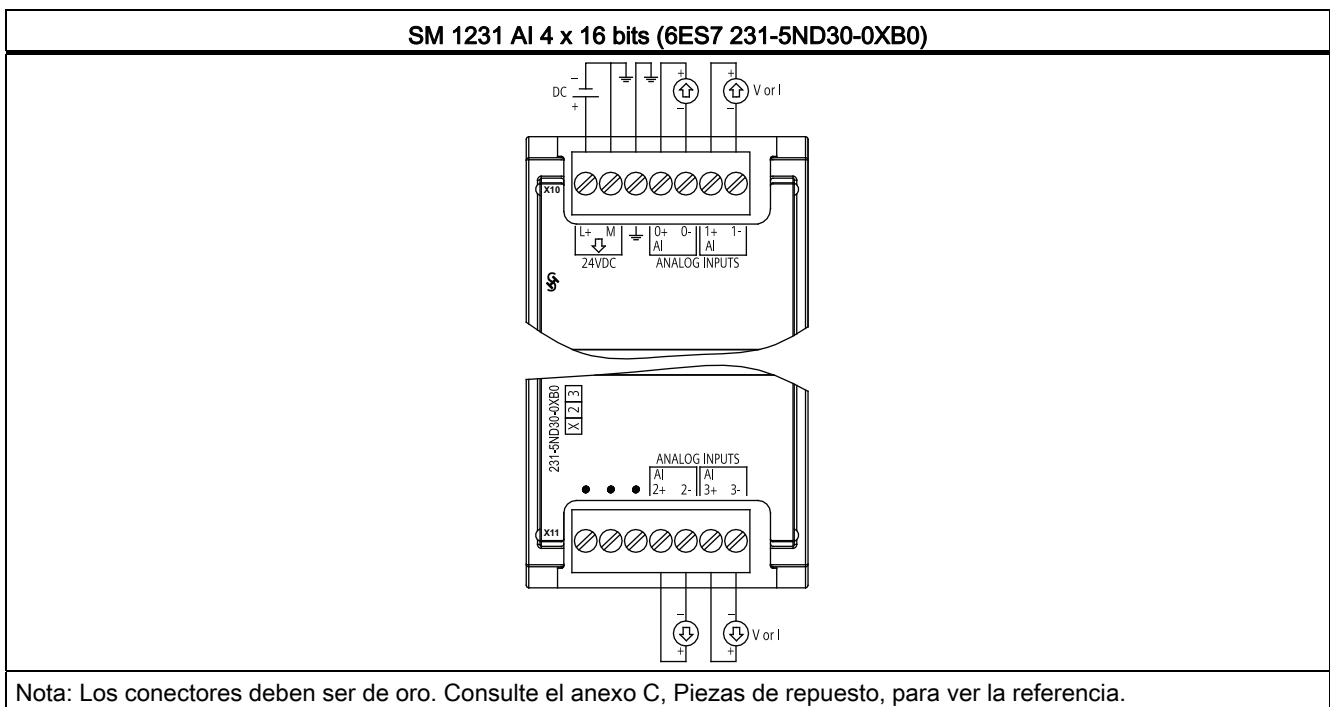


Tabla A- 119 Asignación de pines de conectores para SM 1231 AI 4 x 16 bits (6ES7 231-5ND30-0XB0)

Pin	X10 (oro)	X11 (oro)
1	L+/24 V DC	Sin conexión
2	M/24 V DC	Sin conexión
3	GND	Sin conexión
4	AI 0+	AI 2+
5	AI 0-	AI 2-
6	AI 1+	AI 3+
7	AI 1-	AI 3-

Nota

Las entradas analógicas que no se utilicen deben cortocircuitarse.

Cuando las entradas están configuradas en modo "corriente", no pasará corriente por las entradas a no ser que se suministre alimentación externa al módulo.

A.7.2 Datos técnicos del módulo de salidas analógicas SM 1232

Tabla A- 120 Especificaciones generales

Datos técnicos	SM 1232 AQ 2 x 14 bit	SM 1232 AQ 4 x 14 bit
Referencia	6ES7 232-4HB30-0XB0	6ES7 232-4HD30-0XB0
Dimensiones A x A x P (mm)	45 x 100 x 75	45 x 100 x 75
Peso	180 gramos	180 gramos
Disipación de potencia	1,8 W	2,0 W
Consumo de corriente (bus SM)	80 mA	80 mA
Consumo de corriente (24 V DC)	45 mA (sin carga)	45 mA (sin carga)

Tabla A- 121 Salidas analógicas

Datos técnicos	SM 1232 AQ 2 x 14 bit	SM 1232 AQ 4 x 14 bit
Número de salidas	2	4
Tipo	Tensión o intensidad	Tensión o intensidad
Rango	±10 V ó 0 a 20 mA	±10 V ó 0 a 20 mA
Resolución	Tensión: 14 bits Intensidad: 13 bits	Tensión: 14 bits Intensidad: 13 bits

Datos técnicos

A.7 Módulos de señales analógicas (SMs)

Datos técnicos	SM 1232 AQ 2 x 14 bit	SM 1232 AQ 4 x 14 bit
Rango total (palabra de datos)	Tensión: -27.648 a 27.648; intensidad: 0 a 27.648 Consulte los rangos de salida de tensión e intensidad (Página 815).	
Precisión (25 °C/-20 a 60 °C)	±0,3% / ±0,6% de rango máximo	
Tiempo de estabilización (95% del nuevo valor)	Tensión: 300 μS (R), 750 μS (1 uF) Intensidad: 600 μS (1 mH), 2 ms (10 mH)	
Impedancia de carga	Tensión: ≥ 1000 Ω Intensidad: ≤ 600 Ω	
Reacción al cambiar de RUN a STOP	Último valor o valor sustitutivo (valor predeterminado: 0)	
Aislamiento (campo a lógica)	Ninguno	
Longitud de cable (metros)	100 m, trenzado y apantallado	

Tabla A- 122 Diagnóstico

Datos técnicos	SM 1232 AQ 2 x 14 bit	SM 1232 AQ 4 x 14 bit
Rebase por exceso/defecto	Sí	Sí
Cortocircuito a tierra (sólo en modo de tensión)	Sí	Sí
Rotura de hilo (sólo en modo de intensidad)	Sí	Sí
24 V DC, baja tensión	Sí	Sí

Tabla A- 123 Diagramas de cableado de los SM de salidas analógicas

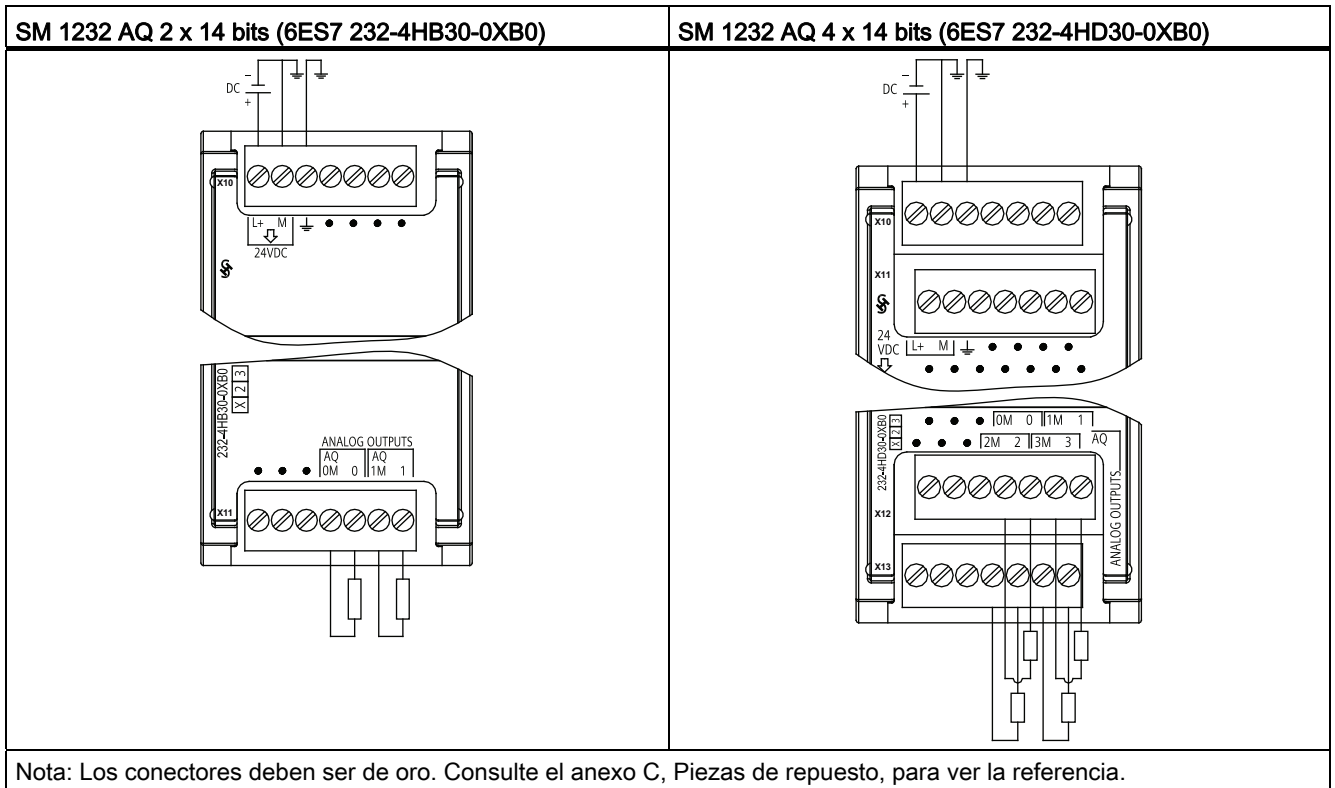


Tabla A- 124 Asignación de pines de conectores para SM 1232 AQ 2 x 14 bits (6ES7 232-4HB30-0XB0)

Pin	X10 (oro)	X11 (oro)
1	L+/24 V DC	Sin conexión
2	M/24 V DC	Sin conexión
3	GND	Sin conexión
4	Sin conexión	AQ 0M
5	Sin conexión	AQ 0
6	Sin conexión	AQ 1M
7	Sin conexión	AQ 1

Tabla A- 125 Asignación de pines de conectores para SM 1232 AQ 4 x 14 bits (6ES7 232-4HD30-0XB0)

Pin	X10 (oro)	X11 (oro)	X12 (oro)	X13 (oro)
1	L+/24 V DC	Sin conexión	Sin conexión	Sin conexión
2	M/24 V DC	Sin conexión	Sin conexión	Sin conexión
3	GND	Sin conexión	Sin conexión	Sin conexión
4	Sin conexión	Sin conexión	AQ 0M	AQ 2M
5	Sin conexión	Sin conexión	AQ 0	AQ 2
6	Sin conexión	Sin conexión	AQ 1M	AQ 3M
7	Sin conexión	Sin conexión	AQ 1	AIQ 3

A.7.3 Datos técnicos del módulo de entradas/salidas analógicas SM 1234

Tabla A- 126 Especificaciones generales

Datos técnicos	SM 1234 AI 4 x 13 bits / AQ 2 x 14 bits
Referencia	6ES7 234-4HE30-0XB0
Dimensiones A x A x P (mm)	45 x 100 x 75
Peso	220 gramos
Disipación de potencia	2,4 W
Consumo de corriente (bus SM)	80 mA
Consumo de corriente (24 V DC)	60 mA (sin carga)

Tabla A- 127 Entradas analógicas

Modelo	SM 1234 AI 4 x 13 bits / AQ 2 x 14 bits
Número de entradas	4
Tipo	Tensión o intensidad (diferencial): Seleccionable en grupos de 2
Rango	± 10 V, ± 5 V, $\pm 2,5$ V ó 0 a 20 mA
Rango total (palabra de datos)	-27.648 a 27.648
Rango de sobreimpulso/subimpulso (palabra de datos)	Tensión: 32.511 a 27.649 / -27.649 a -32.512 Intensidad: 32.511 a 27.649 / 0 a -4864 Consulte el apartado en que aparecen los rangos de entrada de tensión e intensidad (Página 814).
Rebase por exceso/defecto (palabra de datos)	Tensión: 32.767 a 32.512 / -32.513 a -32.768 Intensidad: 32.767 a 32.512 / -4865 a -32.768 Consulte el apartado en que aparecen los rangos de entrada de tensión e intensidad (Página 814).
Resolución	12 bits + bit de signo
Tensión/intensidad soportada máxima	± 35 V / ± 40 mA

Modelo	SM 1234 AI 4 x 13 bits / AQ 2 x 14 bits
Filtrado	Ninguno, débil, medio o fuerte Consulte el apartado en que aparecen los tiempos de respuesta a un escalón (Página 813).
Supresión de ruido	400, 60, 50 ó 10 Hz Consulte el apartado en que aparecen las frecuencias de muestreo (Página 813).
Impedancia de entrada	$\geq 9 \text{ M}\Omega$ (tensión) / 280Ω (intensidad)
Aislamiento (campo a lógica)	Ninguno
Precisión (25 °C/-20 a 60 °C)	$\pm 0,1\%$ / $\pm 0,2\%$ de rango máximo
Tiempo de conversión analógica-digital	625 μs (rechazo de 400 Hz)
Rechazo en modo común	40 dB, DC a 60 Hz
Rango de señales operativo ¹	La tensión de señal más la tensión en modo común debe ser menor que +12 V y mayor que -12 V
Longitud de cable (metros)	100 m, trenzado y apantallado

¹ La aplicación de tensiones que estén fuera del rango operativo a un canal puede causar interferencias en los demás canales.

Tabla A- 128 Salidas analógicas

Datos técnicos	SM 1234 AI 4 x 13 bits / AQ 2 x 14 bits
Número de salidas	2
Tipo	Tensión o intensidad
Rango	$\pm 10 \text{ V}$ ó 0 a 20 mA
Resolución	Tensión: 14 bits; intensidad: 13 bits
Rango total (palabra de datos)	Tensión: -27.648 a 27.648; intensidad: 0 a 27.648 Consulte el apartado en que aparecen los rangos de salida de tensión e intensidad (Página 815).
Precisión (25 °C/-20 a 60 °C)	$\pm 0,3\%$ / $\pm 0,6\%$ de rango máximo
Tiempo de estabilización (95% del nuevo valor)	Tensión: 300 μs (R), 750 μs (1 μF) Intensidad: 600 μs (1 mH), 2 ms (10 mH)
Impedancia de carga	Tensión: $\geq 1000 \Omega$ Intensidad: $\leq 600 \Omega$
Reacción al cambiar de RUN a STOP	Último valor o valor sustitutivo (valor predeterminado: 0)
Aislamiento (campo a lógica)	Ninguno
Longitud de cable (metros)	100 m, trenzado y apantallado

A.7 Módulos de señales analógicas (SMs)

Tabla A- 129 Diagnóstico

Modelo	SM 1234 AI 4 x 13 bits / AQ 2 x 14 bits
Rebase por exceso/defecto	Sí ¹
Cortocircuito a tierra (sólo en modo de tensión)	Sí en las salidas
Rotura de hilo (sólo en modo de intensidad)	Sí en las salidas
24 V DC, baja tensión	Sí

¹ Si se aplica una tensión superior a +30 V DC o inferior a -15 V DC a la entrada, el valor resultante se desconocerá y es posible que no se active el rebase por exceso o por defecto correspondiente.

Tabla A- 130 Diagramas de cableado de los SM de entradas/salidas analógicas

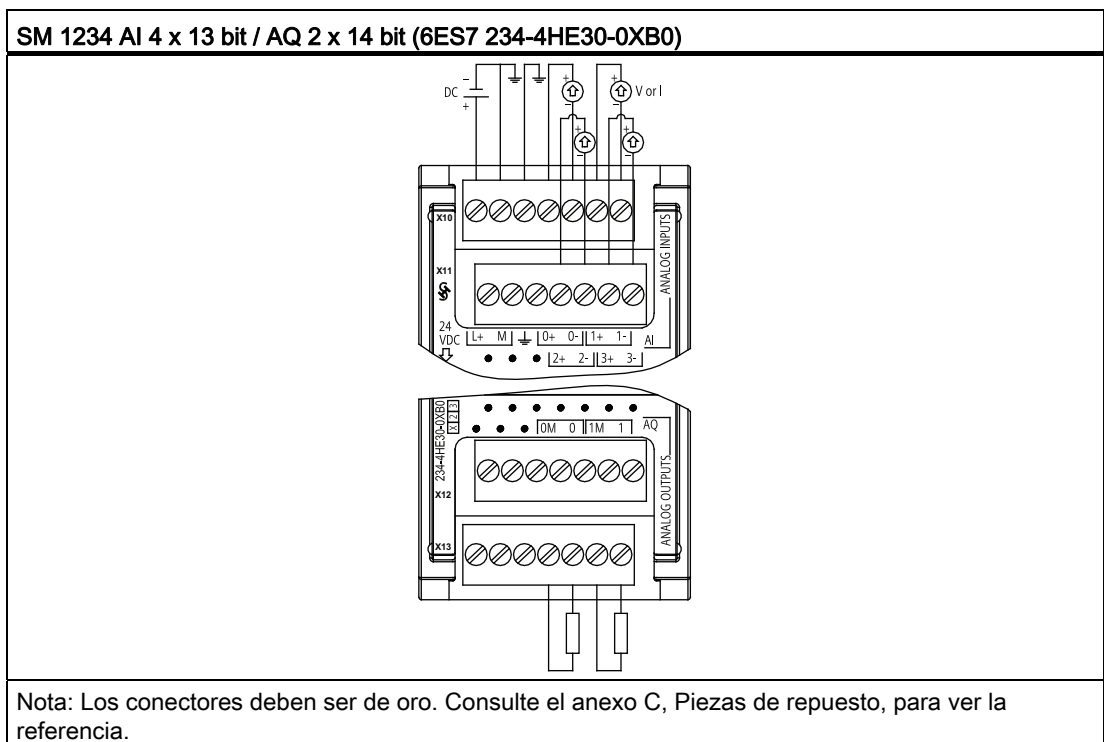


Tabla A- 131 Asignación de pines de conectores para SM 1234 AI 4 x 13 Bits / AQ 2 x 14 bits (6ES7 234-4HE30-0XB0)

Pin	X10 (oro)	X11 (oro)	X12 (oro)	X13 (oro)
1	L+/24 V DC	Sin conexión	Sin conexión	Sin conexión
2	M/24 V DC	Sin conexión	Sin conexión	Sin conexión
3	GND	Sin conexión	Sin conexión	Sin conexión
4	AI 0+	AI 2+	Sin conexión	AQ 0M
5	AI 0-	AI 2-	Sin conexión	AQ 0
6	AI 1+	AI 3+	Sin conexión	AQ 1M
7	AI 1-	AI 3-	Sin conexión	AIQ 1

Nota

Las entradas analógicas que no se utilicen deben cortocircuitarse.

Cuando las entradas están configuradas en modo "corriente", no pasará corriente por las entradas a no ser que se suministre alimentación externa al módulo.

A.7.4 Respuesta a un escalón de las entradas analógicas

Tabla A- 132 Respuesta a un escalón (ms), de 0 a rango máximo, medido al 95%

Selección de filtrado (valor medio de muestreo)	Reducción de ruido/supresión de frecuencias (selección del tiempo de integración)			
	400 Hz (2,5 ms)	60 Hz (16,6 ms)	50 Hz (20 ms)	10 Hz (100 ms)
Ninguno (1 ciclo): Sin media	4 ms	18 ms	22 ms	100 ms
Débil (4 ciclos): 4 muestreos	9 ms	52 ms	63 ms	320 ms
Medio (16 ciclos): 16 muestreos	32 ms	203 ms	241 ms	1200 ms
Fuerte (32 ciclos): 32 muestreos	61 ms	400 ms	483 ms	2410 ms
Tiempo de muestreo				
• 4 AI x 13 bits	• 0,625 ms	• 4,17 ms	• 5 ms	• 25 ms
• 8 AI x 13 bits	• 1,25 ms	• 4,17 ms	• 5 ms	• 25 ms
• 4 AI4 x 16 bits	• 0,417 ms	• 0,397 ms	• 0,400 ms	• 0,400 ms

A.7.5 Tiempo de muestreo y tiempos de actualización para entradas analógicas

Tabla A- 133 Tiempo de muestreo y tiempo de actualización

Supresión de frecuencias (tiempo de integración)	Tiempo de muestreo	Tiempo de actualización del módulo para todos los canales	
		SM de 4 canales	SM de 8 canales
400 Hz (2,5 ms)	<ul style="list-style-type: none"> SM de 4 canales: 0,625 ms SM de 8 canales: 1,250 ms 	0,625 ms	1,250 ms
60 Hz (16,6 ms)	4,170 ms	4,17 ms	4,17 ms
50 Hz (20 ms)	5,000 ms	5 ms	5 ms
10 Hz (100 ms)	25,000 ms	25 ms	25 ms

A.7.6 Rangos de medida de entradas analógicas de tensión

Tabla A- 134 Representación de entradas analógicas para tensión

Sistema		Rango de medida de tensión						
Decimal	Hexadecimal	±10 V	±5 V	±2,5 V	±1,25V	De 0 a 10 V		
32767	7FFF	11,851 V	5,926 V	2,963 V	1,481 V	Rebase por exceso	11,851 V	Rebase por exceso
32512	7F00							
32511	7EFF	11,759 V	5,879 V	2,940 V	1,470 V	Rango de sobreimpulso	11,759 V	Rango de sobreimpulso
27649	6C01							
27648	6C00	10 V	5 V	2,5 V	1,250 V	Rango nominal	10 V	Rango nominal
20736	5100	7,5 V	3,75 V	1,875 V	0,938 V		7,5 V	
1	1	361,7 µV	180,8 µV	90,4 µV	45,2 µV		361,7 µV	
0	0	0 V	0 V	0 V	0 V		0 V	
-1	FFFF						Los valores negativos no se soportan	
-20736	AF00	-7,5 V	-3,75 V	-1,875 V	-0,938 V			
-27648	9400	-10 V	-5 V	-2,5 V	-1,250 V			
-27649	93FF							
-32512	8100	-11,759 V	-5,879 V	-2,940 V	-1,470 V	Rango de subimpulso		
-32513	80FF					Rebase por defecto		
-32768	8000	-11,851 V	-5,926 V	-2,963 V	-1,481 V			

A.7.7 Rangos de medida de las entradas analógicas de intensidad

Tabla A- 135 Representación de entradas analógicas para intensidad

Sistema		Rango de medida de intensidad		
Decimal	Hexadecimal	De 0 mA a 20 mA	De 4 mA a 20 mA	
32767	7FFF	23,70 mA	22,96 mA	Rebase por exceso
32512	7F00			
32511	7EFF	23,52 mA	22,81 mA	Rango de sobreimpulso
27649	6C01			
27648	6C00	20 mA	20 mA	Rango nominal
20736	5100	15 mA	16 mA	
1	1	723,4 nA	4 mA + 578,7 nA	
0	0	0 mA	4 mA	
-1	FFFF			
-4864	ED00	-3,52 mA	1,185 mA	
-4865	ECFF			Rebase por defecto
-32768	8000			

A.7.8 Rangos de medición de salidas (AQ) de tensión e intensidad (SB y SM)

Tabla A- 136 Representación de salidas analógicas para tensión

Sistema		Rango de salida de tensión	
Decimal	Hexadecimal	±10 V	
32767	7FFF	V. nota 1	Rebase por exceso
32512	7F00	V. nota 1	
32511	7EFF	11,76 V	Rango de sobreimpulso
27649	6C01		
27648	6C00	10 V	Rango nominal
20736	5100	7,5 V	
1	1	361,7 μ V	
0	0	0 V	
-1	FFFF	-361,7 μ V	
-20736	AF00	-7,5 V	
-27648	9400	-10 V	
-27649	93FF		
-32512	8100	-11,76 V	Rango de subimpulso
-32513	80FF	V. nota 1	
-32768	8000	V. nota 1	

¹ En una condición de rebase por exceso o por defecto, la reacción de las salidas analógicas corresponderá a las propiedades ajustadas en la configuración de dispositivos para el módulo de señales analógico. En el parámetro "Reacción a STOP de la CPU", seleccione: "Aplicar valor sustitutivo" o "Mantener último valor".

Tabla A- 137 Representación de salidas analógicas para intensidad

Sistema		Rango de salida de intensidad	
Decimal	Hexadecimal	0 mA a 20 mA	
32767	7FFF	V. nota 1	Rebase por exceso
32512	7F00	V. nota 1	
32511	7EFF	23,52 mA	Rango de sobreimpulso
27649	6C01		
27648	6C00	20 mA	Rango nominal
20736	5100	15 mA	
1	1	723,4 nA	
0	0	0 mA	

¹ En una condición de rebase por exceso o por defecto, la reacción de las salidas analógicas corresponderá a las propiedades ajustadas en la configuración de dispositivos para el módulo de señales analógico. En el parámetro "Reacción a STOP de la CPU", seleccione: "Aplicar valor sustitutivo" o "Mantener último valor".

A.8 Módulos de señales RTD y de termopar (SMs)

A.8.1 SM 1231 de termopar

Tabla A- 138 Especificaciones generales

Modelo	SM 1231 AI 4 x 16 bits TC	SM 1231 AI 8 x 16 bits TC
Referencia	6ES7 231-5QD30-0XB0	6ES7 231-5QF30-0XB0
Dimensiones A x A x P (mm)	45 x 100 x 75	45 x 100 x 75
Peso	180 gramos	190 gramos
Disipación de potencia	1,5 W	1,5 W
Consumo de corriente (bus SM)	80 mA	80 mA
Consumo de corriente (24 V DC) ¹	40 mA	40 mA

¹ 20,4 a 28,8 V DC (clase 2, potencia limitada o alimentación de sensor por PLC)

Tabla A- 139 Entradas analógicas

Modelo	SM 1231 AI 4 x 16 bits TC	SM 1231 AI 8 x 16 bits TC
Número de entradas	4	8
Rango	Véase la tabla de selección de termopares (Página 820).	Véase la tabla de selección de termopares (Página 820).
Rango nominal (palabra de datos)		
Rango de saturación máximo/mínimo (palabra de datos)		
Rebase por exceso/defecto (palabra de datos)		
Resolución		
Temperatura	0,1 °C/0,1 °F	0,1 °C/0,1 °F
Tensión	Signo más (+) de 15 bits	Signo más (+) de 15 bits
Tensión soportada máxima	± 35 V	± 35 V
Supresión de ruido	85 dB para el ajuste de filtro seleccionado (10 Hz, 50 Hz, 60 Hz ó 400 Hz)	85 dB para el ajuste de filtro seleccionado (10 Hz, 50 Hz, 60 Hz ó 400 Hz)
Rechazo en modo común	> 120 dB a 120 V AC	> 120 dB a 120 V AC
Impedancia	≥ 10 MΩ	≥ 10 MΩ
Aislamiento galvánico		
De campo a lógica	500 V AC	500 V AC
De campo a 24 V DC	500 V AC	500 V AC
24 V DC a circuito lógico	500 V AC	500 V AC
Entre canales	120 V AC	120 V AC
Precisión	Véase la tabla de selección de termopares (Página 820).	Véase la tabla de selección de termopares (Página 820).
Repetitividad	±0,05% FS	±0,05% FS
Principio de medición	Integrador	Integrador

Modelo	SM 1231 AI 4 x 16 bits TC	SM 1231 AI 8 x 16 bits TC
Tiempo de actualización del módulo	Véase la tabla de selección de reducción de ruido (Página 820).	Véase la tabla de selección de reducción de ruido (Página 820).
Error de unión fría	±1,5°C	±1,5°C
Longitud de cable (metros)	100 metros hasta el sensor (máx.)	100 metros hasta el sensor (máx.)
Resistencia del cable	100 Ω máx.	100 Ω máx.

Tabla A- 140 Diagnóstico

Modelo	SM 1231 AI 4 x 16 bits TC	SM 1231 AI 8 x 16 bits TC
Rebase por exceso/defecto ¹	Sí	Sí
Rotura de hilo (sólo en modo de intensidad) ²	Sí	Sí
Baja tensión 24 V DC ¹	Sí	Sí

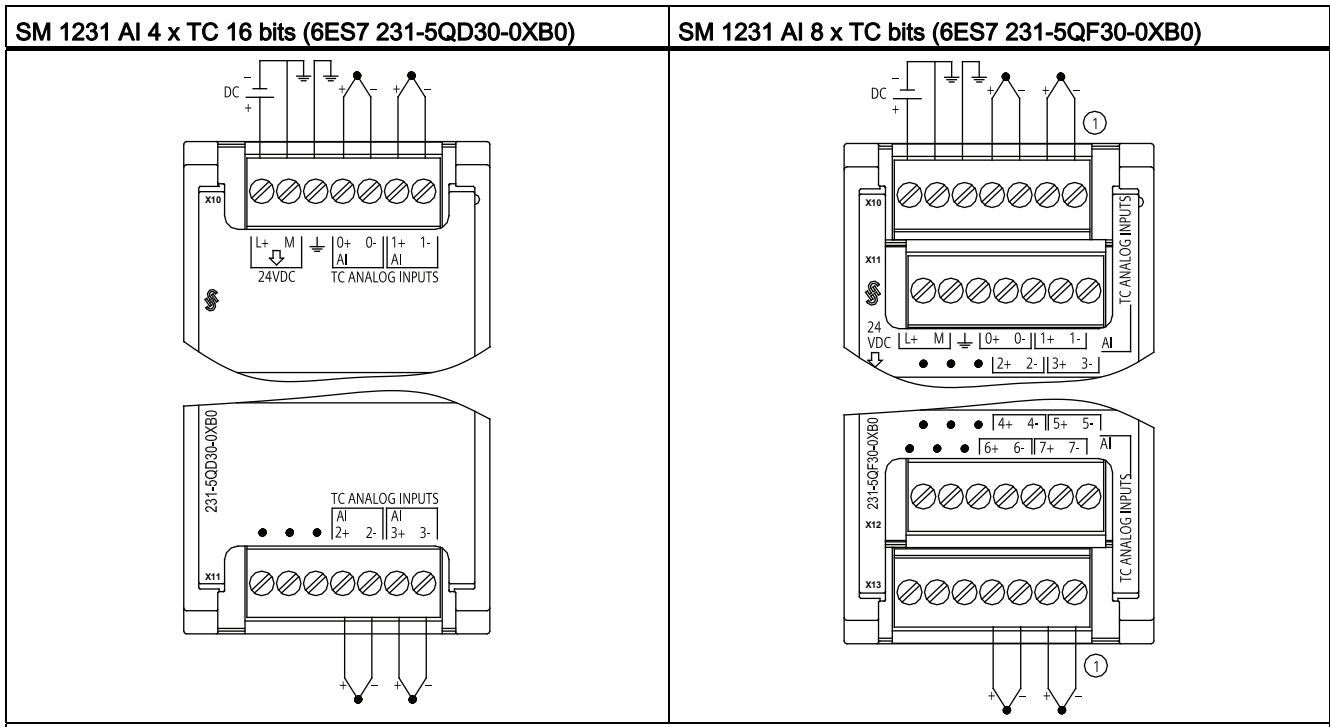
¹ La información de las alarmas de diagnóstico de baja tensión y de rebase por exceso/defecto se indicará en los valores de datos analógicos aunque las alarmas estén deshabilitadas en la configuración del módulo.

² Si la alarma de rotura de hilo está deshabilitada y se presenta una condición de rotura de hilo en la línea del sensor, el módulo puede señalar valores aleatorios.

El módulo de señales analógicas de termopar SM 1231 (TC) mide el valor de la tensión conectada a las entradas del módulo. El tipo de medición de temperatura puede ser "termopar" o "tensión".

- "Termopar": el valor se expresará en grados multiplicados por diez (p. ej. 25,3 grados se expresarán como 253 decimales).
- "Tensión": el valor máximo del rango nominal serán 27648 decimales.

Tabla A- 141 Diagramas de cableado de los SM de termopar



Nota: Los conectores deben ser de oro. Consulte el anexo C, Piezas de repuesto, para ver la referencia.

① Para mejorar la claridad, TC 2, 3, 4 y 5 no se muestran conectados.

Tabla A- 142 Asignación de pines de conectores para SM 1231 AI 4 x TC 16 bits (6ES7 231-5QD30-0XB0)

Pin	X10 (oro)	X11 (oro)
1	L+/24 V DC	Sin conexión
2	M/24 V DC	Sin conexión
3	GND	Sin conexión
4	AI 0+/TC	AI 2+/TC
5	AI 0-/TC	AI 2-/TC
6	AI 1+/TC	AI 3+/TC
7	AI 1-/TC	AI 3-/TC

Tabla A- 143 Asignación de pines de conectores para SM 1231 AI 8 x TC bits (6ES7 231-5QF30-0XB0)

Pin	X10 (oro)	X11 (oro)	X12 (oro)	X13 (oro)
1	L+/24 V DC	Sin conexión	Sin conexión	Sin conexión
2	M/24 V DC	Sin conexión	Sin conexión	Sin conexión
3	GND	Sin conexión	Sin conexión	Sin conexión
4	AI 0+/TC	AI 2+/TC	AI 4 I-/TC	AI 6 I-/TC
5	AI 0-/TC	AI 2-/TC	AI 4 I+/TC	AI 6 I+/TC
6	AI 1+/TC	AI 3+/TC	AI 5 M-/TC	AI 7 M-/TC
7	AI 1-/TC	AI 3-/TC	AI 5 M+/TC	AI 7 M+/TC

Nota

Las entradas analógicas que no se utilicen deben cortocircuitarse.

Los canales de termopar que no se utilizan pueden desactivarse. No se producirá ningún error si se desactiva un canal no utilizado.

A.8.1.1 Funcionamiento básico de un termopar

Los termopares se forman por la unión de dos metales diferentes que se conectan eléctricamente produciendo una tensión. La tensión generada es proporcional a la temperatura de la unión. Se trata de una tensión pequeña; un microvoltio puede representar varios grados. La medición de temperatura con termopares consiste en medir la tensión de un termopar, compensar las uniones adicionales y linealizar posteriormente el resultado.

Cuando se conecta un termopar al módulo de termopar SM 1231, los dos hilos de metales distintos se unen al conector de señales del módulo. El punto en el que los dos hilos diferentes se unen el uno con el otro constituye el termopar del sensor.

Dos termopares adicionales se forman donde los dos hilos diferentes se unen al conector de señales. La temperatura del conector genera una tensión que se suma a la del termopar del sensor. Si no se corrige esta tensión, la temperatura indicada será diferente de la temperatura del sensor.

La compensación de unión fría se utiliza para compensar el termopar del conector. Las tablas de termopares se basan en una temperatura de referencia que, por lo general, es de cero grados centígrados. La compensación de unión fría compensa el conector a cero grados centígrados. La compensación restablece la tensión sumada por los termopares del conector. La temperatura del módulo se mide internamente y se convierte luego a un valor a sumar a la conversión del sensor. La conversión del sensor corregida se linealiza entonces utilizando las tablas de termopares.

Para optimizar el funcionamiento de la compensación de unión fría es necesario colocar el módulo de termopar en un entorno térmicamente estable. Una variación lenta (inferior a 0,1° C/minuto) del módulo a temperatura ambiente se compensa correctamente dentro de las especificaciones del módulo. Si hay corriente de aire a través del módulo también se producirán errores de compensación de unión fría.

A.8 Módulos de señales RTD y de termopar (SMs)

Si se requiere una mejor compensación del error de unión fría, se puede utilizar un bloque de terminales isotérmico. El módulo de termopar permite utilizar un bloque de terminales con una referencia de 0° C o 50° C.

A.8.1.2 Tablas de selección de termopares para el SM 1231

Los rangos y la exactitud de los diferentes tipos de termopares soportados por el módulo de señales de termopar SM 1231 se indican en la tabla siguiente.

Tabla A- 144 Tabla de selección de termopares del SM 1231

Tipo	Rango de saturación mínimo ¹	Rango nominal límite inferior	Rango nominal límite superior	Rango de saturación máximo ²	Exactitud rango ^{3,4} normal a 25 °C	Exactitud rango ^{1, 2} normal de -20 °C a 60 °C
J	-210,0°C	-150,0°C	1200,0°C	1450,0°C	±0,3°C	±0,6°C
K	-270,0°C	-200,0°C	1372,0°C	1622,0°C	±0,4°C	±1,0°C
T	-270,0°C	-200,0°C	400,0°C	540,0°C	±0,5°C	±1,0°C
E	-270,0°C	-200,0°C	1000,0°C	1200,0°C	±0,3°C	±0,6°C
R & S	-50,0°C	100,0°C	1768,0°C	2019,0°C	±1,0°C	±2,5°C
N	-270,0°C	-200,0°C	1300,0°C	1550,0°C	±1,0°C	±1,6°C
C	0,0°C	100,0°C	2315,0°C	2500,0°C	±0,7°C	±2,7°C
TXK/XK(L)	-200,0°C	-150,0°C	800,0°C	1050,0°C	±0,6°C	±1,2°C
Tensión	-32512	-27648 -80mV	27648 80mV	32511	±0,05%	±0,1%

- 1 Los valores de termopar inferiores al valor del rango de saturación mínimo se notifican como -32768.
- 2 Los valores de termopar superiores al valor del rango de saturación máximo se notifican como 32767.
- 3 El error de la unión fría interna es de ±1,5 °C en todos los rangos. Esto debe añadirse al error en esta tabla. Para cumplir estas especificaciones, el módulo requiere como mínimo 30 minutos de calentamiento.
- 4 En presencia de radiofrecuencia radiada de 970 MHz a 990 MHz, la exactitud del SM 1231 AI 4 x 16 bits TC puede verse reducida.

Tabla A- 145 Reducción de ruido y tiempos de actualización para el SM 1231 termopar

Selección de supresión de frecuencias	Tiempo de integración	Tiempo de actualización de módulo de 4 canales (segundos)	Tiempo de actualización de módulo de 8 canales (segundos)
400 Hz (2,5 ms)	10 ms ¹	0,143	0,285
60 Hz (16,6 ms)	16,67 ms	0,223	0,445
50 Hz (20 ms)	20 ms	0,263	0,525
10 Hz (100 ms)	100 ms	1,225	2,450

- 1 Para mantener la resolución y exactitud del módulo con la supresión de 400 Hz, el tiempo de integración es de 10 ms. Esta selección también suprime perturbaciones de 100 Hz y 200 Hz.

Para medir termopares se recomienda utilizar un tiempo de integración de 100 ms. El uso de tiempos de integración inferiores aumentará el error de repetibilidad de las lecturas de temperatura.

Nota

Después de aplicar tensión al módulo, éste lleva a cabo una calibración interna del convertor de analógico a digital. Durante este tiempo, el módulo indica un valor de 32767 en cada canal hasta que haya información válida disponible en el canal respectivo. Es posible que el programa de usuario deba autorizar este tiempo de inicialización. Como la configuración del módulo puede modificar la longitud del tiempo de inicialización, es conveniente verificar el comportamiento del módulo en la configuración. Si es necesario, se puede incluir lógica en el programa de usuario para adaptarse al tiempo de inicialización del módulo.

Representación de los valores analógicos de termopar tipo J

En la tabla siguiente se muestra una representación de los valores analógicos de termopares tipo J.

Tabla A- 146 Representación de los valores analógicos de termopares tipo J

Tipo J en °C	Unidades		Tipo J en °F	Unidades		Rango
	Decimal	Hexadecimal		Decimal	Hexadecimal	
> 1450.0	32767	7FFF	> 2642.0	32767	7FFF	Desbordamiento
1450.0	14500	38A4	2642.0	26420	6734	Sobre rango
:	:	:	:	:	:	
1200.1	12001	2EE1	2192.2	21922	55A2	Rango nominal
:	:	:	:	:	:	
1200.0	12000	2EE0	2192.0	21920	55A0	Rango nominal
:	:	:	:	:	:	
-150.0	-1500	FA24	-238.0	-2380	F6B4	Rebase por defecto ¹
< -150.0	-32768	8000	< -238.0	-32768	8000	

¹ Un cableado defectuoso (por ejemplo, inversión de polaridad o entradas abiertas) o error del sensor en el rango negativo (por ejemplo, tipo erróneo de termopar) pueden provocar que el módulo de termopar señale un rebase por defecto.

A.8.2 SM 1231 RTD

Datos técnicos del SM 1231 RTD

Tabla A- 147 Especificaciones generales

Datos técnicos	SM 1231 AI 4 x RTD x 16 bits	SM 1231 AI 8 x RTD x 16 bits
Referencia	6ES7 231-5PD30-0XB0	6ES7 231-5PF30-0XB0
Dimensiones A x A x P (mm)	45 x 100 x 75	70 x 100 x 75
Peso	220 gramos	270 gramos
Disipación de potencia	1,5 W	1,5 W
Consumo de corriente (bus SM)	80 mA	90 mA
Consumo de corriente (24 V DC) ¹	40 mA	40 mA

¹ 20,4 a 28,8 V DC (clase 2, potencia limitada o alimentación de sensor de la CPU)

Tabla A- 148 Entradas analógicas

Datos técnicos	SM 1231 AI 4 x RTD x 16 bits	SM 1231 AI 8 x RTD x16 bits
Número de entradas	4	8
Tipo	RTD y Ω indicado por el módulo	RTD y Ω indicado por el módulo
Rango	Véase la tabla de selección de sensores RTD (Página 825).	Véase la tabla de selección de sensores RTD (Página 825).
Rango nominal (palabra de datos)		
Rango de sobreimpulso/subimpulso (palabra de datos)		
Rebase por exceso/defecto (palabra de datos)		
Resolución		
Temperatura	0,1 °C/0,1 °F	0,1 °C/0,1 °F
Resistencia	Signo más (+) de 15 bits	Signo más (+) de 15 bits
Tensión soportada máxima	± 35 V	± 35 V
Supresión de ruido	85 dB para la reducción de ruido seleccionada (10 Hz, 50 Hz, 60 Hz o 400 Hz)	85 dB para la reducción de ruido seleccionada (10 Hz, 50 Hz, 60 Hz o 400 Hz)
Rechazo en modo común	> 120dB	> 120dB
Impedancia	≥ 10 M Ω	≥ 10 M Ω
Aislamiento galvánico		
De campo a lógica	500 V AC	500 V AC
De campo a 24 V DC	500 V AC	500 V AC
24 V DC a circuito lógico	500 V AC	500 V AC
Aislamiento entre canales	Ninguno	Ninguno
Precisión	Véase la tabla de selección de sensores RTD (Página 825).	Véase la tabla de selección de sensores RTD (Página 825).
Repetitividad	±0,05% FS	±0,05% FS
Disipación máxima del sensor	0,5 mW	0,5 mW

Datos técnicos	SM 1231 AI 4 x RTD x 16 bits	SM 1231 AI 8 x RTD x16 bits
Principio de medición	Integrador	Integrador
Tiempo de actualización del módulo	Véase la tabla de selección de reducción de ruido (Página 825).	Véase la tabla de selección de reducción de ruido (Página 825).
Longitud de cable (metros)	100 metros hasta el sensor (máx.)	100 metros hasta el sensor (máx.)
Resistencia del cable	20 Ω, 2,7 Ω para 10 Ω RTD máx.	20 Ω, 2,7 Ω para 10 Ω RTD máx.

Tabla A- 149 Diagnóstico

Datos técnicos	SM 1231 AI 4 x RTD x 16 bits	SM 1231 AI 8 x RTD x16 bits
Rebase por exceso/defecto ^{1,2}	Sí	Sí
Rotura de hilo ³	Sí	Sí
Baja tensión 24 V DC ¹	Sí	Sí

¹ La información de las alarmas de diagnóstico de baja tensión y de rebase por exceso/defecto se indicará en los valores de datos analógicos aunque las alarmas estén deshabilitadas en la configuración del módulo.

² La detección de rebase por defecto nunca está habilitada para los rangos de resistencia.

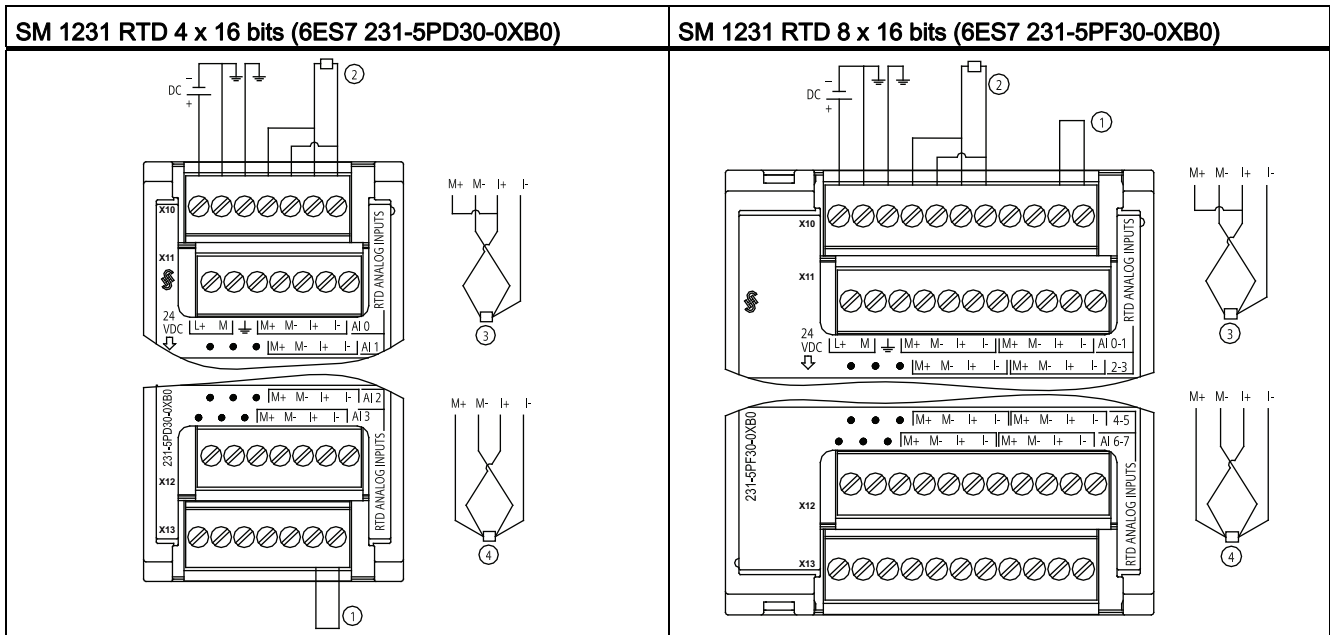
³ Si la alarma de rotura de hilo está deshabilitada y se presenta una condición de rotura de hilo en la línea del sensor, el módulo puede señalar valores aleatorios.

El módulo de señales analógico SM 1231 RTD mide el valor de la resistencia conectada a las entradas del módulo. El tipo de medición puede elegirse entre "resistor" y "resistor térmico".

- "Resistor": el valor máximo del rango nominal serán 27648 decimales.
- "Resistor térmico": el valor se expresará en grados multiplicados por diez (p. ej. 25,3 grados se expresarán como 253 decimales).

El módulo SM 1231 RTD soporta mediciones con base en conexiones a 2, 3 y 4 hilos que van al sensor de resistencia.

Tabla A- 150 Diagramas de cableado de SM RTD



- ① Entradas de bucle RTD no utilizadas
- ② RTD a 2 hilos ③ RTD a 3 hilos ④ RTD a 4 hilos

NOTA: Nota: Los conectores deben ser de oro. Consulte el anexo C, Piezas de repuesto, para ver la referencia.

Tabla A- 151 Asignación de pines de conectores para SM 1231 RTD 4 x 16 bits (6ES7 231-5PD30-0XB0)

Pin	X10 (oro)	X11 (oro)	X12 (oro)	X13 (oro)
1	L+/24 V DC	Sin conexión	Sin conexión	Sin conexión
2	M/24 V DC	Sin conexión	Sin conexión	Sin conexión
3	GND	Sin conexión	Sin conexión	Sin conexión
4	AI 0 M+/RTD	AI 1 M+/RTD	AI 2 M+/RTD	AI 3 M+/RTD
5	AI 0 M-/RTD	AI 1 M-/RTD	AI 2 M-/RTD	AI 3 M-/RTD
6	AI 0 I+/RTD	AI 1 I+/RTD	AI 2 I+/RTD	AI 3 I+/RTD
7	AI 0 I-/RTD	AI 1 I-/RTD	AI 2 I-/RTD	AI 3 I-/RTD

Tabla A- 152 Asignación de pines de conectores para SM 1231 RTD 8 x 16 bits (6ES7 231-5PF30-0XB0)

Pin	X10 (oro)	X11 (oro)	X12 (oro)	X13 (oro)
1	L+/24 V DC	Sin conexión	Sin conexión	Sin conexión
2	M/24 V DC	Sin conexión	Sin conexión	Sin conexión
3	GND	Sin conexión	Sin conexión	Sin conexión
4	AI 0 M+/RTD	AI 2 M+/RTD	AI 4 M+/RTD	AI 6 M+/RTD
5	AI 0 M-/RTD	AI 2 M-/RTD	AI 4 M-/RTD	AI 6 M-/RTD
6	AI 0 I+/RTD	AI 2 I+/RTD	AI 4 I+/RTD	AI 6 I+/RTD
7	AI 0 I-/RTD	AI 2 I-/RTD	AI 4 I-/RTD	AI 6 I-/RTD
8	AI 1 M+/RTD	AI 3 M+/RTD	AI 5 M+/RTD	AI 7 M+/RTD
9	AI 1 M-/RTD	AI 3 M-/RTD	AI 5 M-/RTD	AI 7 M-/RTD
10	AI 1 I+/RTD	AI 3 I+/RTD	AI 5 I+/RTD	AI 7 I+/RTD
11	AI 1 I-/RTD	AI 3 I-/RTD	AI 5 I-/RTD	AI 7 I-/RTD

Nota

Los canales RTD que no se utilizan pueden desactivarse. No se producirá ningún error si se desactiva un canal no utilizado.

El módulo RTD necesita un bucle de corriente continuo para eliminar el tiempo de estabilización suplementario que se agrega automáticamente a un canal no utilizado que no se ha desactivado. El módulo RTD debe tener conectado un resistor para fines de coherencia (como la conexión RTD a 2 hilos).

A.8.2.1 Tablas de selección para el SM 1231 RTD

Tabla A- 153 Rangos y precisión de los diferentes sensores soportados por los módulos RTD

Coefficiente de temperatura	Tipo de RTD	Rango de saturación mínimo ¹	Rango nominal límite inferior	Rango nominal límite superior	Rango de saturación máximo ²	Exactitud del rango normal a 25 °C	Exactitud del rango normal de -20 °C a 60 °C
Pt 0,003850 ITS90 DIN EN 60751	Pt 10	-243,0°C	-200,0°C	850,0°C	1000,0°C	±1,0°C	±2,0°C
	Pt 50	-243,0°C	-200,0°C	850,0°C	1000,0°C	±0,5°C	±1,0°C
	Pt 100						
	Pt 200						
	Pt 500						
Pt 0,003902 Pt 0,003916 Pt 0,003920	Pt 100	-243,0°C	-200,0°C	850,0°C	1000,0°C	±0,5°C	±1,0°C
	Pt 200	-243,0°C	-200,0°C	850,0°C	1000,0°C	±0,5°C	±1,0°C
	Pt 500						

A.8 Módulos de señales RTD y de termopar (SMs)

Coefficiente de temperatura	Tipo de RTD	Rango de saturación mínimo ¹	Rango nominal límite inferior	Rango nominal límite superior	Rango de saturación máximo ²	Exactitud del rango normal a 25 °C	Exactitud del rango normal de -20 °C a 60 °C
	Pt 1000						
Pt 0,003910	Pt 10	-273,2°C	-240,0°C	1100,0°C	1295°C	±1,0°C	±2,0°C
	Pt 50	-273,2°C	-240,0°C	1100,0°C	1295°C	±0,8°C	±1,6°C
	Pt 100						
	Pt 500						
Ni 0,006720 Ni 0,006180	Ni 100	-105,0°C	-60,0°C	250,0°C	295,0°C	±0,5°C	±1,0°C
	Ni 120						
	Ni 200						
	Ni 500						
	Ni 1000						
LG-Ni 0,005000	LG-Ni 1000	-105,0°C	-60,0°C	250,0°C	295,0°C	±0,5°C	±1,0°C
Ni 0,006170	Ni 100	-105,0°C	-60,0°C	180,0°C	212,4°C	±0,5°C	±1,0°C
Cu 0,004270	Cu 10	-240,0°C	-200,0°C	260,0°C	312,0°C	±1,0°C	±2,0°C
Cu 0,004260	Cu 10	-60,0°C	-50,0°C	200,0°C	240,0°C	±1,0°C	±2,0°C
	Cu 50						
	Cu 100						
Cu 0,004280	Cu 10	-240,0°C	-200,0°C	200,0°C	240,0°C	±1,0°C	±2,0°C
	Cu 50	-240,0°C	-200,0°C	200,0°C	240,0°C	±0,7°C	±1,4°C
	Cu 100						

¹ Los valores de RTD inferiores al valor del rango de saturación mínimo se notifican como -32768.

² Los valores de RTD superiores al valor del rango de saturación máximo se notifican como +32767.

Tabla A- 154 Resistencia

Rango	Rango de saturación mínimo	Rango nominal límite inferior	Rango nominal límite superior	Rango de saturación máximo ¹	Exactitud del rango normal a 25 °C	Exactitud del rango normal de -20 °C a 60 °C
150 Ω	n/a	0 (0 Ω)	27648 (150 Ω)	176,383 Ω	±0,05%	±0,1%
300 Ω	n/a	0 (0 Ω)	27648 (300 Ω)	352,767 Ω	±0,05%	±0,1%
600 Ω	n/a	0 (0 Ω)	27648 (600 Ω)	705,534 Ω	±0,05%	±0,1%

¹ Los valores de resistencia superiores al valor del rango de saturación máximo se notifican como +32767.

Nota

El módulo reporta 32767 en todo canal que esté activado y que no tenga conectado ningún sensor. Si la detección de rotura de hilo también está habilitada, en el módulo se encenderán de forma intermitente los LEDs rojos correspondientes.

En caso de utilizar rangos de 500 Ω y 1000 Ω del RTD con otros resistores de menor valor, el error puede aumentar al doble del error especificado.

La exactitud más elevada se alcanza para los rangos de 10 Ω del RTD con conexiones a 4 hilos.

La resistencia de los hilos en el modo a 2 hilos dará lugar a un error de lectura en el sensor y por ello no se garantiza la exactitud.

Tabla A- 155 Reducción de ruido y tiempos de actualización para los módulos RTD

Selección de supresión de frecuencias	Tiempo de integración	Tiempo de actualización (segundos)	
		Módulo de 4 canales	Módulo de 8 canales
400 Hz (2,5 ms)	10 ms ¹	4/2 hilos: 0,142 3 hilos: 0,285	4/2 hilos: 0,285 3 hilos: 0,525
60 Hz (16,6 ms)	16,67 ms	4/2 hilos: 0,222 3 hilos: 0,445	4/2 hilos: 0,445 3 hilos: 0,845
50 Hz (20 ms)	20 ms	4/2 hilos: 0,262 3 hilos: 0,505	4/2 hilos: 0,524 3 hilos: 1,015
10 Hz (100 ms)	100 ms	4/2 hilos: 1,222 3 hilos: 2,445	4/2 hilos: 2,425 3 hilos: 4,845

¹ Para mantener la resolución y la exactitud del módulo seleccionando el filtro de 400 Hz, el tiempo de integración es de 10 ms. Esta selección también rechaza perturbaciones de 100 Hz y 200 Hz.

ATENCIÓN

Después de aplicar tensión al módulo, éste lleva a cabo una calibración interna del convertor de analógico a digital. Durante este tiempo, el módulo indica un valor de 32767 en cada canal hasta que haya información válida disponible en el canal respectivo. Es posible que el programa de usuario deba autorizar este tiempo de inicialización. Como la configuración del módulo puede modificar la longitud del tiempo de inicialización, es conveniente verificar el comportamiento o el módulo en la configuración. Si es necesario, se puede incluir lógica en el programa de usuario para adaptarse al tiempo de inicialización del módulo.

Representación de valores analógicos para RTD

En las tablas siguientes se muestra una representación del valor medido digitalizado para los sensores de rango de temperatura estándar RTD.

Tabla A- 156 Representación de los valores analógicos para termorresistencias PT 100, 200, 500, 1000 y PT 10, 50, 100, 500 GOST (0,003850) estándar

Pt x00 estándar en °C (1 dígito = 0,1 °C)	Unidades		Pt x00 estándar en °F (1 dígito = 0,1 °F)	Unidades		Rango
	Decimal	Hexadecimal		Decimal	Hexadecimal	
> 1000.0	32767	7FFF	> 1832.0	32767	7FFF	Desbordamiento
1000.0	10000	2710	1832.0	18320	4790	Sobre rango
:	:	:	:	:	:	
850.1	8501	2135	1562.1	15621	3D05	Rango nominal
850.0	8500	2134	1562.0	15620	3D04	
:	:	:	:	:	:	Bajo rango
-200.0	-2000	F830	-328.0	-3280	F330	
-200.1	-2001	F82F	-328.1	-3281	F32F	Rebase por defecto
:	:	:	:	:	:	
-243.0	-2430	F682	-405.4	-4054	F02A	
< -243.0	-32768	8000	< -405.4	-32768	8000	

A.9 Signal Boards digitales (SBs)

A.9.1 Datos técnicos de la SB 1221 de entradas digitales 200 kHz

Tabla A- 157 Especificaciones generales

Datos técnicos	SB 1221 DI 4 x 24 V DC, 200 kHz	SB 1221 DI 4 x 5 V DC, 200 kHz
Referencia	6ES7 221-3BD30-0XB0	6ES7 221-3AD30-0XB0
Dimensiones A x A x P (mm)	38 x 62 x 21	38 x 62 x 21
Peso	35 gramos	35 gramos
Disipación de potencia	1,5 W	1,0 W
Consumo de corriente (bus SM)	40 mA	40 mA
Consumo de corriente (24 V DC)	7 mA / entrada + 20 mA	15 mA / entrada + 15 mA

Tabla A- 158 Entradas digitales

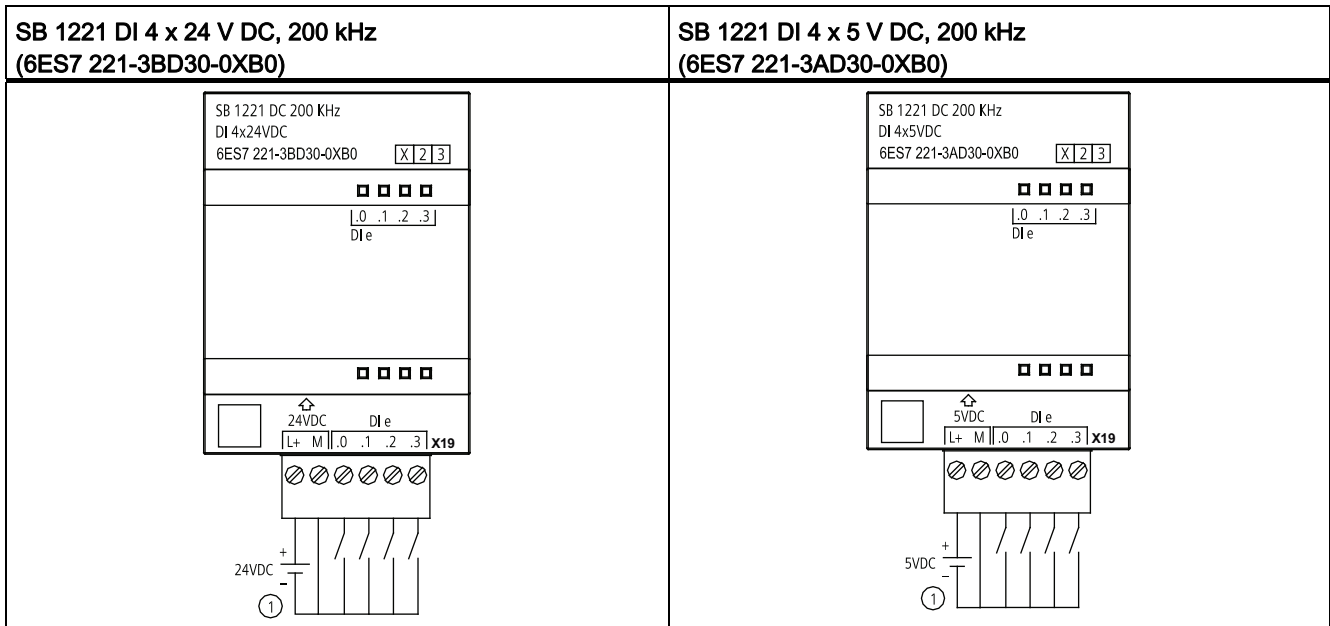
Datos técnicos	SB 1221 DI 4 x 24 V DC, 200 kHz	SB 1221 DI 4 x 5 V DC, 200 kHz
Número de entradas	4	4
Tipo	Fuente	Fuente
Tensión nominal	24 V DC a 7 mA, nominal	5 V DC a 15 mA, nominal
Tensión continua admisible	28,8 V DC	6 V DC
Sobretensión transitoria	35 V DC durante 0,5 s	6 V
Señal 1 lógica (mín.)	L+ menos 10 V DC a 2,9 mA	L+ menos 2.0 V DC a 5,1 mA
Señal 0 lógica (máx.)	L+ menos 5 V DC a 1,4 mA	L+ menos 1,0 V DC a 2,2 mA
Frecuencias de entrada de reloj HSC (máx.)	Fase simple: 200 kHz Fase en cuadratura: 160 kHz	Fase simple: 200 kHz Fase en cuadratura: 160 kHz
Aislamiento (campo a lógica)	500 V AC durante 1 minuto	500 V AC durante 1 minuto
Grupos de aislamiento	1	1
Tiempos de filtro	0,2, 0,4, 0,8, 1,6, 3,2, 6,4 y 12,8 ms; seleccionable en grupos de 4	0,2, 0,4, 0,8, 1,6, 3,2, 6,4 y 12,8 ms; seleccionable en grupos de 4
Número de entradas ON simultáneamente	<ul style="list-style-type: none"> • 2 (no adyacentes) a 60 °C en horizontal o 50 °C en vertical • 4 a 55 °C en horizontal o 45 °C en vertical 	4
Longitud de cable (metros)	50, par trenzado apantallado	50, par trenzado apantallado

ATENCIÓN

En caso de conmutar frecuencias superiores a 20 kHz, es importante que las entradas digitales reciban una forma de onda cuadrada. Tenga en cuenta las siguientes posibilidades para mejorar la calidad de señal hacia las entradas:

- Minimizar la longitud del cable
- Cambiar el driver tipo sumidero NPN por uno tipo sumidero PNP y fuente PNP
- Cambiar el cable por uno de mayor calidad
- Sustituir el circuito/los componentes de 24 V por unos de 5 V
- Añadir una carga externa a la entrada

Tabla A- 159 Diagramas de cableado de las SB de entradas digitales 200 kHz



① Admite únicamente entradas en fuente

Tabla A- 160 Asignación de pines de conectores para SB 1221 DI 4 x 24 V DC, 200 kHz (6ES7 221-3BD30-0XB0)

Pin	X19
1	L+/24 V DC
2	M/24 V DC
3	DI e.0
4	DI e.1
5	DI e.2
6	DI e.3

Tabla A- 161 Asignación de pines de conectores para SB 1221 DI 4 x 5 V DC, 200 kHz (6ES7 221-3AD30-0XB0)

Pin	X19
1	L+/5 V DC
2	M/5 V DC
3	DI e.0
4	DI e.1
5	DI e.2
6	DI e.3

A.9.2 Datos técnicos de la SB 1222 de salidas digitales 200 kHz

Tabla A- 162 Especificaciones generales

Datos técnicos	SB 1222 DQ 4 x 24 V DC, 200 kHz	SB 1222 DQ 4 x 5 V DC, 200 kHz
Referencia	6ES7 222-1BD30-0XB0	6ES7 222-1AD30-0XB0
Dimensiones A x A x P (mm)	38 x 62 x 21	38 x 62 x 21
Peso	35 gramos	35 gramos
Disipación de potencia	0,5 W	0,5 W
Consumo de corriente (bus SM)	35 mA	35 mA
Consumo de corriente (24 V DC)	15 mA	15 mA

Tabla A- 163 Salidas digitales

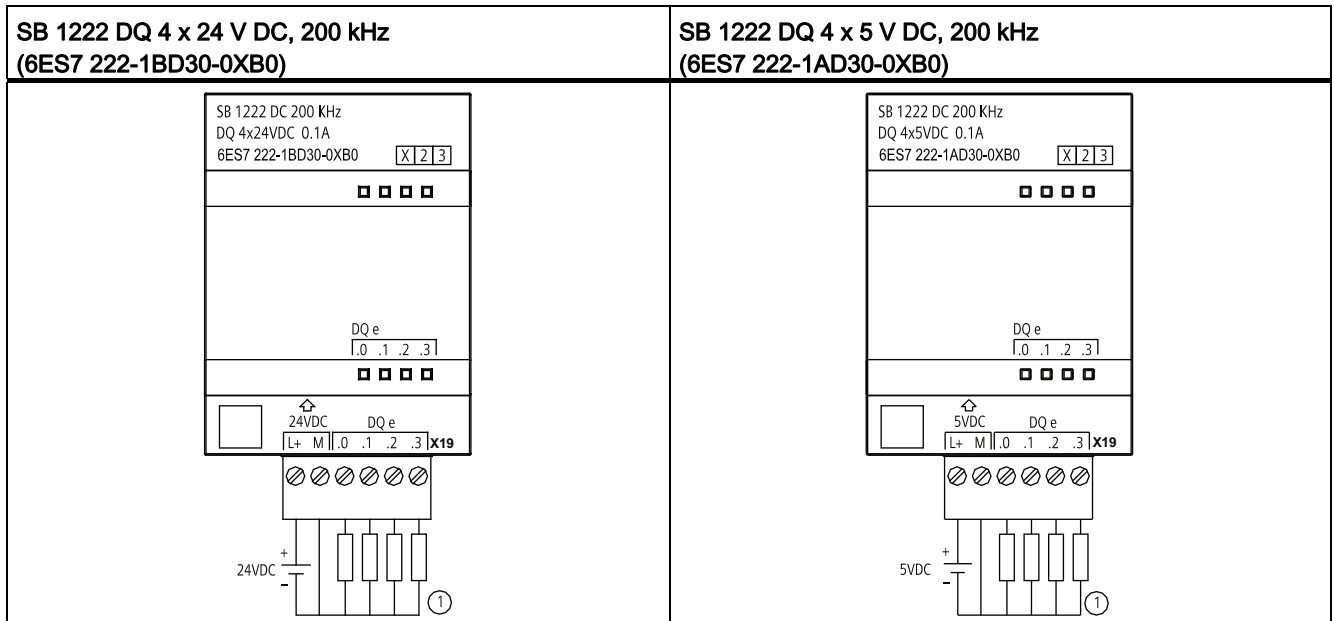
Datos técnicos	SB 1222 DQ 4 x 24 V DC, 200 kHz	SB 1222 DQ 4 x 5 V DC, 200 kHz
Número de salidas	4	4
Tipo de salida	Estado sólido, MOSFET (sumidero y fuente) ¹	Estado sólido, MOSFET (sumidero y fuente) ¹
Rango de tensión	20,4 a 28,8 V DC	4,25 a 6,0 V DC
Señal 1 lógica a intensidad máx.	L+ menos 1,5 V	L+ menos 0,7 V
Señal 0 lógica a intensidad máx.	1,0 V DC, máx.	0,2 V DC, máx.
Intensidad (máx.)	0,1 A	0,1 A
Carga de lámparas	--	--
Resistencia en estado ON (contactos)	11 Ω máx.	7 Ω máx.
Resistencia en estado desact.	6 Ω máx.	0,2 Ω máx.
Corriente de fuga por salida	--	--
Frecuencia de tren de impulsos	200 kHz máx., 2 Hz mín.	200 kHz máx., 2 Hz mín.
Sobrecorriente momentánea	0,11 A	0,11 A
Protección contra sobrecargas	No	No
Aislamiento (campo a lógica)	500 V AC durante 1 minuto	500 V AC durante 1 minuto
Grupos de aislamiento	1	1
Intensidad por neutro	0,4 A	0,4 A
Tensión de bloqueo inductiva	Ninguno	Ninguno
Retardo de conmutación	1,5 μ s + 300 ns asc. 1,5 μ s + 300 ns desc.	200 ns + 300 ns asc. 200 ns + 300 ns desc.
Reacción al cambiar de RUN a STOP	Último valor o valor sustitutivo (valor predeterminado: 0)	Último valor o valor sustitutivo (valor predeterminado: 0)

Datos técnicos	SB 1222 DQ 4 x 24 V DC, 200 kHz	SB 1222 DQ 4 x 5 V DC, 200 kHz
Número de salidas ON simultáneamente	<ul style="list-style-type: none">• 2 (no adyacentes) a 60 °C en horizontal o 50 °C en vertical• 4 a 55 °C en horizontal o 45 °C en vertical	4
Longitud de cable (metros)	50, par trenzado apantallado	50, par trenzado apantallado

¹ Dado que la misma circuitería admite configuraciones en sumidero y en fuente, el estado activo de una carga en fuente es opuesto al de una carga en sumidero. Una salida de fuente es de lógica positiva (el bit Q y los LED están en ON cuando por la carga pasa corriente); una salida de sumidero es de lógica negativa (el bit Q y los LED están en OFF cuando por la carga pasa corriente). Si el módulo está conectado sin programa de usuario, de forma predeterminada estará a 0 V para este módulo, lo que significa que una carga en sumidero estará ON.

<p>ATENCIÓN</p> <p>En caso de conmutar frecuencias superiores a 20 kHz, es importante que las entradas digitales reciban una forma de onda cuadrada. Tenga en cuenta las siguientes posibilidades para mejorar la calidad de señal hacia las entradas:</p> <ul style="list-style-type: none">• Minimizar la longitud del cable• Cambiar el driver tipo sumidero NPN por uno tipo sumidero NPN y fuente PNP• Cambiar el cable por uno de mayor calidad• Reducir el circuito/los componentes de 24 V por 5 V• Añadir una carga externa a la entrada
--

Tabla A- 164 Diagramas de cableado de las SB de salidas digitales 200 kHz



① Para salidas en fuente, conectar "Carga" a "-" (como se indica). Para salidas en sumidero, conecte "Carga" a "+". Dado que la misma circuitería admite configuraciones en sumidero y en fuente, el estado activo de una carga en fuente es opuesto al de una carga en sumidero. Una salida de fuente es de lógica positiva (el bit Q y los LED están en ON cuando por la carga pasa corriente); una salida de sumidero es de lógica negativa (el bit Q y los LED están en OFF cuando por la carga pasa corriente). Si el módulo está conectado sin programa de usuario, de forma predeterminada estará a 0 V para este módulo, lo que significa que una carga en sumidero estará ON.

Tabla A- 165 Asignación de pines de conectores para SB 1222 DQ 4 x 24 V DC, 200 kHz (6ES7 222-1BD30-0XB0)

Pin	X19
1	L+/24 V DC
2	M/24 V DC
3	DQ e.0
4	DQ e.1
5	DQ e.2
6	DQ e.3

Tabla A- 166 Asignación de pines de conectores para SB 1222 DQ 4 x 5 V DC, 200 kHz (6ES7 222-1AD30-0XB0)

Pin	X19
1	L+/5 V DC
2	M/5 V DC
3	DQ e.0
4	DQ e.1
5	DQ e.2
6	DQ e.3

A.9.3 Datos técnicos de la SB 1223 de entradas/salidas digitales 200 kHz

Tabla A- 167 Especificaciones generales

Datos técnicos	SB 1223 DI 2 x 24 V DC / DQ 2 x 24 V DC, 200 kHz	SB 1223 DI 2 x 5 V DC / DQ 2 x 5 V DC, 200 kHz
Referencia	6ES7 223-3BD30-0XB0	6ES7 223-3AD30-0XB0
Dimensiones A x A x P (mm)	38 x 62 x 21	38 x 62 x 21
Peso	35 gramos	35 gramos
Disipación de potencia	1,0 W	0,5 W
Consumo de corriente (bus SM)	35 mA	35 mA
Consumo de corriente (24 V DC)	7 mA / entrada + 30 mA	15 mA / entrada + 15 mA

Tabla A- 168 Entradas digitales

Datos técnicos	SB 1223 DI 2 x 24 V DC / DQ 2 x 24 V DC, 200 kHz	SB 1223 DI 2 x 5 V DC / DQ 2 x 5 V DC, 200 kHz
Número de entradas	2	2
Tipo	Fuente	Fuente
Tensión nominal	24 V DC a 7 mA, nominal	5 V DC a 15 mA, nominal
Tensión continua admisible	28,8 V DC	6 V DC
Sobretensión transitoria	35 V DC durante 0,5 seg.	6 V
Señal 1 lógica (mín.)	L+ menos 10 V DC a 2,9 mA	L+ menos 2,0 V DC a 5,1 mA
Señal 0 lógica (máx.)	L+ menos 5 V DC a 1,4 mA	L+ menos 1,0 V DC a 2,2 mA
Frecuencias de entrada de reloj HSC (máx.)	Fase simple: 200 kHz Fase en cuadratura: 160 kHz	Fase simple: 200 kHz Fase en cuadratura: 160 kHz
Aislamiento (campo a lógica)	500 V AC durante 1 minuto	500 V AC durante 1 minuto
Grupos de aislamiento	1 (no aislado hacia salidas)	1 (no aislado hacia salidas)
Tiempos de filtro	0,2, 0,4, 0,8, 1,6, 3,2, 6,4 y 12,8 ms; seleccionable en grupos de 4	0,2, 0,4, 0,8, 1,6, 3,2, 6,4 y 12,8 ms; seleccionable en grupos de 4

Datos técnicos	SB 1223 DI 2 x 24 V DC / DQ 2 x 24 V DC, 200 kHz	SB 1223 DI 2 x 5 V DC / DQ 2 x 5 V DC, 200 kHz
Número de entradas ON simultáneamente	2	2
Longitud de cable (metros)	50, par trenzado apantallado	50, par trenzado apantallado

Tabla A- 169 Salidas digitales

Datos técnicos	SB 1223 DI 2 x 24 V DC / DQ 2 x 24 V DC, 200 kHz	SB 1223 DI 2 x 5 V DC / DQ 2 x 5 V DC, 200 kHz
Número de salidas	2	2
Tipo de salida	Estado sólido, MOSFET (sumidero y fuente) ¹	Estado sólido, MOSFET (sumidero y fuente) ¹
Rango de tensión	20,4 a 28,8 V DC	4,25 a 6,0 V DC
Valor nominal	24 V DC	5 V DC
Señal 1 lógica a intensidad máx.	L+ menos 1,5 V	L+ menos 0,7 V
Señal 0 lógica a intensidad máx.	1,0 V DC, máx.	0,2 V DC, máx.
Intensidad (máx.)	0,1 A	0,1 A
Carga de lámparas	--	--
Resistencia en estado ON (contactos)	11 Ω máx.	7 Ω máx.
Resistencia en estado desact.	6 Ω máx.	0,2 Ω máx.
Corriente de fuga por salida	--	--
Frecuencia de tren de impulsos	200 kHz máx., 2 Hz mín.	200 kHz máx., 2 Hz mín.
Sobrecorriente momentánea	0,11 A	0,11 A
Protección contra sobrecargas	No	No
Aislamiento (campo a lógica)	500 V AC durante 1 minuto	500 V AC durante 1 minuto
Grupos de aislamiento	1 (no aislado hacia entradas)	1 (no aislado hacia entradas)
Intensidad por neutro	0,2 A	0,2 A
Tensión de bloqueo inductiva	Ninguno	Ninguno
Retardo de conmutación	1,5 μ s + 300 ns asc. 1,5 μ s + 300 ns desc.	200 ns + 300 ns asc. 200 ns + 300 ns desc.
Reacción al cambiar de RUN a STOP	Último valor o valor sustitutivo (valor predeterminado 0)	Último valor o valor sustitutivo (valor predeterminado 0)
Número de salidas ON simultáneamente	2	2
Longitud de cable (metros)	50, par trenzado apantallado	50, par trenzado apantallado

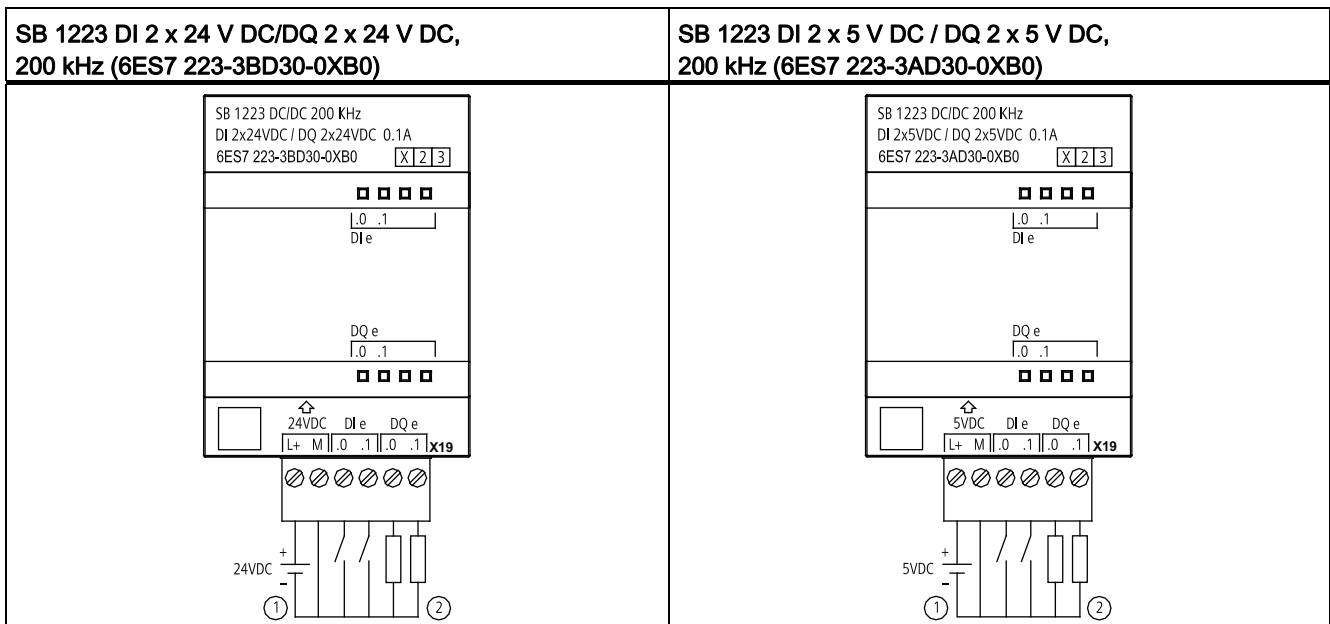
¹ Dado que la misma circuitería admite configuraciones en sumidero y en fuente, el estado activo de una carga en fuente es opuesto al de una carga en sumidero. Una salida de fuente es de lógica positiva (el bit Q y los LED están en ON cuando por la carga pasa corriente); una salida de sumidero es de lógica negativa (el bit Q y los LED están en OFF cuando por la carga pasa corriente). Si el módulo está conectado sin programa de usuario, de forma predeterminada estará a 0 V para este módulo, lo que significa que una carga en sumidero estará ON.

ATENCIÓN

En caso de conmutar frecuencias superiores a 20 kHz, es importante que las entradas digitales reciban una forma de onda cuadrada. Tenga en cuenta las siguientes posibilidades para mejorar la calidad de señal hacia las entradas:

- Minimizar la longitud del cable
- Cambiar el driver tipo sumidero NPN por uno tipo sumidero PNP y fuente PNP
- Cambiar el cable por uno de mayor calidad
- Reducir el circuito/los componentes de 24 V por 5 V
- Añadir una carga externa a la entrada

Tabla A- 170 Diagramas de cableado de las SB de entradas/salidas digitales 200 kHz



- ① Admite únicamente entradas en fuente
- ② Para salidas en fuente, conectar "Carga" a "-" (como se indica). Para salidas en sumidero, conecte "Carga" a "+".¹ Dado que la misma circuitería admite configuraciones en sumidero y en fuente, el estado activo de una carga en fuente es opuesto al de una carga en sumidero. Una salida de fuente es de lógica positiva (el bit Q y los LED están en ON cuando por la carga pasa corriente); una salida de sumidero es de lógica negativa (el bit Q y los LED están en OFF cuando por la carga pasa corriente). Si el módulo está conectado sin programa de usuario, de forma predeterminada estará a 0 V para este módulo, lo que significa que una carga en sumidero estará ON.

Tabla A- 171 Asignación de pines de conectores para SB 1223 DI 2 x 24 V DC/DQ 2 x 24 V DC, 200 kHz (6ES7 223-3BD30-0XB0)

Pin	X19
1	L+/24 V DC
2	M/24 V DC
3	DI e.0
4	DI e.1
5	DQ e.0
6	DQ e.1

Tabla A- 172 Asignación de pines de conectores para SB 1223 DI 2 x 5 V DC / DQ 2 x 5 V DC, 200 kHz (6ES7 223-3AD30-0XB0)

Pin	X19
1	L+/5 V DC
2	M/5 V DC
3	DI e.0
4	DI e.1
5	DQ e.0
6	DQ e.1

A.9.4 Datos técnicos de la SB 1223 2 entradas x 24 V DC / 2 salidas x 24 V DC

Tabla A- 173 Especificaciones generales

Datos técnicos	SB 1223 DI 2 x 24 V DC, DQ 2 x 24 V DC
Referencia	6ES7 223-0BD30-0XB0
Dimensiones A x A x P (mm)	38 x 62 x 21
Peso	40 gramos
Disipación de potencia	1,0 W
Consumo de corriente (bus SM)	50 mA
Consumo de corriente (24 V DC)	4 mA/entrada utilizada

Tabla A- 174 Entradas digitales

Datos técnicos	SB 1223 DI 2 x 24 V DC, DQ 2 x 24 V DC
Número de entradas	2
Tipo	Tipo 1 IEC sumidero
Tensión nominal	24 V DC a 4 mA, nominal
Tensión continua admisible	30 V DC, máx.

Datos técnicos

A.9 Signal Boards digitales (SBs)

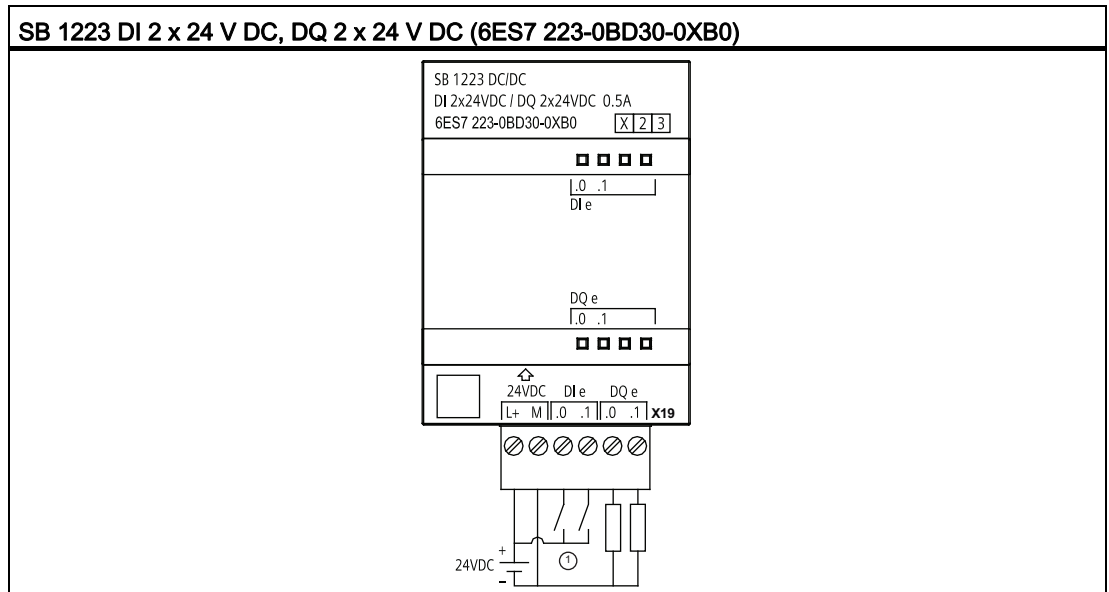
Datos técnicos	SB 1223 DI 2 x 24 V DC, DQ 2 x 24 V DC
Sobretensión transitoria	35 V DC durante 0,5 seg.
Señal 1 lógica (mín.)	15 V DC a 2,5 mA
Señal 0 lógica (máx.)	5 V DC a 1 mA
Frecuencias de entrada de reloj HSC (máx.)	20 kHz (15 a 30 V DC) 30 kHz (15 a 26 V DC)
Aislamiento (campo a lógica)	500 V AC durante 1 minuto
Grupos de aislamiento	1
Tiempos de filtro	0,2, 0,4, 0,8, 1,6, 3,2, 6,4 y 12,8 ms Seleccionable en grupos de 2
Número de entradas ON simultáneamente	2
Longitud de cable (metros)	500 apantallado, 300 no apantallado

Tabla A- 175 Salidas digitales

Datos técnicos	SB 1223 DI 2 x 24 V DC, DQ 2 x 24 V DC
Número de salidas	2
Tipo de salida	Estado sólido MOSFET (fuente)
Rango de tensión	20,4 a 28,8 V DC
Señal 1 lógica a intensidad máx.	20 V DC mín.
Señal 0 lógica con carga de 10K Ω	0,1 V DC máx.
Intensidad (máx.)	0,5 A
Carga de lámparas	5 W
Resistencia en estado ON (contactos)	0,6 Ω máx.
Corriente de fuga por salida	10 μ A máx.
Frecuencia de tren de impulsos (PTO)	20 kHz máx., 2 Hz mín. ¹
Sobrecorriente momentánea	5 A durante máx. 100 ms
Protección contra sobrecargas	No
Aislamiento (campo a lógica)	500 V AC durante 1 minuto
Grupos de aislamiento	1
Intensidad por neutro	1 A
Tensión de bloqueo inductiva	L+ menos 48 V, disipación de 1 W
Retardo de conmutación	2 μ s máx. OFF a ON 10 μ s máx. ON a OFF
Reacción al cambiar de RUN a STOP	Último valor o valor sustitutivo (valor predeterminado: 0)
Número de salidas ON simultáneamente	2
Longitud de cable (metros)	500 m apantallado, 150 m no apantallado

¹ En función del receptor de impulsos y del cable, un resistor de carga adicional (al menos 10% de la intensidad nominal) puede mejorar la calidad de la señal de los impulsos y la inmunidad a interferencias.

Tabla A- 176 Diagrama de cableado de la SB de entradas/salidas digitales



① Soporta únicamente entradas NPN

Tabla A- 177 Asignación de pines de conectores para SB 1223 DI 2 x 24 V DC, DQ 2 x 24 V DC (6ES7 223-0BD30-0XB0)

Pin	X19
1	L+/24 V DC
2	M/24 V DC
3	DI e.0
4	DI e.1
5	DQ e.0
6	DQ e.1

A.10 Signal Boards analógicas (SBs)

A.10.1 Datos técnicos de la SB 1231 de 1 entrada analógica

Nota

Para utilizar esta SB el firmware de la CPU debe tener la versión 2.0 o superior.

Datos técnicos

A.10 Signal Boards analógicas (SBs)

Tabla A- 178 Especificaciones generales

Datos técnicos	SB 1231 AI 1 x 12 bits
Referencia	6ES7 231-4HA30-0XB0
Dimensiones A x A x P (mm)	38 x 62 x 21
Peso	35 gramos
Disipación de potencia	0,4 W
Consumo de corriente (bus SM)	55 mA
Consumo de corriente (24 V DC)	Ninguno

Tabla A- 179 Entradas analógicas

Datos técnicos	SB 1231 AI 1x12 bits
Número de entradas	1
Tipo	Tensión o intensidad (diferencial)
Rango	$\pm 10V$, $\pm 5V$, $\pm 2,5$ ó 0 a 20 mA
Resolución	11 bits + bit de signo
Rango total (palabra de datos)	-27.648 a 27.648
Rango de saturación máximo/mínimo (palabra de datos)	Tensión: 32.511 a 27.649 / -27.649 a -32.512 Intensidad: 32.511 a 27.649 / 0 a -4.864 (Consulte Representación de entradas analógicas para tensión y representación de entradas analógicas para intensidad.)
Rebase por exceso/defecto (palabra de datos)	Tensión: 32.767 a 32.512 / -32.513 a -32.768 Intensidad: 32.767 a 32.512 / -4.865 a -32.768 (Consulte Representación de entradas analógicas para tensión y representación de entradas analógicas para intensidad.)
Tensión/intensidad soportada máxima	$\pm 35V$ / ± 40 mA
Filtrado	Ninguno, débil, medio o fuerte (consulte los tiempos de respuesta de las entradas analógicas para más detalles sobre el tiempo de respuesta a un escalón.)
Supresión de ruido	400, 60, 50 ó 10 Hz (consulte las frecuencias de muestreo en Tiempos de respuesta de las entradas analógicas.)
Precisión (25 °C/-20 a 60 °C)	$\pm 0,3\%$ / $\pm 0,6\%$ de rango máximo
Impedancia de entrada	
Diferencial	Tensión: 220 k Ω ; intensidad: 250 Ω
Modo común	Tensión: 55 k Ω ; intensidad: 55 k Ω
Reacción al cambiar de RUN a STOP	Último valor o valor sustitutivo (valor predeterminado: 0)
Principio de medición	Conversión de valor real
Rechazo en modo común	400 dB, DC a 60 Hz
Rango de señales operativo	La tensión de señal más la tensión en modo común debe ser menor que +35 V y mayor que -35 V
Aislamiento (campo a lógica)	Ninguno
Longitud de cable (metros)	100 m, trenzado y apantallado

Tabla A- 180 Diagnóstico

Datos técnicos	SB 1231 AI 1 x 12 bits
Rebase por exceso/defecto	Sí
24 V DC, baja tensión	No

Tabla A- 181 Diagrama de cableado de la SB de entrada analógica

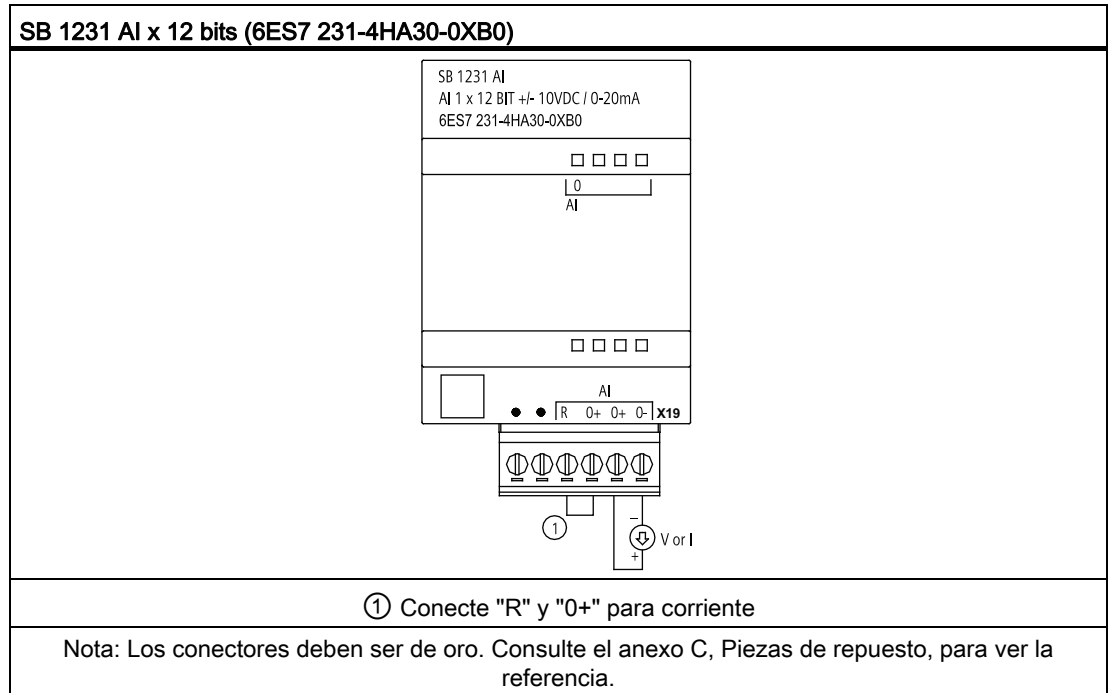


Tabla A- 182 Asignación de pines de conectores para SB 1231 AI x 12 bits (6ES7 231-4HA30-0XB0)

Pin	X19 (oro)
1	Sin conexión
2	Sin conexión
3	AI R
4	AI 0+
5	AI 0+
6	AI 0-

A.10.2 Datos técnicos de la SB 1232 de 1 salida analógica

Tabla A- 183 Especificaciones generales

Datos técnicos	SB 1232 AQ 1 x 12 bits
Referencia	6ES7 232-4HA30-0XB0
Dimensiones A x A x P (mm)	38 x 62 x 21
Peso	40 gramos
Disipación de potencia	1,5 W
Consumo de corriente (bus SM)	15 mA
Consumo de corriente (24 V DC)	40 mA (sin carga)

Tabla A- 184 Salidas analógicas

Datos técnicos	SB 1232 AQ 1 x 12 bits
Número de salidas	1
Tipo	Tensión o intensidad
Rango	±10 V ó 0 a 20 mA
Resolución	Tensión: 12 bits Intensidad: 11 bits
Rango total (palabra de datos)	Tensión: -27.648 a 27.648
Consulte los rangos de salida de tensión e intensidad (Página 846).	Intensidad: 0 a 27.648
Precisión (25 °C/-20 a 60 °C)	±0,5% / ±1% de rango máximo
Tiempo de estabilización (95% del nuevo valor)	Tensión: 300 µS (R), 750 µS (1 uF) Intensidad: 600 µS (1 mH), 2 ms (10 mH)
Impedancia de carga	Tensión: ≥ 1000 Ω Intensidad: ≤ 600 Ω
Reacción al cambiar de RUN a STOP	Último valor o valor sustitutivo (valor predeterminado: 0)
Aislamiento (campo a lógica)	Ninguno
Longitud de cable (metros)	100 m, trenzado y apantallado

Tabla A- 185 Diagnóstico

Datos técnicos	SB 1232 AQ 1 x 12 bits
Rebase por exceso/defecto	Sí
Cortocircuito a tierra (sólo en modo de tensión)	Sí
Rotura de hilo (sólo en modo de intensidad)	Sí

Tabla A- 186 Diagrama de cableado de la SB 1232 AQ 1 x 12 bits

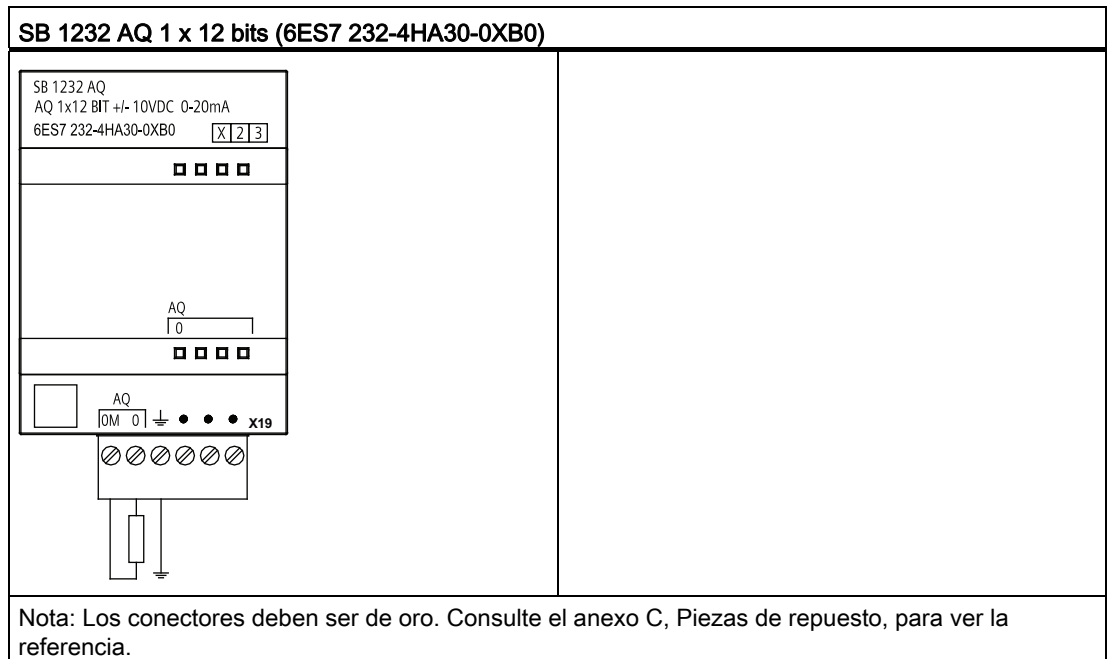


Tabla A- 187 Asignación de pines de conectores para SB 1232 AQ 1 x 12 bits (6ES7 232-4HA30-0XB0)

Pin	X19 (oro)
1	AQ 0M
2	AQ 0
3	GND
4	Sin conexión
5	Sin conexión
6	Sin conexión

A.10.3 Rangos de medida para entradas y salidas analógicas

A.10.3.1 Respuesta a un escalón de las entradas analógicas

Tabla A- 188 Respuesta a un escalón (ms), 0V a 10V medido a 95%

Selección de filtrado (valor medio de muestreo)	Selección del tiempo de integración			
	400 Hz (2,5 ms)	60 Hz (16,6 ms)	50 Hz (20 ms)	10 Hz (100 ms)
Ninguno (1 ciclo): Sin media	4,5 ms	18,7 ms	22,0 ms	102 ms
Débil (4 ciclos): 4 muestreos	10,6 ms	59,3 ms	70,8 ms	346 ms
Medio (16 ciclos): 16 muestreos	33,0 ms	208 ms	250 ms	1240 ms
Fuerte (32 ciclos): 32 muestreos	63,0 ms	408 ms	490 ms	2440 ms
Tiempo de muestreo	0,156 ms	1,042 ms	1,250 ms	6,250 ms

A.10.3.2 Tiempo de muestreo y tiempos de actualización para entradas analógicas

Tabla A- 189 Tiempo de muestreo y tiempo de actualización

Selección	Tiempo de muestreo	Tiempo de actualización de la SB
400 Hz (2,5 ms)	0,156 ms	0,156 ms
60 Hz (16,6 ms)	1,042 ms	1,042 ms
50 Hz (20 ms)	1,250 ms	1,25 ms
10 Hz (100 ms)	6,250 ms	6,25 ms

A.10.3.3 Rangos de medida de entradas analógicas de tensión

Tabla A- 190 Representación de entradas analógicas para tensión

Sistema		Rango de medida de tensión						
Decimal	Hexadecimal	±10 V	±5 V	±2,5 V	±1,25V		De 0 a 10 V	
32767	7FFF	11,851 V	5,926 V	2,963 V	1,481 V	Rebase por exceso	11,851 V	Rebase por exceso
32512	7F00							
32511	7EFF	11,759 V	5,879 V	2,940 V	1,470 V	Rango de sobreimpulso	11,759 V	Rango de sobreimpulso
27649	6C01							
27648	6C00	10 V	5 V	2,5 V	1,250 V	Rango nominal	10 V	Rango nominal
20736	5100	7,5 V	3,75 V	1,875 V	0,938 V		7,5 V	
1	1	361,7 µV	180,8 µV	90,4 µV	45,2 µV		361,7 µV	
0	0	0 V	0 V	0 V	0 V		0 V	
-1	FFFF						Los valores negativos no se soportan	
-20736	AF00	-7,5 V	-3,75 V	-1,875 V	-0,938 V			
-27648	9400	-10 V	-5 V	-2,5 V	-1,250 V			
-27649	93FF					Rango de subimpulso		
-32512	8100	-11,759 V	-5,879 V	-2,940 V	-1,470 V			
-32513	80FF					Rebase por defecto		
-32768	8000	-11,851 V	-5,926 V	-2,963 V	-1,481 V			

A.10.3.4 Rangos de medida de las entradas analógicas de intensidad

Tabla A- 191 Representación de entradas analógicas para intensidad

Sistema		Rango de medida de intensidad		
Decimal	Hexadecimal	De 0 mA a 20 mA	De 4 mA a 20 mA	
32767	7FFF	23,70 mA	22,96 mA	Rebase por exceso
32512	7F00			
32511	7EFF	23,52 mA	22,81 mA	Rango de sobreimpulso
27649	6C01			
27648	6C00	20 mA	20 mA	Rango nominal
20736	5100	15 mA	16 mA	
1	1	723,4 nA	4 mA + 578,7 nA	
0	0	0 mA	4 mA	
-1	FFFF			
-4864	ED00	-3,52 mA	1,185 mA	Rango de subimpulso
-4865	ECFF			
-32768	8000			
				Rebase por defecto

A.10.3.5 Rangos de medición de salidas (AQ) de tensión e intensidad (SB y SM)

Tabla A- 192 Representación de salidas analógicas para tensión

Sistema		Rango de salida de tensión	
Decimal	Hexadecimal	±10 V	
32767	7FFF	V. nota 1	Rebase por exceso
32512	7F00	V. nota 1	
32511	7EFF	11,76 V	Rango de sobreimpulso
27649	6C01		
27648	6C00	10 V	Rango nominal
20736	5100	7,5 V	
1	1	361,7 μ V	
0	0	0 V	
-1	FFFF	-361,7 μ V	
-20736	AF00	-7,5 V	
-27648	9400	-10 V	
-27649	93FF		Rango de subimpulso
-32512	8100	-11,76 V	
-32513	80FF	V. nota 1	Rebase por defecto
-32768	8000	V. nota 1	

¹ En una condición de rebase por exceso o por defecto, la reacción de las salidas analógicas corresponderá a las propiedades ajustadas en la configuración de dispositivos para el módulo de señales analógico. En el parámetro "Reacción a STOP de la CPU", seleccione: "Aplicar valor sustitutivo" o "Mantener último valor".

Tabla A- 193 Representación de salidas analógicas para intensidad

Sistema		Rango de salida de intensidad	
Decimal	Hexadecimal	0 mA a 20 mA	
32767	7FFF	V. nota 1	Rebase por exceso
32512	7F00	V. nota 1	
32511	7EFF	23,52 mA	Rango de sobreimpulso
27649	6C01		
27648	6C00	20 mA	Rango nominal
20736	5100	15 mA	
1	1	723,4 nA	
0	0	0 mA	

¹ En una condición de rebase por exceso o por defecto, la reacción de las salidas analógicas corresponderá a las propiedades ajustadas en la configuración de dispositivos para el módulo de señales analógico. En el parámetro "Reacción a STOP de la CPU", seleccione: "Aplicar valor sustitutivo" o "Mantener último valor".

A.10.4 SBs de termopar

A.10.4.1 Datos técnicos de la SB 1231 de termopar de 1 entrada analógica

Nota

Para utilizar esta SB el firmware de la CPU debe tener la versión 2.0 o superior.

Tabla A- 194 Especificaciones generales

Datos técnicos	SB 1231 AI 1 x 16 bits de termopar
Referencia	6ES7 231-5QA30-0XB0
Dimensiones A x A x P (mm)	38 x 62 x 21
Peso	35 gramos
Disipación de potencia	0,5 W
Consumo de corriente (bus SM)	5 mA
Consumo de corriente (24 V DC)	20 mA

Tabla A- 195 Entradas analógicas

Datos técnicos	SB 1231 AI 1x16 bits de termopar				
Número de entradas	1				
Tipo	TC flotante y mV				
Rango	Véase la tabla de selección de filtros de termopar (Página 848).				
	<ul style="list-style-type: none"> Rango nominal (palabra de datos) Rango de saturación máximo/mínimo (palabra de datos) Rebase por exceso/defecto (palabra de datos) 				
Resolución	<table border="0"> <tr> <td>Temperatura</td> <td>0,1 °C / 0,1 °F</td> </tr> <tr> <td>Tensión</td> <td>Signo más (+) de 15 bits</td> </tr> </table>	Temperatura	0,1 °C / 0,1 °F	Tensión	Signo más (+) de 15 bits
Temperatura	0,1 °C / 0,1 °F				
Tensión	Signo más (+) de 15 bits				
Tensión soportada máxima	±35 V				
Supresión de ruido	85 dB para el ajuste de filtro seleccionado (10 Hz, 50 Hz, 60 Hz, 400 Hz)				
Rechazo en modo común	> 120 dB a 120 V AC				
Impedancia	≥ 10 M Ω				
Precisión	Véase la tabla de selección de termopares (Página 848).				
Repetitividad	±0,05% FS				
Principio de medición	Integrador				
Tiempo de actualización del módulo	Véase la tabla de selección de filtros de termopar (Página 848).				
Error de unión fría	±1,5 °C				

A.10 Signal Boards analógicas (SBs)

Datos técnicos	SB 1231 AI 1x16 bits de termopar
Aislamiento (campo a lógica)	500 V AC
Longitud de cable (metros)	100 m hasta el sensor (máx.)
Resistencia del cable	100 Ω máx.

Tabla A- 196 Diagnóstico

Datos técnicos	SB 1231 AI 1 x 16 bits de termopar
Rebase por exceso/defecto ¹	Sí
Rotura de hilo ²	Sí

¹ La información de las alarmas de rebase por exceso/defecto se indicará en los valores de datos analógicos aunque las alarmas estén deshabilitadas en la configuración del módulo.

² Si la alarma de rotura de hilo está deshabilitada y se presenta una condición de rotura de hilo en la línea del sensor, el módulo puede señalar valores aleatorios.

El módulo de señales analógicas de termopar SM 1231 (TC) mide el valor de la tensión conectada a las entradas del módulo.

La Signal Board analógica de termopar SB 1231 mide el valor de la tensión conectada a las entradas de la Signal Board. El tipo de medición de temperatura puede ser "termopar" o "tensión".

- "Termopar": el valor se expresará en grados multiplicados por diez (p. ej. 25,3 grados se expresarán como 253 decimales).
- "Tensión": el valor máximo del rango nominal serán 27648 decimales.

A.10.4.2 Funcionamiento básico de un termopar

Los termopares se forman por la unión de dos metales diferentes que se conectan eléctricamente produciendo una tensión. La tensión generada es proporcional a la temperatura de la unión. Se trata de una tensión pequeña; un microvoltio puede representar varios grados. La medición de temperatura con termopares consiste en medir la tensión de un termopar, compensar las uniones adicionales y linealizar posteriormente el resultado.

Cuando se conecta un termopar al módulo de termopar SM 1231, los dos hilos de metales distintos se unen al conector de señales del módulo. El punto en el que los dos hilos diferentes se unen el uno con el otro constituye el termopar del sensor.

Dos termopares adicionales se forman donde los dos hilos diferentes se unen al conector de señales. La temperatura del conector genera una tensión que se suma a la del termopar del sensor. Si no se corrige esta tensión, la temperatura indicada será diferente de la temperatura del sensor.

La compensación de unión fría se utiliza para compensar el termopar del conector. Las tablas de termopares se basan en una temperatura de referencia que, por lo general, es de cero grados centígrados. La compensación de unión fría compensa el conector a cero grados centígrados. La compensación restablece la tensión sumada por los termopares del conector. La temperatura del módulo se mide internamente y se convierte luego a un valor a sumar a la conversión del sensor. La conversión del sensor corregida se linealiza entonces utilizando las tablas de termopares.

Para optimizar el funcionamiento de la compensación de unión fría es necesario colocar el módulo de termopar en un entorno térmicamente estable. Una variación lenta (inferior a 0,1° C/minuto) del módulo a temperatura ambiente se compensa correctamente dentro de las especificaciones del módulo. Si hay corriente de aire a través del módulo también se producirán errores de compensación de unión fría.

Si se requiere una mejor compensación del error de unión fría, se puede utilizar un bloque de terminales isotérmico. El módulo de termopar permite utilizar un bloque de terminales con una referencia de 0° C o 50° C.

Tabla de selección de termopares para la SB 1231

Los rangos y la exactitud de los diferentes tipos de termopares soportados por la Signal Board de termopar SB 1231 se indican en la tabla siguiente.

Tabla A- 197 Tabla de selección de termopares de la SB 1231

Tipo de termopar	Rango de saturación mínimo ¹	Rango nominal límite inferior	Rango nominal límite superior	Rango de saturación máximo ²	Exactitud rango ³ normal a 25°C	Exactitud rango ³ normal de -20 °C a 60 °C
J	-210,0°C	-150,0°C	1200,0°C	1450,0°C	±0,3°C	±0,6°C
K	-270,0°C	-200,0°C	1372,0°C	1622,0°C	±0,4°C	±1,0°C
T	-270,0°C	-200,0°C	400,0°C	540,0°C	±0,5°C	±1,0°C
E	-270,0°C	-200,0°C	1000,0°C	1200,0°C	±0,3°C	±0,6°C
R & S	-50,0°C	100,0°C	1768,0°C	2019,0°C	±1,0°C	±2,5°C
N	-270,0°C	0,0°C	1300,0°C	1550,0°C	±1,0°C	±1,6°C
C	0,0°C	100,0°C	2315,0°C	2500,0°C	±0,7°C	±2,7°C
TXK/XK(L)	-200,0°C	-150,0°C	800,0°C	1050,0°C	±0,6°C	±1,2°C
Tensión	-32511	-27648 -80mV	27648 80mV	32511	±0,05%	±0,1%

¹ Los valores de termopar inferiores al valor del rango de saturación mínimo se notifican como -32768.

² Los valores de termopar superiores al valor del rango de saturación máximo se notifican como 32767.

³ El error de la unión fría interna es de ±1,5 °C en todos los rangos. Esto debe añadirse al error en esta tabla. Para cumplir estas especificaciones, la Signal Board requiere como mínimo 30 minutos de calentamiento.

Tabla A- 198 Tabla de selección de filtros para el termopar de la SB 1231

Supresión de frecuencias (Hz)	Tiempo de integración (ms)	Tiempo de actualización de la Signal Board (segundos)
10	100	0.306
50	20	0.066
60	16.67	0.056
400 ¹	10	0.036

¹ Para mantener la resolución y exactitud del módulo con la supresión de 400 Hz, el tiempo de integración es de 10 ms. Esta selección también suprime perturbaciones de 100 Hz y 200 Hz.

Para medir termopares se recomienda utilizar un tiempo de integración de 100 ms. El uso de tiempos de integración inferiores aumentará el error de repetibilidad de las lecturas de temperatura.

Nota

Después de aplicar tensión al módulo, éste lleva a cabo una calibración interna del convertidor analógico a digital. Durante este tiempo el módulo reporta un valor de 32767 en cada canal hasta que haya información válida disponible en el canal respectivo. Es posible que el programa de usuario deba autorizar este tiempo de inicialización.

Tabla A- 199 Diagrama de cableado de la SB de termopar de entrada analógica

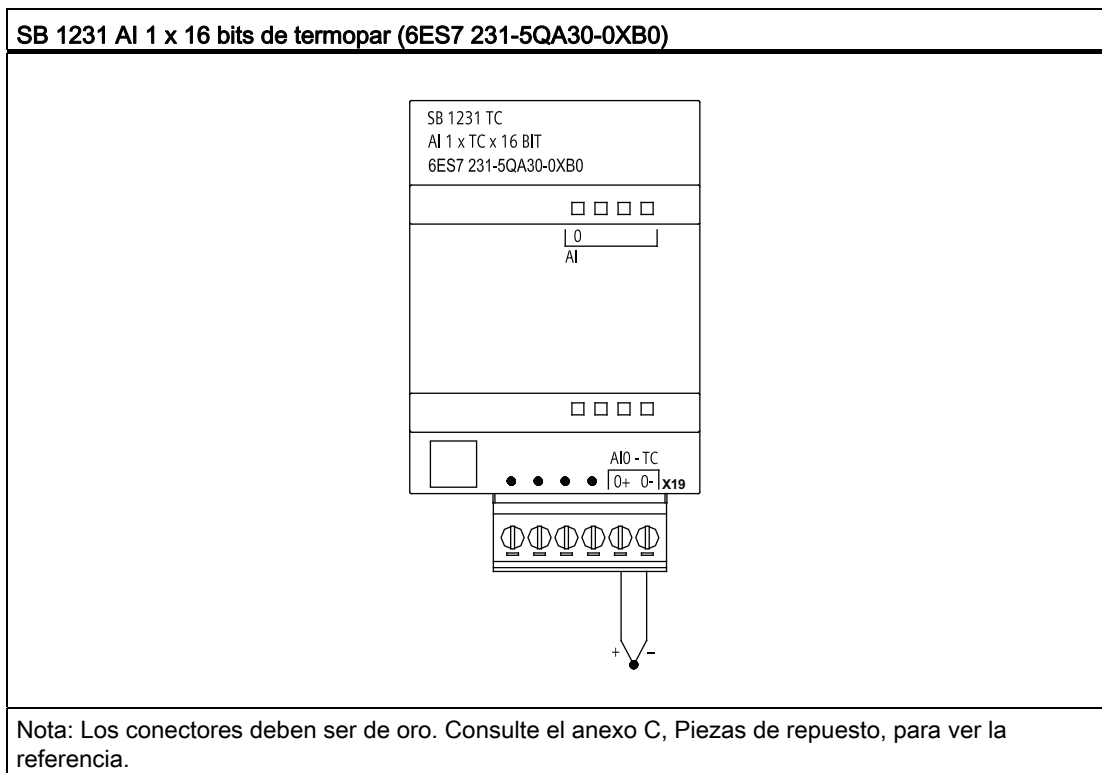


Tabla A- 200 Asignación de pines de conectores para SB 1231 AI 1 x 16 bits termopar (6ES7 231-5QA30-0XB0)

Pin	X19 (oro)
1	Sin conexión
2	Sin conexión
3	Sin conexión
4	Sin conexión
5	AI 0-/TC
6	AI 0+/TC

A.10.5 SBs RTD

A.10.5.1 Datos técnicos de la SB 1231 de 1 entrada analógica RTD

Nota

Para utilizar esta SB el firmware de la CPU debe tener la versión 2.0 o superior.

Tabla A- 201 Especificaciones generales

Datos técnicos	SB 1231 AI 1 x 16 bits RTD
Referencia	6ES7 231-5PA30-0XB0
Dimensiones A x A x P (mm)	38 x 62 x 2
Peso	35 gramos
Disipación de potencia	0,7 W
Consumo de corriente (bus SM)	5 mA
Consumo de corriente (24 V DC)	25 mA

Tabla A- 202 Entradas analógicas

Datos técnicos	SB 1231 AI 1 x 16 bits RTD				
Número de entradas	1				
Tipo	RTD y ohmios indicados por el módulo				
Rango	Véanse las tablas de selección (Página 854). <ul style="list-style-type: none"> • Rango nominal (palabra de datos) • Rango de saturación máximo/mínimo (palabra de datos) • Rebase por exceso/defecto (palabra de datos) 				
Resolución	<table border="0"> <tr> <td>Temperatura</td> <td>0,1 °C / 0,1 °F</td> </tr> <tr> <td>Tensión</td> <td>Signo más (+) de 15 bits</td> </tr> </table>	Temperatura	0,1 °C / 0,1 °F	Tensión	Signo más (+) de 15 bits
Temperatura	0,1 °C / 0,1 °F				
Tensión	Signo más (+) de 15 bits				
Tensión soportada máxima	±35 V				
Supresión de ruido	85 dB (10 Hz, 50 Hz, 60 Hz, 400 Hz)				
Rechazo en modo común	> 120 dB				
Impedancia	≥ 10 MΩ				
Precisión	Véanse las tablas de selección (Página 854).				
Repetitividad	±0,05% FS				
Disipación máxima del sensor	0,5 m W				
Principio de medición	Integrador				
Tiempo de actualización del módulo	Véase la tabla de selección (Página 854).				
Aislamiento (campo a lógica)	500 V AC				

Datos técnicos

A.10 Signal Boards analógicas (SBs)

Datos técnicos	SB 1231 AI 1 x 16 bits RTD
Longitud de cable (metros)	100 m hasta el sensor (máx.)
Resistencia del cable	20 Ω , 2,7 para 10 Ω RTD máx.

Tabla A- 203 Diagnóstico

Datos técnicos	SB 1231 AI 1 x 16 bits RTD
Rebase por exceso/defecto ^{1, 2}	Sí
Rotura de hilo ³	Sí

¹ La información de las alarmas de rebase por exceso/defecto se indicará en los valores de datos analógicos aunque las alarmas estén deshabilitadas en la configuración del módulo.

² La detección de rebase por defecto nunca está habilitada para los rangos de resistencia.

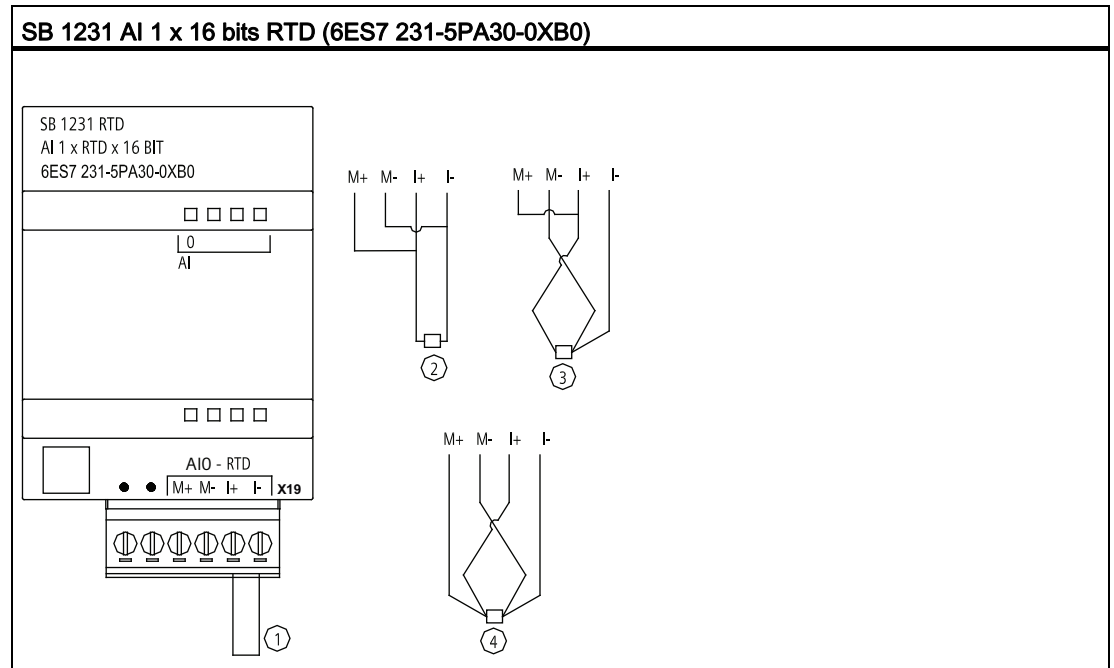
³ Si la alarma de rotura de hilo está deshabilitada y se presenta una condición de rotura de hilo en la línea del sensor, el módulo puede señalar valores aleatorios.

La Signal Board analógica SB 1231 RTD mide el valor de la resistencia conectada a las entradas de la Signal Board. El tipo de medición puede elegirse entre "resistor" y "resistor térmico".

- "Resistor": el valor máximo del rango nominal serán 27648 decimales.
- "Resistor térmico": el valor se expresará en grados multiplicados por diez (p. ej. 25,3 grados se expresarán como 253 decimales).

La Signal Board SB 1231 RTD soporta mediciones con base en conexiones de 2, 3 y 4 hilos que van al sensor de resistencia.

Tabla A- 204 Diagrama de cableado de la SB 1231 AI 1 x 16 bits RTD



- ① Entrada de bucle RTD no utilizada
- ② RTD de dos hilos
- ③ RTD de tres hilos
- ④ RTD de cuatro hilos

Nota: Los conectores deben ser de oro. Consulte el anexo C, Piezas de repuesto, para ver la referencia.

Tabla A- 205 Asignación de pines de conectores para SB 1231 AI 1 x 16 bits RTD (6ES7 231-5PA30-0XB0)

Pin	X19 (oro)
1	Sin conexión
2	Sin conexión
3	AI 0 M+/RTD
4	AI 0 M-/RTD
5	AI 0 I+/RTD
6	AI 0 I-/RTD

A.10.5.2 Tablas de selección para la SB 1231 RTD

Tabla A- 206 Rangos y precisión de los diferentes sensores soportados por los módulos RTD

Coeficiente de temperatura	Tipo de RTD	Rango de saturación mínimo ¹	Rango nominal límite inferior	Rango nominal límite superior	Rango de saturación máximo ²	Exactitud del rango normal a 25 °C	Exactitud del rango normal de -20 °C a 60 °C
Pt 0,003850 ITS90 DIN EN 60751	Pt 10	-243,0°C	-200,0°C	850,0°C	1000,0°C	±1,0°C	±2,0°C
	Pt 50	-243,0°C	-200,0°C	850,0°C	1000,0°C	±0,5°C	±1,0°C
	Pt 100						
	Pt 200						
	Pt 500						
Pt 0,003902 Pt 0,003916 Pt 0,003920	Pt 100	-243,0°C	-200,0°C	850,0°C	1000,0°C	±0,5°C	±1,0°C
	Pt 200						
	Pt 500						
	Pt 1000						
Pt 0,003910	Pt 10	-273,2°C	-240,0°C	1100,0°C	1295°C	±1,0°C	±2,0°C
	Pt 50	-273,2°C	-240,0°C	1100,0°C	1295°C	±0,8°C	±1,6°C
	Pt 100						
	Pt 500						
Ni 0,006720 Ni 0,006180	Ni 100	-105,0°C	-60,0°C	250,0°C	295,0°C	±0,5°C	±1,0°C
	Ni 120						
	Ni 200						
	Ni 500						
	Ni 1000						
LG-Ni 0,005000	LG-Ni 1000	-105,0°C	-60,0°C	250,0°C	295,0°C	±0,5°C	±1,0°C
Ni 0,006170	Ni 100	-105,0°C	-60,0°C	180,0°C	212,4°C	±0,5°C	±1,0°C
Cu 0,004270	Cu 10	-240,0°C	-200,0°C	260,0°C	312,0°C	±1,0°C	±2,0°C
Cu 0,004260	Cu 10	-60,0°C	-50,0°C	200,0°C	240,0°C	±1,0°C	±2,0°C
	Cu 50	-60,0°C	-50,0°C	200,0°C	240,0°C	±0,6°C	±1,2°C
	Cu 100						
Cu 0,004280	Cu 10	-240,0°C	-200,0°C	200,0°C	240,0°C	±1,0°C	±2,0°C
	Cu 50	-240,0°C	-200,0°C	200,0°C	240,0°C	±0,7°C	±1,4°C
	Cu 100						

¹ Los valores de RTD inferiores al valor del rango de saturación mínimo se notifican como -32768.

² Los valores de RTD superiores al valor del rango de saturación máximo se notifican como +32768.

Tabla A- 207 Resistencia

Rango	Rango de saturación mínimo	Rango nominal límite inferior	Rango nominal límite superior	Rango de saturación máximo ¹	Exactitud del rango normal a 25 °C	Exactitud del rango normal de -20 °C a 60 °C
150 Ω	n/a	0 (0 Ω)	27648 (150 Ω)	176,383 Ω	±0,05%	±0,1%
300 Ω	n/a	0 (0 Ω)	27648 (300 Ω)	352,767 Ω	±0,05%	±0,1%
600 Ω	n/a	0 (0 Ω)	27648 (600 Ω)	705,534 Ω	±0,05%	±0,1%

¹ Los valores de resistencia superiores al valor del rango de saturación máximo se notifican como 32767.

Nota

El módulo reporta 32767 en todo canal que esté activado y que no tenga conectado ningún sensor. Si la detección de rotura de hilo también está habilitada, en el módulo se encenderán de forma intermitente los LEDs rojos correspondientes.

En caso de utilizar rangos de 500 Ω y 1000 Ω del RTD con otros resistores de menor valor, el error puede aumentar al doble del error especificado.

La exactitud más elevada se alcanza para los rangos de 10 Ω del RTD con conexiones a 4 hilos.

La resistencia de los hilos en el modo a 2 hilos dará lugar a un error de lectura en el sensor y por ello no se garantiza la exactitud.

Tabla A- 208 Reducción de ruido y tiempos de actualización para los módulos RTD

Selección de supresión de frecuencias	Tiempo de integración	Módulo de 4/2 hilos y 1 canal Tiempo de actualización (segundos)	Módulo de 3 hilos y 1 canal Tiempo de actualización (segundos)
400 Hz (2,5 ms)	10 ms ¹	0,036	0,071
60 Hz (16,6 ms)	16,67 ms	0,056	0,111
50 Hz (20 ms)	20 ms	0,066	1,086
10 Hz (100 ms)	100 ms	0,306	0,611

¹ Para mantener la resolución y la exactitud del módulo seleccionando el filtro de 400 Hz, el tiempo de integración es de 10 ms. Esta selección también rechaza perturbaciones de 100 Hz y 200 Hz.

ATENCIÓN

Después de aplicar tensión al módulo, éste lleva a cabo una calibración interna del conversor de analógico a digital. Durante este tiempo, el módulo indica un valor de 32767 en cada canal hasta que haya información válida disponible en el canal respectivo. Es posible que el programa de usuario deba autorizar este tiempo de inicialización. Como la configuración del módulo puede modificar la longitud del tiempo de inicialización, es conveniente verificar el comportamiento o el módulo en la configuración. Si es necesario, se puede incluir lógica en el programa de usuario para adaptarse al tiempo de inicialización del módulo.

A.11 BB 1297 Battery Board

BB 1297 Battery Board

La Battery Board (placa de batería) BB 1297 de S7-1200 está pensada para proporcionar respaldo a largo plazo del reloj de tiempo real. Se puede conectar en la ranura para Signal Board de la CPU S7-1200 (firmware 3.0 y posteriores). Debe agregar la BB 1297 a la configuración de dispositivo y descargar la configuración de hardware en la CPU para que la BB esté operativa.

La batería (tipo CR1025) no se suministra con la BB 1297 y debe adquirirla el usuario.

Nota

La BB 1297 tiene un diseño mecánico que permite que las CPU cuenten con el firmware 3.0 y versiones posteriores.

No utilice la BB 1297 con CPU de versiones anteriores, ya que el conector de la BB 1297 no se podrá enchufar a la CPU.

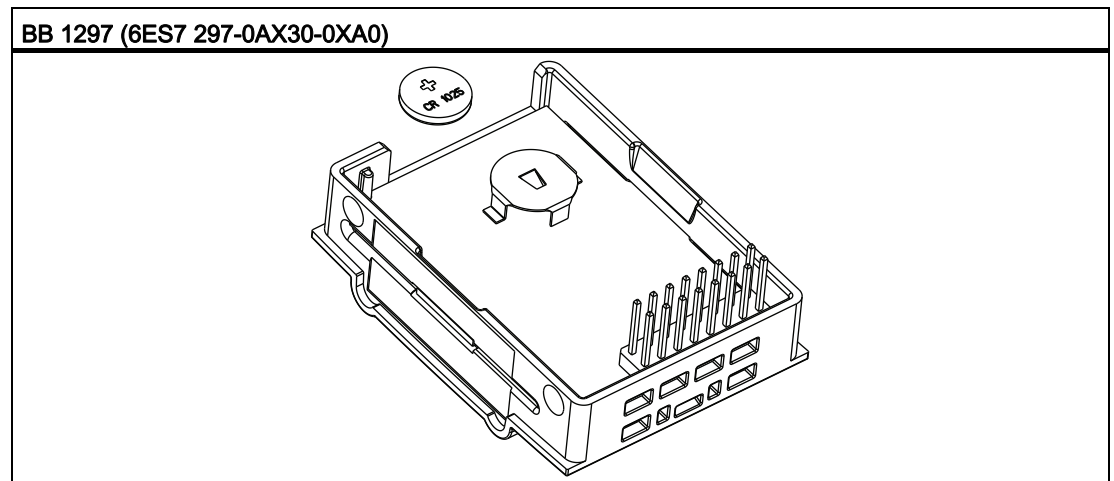
Tabla A- 209 Especificaciones generales

Datos técnicos	BB 1297 Battery Board
Referencia	6ES7 297-0AX30-0XA0
Dimensiones A x A x P (mm)	38 x 62 x 21
Peso	28 gramos
Disipación de potencia	0,5 W
Consumo de corriente (bus SM)	11 mA
Consumo de corriente (24 V DC)	ninguno

Batería (no incluida)	BB 1297 Battery Board
Tiempo de retención	Aproximadamente 1 año
Tipo de batería	CR1025 Consulte Instalar o sustituir la batería en la BB 1297 (Página 49)
Tensión nominal	3 V
Capacidad nominal	30 mAh mínimo

Diagnóstico	BB 1297 Battery Board
Nivel crítico de batería	< 2,5 V
Diagnóstico de batería	<p>Indicador de baja tensión:</p> <ul style="list-style-type: none"> La baja tensión de la batería hace que el LED MAINT de la CPU se encienda con una luz ámbar continua. Evento de búfer de diagnóstico: 16#06:2700 "Mantenimiento solicitado de submódulo: Como mínimo hay una batería agotada (BATTF)"
Estado de la batería	<p>Hay un bit de estado de la batería</p> <p>0 = Batería OK</p> <p>1 = Batería baja</p>
Actualización del estado de la batería	El estado de la batería se actualiza durante el encendido y después, diariamente mientras la CPU está en estado operativo RUN.

Tabla A- 210 Diagrama de inserción para la BB 1297



A.12 Interfaces de comunicación

A.12.1 PROFIBUS

A.12.1.1 CM 1242-5

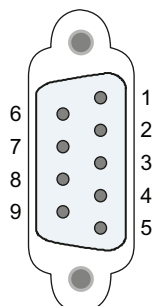
Tabla A- 211 Datos técnicos del CM 1242-5

Datos técnicos	
Referencia	6GK7 242-5DX30-0XE0
Interfaces	
Conexión a PROFIBUS	Conector hembra Sub-D de 9 polos
Consumo máximo de corriente en la interfaz PROFIBUS en caso de conexión de componentes de red (por ejemplo, componentes ópticos)	15 mA a 5 V (sólo para la terminación de bus) *)
Condiciones ambientales permitidas	
Temperatura ambiente	
<ul style="list-style-type: none"> • durante el almacenamiento • durante el transporte • durante la fase de servicio en caso de instalación vertical (perfil horizontal) • durante la fase de servicio en caso de instalación horizontal (perfil vertical) 	<ul style="list-style-type: none"> • -40 °C... + 70 °C • -40 °C... + 70 °C • 0 °C... + 55 °C • 0 °C... + 45 °C
Humedad relativa de 25 °C durante la fase de servicio, sin condensación, como máximo	95 %
Clase de protección	IP20
Alimentación eléctrica, consumo de corriente, potencia perdida	
Clase de la alimentación eléctrica	DC
Alimentación eléctrica del bus de panel posterior	5 V
Corriente consumida (típica)	150 mA
Potencia real perdida (típica)	0,75 W
Medidas y peso	
<ul style="list-style-type: none"> • Ancho • Alto • Profundo 	<ul style="list-style-type: none"> • 30 mm • 100 mm • 75 mm
Peso	
<ul style="list-style-type: none"> • Peso neto • Peso incluyendo el embalaje 	<ul style="list-style-type: none"> • 115 g • 152 g

*) La carga de corriente mediante un consumidor externo, que se conecta entre VP (pin 6) y DGND (pin 5), puede ser de 15 mA como máximo para la terminación del bus (resistente a cortocircuito).

Interfaz PROFIBUS

Tabla A- 212 Asignación de contactos del conector hembra Sub-D



Pin	Descripción	Pin	Descripción
1	- libre -	6	P5V2: alimentación eléctrica de +5V
2	- libre -	7	- libre -
3	RxD/TxD-P: Conductor de datos B	8	RxD/TxD-N: Conductor de datos A
4	RTS	9	- libre -
5	M5V2: tierra de referencia de datos (masa DGND)	Carcasa	Conexión de tierra

A.12.1.2 CM 1243-5

Tabla A- 213 Datos técnicos del CM 1243-5

Datos técnicos	
Referencia	6GK7 243-5DX30-0XE0
Interfaces	
Conexión a PROFIBUS	Conector hembra Sub-D de 9 polos
Consumo máximo de corriente en la interfaz PROFIBUS en caso de conexión de componentes de red (por ejemplo, componentes ópticos)	15 mA a 5 V (sólo para la terminación de bus) *)
Condiciones ambientales permitidas	
Temperatura ambiente	
<ul style="list-style-type: none"> durante el almacenamiento durante el transporte durante la fase de servicio en caso de instalación vertical (perfil horizontal) durante la fase de servicio en caso de instalación horizontal (perfil vertical) 	<ul style="list-style-type: none"> -40 °C... + 70 °C -40 °C... + 70 °C 0 °C... + 55 °C 0 °C... + 45 °C
Humedad relativa de 25 °C durante la fase de servicio, sin condensación, como máximo	95 %
Clase de protección	IP20
Alimentación eléctrica, consumo de corriente, potencia perdida	
Clase de la alimentación eléctrica	DC
Tensión de alimentación / externa	24 V
<ul style="list-style-type: none"> mínima máxima 	<ul style="list-style-type: none"> 19,2 V 28,8 V
Corriente consumida (típica)	
<ul style="list-style-type: none"> de DC 24 V desde el bus de panel posterior de la S7-1200 	<ul style="list-style-type: none"> 100 mA 0 mA

Datos técnicos

Potencia real perdida (típica)

- de DC 24 V
- desde el bus de panel posterior de S7-1200
- 2,4 W
- 0 W

Alimentación eléctrica de DC 24 V / externa

- Sección de cable mínima
- Sección de cable máxima
- Momento de apriete de los bornes roscados
- mín.: 0,14 mm² (AWG 25)
- máx.: 1,5 mm² (AWG 15)
- 0,45 Nm (4 lb.in.)

Medidas y peso

- Ancho
- Alto
- Profundo
- 30 mm
- 100 mm
- 75 mm

Peso

- Peso neto
- Peso incluyendo el embalaje
- 134 g
- 171 g

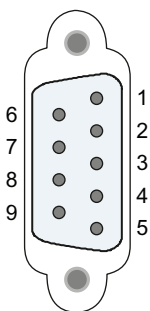
*) La carga de corriente mediante un consumidor externo, que se conecta entre VP (pin 6) y DGND (pin 5), puede ser de 15 mA como máximo para la terminación del bus (resistente a cortocircuito).

Nota

El CM 1243-5 (módulo maestro PROFIBUS) requiere alimentación de la alimentación de sensores de 24 V DC de la CPU.

Interfaz PROFIBUS

Tabla A- 214 Asignación de contactos del conector hembra Sub-D



Pin	Descripción	Pin	Descripción
1	- libre -	6	VP: alimentación de tensión +5 V sólo para resistencias de cierre de bus (terminación); no para la alimentación de aparatos externos
2	- libre -	7	- libre -
3	RxD/TxD-P: Conductor de datos B	8	RxD/TxD-N: Conductor de datos A
4	CNTR-P: RTS	9	- libre -
5	DGND: masa para señales de datos y VP	Carcasa	Conexión de tierra

Cable PROFIBUS

ATENCIÓN
Colocación del apantallado del cable PROFIBUS
El apantallado del cable PROFIBUS debe estar colocado.
Aísle a tal fin un extremo del cable PROFIBUS y una la pantalla con la puesta a tierra de la función.

A.12.2 GPRS

Nota

El CP GPRS no está aprobado para aplicaciones marítimas

Los módulos siguientes no tienen aprobación marítima:

- Módulo CP 1242-7 GPRS

Nota

Para utilizar estos módulos el firmware de la CPU debe tener la versión 2.0 o superior.

A.12.2.1 CP 1242-7

Tabla A- 215 Datos técnicos del CP 1242-7

Datos técnicos	
Referencia	6GK7 242-7KX30-0XE0
Interfaz de radiofrecuencia	
Conexión de antena	Conector SMA
Impedancia nominal	50 Ohm
Radioconexión	
Potencia de emisión máxima	<ul style="list-style-type: none"> • GSM 850, Class 4: +33 dBm ±2dBm • GSM 900, Class 4: +33 dBm ±2dBm • GSM 1800, Class 1: +30 dBm ±2dBm • GSM 1900, Class 1: +30 dBm ±2dBm
GPRS	Clase de Multislot 10 Clase de equipo terminal B Esquema de codificación 1...4 (GMSK)
SMS	Modo operativo saliente: MO Servicio: punto a punto

Datos técnicos	
Condiciones ambientales permitidas	
Temperatura ambiente	
• durante el almacenamiento	• -40 °C... + 70 °C
• durante el transporte	• -40 °C... + 70 °C
• durante la fase de servicio en caso de instalación vertical (perfil horizontal)	• 0 °C... + 55 °C
• durante la fase de servicio en caso de instalación horizontal (perfil vertical)	• 0 °C... + 45 °C
Humedad relativa de 25 °C durante la fase de servicio, sin condensación, como máximo	95 %
Clase de protección	IP20
Alimentación eléctrica, consumo de corriente, potencia perdida	
Clase de la alimentación eléctrica	DC
Tensión de alimentación / externa	24 V
• mínima	• 19,2 V
• máxima	• 28,8 V
Corriente consumida (típica)	
• de DC 24 V	• 100 mA
• desde el bus de panel posterior de S7-1200	• 0 mA
Potencia real perdida (típica)	
• de DC 24 V	• 2,4 W
• desde el bus de panel posterior de la S7-1200	• 0 W
Alimentación eléctrica de DC 24 V	
• Sección de cable mínima	• mín.: 0,14 mm ² (AWG 25)
• Sección de cable máxima	• máx.: 1,5 mm ² (AWG 15)
• Momento de apriete de los bornes roscados	• 0,45 Nm (4 lb.in.)
Medidas y peso	
• Ancho	• 30 mm
• Alto	• 100 mm
• Profundo	• 75 mm
Peso	
• Peso neto	• 133 g
• Peso incluyendo el embalaje	• 170 g

Datos técnicos de la antena de GSM/GPRS ANT794-4MR

ANT794-4MR	
Referencia	6NH9860-1AA00
Redes de telefonía móvil	GSM / GPRS
Gamas de frecuencia	<ul style="list-style-type: none"> • 824...960 MHz (GSM 850, 900) • 1 710...1 880 MHz (GSM 1 800) • 1 900...2 200 MHz (GSM / UMTS)
Característica	omnidireccional
Ganancia de la antena	0 dB
Impedancia	50 Ohm
Relación de ondas estacionarias (SWR)	< 2,0
Potencia máx.	20 W
Polaridad	lineal vertical
Conector	SMA
Longitud del cable de antena	5 m
Material exterior	PVC duro, resistente a UV
Clase de protección	IP20
Condiciones ambientales permitidas	<ul style="list-style-type: none"> • Temperatura de servicio • Temperatura de transporte/almacenaje • Humedad relativa
	<ul style="list-style-type: none"> • -40 °C hasta +70 °C • -40 °C hasta +70 °C • 100 %
Material exterior	PVC duro, resistente a UV
Construcción	Antena con cable de 5 m unido fijo y conector macho SMA
Medidas (D x H) en mm	25 x 193
Peso	<ul style="list-style-type: none"> • Antena incl. cable • Piezas para montaje
	<ul style="list-style-type: none"> • 310 g • 54 g
Montaje	Con escuadra adjuntada

Datos técnicos de la antena plana ANT794-3M

Referencia	6NH9870-1AA00	
Redes de telefonía móvil	GSM 900	GSM 1800/1900
Gamas de frecuencia	890 - 960 MHz	1710 - 1990 MHz
Relación de ondas estacionarias (VSWR)	≤ 2:1	≤ 1,5:1
Pérdida de retorno (Tx)	≈ 10 dB	≈ 14 dB
Ganancia de la antena	0 dB	
Impedancia	50 Ohm	

A.12 Interfaces de comunicación

Potencia máx.	10 W
Cable de la antena	Cable HF RG 174 (conectado fijamente) con un conector SMA
Longitud del cable	1,2 m
Clase de protección	IP 64
Margen de temperatura permitido	-40°C hasta +75°C
Inflamabilidad	UL 94 V2
Material exterior	ABS Polylac PA-765, gris luminoso (RAL 7035)
Medidas (An x L x Al) en mm	70,5 x 146,5 x 20,5
Peso	130 g

A.12.3 CM 1243-2 AS-i Master

A.12.3.1 Datos técnicos del maestro AS-i CM 1243-2

Tabla A- 216 Datos técnicos del maestro AS-i CM 12432-2

Datos técnicos	
Referencia	3RK7243-2AA30-0XB0
Interfaces	
Consumo máximo de corriente	
Del bus de fondo SIMATIC	Máx. 250 mA, Tensión de alimentación 5 V DC del bus de fondo SIMATIC
Del cable AS-i	Máx. 100 mA
Ocupación de los pines	Véase el apartado Conexiones eléctricas del maestro AS-i CM 1243-2 (Página 865)
Sección del conductor	0,2 mm ² (AWG 24) ... 3,3 mm ² (AWG 12)
Par de apriete del conector ASI	0,56 Nm
Condiciones ambientales admisibles	
Temperatura ambiente	
Durante el almacenamiento	-40 °C a 70 °C
Durante el transporte	-40°C a 70 °C
Durante la fase operativa con instalación vertical (raíl estándar horizontal)	0 °C a 55 °C
Durante la fase operativa con instalación horizontal (raíl estándar vertical)	0 °C a 45 °C
Humedad relativa a 25 °C durante la fase operativa, sin condensación, máxima	95 %
Grado de protección	IP20
Fuente de alimentación, consumo de corriente, pérdidas	
Tipo de fuente de alimentación	DC

Datos técnicos	
Consumo de corriente (típico)	
Del bus de fondo S7-1200	200 mA
Pérdidas (típicas)	
Del bus de fondo S7-1200	2,4 W de AS-i 0,5 W
Dimensiones y peso	
Ancho	30 mm
Altura	100 mm
Profundidad	75 mm
Peso	
Peso neto	122 g
Peso con embalaje	159 g

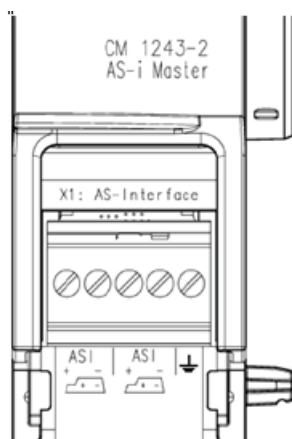
A.12.3.2 Conexiones eléctricas del maestro AS-i CM 1243-2

Alimentación del maestro AS-i CM 1243-2

El maestro AS-i CM 1243-2 se alimenta a través del bus de comunicación del S7-1200. En caso de fallar la alimentación del bus AS-i ello permite enviar un aviso de diagnóstico al S7-1200. El conector de conexión al bus de comunicación se encuentra en el costado derecho del maestro AS-i CM 1243-2.

Bornes de conexión a AS-Interface

El bloque de bornes desmontable para conectar el cable AS-i se encuentra detrás de la tapa inferior del frente del maestro AS-i CM 1243-2.




Si usa el cable perfilado AS-i, el símbolo  le permite reconocer la polaridad correcta del cable de conexión.

Más detalles relativos a cómo desmontar y volver a montar el bloque de bornes figuran en el manual de sistema "SIMATIC S7 Controlador programable S7-1200" (referencia: 6ES7298-8FA30-8DH0).

ATENCIÓN
<p>Intensidad máxima admisible en los contactos de conexión</p> <p>La intensidad máxima admisible en los contactos de conexión es de 8 A. Si se supera este valor por el cable AS-i, entonces el maestro AS-i CM 1243-2 no debe "intercalarse" en el cable AS-i, sino deberá conectarse mediante un cable derivado (sólo se ocupa un par de conexiones del maestro AS-i CM 1243-2).</p> <p>Más detalles sobre la conexión del cable AS-i figuran en el capítulo "Montaje, conexión y puesta en marcha del los módulos" del manual "Maestro AS-i CM 1243-2 y módulo de desacoplamiento de datos AS-i DCM 1271 para SIMATIC S7-1200".</p>

Asignación de bornes

Rotulación	Significado
ASI +	Conexión AS-i, polos positivo
ASI -	Conexión AS-i, polos negativo
	Tierra funcional

A.12.4 RS232, RS422 y RS485

A.12.4.1 Datos técnicos de CB 1241 RS485

Nota

Para utilizar esta CB el firmware de la CPU debe tener la versión 2.0 o superior.

Tabla A- 217 Especificaciones generales

Datos técnicos	CB 1241 RS485
Referencia	6ES7 241-1CH30-1XB0
Dimensiones A x A x P (mm)	38 x 62 x 21
Peso	40 gramos

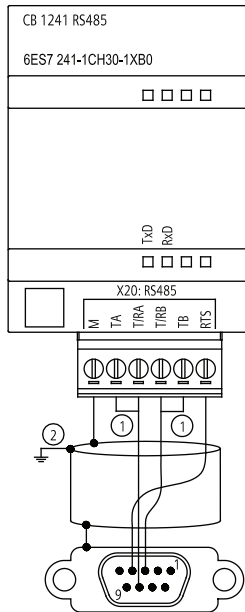
Tabla A- 218 Transmisor y receptor

Datos técnicos	CB 1241 RS485
Tipo	RS485 (semidúplex de 2 hilos)
Rango de tensión en modo común	-7 V a +12 V, 1 segundo, 3 VRMS continuo
Tensión de salida diferencial del transmisor	2 V mín. a $R_L = 100 \Omega$ 1,5 V mín. a $R_L = 54 \Omega$
Terminación y polarización	10 K a +5 V en B, pin 3 RS485 10K a GND en A, pin 4 RS485
Terminación opcional	Conexión del pin TB al pin T/RB, la impedancia de terminación efectiva es de 127Ω , se conecta al pin 3 RS485 Conexión del pin TA al pin T/RA, la impedancia de terminación efectiva es de 127Ω , se conecta al pin 4 RS485
Impedancia de entrada del receptor	5,4K Ω mín. incluyendo terminación
Umbral/sensibilidad del receptor	+/- 0,2 V mín., 60 mV de histéresis típica
Aislamiento Señal RS485 a conexión a masa Señal RS485 a lógica de la CPU	500 V AC durante 1 minuto
Longitud de cable, apantallado	1000 m máx.
Velocidad de transferencia	300 baudios, 600 baudios, 1,2 kbits, 2,4 kbits, 4,8 kbits, 9,6 kbits (valor predeterminado), 19,2 kbits, 38,4 kbits, 57,6 kbits, 76,8 kbits, 115,2 kbits
Paridad	Sin paridad (valor predeterminado), par, impar, marca (bit de paridad siempre puesto a 1), espacio (bit de paridad siempre puesto a 0)
Número de bits de parada	1 (valor predeterminado), 2
Control de flujo	No soportado
Tiempo de espera	0 a 65535 ms

Tabla A- 219 Alimentación eléctrica

Datos técnicos	CB 1241 RS485
Pérdida de potencia (disipación)	1,5 W
Consumo de corriente (bus SM), máx.	50 mA
Consumo de corriente (24 V DC) máx.	80 mA

CB 1241 RS485 (6ES7 241-1CH30-1XB0)



① Conecte "TA" y "TB" tal y como se indica para finalizar la red. (En la red RS485 sólo deben finalizarse los dispositivos terminadores.)

② Utilice un cable de par trenzado apantallado y conéctelo a tierra.

Sólo se terminan los dos extremos de la red RS485. Los dispositivos que están entre los dos dispositivos terminadores no se terminan ni polarizan. Consulte la sección "Polarizar y terminar un conector de red RS485" del Manual de sistema S7-1200.

Tabla A- 220 Asignación de pines de conectores para CB 1241 RS485 (6ES7 241-1CH30-1XB0)

Pin	Conector de 9 pines	X20
1	RS485/GND de lógica	--
2	RS485/No utilizado	--
3	RS485/TxD+	3 - T/RB
4	RS485/RTS	1 - RTS
5	RS485/GND de lógica	--
6	RS485/Alimentación 5 V	--
7	RS485/No utilizado	--
8	RS485/TxD-	4 - T/RA
9	RS485/No utilizado	--
Carcasa		7 - M

Consulte también

Polarizar y terminar un conector de red RS485 (Página 590)

A.12.4.2 Datos técnicos del CM 1241 RS232

Tabla A- 221 Especificaciones generales

Datos técnicos	CM 1241 RS232
Referencia	6ES7 241-1AH30-0XB0
Dimensiones (mm)	30 x 100 x 75
Peso	150 gramos

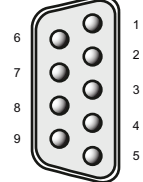
Tabla A- 222 Transmisor y receptor

Datos técnicos	CM 1241 RS232
Tipo	RS232 (dúplex completo)
Tensión de salida del transmisor	+/- 5 V mín. a $R_L = 3K \Omega$
Tensión de salida del transmisor	+/- 15 V DC máx.
Impedancia de entrada del receptor	3 K Ω mín.
Umbral/sensibilidad del receptor	0,8 V mín. bajo, 2,4 máx. alto histéresis típica: 0,5 V
Tensión de entrada del receptor	+/- 30 V DC máx.
Aislamiento	500 V AC durante 1 minuto
Señal RS 232 a conexión a masa	
Señal RS 232 a lógica de la CPU	
Longitud de cable, apantallado	10 m máx.
Velocidad de transferencia	300 baudios, 600 baudios, 1,2 kbits, 2,4 kbits, 4,8 kbits, 9,6 kbits (valor predeterminado), 19,2 kbits, 38,4 kbits, 57,6 kbits, 76,8 kbits, 115,2 kbits
Paridad	Sin paridad (valor predeterminado), par, impar, marca (bit de paridad siempre establecido a 1), espacio (bit de paridad siempre establecido a 0)
Número de bits de parada	1 (valor predeterminado), 2
Control de flujo	Hardware, software
Tiempo de espera	0 a 65535 ms

Tabla A- 223 Alimentación eléctrica

Datos técnicos	CM 1241 RS232
Pérdida de potencia (disipación)	1,1 W
De +5 V DC	220 mA

Tabla A- 224 Conector RS232 (macho)

Pin	Descripción	Conector (macho)	Pin	Descripción
1 DCD	Detección de portadora de datos: Entrada		6 DSR	Equipo de datos listo: Entrada
2 RxD	Datos recibidos de DCE: Entrada		7 RTS	Petición de transmitir Salida
3 TxD	Datos transmitidos a DCE: Salida		8 CTS	Listo para transmitir: Entrada
4 DTR	Terminal de datos disponible: Salida		9 RI	Indicación de timbre (no utilizado)
5 GND	Masa lógica		SHELL	Conexión a masa

A.12.4.3 Datos técnicos del CM 1241 RS422/485

Datos técnicos del CM 1241 RS422/485

Tabla A- 225 Especificaciones generales

Datos técnicos	CM 1241 RS422/485
Referencia	6ES7 241-1CH31-0XB0
Dimensiones A x A x P (mm)	30 x 100 x 75
Peso	155 gramos

Tabla A- 226 Transmisor y receptor

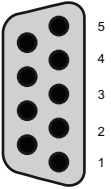
Datos técnicos	CM 1241 RS422/485
Tipo	RS422 o RS485, conector hembra sub-D de 9 polos
Rango de tensión en modo común	-7 V a +12 V, 1 segundo, 3 VRMS continuo
Tensión de salida diferencial del transmisor	2 V mín. a $R_L = 100 \Omega$ 1,5 V mín. a $R_L = 54 \Omega$
Terminación y polarización	10K Ω a +5 V en B, pin PROFIBUS 3 10K Ω a GND en A, pin PROFIBUS 8 Opciones de polarización interna disponibles o bien sin polarización interna. En todos los casos se requiere terminación externa; consulte Polarizar y terminar un conector de red RS485 (Página 590) y Configurar RS422 y RS485 en el Manual de sistema S7-1200. (Página 629)
Impedancia de entrada del receptor	5,4K Ω mín. incluyendo terminación
Umbral/sensibilidad del receptor	+/- 0,2 V mín., 60 mV de histéresis típica
Aislamiento	500 V AC durante 1 minuto
Señal RS485 a conexión a masa	
Señal RS485 a lógica de la CPU	

Datos técnicos	CM 1241 RS422/485
Longitud de cable, apantallado	máx. 1000 m (en función de la velocidad de transferencia)
Velocidad de transferencia	300 baudios, 600 baudios, 1,2 kbits, 2,4 kbits, 4,8 kbits, 9,6 kbits (valor predeterminado), 19,2 kbits, 38,4 kbits, 57,6 kbits, 76,8 kbits, 115,2 kbits
Paridad	Sin paridad (valor predeterminado), par, impar, marca (bit de paridad siempre puesto a 1), espacio (bit de paridad siempre puesto a 0)
Número de bits de parada	1 (valor predeterminado), 2
Control de flujo	Se soporta XON/XOFF para el modo RS422
Tiempo de espera	0 a 65535 ms

Tabla A- 227 Alimentación eléctrica

Datos técnicos	CM 1241 RS422/485
Pérdida de potencia (disipación)	1,2 W
De +5 V DC	240 mA

Tabla A- 228 Conector RS485 o RS422 (hembra)

Pin	Descripción	Conector (hembra)	Pin	Descripción
1	Masa lógica o de comunicación		6 PWR	+5V con resistor en serie de 100 ohmios: Salida
2 TxD+ ¹	Conectada para RS422 Sin uso para RS485: Salida		7	Sin conexión
3 TxD+	Señal B (RxD/TxD+): Entrada/salida		8 TXD-	Señal A (RxD/TxD-): Entrada/salida
4 RTS ²	Petición de transmitir (nivel TTL) salida		9 TXD- ¹	Conectada para RS422 Sin uso para RS485: Salida
5 GND	Masa lógica o de comunicación		SHELL	Conexión a masa

¹ Los pines 2 y 9 solo se utilizan para transmitir señales para RS422.

² RTS es una señal de nivel TTL y se puede emplear para controlar otro dispositivo semidúplex basado en esta señal. Se habilita al transmitir datos y se deshabilita el resto del tiempo.

A.13 TeleService (TS Adapter y TS Adapter modular)

Los manuales siguientes contienen los datos técnicos del TS Adapter IE Basic y el TS Adapter modular:

- Herramientas de ingeniería para software industrial TS Adapter modular
- Herramientas de ingeniería para software industrial TS Adapter IE Basic

A.14 SIMATIC Memory Cards

Referencia	Capacidad
6ES7 954-8LF01-0AA0	24 MB
6ES7 954-8LE01-0AA0	12 MB
6ES7 954-8LB01-0AA0	2 MB

A.15 Simuladores de entradas

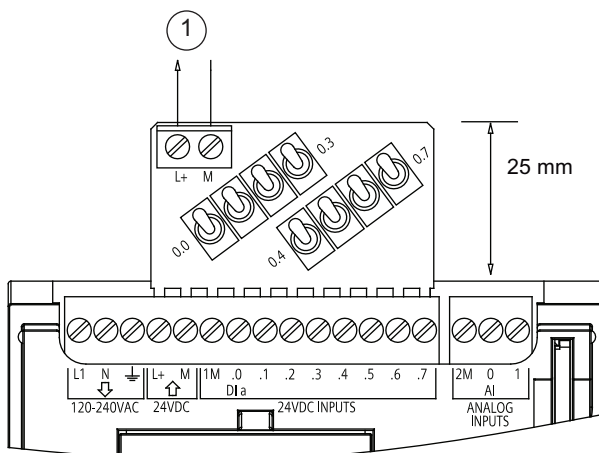
Tabla A- 229 Especificaciones generales

Datos técnicos	Simulador de 8 entradas	Simulador de 14 entradas
Referencia	6ES7 274-1XF30-0XA0	6ES7 274-1XH30-0XA0
Dimensiones A x A x P (mm)	43 x 35 x 23	67 x 35 x 23
Peso	20 gramos	30 gramos
Entradas	8	14
Uso con CPU	CPU 1211C, CPU 1212C	CPU 1214C

⚠ ADVERTENCIA

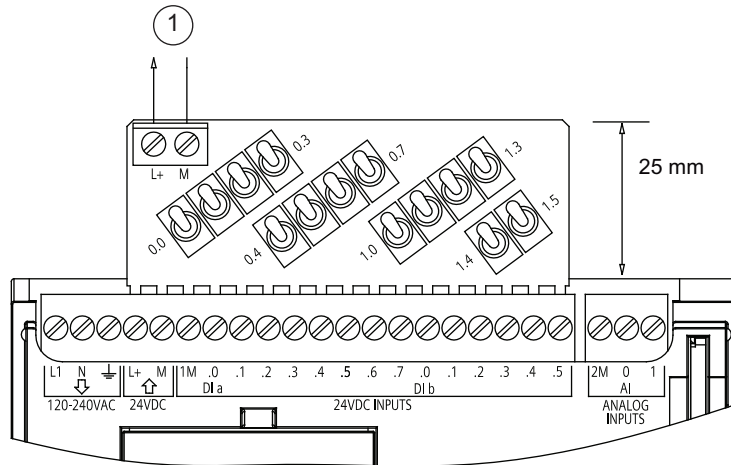
Estos simuladores de entradas no están aprobados para ser utilizados en ubicaciones peligrosas ("hazardous locations") Class I DIV 2 o Class I Zone 2. Los interruptores pueden producir chispas o explotar si se utilizan en ubicaciones Class I DIV 2 o Class I Zone 2.

Simulador de 8 entradas (6ES7 274-1XF30-0XA0)



① Alimentación de sensores de 24 V DC

Simulador de 14 entradas (6ES7 274-1XF30-0XA0)



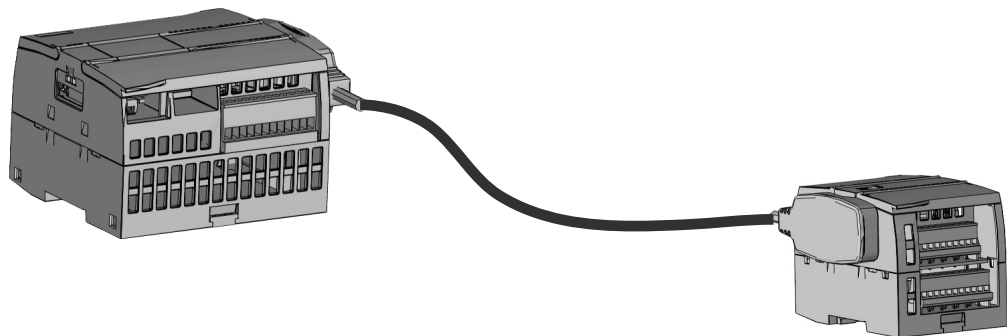
① Alimentación de sensores de 24 V DC

A.16 Cable para módulos de ampliación

Datos técnicos

Referencia	6ES7 290-6AA30-0XA0
Longitud del cable	2 m
Peso	200 g

Consulte el apartado de montaje (Página 55) para más información acerca del montaje y desmontaje del cable de ampliación del S7-1200.



A.17 Productos adicionales

A.17.1 Módulo de alimentación PM 1207

El PM 1207 es un módulo de alimentación para SIMATIC S7-1200. Ofrece las funciones siguientes:

- Entrada 120/230 V AC, salida 24 V DC/2,5 A
- Referencia: 6ESP 332-1SH71

Para obtener más información sobre este producto y consultar la documentación del producto, visite la página web de atención al cliente (<http://www.siemens.com/automation/>).

A.17.2 Módulo de conmutación compacto CSM 1277

El CSM1277 es un módulo de conmutación compacto Industrial Ethernet. Se puede emplear para multiplicar la interfaz Ethernet del S7-1200 y permitir así una comunicación simultánea con paneles de operador, programadoras u otros controladores. Ofrece las funciones siguientes:

- 4 sockets RJ45 para la conexión a Industrial Ethernet
- 3 bujías polares en la placa de bornes para la conexión de la alimentación externa de 24 V DC encima
- LEDs para mostrar el estado y el diagnóstico de puertos Industrial Ethernet
- Referencia: 6GK7 277-1AA00-0AA0

Para obtener más información sobre este producto y consultar la documentación del producto, visite la página web de atención al cliente (<http://www.siemens.com/automation/>).

Calcular la corriente necesaria

La CPU incorpora una fuente de alimentación interna capaz de abastecer la CPU, los módulos de ampliación y otros consumidores de 24 V DC.

Hay cuatro tipos de módulos de ampliación, a saber:

- Los módulos de señales (SM) se montan a la derecha de la CPU. Toda CPU permite conectar un número máximo posible de módulos de señales, sin considerar la corriente disponible.
 - La CPU 1214C y la CPU 1215C permiten conectar 8 módulos de señales.
 - La CPU 1212C permite conectar 2 módulos de señales.
 - La CPU 1211C no permite conectar módulos de señales.
- Los módulos de comunicación (CM) se montan a la izquierda de la CPU. Se permiten como máximo 3 módulos de comunicación para cualquier CPU, sin considerar la corriente disponible.
- Las Signal Boards (SB), placas de comunicación (CB) y Battery Boards (BB) se montan en el lado superior de la CPU. Se permite como máximo 1 Signal Board, Communication Board o Battery Board para cualquier CPU.

Utilice la información siguiente como guía para determinar cuánta energía (o corriente) puede suministrar la CPU a la configuración.

Toda CPU suministra alimentación tanto de 5 V DC como de 24 V DC:

- La CPU suministra 5 V DC a los módulos de ampliación cuando son conectados. Si el consumo de 5 V DC de los módulos de ampliación excede la corriente que ofrece la CPU, habrá que desconectar tantos módulos de ampliación como sea necesario para no excederla.
- Toda CPU incorpora una fuente de alimentación de sensores de 24 V DC que puede suministrar 24 V DC a las entradas locales, o bien a las bobinas de relé de los módulos de ampliación. Si el consumo de 24 V DC excede la corriente disponible de la CPU, es posible agregar una fuente de alimentación externa de 24 V DC para suministrar 24 V DC a los módulos de ampliación. La alimentación de 24 V DC debe conectarse manualmente a las entradas o bobinas de relé.

 **ADVERTENCIA**

Si se conecta una fuente de alimentación externa de 24 V DC en paralelo con la fuente de alimentación DC de sensores, podría surgir un conflicto entre ambas fuentes, ya que cada una intenta establecer su propio nivel de tensión de salida.

Este conflicto puede reducir la vida útil u ocasionar la avería inmediata de una o ambas fuentes de alimentación y, en consecuencia, el funcionamiento imprevisible del sistema PLC. El funcionamiento imprevisible puede producir la muerte, lesiones corporales graves y/o daños materiales.

La fuente de alimentación DC de sensores de la CPU y cualquier fuente de alimentación externa deben alimentar diferentes puntos. Se permite una conexión común de los cables neutros.

Algunos puertos de entrada de alimentación de 24 V del sistema PLC están interconectados y tienen un circuito lógico común que conecta varios bornes M. La fuente de alimentación de 24V de la CPU, la entrada de alimentación de las bobinas de relé de un SM, o bien una fuente de alimentación analógica sin aislamiento galvánico son ejemplos de circuitos interconectados si no tienen aislamiento galvánico según las hojas de datos técnicos. Todos los bornes M sin aislamiento galvánico deben conectarse al mismo potencial de referencia externo.

 **ADVERTENCIA**

Si los bornes M sin aislamiento galvánico se conectan a diferentes potenciales de referencia, circularán corrientes indeseadas que podrían averiar o causar reacciones inesperadas en el PLC y los equipos conectados.

Las averías o reacciones inesperadas podrían producir la muerte, lesiones corporales graves y/o daños materiales.

Asegúrese que todos los bornes M sin aislamiento galvánico de un sistema PLC están conectados al mismo potencial de referencia.

Para más información sobre la corriente disponible de las distintas CPUs y el consumo de corriente de los módulos de señales, consulte los datos técnicos (Página 737).

Nota

Si se excede la corriente disponible de la CPU, es posible que no se pueda conectar el número máximo de módulos soportado.

Ejemplo de corriente disponible

El ejemplo siguiente muestra el cálculo del consumo de corriente de una configuración que incluye una CPU 1214C AC/DC/relé, una Signal Board SB 1223 de 2 entradas 24 V DC / 2 salidas 24 V DC, un módulo de comunicación CM 1241, tres módulos de señales SM 1223 de 8 entradas DC y 8 salidas de relé, así como un módulo de señales SM 1221 de 8 entradas DC. Este ejemplo incluye 48 entradas y 36 salidas en total.

Nota

La CPU ya ha asignado la corriente necesaria para accionar las bobinas de relé internas. Por tanto, no es necesario incluir el consumo de corriente de las bobinas de relé internas en el cálculo.

La CPU de este ejemplo suministra suficiente corriente de 5 V DC a los SMs, pero la alimentación de sensores no suministra suficiente corriente de 24 V DC a todas las entradas y bobinas de relé de ampliación. Las E/S requieren 456 mA, pero la CPU sólo puede suministrar 400 mA. Para esta configuración se necesita una fuente de alimentación adicional de 56 mA (como mínimo) a 24 V DC para operar todas las entradas y salidas de 24 V DC.

Tabla B- 1 Ejemplo de corriente disponible

Corriente disponible de la CPU	5 V DC	24 V DC
CPU 1214C AC/DC/relé	1600 mA	400 mA
<i>Menos</i>		
Consumo del sistema	5 V DC	24 V DC
CPU 1214C, 14 entradas	-	14 * 4 mA = 56 mA
1 SB 1223 2 entradas x 24 V DC / 2 salidas x 24 V DC	50 mA	2 * 4 mA = 8 mA
1 CM 1241 RS422/485, 5 V de corriente	220 mA	
3 SM 1223, 5 V de corriente	3 * 145 mA = 435 mA	-
1 SM 1221, 5 V de corriente	1 * 105 mA = 105 mA	-
3 SM 1223, 8 entradas c/u	-	3 * 8 * 4 mA = 96 mA
3 SM 1223, 8 salidas de relé c/u	-	3 * 8 * 11 mA = 264 mA
1 SM 1221, 8 entradas c/u	-	8 * 4 mA = 32 mA
Consumo total	810 mA	456 mA
<i>Igual a</i>		

Corriente disponible de la CPU	5 V DC	24 V DC
Balance de corriente	5 V DC	24 V DC
Balance total de corriente	790 mA	(56 mA)

Forma de calcular la corriente necesaria

Utilice la tabla siguiente para determinar cuánta energía (o corriente) puede suministrar la CPU S7-1200 a la configuración en cuestión. En los datos técnicos (Página 737) encontrará información sobre la corriente disponible de la CPU y el consumo de los módulos de señales.

Tabla B- 2 Cálculo de la corriente necesaria

Corriente disponible de la CPU	5 V DC	24 V DC
<i>Menos</i>		
Consumo del sistema	5 V DC	24 V DC
Consumo total		
<i>Igual a</i>		
Balance de corriente	5 V DC	24 V DC
Balance total de corriente		

Referencias

C.1 Módulos CPU

Tabla C- 1 CPUs S7-1200

Modelos de CPUs		Referencia
CPU 1211C	CPU 1211C DC/DC/DC	6ES7 211-1AE31-0XB0
	CPU 1211C AC/DC/relé	6ES7 211-1BE31-0XB0
	CPU 1211C DC/DC/relé	6ES7 211-1HE31-0XB0
CPU 1212C	CPU 1212C DC/DC/DC	6ES7 212-1AE31-0XB0
	CPU 1212C AC/DC/relé	6ES7 212-1BE31-0XB0
	CPU 1212C DC/DC/relé	6ES7 212-1HE31-0XB0
CPU 1214C	CPU 1214C DC/DC/DC	6ES7 214-1AG31-0XB0
	CPU 1214C AC/DC/relé	6ES7 214-1BG31-0XB0
	CPU 1214C DC/DC/relé	6ES7 214-1HG31-0XB0
CPU 1215C	CPU 1215C DC/DC/DC	6ES7 215-1AG31-0XB0
	CPU 1215C AC/DC/relé	6ES7 215-1BG31-0XB0
	CPU 1215C DC/DC/relé	6ES7 215-1HG31-0XB0

C.2 Módulos de señales (SM), Signal Boards (SB) y Battery Boards (BB)

Tabla C- 2 Módulos de señales (SM)

Módulos de señales		Referencia
Entrada digital	SM 1221 8 entradas de 24 V DC (NPN/PNP)	6ES7 221-1BF30-0XB0
	SM 1221 16 entradas de 24 V DC (NPN/PNP)	6ES7 221-1BH30-0XB0
Salida digital	SM 1222 8 salidas de 24 V DC (PNP)	6ES7 222-1BF30-0XB0
	SM 1222 16 salidas de 24 V DC (PNP)	6ES7 222-1BH30-0XB0
	SM 1222 8 salidas de relé	6ES7 222-1HF30-0XB0
	SM 1222 8 salidas de relé (inversor)	6ES7 222-1XF30-0XB0
	SM 1222 16 salidas de relé	6ES7 222-1HH30-0XB0
Entrada/salida digital	SM 1223 8 entradas de 24 V DC (NPN/PNP) / 8 salidas de 24 V DC (PNP)	6ES7 223-1BH30-0XB0
	SM 1223 16 entradas de 24 V DC (NPN/PNP) / 16 salidas de 24 V DC (PNP)	6ES7 223-1BL30-0XB0
	SM 1223 8 entradas de 24 V DC (NPN/PNP) / 8 salidas de relé	6ES7 223-1PH30-0XB0
	SM 1223 16 entradas de 24 V DC (NPN/PNP) / 16 salidas de relé	6ES7 223-1PL30-0XB0

Referencias

C.2 Módulos de señales (SM), Signal Boards (SB) y Battery Boards (BB)

Módulos de señales		Referencia
	SM 1223 8 entradas de 120/230 V AC (NPN/PNP) / 8 salidas de relé	6ES7 223-1QH30-0XB0
Entrada analógica	SM 1231 4 entradas analógicas	6ES7 231-4HD30-0XB0
	SM 1231 8 entradas analógicas	6ES7 231-4HF30-0XB0
	SM 1231 4 x entradas analógicas x 16 bits (high feature)	6ES7 231-5ND30-0XB0
Salida analógica	SM 1232 2 salidas analógicas	6ES7 232-4HB30-0XB0
	SM 1232 4 salidas analógicas	6ES7 232-4HD30-0XB0
Entrada/salida analógica	SM 1234 4 entradas analógicas / 2 salidas analógicas	6ES7 234-4HE30-0XB0
RTD y termopar	SM 1231 TC 4 x 16 bits	6ES7 231-5QD30-0XB0
	SM 1231 TC 8 x 16 bits	6ES7 231-5QF30-0XB0
	SM 1231 RTD 4 x 16 bits	6ES7 231-5PD30-0XB0
	SM 1231 RTD 8 x 16 bits	6ES7 231-5PF30-0XB0

Tabla C- 3 Signal Boards (SB) y Battery Boards (BB)

Signal Boards y Battery Boards		Referencia
Entrada digital	SB 1221 200 KHz 4 entradas de 24 V DC (PNP),	6ES7 221-3BD30-0XB0
	SB 1221 200 KHz 4 entradas de 5 V DC (PNP)	6ES7 221-3AD30-0XB0
Salida digital	SB 1222 200 KHz 4 salidas de 24 V DC (NPN/PNP)	6ES7 222-1BD30-0XB0
	SB 1222 200 KHz 4 salidas de 5 V DC (NPN/PNP)	6ES7 222-1AD30-0XB0
Entrada/salida digital	SB 1223 2 entradas de 24 V DC (NPN) / 2 salidas de 24 V DC (PNP)	6ES7 223-0BD30-0XB0
	SB 1223 200 KHz 2 entradas de 24 V DC (PNP) / 2 salidas de 24 V DC (NPN/PNP)	6ES7 223-3BD30-0XB0
	SB 1223 200 KHz 2 entradas de 5 V DC (PNP) / 2 salidas de 5 V DC (NPN/PNP)	6ES7 223-3AD30-0XB0
Analógico	SB 1232 de 1 salida analógica	6ES7 232-4HA30-0XB0
	SB 1231 de 1 entrada analógica	6ES7 231-4HA30-0XB0
	SB 1231 de termopar de 1 entrada analógica	6ES7 231-5QA30-0XB0
	SB 1231 de 1 entrada analógica RTD	6ES7 231-5PA30-0XB0
Batería	BB 1297 Battery	6ES7 297-0AX30-0XA0

C.3 Comunicación

Tabla C- 4 Módulo de comunicación (CM)

Módulo de comunicación (CM)			Referencia
RS232, RS422 y RS485	CM 1241 RS232	RS232	6ES7 241-1AH30-0XB0
	CM 1241 RS422/485	RS422/485	6ES7 241-1CH31-0XB0
PROFIBUS	CM 1243-5	Maestro PROFIBUS	6GK7 243-5DX30-0XE0
	CM 1242-5	Esclavo PROFIBUS	6GK7 242-5DX30-0XE0
AS-i Master	CM 1243-2	AS-i Master	3RK7 243-2AA30-0XB0

Tabla C- 5 Placa de comunicación (CB)

Placa de comunicación (CB)			Referencia
RS485	CB 1241 RS485	RS485	6ES7 241-1CH30-1XB0

Tabla C- 6 Procesador de comunicaciones (CP)

Procesador de comunicaciones (CP)		Referencia
CP 1242-7	GPRS	6GK7 242-7KX30-0XE0

Tabla C- 7 TeleService

TS Adapter	Referencia
TS Adapter IE Basic	6ES7 972-0EB00-0XA0
Módulo TS GSM	6GK7 972-0MG00-0XA0
Módulo TS RS232	6ES7 792-0MS00-0XA0
Módulo TS Módem	6ES7 972-0MM00-0XA0
Módulo TS RDSI	6ES7 972-0MD00-0XA0

Tabla C- 8 Accesorios

Accesorio			Referencia
Antena	ANT794-4MR	Antena GSM/GPRS	6NH9 860-1AA00
	ANT794-3M	Antena plana	6NH9 870-1AA00

Tabla C- 9 Conectores

Tipo de conector		Referencia
RS485	Salida del cable de 35 grados, conexión de terminal roscado	6ES7 972-0BA42-0XA0
	Salida del cable de 35 grados, conexión FastConnect	6ES7 972-0BA60-0XA0

C.4 Otros módulos

Tabla C- 10 Productos adicionales

Producto		Referencia
Módulo de alimentación	Fuente de alimentación PM 1207	6EP1 332-1SH71
Switch Ethernet	Switch Ethernet CSM 1277 - 4 puertos	6GK7 277-1AA10-0AA0

C.5 Memory Cards

Tabla C- 11 Memory Cards

SIMATIC Memory Cards	Referencia
SIMATIC MC 2 MB	6ES7 954-8LB01-0AA0
SIMATIC MC 12 MB	6ES7 954-8LE01-0AA0
SIMATIC MC 24 MB	6ES7 954-8LF01-0AA0

C.6 Dispositivos HMI Basic

Tabla C- 12 Dispositivos HMI

Paneles HMI Basic	Referencia
KTP400 Basic (Mono, PN)	6AV6 647-0AA11-3AX0
KTP600 Basic (Mono, PN)	6AV6 647-0AB11-3AX0
KTP600 Basic (Color, PN)	6AV6 647-0AD11-3AX0
KTP1000 Basic (Color, PN)	6AV6 647-0AF11-3AX0
TP1500 Basic (Color, PN)	6AV6 647-0AG11-3AX0

C.7 Repuestos y hardware adicional

Tabla C- 13 Cables de ampliación, simuladores y bloques de conectores

Producto		Referencia	
Cable para módulos de ampliación	Cable para módulos de ampliación, 2 m	6ES7 290-6AA30-0XA0	
Simulador E/S	Simulador (1214C/1211C - 8 entradas)	6ES7 274-1XF30-0XA0	
	Simulador (1214C - 14 entradas)	6ES7 274-1XH30-0XA0	
Kit de tapas de repuesto	CPU 1211C/1212C	6ES7 291-1AA30-0XA0	
	CPU 1214C	6ES7 291-1AB30-0XA0	
	CPU 1215C	6ES7 291-1AC30-0XA0	
	Módulo de señales, 45 mm	6ES7 291-1BA30-0XA0	
	Módulo de señales, 70 mm	6ES7 291-1BB30-0XA0	
	Módulo de comunicación	6ES7 291-1CC30-0XA0	
Bloque de conectores	Estaño	7 terminales, 4/pk	6ES7 292-1AG30-0XA0
		8 terminales, 4/pk	6ES7 292-1AH30-0XA0
		11 terminales, 4/pk	6ES7 292-1AL30-0XA0
		12 terminales, 4/pk	6ES7 292-1AM30-0XA0
		14 terminales, 4/pk	6ES7 292-1AP30-0XA0
		20 terminales, 4/pk	6ES7 292-1AV30-0XA0
	Oro	3 terminales, 4/pk (para CPU analógica)	6ES7 292-1BC30-0XA0
		6 terminales, 4/pk (para Signal Board)	6ES7 292-1BF30-0XA0
		6 terminales, 4/pk (para CPU analógica)	6ES7 292-1BF30-0XB0
		7 terminales, 4/pk (para módulo de señales analógicas)	6ES7 292-1BG30-0XA0
		11 terminales, 4/pk (para módulo de señales analógicas)	6ES7 292-1BL30-0XA0

C.8 Software de programación

Tabla C- 14 Software de programación

Software SIMATIC		Referencia
Software de programación	STEP 7 Basic V11	6ES7 822-0AA01-0YA0
	STEP 7 Professional V11	6ES7 822-1AA01-0YA5
Software de visualización	WinCC Basic V11	6AV2100-0AA01-0AA0
	WinCC Comfort V11	6AV2101-0AA01-0AA5
	WinCC Advanced V11	6AV2102-0AA01-0AA5
	WinCC Professional 512 PowerTags V11	6AV2103-0DA01-0AA5
	WinCC Professional 4096 PowerTags V11	6AV2103-0HA01-0AA5
	WinCC Professional max. PowerTags V11	6AV2103-0XA01-0AA5

C.9 Documentación

Tabla C- 15 Documentación del S7-1200

Documentación impresa	Idioma	Referencia
Manual de sistema del controlador programable S7-1200	Alemán	6ES7 298-8FA30-8AH0
	Inglés	6ES7 298-8FA30-8BH0
	Francés	6ES7 298-8FA30-8CH0
	Español	6ES7 298-8FA30-8DH0
	Italiano	6ES7 298-8FA30-8EH0
	Chino	6ES7 298-8FA30-8KH0
S7-1200 Easy Book	Alemán	6ES7 298-8FA30-8AQ0
	Inglés	6ES7 298-8FA30-8BQ0
	Francés	6ES7 298-8FA30-8CQ0
	Español	6ES7 298-8FA30-8DQ0
	Italiano	6ES7 298-8FA30-8EQ0
	Chino	6ES7 298-8FA30-8KQ0

Índice alfabético

A

ABS (Calcular valor absoluto), 209
AC
 Cargas inductivas, 65
 Directrices de aislamiento, 62
 Puesta a tierra, 63
 Reglas de cableado, 61, 63
Acceder
 páginas web definidas por el usuario, 563
 registros del PC, 539
ACOS (arcocoseno o coseno inverso), 212
Activar salida, 182
Actualización de firmware, 119
Actualizar páginas web definidas por el usuario, 545
Acumulador de tiempo (TONR), 187
 Operación, 191
ADD (sumar), 206
Agregar dispositivo
 CPU, 124
Agregar entradas o salidas a instrucciones KOP o FUP, 35
Agregar nuevo dispositivo
 CPU sin especificar, 125
 Detectar hardware existente, 125
Alarma, 309
 acceso a la periferia, 490
Alarma de retardo, 307
Alarmas
 ATTACH y DETACH, 301
 CAN_DINT (Anular alarma de retardo), 307
 Latencia de alarmas, 76
 SRT_DINT (Iniciar alarma de retardo), 307
 Vista general, 74
Alarmas de acceso a la periferia, 490
Alias en páginas web definidas por el usuario, 552
Ambiental
 condiciones de manejo, 740
 Condiciones de transporte y almacenamiento, 740
 entornos industriales, 739
Ampliar las prestaciones del S7-1200, 22
Aprobación C-Tick, 739
Aprobación marina, 739
Áreas de memoria
 Acceso inmediato, 89
 Direccionamiento de valores booleanos o de bit, 90

 Imagen de proceso, 89
Arquitectura de sondeo, 625
Arquitectura de sondeo del esclavo, 625
Arquitectura de sondeo del maestro, 625
Arranque en caliente, 70
Arranque tras POWER ON, 70
 Procesamiento del arranque, 72
AS-i
 Agregar esclavo AS-i, 500
 Agregar módulo maestro AS-i CM1243-2, 500
 Asignación de sistema, 503
 Asignación del sistema de direcciones de esclavos, 503
 Conexión de red, 501
 Dirección AS-i, 502
 Instrucciones de E/S descentralizadas, 285
 Propiedades de la dirección AS-i, 503
 RDREC, 286
 Transferir valores analógicos, 505
 Transferir valores digitales, 505
 WRREC, 286
AS-I
 Configuración de esclavos con STEP 7, 505
 Configuración de esclavos sin STEP 7, 503
Asignar tipos de enumeración, páginas web definidas por el usuario, 554
ASIN (arcoseno o seno inverso), 212
Asistente de importación de certificados, 586
ATH (ASCII a hexadecimal), 276
ATTACH, 301
AWP_Enum_Def, 553
AWP_Enum_Ref, 554
AWP_Import_Fragment, 556
AWP_In_Variable, 547, 551
AWP_Out_Variable, 549
AWP_Start_Fragment, 555

B

Barra de herramientas Favoritos, 32
Bits de parada, 612
Bloque
 comprobación de coherencia, 176
Bloque de administración de datos (DHB), 156
Bloque de datos
 Bloque de datos de instancia, 89
 Bloque de datos global, 89, 156
 Bloque de organización (OB), 153

- CONF_DATA, 473
- FB único con varios DBs de instancia, 156
- Números válidos de DB, 67
- Obtener valores, 721
- Restablecer los valores iniciales, 721
- Vista general, 67, 156
- Bloque de datos de instancia, 89
- Bloque de datos global, 89, 156
- Bloque de función (FB)
 - Bloque de datos de instancia, 155
 - FB único con varios DBs de instancia, 156
 - Llamar bloques lógicos en el programa de usuario, 152
 - Números válidos de FB, 67
 - Parámetros de salida, 155
 - Programas lineales y estructurados, 150
 - Protección de know-how, 170
 - Valor inicial, 155
 - Vista general, 67, 155
- Bloque de organización
 - Clases de prioridad, 74
 - Configurar el funcionamiento, 154
 - Crear, 154
 - De ciclo, varios, 154
 - Función, 74
 - Llamada, 74
 - Llamar bloques lógicos en el programa de usuario, 152
 - Procesamiento del arranque, 72
 - Procesar, 153
 - Programación lineal y estructurada, 150
 - Protección de know-how, 170
 - Vista general, 67
- Bloque de transferencia (bloque T), 484
- Bloque lógico
 - Alarmas, 21, 745, 756, 766, 776
 - Bloque de datos (DB) de instancia, 155
 - Bloque de organización (OB), 153
 - Bloques de organización (OB), 21, 745, 756, 766, 776
 - Contadores (requisitos de memoria y cantidad), 21, 746, 756, 766, 776
 - DB (bloque de datos), 67, 156
 - Enlazar a una CPU o Memory Card, 171
 - FB (bloque de función), 67, 155
 - FC (función), 67, 154
 - Llamadas de bloque, 67
 - Llamar bloques lógicos en el programa de usuario, 152
 - Número de bloques lógicos, 21, 745, 755, 765, 776
 - Número de OB, 21, 745, 756, 766, 776
 - Números válidos de FC, FB y DB, 67
 - Observar, 21, 745, 756, 766, 776
 - Profundidad de anidamiento, 21, 745, 756, 765, 776
 - Programas lineales y estructurados, 150
 - Protección anticopia, 171
 - Protección de know-how, 170
 - Tamaño del programa de usuario, 21, 745, 755, 765, 776
 - Temporizadores (requisitos de memoria y cantidad), 21, 746, 756, 766, 776
 - Tipos de bloques lógicos, 67
 - Valor inicial de un FB, 155
- Bloques
 - Alarmas, 21, 76, 745, 756, 766, 776
 - Bloque de datos (DB), 67
 - Bloque de datos (DB) de instancia, 155
 - Bloque de función (FB), 67, 155
 - Bloques de organización (OB), 21, 67, 74, 76, 745, 756, 766, 776
 - Bloques de organización (OBs), 67
 - Cargar, 172
 - Contadores (requisitos de memoria y cantidad), 21, 746, 756, 766, 776
 - copiar bloques desde una CPU online, 172
 - DB de instancia única o multiinstancia, 155
 - Eventos, 76
 - Función (FC), 67, 154
 - Llamadas de bloque, 67
 - Llamar un FB o una FC con SCL, 161
 - Número de bloques lógicos, 21, 745, 755, 765, 776
 - Número de OB, 21, 76, 745, 756, 766, 776
 - Números válidos de FC, FB y DB, 67
 - OB de arranque, 76
 - Observar, 21, 745, 756, 766, 776
 - Profundidad de anidamiento, 21, 67, 745, 756, 765, 776
 - Programas lineales y estructurados, 150
 - Protección por contraseña, 170
 - Tamaño del programa de usuario, 21, 67, 745, 755, 765, 776
 - Temporizadores (requisitos de memoria y cantidad), 21, 746, 756, 766, 776
 - Tipos de bloques lógicos, 67
 - Valor inicial de un FB, 155
- Bloques de datos para páginas web definidas por el usuario
 - importar fragmentos, 556
- Bloques lógicos, 152
- Botones RUN/STOP, 37
- Búfer de diagnóstico, 87, 717
- Byte de marcas de sistema, 86

C

Cable de ampliación
 Desmontaje, 55
 Instalación, 55
 CALCULAR, 33, 205
 Escalar valores analógicos, 34
 Calendario, 257
 Cambiar idiomas, páginas web definidas por el usuario, 576
 Cambiar los ajustes de STEP 7, 36
 CAN_DINT (Anular alarma de retardo), 307
 Carácter de fin del mensaje, 621
 Carácter de inicio del mensaje, 617
 Caracteres especiales
 Páginas web definidas por el usuario, 557
 Cargar
 Actualización de firmware, 119
 Certificado de seguridad Siemens a PC, 530, 586
 copiar bloques desde una CPU online, 172
 DBs de páginas web definidas por el usuario, 563
 Programa de usuario, 172
 Proyecto, 172
 Visualizar las direcciones MAC e IP, 145
 Cargas de lámpara, 64
 Cargas inductivas, 65
 Carpetas, idiomas para páginas web definidas por el usuario, 576
 CEIL (redondear número en coma flotante a entero superior), 226
 Certificado de seguridad Siemens, páginas web, 530, 586

Ch

Char (tipo de datos Character), 100

C

Ciclo
 Forzar permanentemente, 725
 Operación de forzado permanente, 726
 Clase de prioridad, 74
 Clase de protección, 741
 Clases de error PtP, 594
 CM 1241
 Datos técnicos de RS232, 870
 Datos técnicos de RS422/RS485, 870
 Códigos de error
 Errores comunes de las instrucciones avanzadas, 351
 RALRM, 292

RDREC, 292
 WRREC, 292
 Colas de espera, 76
 Columnas y encabezados en Task Cards, 36
 Comandos AWP, 545
 combinar definiciones, 557
 definir un tipo de enumeración, 553
 escribir variables, 547
 escribir variables especiales, 551
 generar fragmentos, 555
 importar fragmentos, 556
 leer variables especiales, 549
 referenciar un tipo de enumeración, 554
 utilizar un alias, 552
 Comparación, 202
 Comparar bloques lógicos, 718
 Comparar y sincronizar CPU online/offline, 718
 Compatibilidad electromagnética (CEM), 740
 Compensación de unión fría
 Termopar, 820, 849
 Complemento a uno (INV), 249
 Comprobación de coherencia, 176
 Comprobar el programa, 175
 Comunicación
 Activa/pasiva, 132, 134, 514
 Arquitectura de sondeo, 625
 Carga de comunicación, 83
 Conexión de hardware, 479
 Conexión de red, 131
 Configuración, 132, 134, 514
 Control de flujo, 613
 Dirección AS-i, 502
 Dirección IP, 141
 Dirección MAC, 141
 Dirección PROFIBUS, 498
 estadística, 535
 ID de conexión, 445
 Número de conexiones (PROFINET), 442
 Parámetros, 134, 535
 Parámetros de transmisión y recepción, 614
 PROFINET y PROFIBUS, 441
 Propiedad de sincronización horaria (PROFINET), 147
 Red, 478
 TCON_Param, 134
 Tiempo de ciclo, 83
 Comunicación activa/pasiva
 Configurar los interlocutores, 132, 514
 ID de conexión, 445
 Parámetros, 134
 Comunicación de CPU, 444
 Comunicación de red, 478

- Comunicación pasiva/activa
 - Configurar los interlocutores, 132, 514
 - ID de conexión, 445
 - Parámetros, 134
- Comunicación PtP, 591
 - configuración del programa de ejemplo, 627
 - configurar parámetros, 614
 - configurar puertos, 611
 - emulador de terminal para el programa de ejemplo, 633
 - programa de ejemplo, 626
 - programa de ejemplo, en proceso, 634
 - programa de ejemplo, programación de STEP 7, 632
 - Programación, 624
- Comunicación punto a punto, 591
- Comunicación S7
 - Configurar la conexión, 133
- Comunicación serie, 591
- Comunicación TCP/IP, 448
- Comunicación Teleservice
 - TM_MAIL, 701
- CONCAT (agrupar), 279
- Condiciones de fin, 619
- Condiciones de inicio, 616
- Conector
 - Instalación y desmontaje, 54
- Conector de bus, 25
- Conector del bloque de terminales, 54
- Conexión activa/pasiva, 443
- Conexión de red
 - Configuración, 131
 - Varias CPUs, 482, 483, 487, 497, 501
- Conexión local/interlocutor, 443
- Conexiones
 - Conexión S7, 513
 - Configuración, 134
 - ID de conexión, 445
 - Interlocutores, 132, 514
 - Número de conexiones (PROFINET), 442
 - Protocolos Ethernet, 513
 - Tipos de comunicación, 441
 - Tipos, conexiones multinodo, 513
- Conexiones HTTP, servidor web, 584
- Conexiones multinodo
 - Protocolos Ethernet, 513
 - tipos de conexión, 513
- Conexiones servidor web máximas, 584
- Conexiones, servidor web, 584
- Configuración, 36
 - Agregar módulos, 126
 - AS-i, 502
 - Carga de comunicación, 83
 - Cargar, 172
 - Comunicación entre PLCs, 482
 - Conexión de red, 131
 - Determinar, 125
 - Dirección AS-i, 502
 - Dirección IP, 141
 - Dirección MAC, 141
 - Dirección PROFIBUS, 498
 - HSC (contador rápido), 362
 - Interfaces de comunicación, 611
 - Módulos, 130
 - páginas web definidas por el usuario, 560
 - páginas web definidas por el usuario, (idiomas múltiples), 579
 - Parámetros de arranque, 113
 - Parámetros de la CPU, 127
 - PROFIBUS, 497
 - PROFINET, 141
 - Propiedad de sincronización horaria (PROFINET), 147
 - Puerto AS-i, 501
 - Puerto Ethernet, 141
 - Puerto PROFIBUS, 497
 - Puertos, 611
 - Recepción de mensajes, 616
 - Tiempo de ciclo, 81
- Configuración de dispositivos, 123, 479
 - Agregar dispositivo, 124
 - Agregar módulos, 126
 - AS-i, 502
 - Cargar, 172
 - Conexión de red, 131
 - Configurar la CPU, 127
 - Configurar los módulos, 130
 - Determinar, 125
 - Módulos no enchufados, 40
 - PROFIBUS, 497
 - PROFINET, 141
 - Propiedad de sincronización horaria (PROFINET), 147
 - Puerto AS-i, 502
 - Puerto Ethernet, 141
 - Puerto PROFIBUS, 497
- Configuración de hardware, 123
 - Agregar dispositivo, 124
 - Agregar módulos, 126
 - AS-i, 502
 - Conexión de red, 131
 - Configurar la CPU, 127
 - PROFINET, 141
 - Puerto AS-i, 502

- Puerto Ethernet, 141
- Configuración de la instrucción TRCV_C, 486
- Configuración de la instrucción TSEND_C, 485
- Configuración de la recepción de mensajes, 616
 - programa de ejemplo PtP, 628
- Configuración de la transferencia de mensajes, 615
 - programa de ejemplo PtP, 627
- Configuración de los parámetros de recepción, 486
- Configuración de mensajes
 - Instrucciones, 624
 - Recibir, 616
 - Transmitir, 615
- Configuración de parámetros
 - LENGH y BUFFER para SEND_PTP, 605
 - Recibir, 486
 - Transmitir, 485
- Configuración de parámetros de transmisión, 132, 485, 514
- Configuración del envío de mensajes, 615
- Configuración del mensaje transmitido, 615
- Configuración del puerto, 611
 - Errores, 596
 - Instrucciones, 624
 - programa de ejemplo PtP, 627
- Configuración hardware
 - Cargar, 172
 - Configurar los módulos, 130
 - Determinar, 125
 - PROFIBUS, 497
 - Puerto PROFIBUS, 497
- Consumo de corriente
 - calcular la corriente necesaria, 877
 - Corriente disponible, 875
 - forma de cálculo, 878
- Contactos de conexión
 - intensidad máxima admisible, 866
- Contador de horas de funcionamiento, 261
- Contador rápido
 - Configuración, 362
 - Funcionamiento, 355
 - HSC, 353
- Contador rápido (HSC)
 - No se puede forzar permanentemente, 727
- Contadores
 - Cantidad, 21, 746, 756, 766, 776
 - Configuración del HSC, 362
 - HSC (contador rápido), 353
 - Operación HSC, 355
 - Tamaño, 22, 746, 756, 766, 776
- Contraseña olvidada, 122
- Control de bloques de datos, 348
- Control de flujo, 612, 613
 - Configuración, 612
 - Control de flujo por hardware, 613
 - Control de flujo por software, 614
 - Control del programa (SCL), 230
 - CASE, 232
 - CONTINUE, 236
 - EXIT, 236
 - FOR, 233
 - GO TO, 237
 - IF-THEN, 231
 - REPEAT, 235
 - RETURN, 238
 - WHILE, 234
 - Control manual de DB de fragmentos, 580
 - CONV (convertir), 221
 - Convenciones para comillas, servidor web, 557
 - Conversión (instrucciones SCL), 222
 - Cookie, siemens_automation_language, 577
 - Copiar área (MOVE_BLK), 214
 - Copiar área sin interrupciones (UMOVE_BLK), 214
 - Copiar bloques desde una CPU online, 172
 - Corriente disponible, 42
 - Ejemplo, 877
 - forma de cálculo, 878
 - Vista general, 875
 - COS (coseno), 212
 - CPU
 - Agregar dispositivo, 124
 - Agregar módulos, 126
 - AS-i, 501
 - Asignar una dirección IP a una CPU online, 139
 - Botones RUN/STOP, 38
 - Cable de ampliación, 55
 - Carga de comunicación, 83
 - Cargar, 172
 - Cargar en dispositivo, 145
 - Cargas de lámparas, 64
 - Cargas inductivas, 65
 - comparar y sincronizar bloques, 718
 - Conectar online, 711
 - Conector de bloque de terminales, 54
 - Conexión de red, 131
 - Configuración de dispositivos, 123
 - Configuración del HSC, 362
 - Configuración del tiempo de ciclo, 83
 - Configurar canales de impulsos, 327
 - Configurar la comunicación con HMI, 480
 - Configurar los módulos, 130
 - Configurar parámetros, 127
 - Configurar varias, 482
 - Consumo de corriente, 875
 - Contador de horas de funcionamiento, 261

Contraseña olvidada, 122
copiar bloques desde una CPU online, 172
Corriente disponible, 42
CPU sin especificar, 125
Crear una tarjeta de programa, 117
Crear una tarjeta de transferencia, 114
Datos técnicos 1211C, 744
Datos técnicos 1212C, 754
Datos técnicos 1214C, 764
Datos técnicos 1215C, 774
desbloquear las salidas en modo STOP, 725
Diagramas de cableado 1211C, 753
Diagramas de cableado 1212C, 763
Diagramas de cableado 1214C, 773
Diagramas de cableado 1215C, 785
Dirección AS-i, 502
Dirección IP, 141
Dirección MAC, 141, 145
Dirección PROFIBUS, 498
Directrices de aislamiento, 62
Dispositivos HMI, 27
Ejecución del programa, 67
Estados operativos, 70
Forzado permanente, 726
Forzar permanentemente, 725
Indicadores LED, 709
Inserción de la Memory Card, 112
Instalación, 46, 47
Memory Card, 111, 872
Modo STOP y RUN, 715
Niveles de protección, 168
Número de conexiones de comunicación, 442
observar, 720
Obtener valores de un DB, 721
online, 720
Online, 714
Panel de control (CPU online), 715
Panel de mando, 38
Parámetros de arranque, 113
Placa de comunicación (CB), 25
Procesamiento del arranque, 72
Procesar los OBs, 153
PROFIBUS, 497
PROFINET, 141
PROFINET IO, 486
Propiedad de sincronización horaria, 147
Protección de acceso, 168
Protección de know-how, 170
Protección por contraseña, 168
Puerto AS-i, 501
Puerto Ethernet, 141
Puerto PROFIBUS, 497

Puesta a tierra, 63
RD_LOC_T (leer hora local), 259
RD_SYS_T (leer hora del sistema), 259
Recuperación de una contraseña perdida, 122
Reglas de cableado, 61, 63
Representación de entradas analógicas (tensión), 814, 845
Restablecer los ajustes de fábrica, 714
Restablecer los valores iniciales de un DB, 721
Salidas de impulsos, 325
Signal Board (SB), 25
Tabla de comparación, 20
tabla de observación, 722
Tarjeta de programa, 111, 117
Tarjeta de transferencia, 111, 114
Tarjeta de transferencia vacía, 122
Tiempos de respuesta a un escalón, 750, 760, 770, 780
Tipos de comunicación, 441
Vigilancia del tiempo de ciclo, 81
Vista general, 19
Visualizar las direcciones MAC e IP, 145
WR_SYS_T (escribir hora del sistema), 259
Zona de disipación, 42, 45
CPU sin especificar, 125
Creación de páginas web definidas por el usuario, 544
Crear DB de páginas web definidas por el usuario, 561
Crear una conexión de red, 131
CTD (decrementar contador), 196
CTRL_PWM, 323
CTS, 613
CTU (incrementar contador), 196
CTUD (incrementar y decrementar contador), 196
Customer support, 3

D

Data log
Programa de ejemplo, 344
Date
DTL (tipo de datos Data and Time Long), 99
T_ADD (sumar tiempos), 258
T_COMBINE (combinar tiempos), 259
T_CONV (convertir y extraer tiempos), 257
T_DIFF (diferencia de tiempo), 258
T_SUB (restar tiempos), 258
Tipo de datos Date, 99
Datos de diagnóstico
Lectura con GET_DIAG, 316
Datos técnicos, 737
Aprobación C-Tick, 739
Aprobación marina, 739

- CB 1241 RS485, 867
- CM 1241 RS232, 870
- Compatibilidad electromagnética (CEM), 740
- Condiciones ambientales, 740
- CPU 1211C, 744
- CPU 1212C, 754
- CPU 1214C, 764
- CPU 1215C, 774
- datos técnicos generales, 737
- Diagrama de cableado de la SB 1231 AI 1 x 16 bits de termopar, 850
- Diagrama de cableado de la SB 1231 AI 1 x 16 bits RTD, 853
- Diagrama de cableado de la SB 1231 AI x 12 bits, 841
- Diagrama de cableado del SM 1221, 787
- Diagrama de cableado del SM 1222, 792
- Diagrama de cableado del SM 1223, 797, 800
- Diagrama de cableado del SM 1231 AI 4 x 16 bits TC, 818
- Diagrama de cableado del SM 1231 AI 8 x 16 bits TC, 818
- Diagrama de cableado del SM 1231 RTD 4 x 16 bits, 824
- Diagrama de cableado del SM 1231 RTD 8 x 16 bits, 824
- Diagramas de cableado del SM 1231 con entrada analógica, 807
- Diagramas de cableado del SM 1232 con salida analógica, 809
- Diagramas de cableado del SM 1234 con entrada/salida analógica, 812
- entornos industriales, 739
- Homologación ATEX, 738
- Homologación CE, 737
- Homologación cULus, 738
- Homologación FM, 738
- Memory Cards, 872
- Módulo de señales SM 1221, 786
- Módulo de señales SM 1222, 790, 791
- Módulo de señales SM 1223, 797, 800
- Protección, 741
- Representación de entradas analógicas (intensidad), 814, 845
- Representación de entradas analógicas (tensión), 814, 845
- Representación de salidas analógicas (intensidad), 815, 846
- Representación de salidas analógicas (tensión), 815, 846
- SB 1221 DI 4, 200 kHz, 828
- SB 1222 DQ 4, 200 kHz, 831
- SB 1223 DI 2 / DQ 2, 837
- SB 1223 DI 2 x / DQ 2, 200 kHz, 834
- SB 1231 AI 1 x 12 bits, 840
- SB 1231 AI 1 x 16 bits de termopar, 847
- SB 1231 AI 1 x 16 bits RTD, 851
- SB 1232 AQ 1x12 bits, 842
- Simuladores de entradas, 873
- SM 1222 DQ8 RLY inversor, 790
- SM 1231 AI 4 x 13 bits, 803
- SM 1231 AI 4 x 16 bits módulo de señales TC, 816
- SM 1231 AI 4 x RTD x 16 bits módulo de señales, 822
- SM 1231 AI 8 x 16 bits módulo de señales TC, 816
- SM 1231 AI 8 x RTD x 16 bits módulo de señales, 822
- SM 1232 AQ 2 x 14 bits, 807
- SM 1232 AQ 4 x 14 bits, 807
- SM 1234 AI 4 x 13 bits / AQ 2 x 14 bits, 810
- Tensiones nominales, 742
- Tiempos de respuesta a un escalón (CPU), 750, 760, 770, 780
- Tiempos de respuesta a un escalón (SB), 844
- Tiempos de respuesta a un escalón (SM), 813
- Vida útil de los relés, 743
- Datos técnicos de CB 1241 RS485, 867
- Datos técnicos de la Signal Board (SB) digital
 - SB 1221 DI 4, 200 kHz, 828
 - SB 1222 DQ 4, 200 kHz, 831
 - SB 1223 DI 2 / DQ 2, 837
 - SB 1223 DI 2 / DQ 2, 200 kHz, 834
- Datos técnicos generales, 737
- DB (bloque de datos), 67, 156
 - Números válidos de DB, 67
 - Obtener valores, 721
 - Restablecer los valores iniciales, 721
- DB de control para páginas web definidas por el usuario
 - comandos globales, 580
 - comandos y estados de petición, 580
 - parámetro para instrucción WWW, 561
- DBs de fragmentos (páginas web definidas por el usuario)
 - generar, 561
- DC
 - Cargas inductivas, 65
 - Directrices de aislamiento, 62
 - Puesta a tierra, 63
 - Reglas de cableado, 61, 63
- DEC (Decrementar), 209
- DECO (descodificar), 249
- Definiciones múltiples de variables AWP, 557

- Definir tipos de enumeración, páginas web definidas por el usuario, 553
 - DELETE (Borrar caracteres de una cadena), 281
 - Depuración
 - Cargar en estado operativo RUN, 728, 734
 - Desactivar salida, 182
 - DETACH, 301
 - Determinar para cargar una CPU online, 125
 - DeviceStates, 313
 - Diagnóstico
 - búfer de diagnóstico, 717
 - carga de la memoria, 716
 - DeviceStates, 313
 - estado, 490
 - GET_DIAG, 316
 - Indicador de estado, 86
 - Indicadores LED, 709
 - Instrucción LED, 312
 - ModuleStates, 314
 - OB de alarma, 490
 - tabla de observación, 722
 - Tiempo de ciclo, 716
 - Diagramas de cableado
 - CPU 1211C, 753
 - CPU 1212C, 763
 - CPU 1214C, 773
 - CPU 1215C, 785
 - Módulo de señales SM 1221, 787
 - Módulo de señales SM 1222, 792
 - Módulo de señales SM 1223, 797, 800
 - SB 1221 DI 4, 200 kHz, 830
 - SB 1222 DQ 4, 200 kHz, 833
 - SB 1223 DI 2 / DQ 2, 839
 - SB 1223 DI 2 / DQ 2, 200 kHz, 836
 - SB 1231 AI 1 x 16 bits de termopar, 850
 - SB 1231 AI 1 x 16 bits RTD, 853
 - SB 1231 AI x 12 bits, 841
 - SB 1232 AQ 1 x 12 bits, 843
 - SM 1231 AI 4 x 16 bits TC, 818
 - SM 1231 AI 8 x 16 bits TC, 818
 - SM 1231 con entrada analógica, 807
 - SM 1231 RTD 4 x 16 bits, 824
 - SM 1231 RTD 8 x 16 bits, 824
 - SM 1232 con salida analógica, 809
 - SM 1234 con entrada/salida analógica, 812
 - Dirección AS-i, 502, 503
 - Configurar, 502
 - Dirección IP, 141, 142
 - Asignar, 137, 144
 - Asignar online, 139
 - Configuración de dispositivos, 127
 - Configurar, 141
 - configurar CPU online, 714
 - Dirección MAC, 141
 - Dirección IP del router, 142
 - Dirección MAC, 141, 145
 - Dirección PROFIBUS, 498
 - Configurar, 498
 - Direccionamiento
 - Áreas de memoria, 89
 - Entradas (I) o salidas (Q) individuales, 90
 - Imagen de proceso, 89
 - Valores booleanos o de bit, 90
 - Directorios, idiomas para páginas web definidas por el usuario, 576
 - Directrices
 - Aislamiento, 62
 - Cargas de lámparas, 64
 - Cargas inductivas, 65
 - Instalación, 41
 - Instalación de la CPU, 47
 - Procedimientos de instalación, 46
 - Puesta a tierra, 63
 - Reglas de cableado, 61, 63
 - Directrices de aislamiento galvánico, 62
 - Directrices de cableado, 63
 - Espacio libre para ventilación y refrigeración, 42
 - Puesta a tierra, 63
 - Requisitos previos, 61
 - DIS_AIRT (Retardar tratamiento de eventos de alarma y errores asíncronos de mayor prioridad), 309
 - Diseñar un sistema PLC, 149, 150
 - Disparar valores en la tabla de observación, 724
 - Dispositivo
 - PROFINET IO, 486
 - Dispositivos HMI
 - Conexión de red, 131
 - Configurar la comunicación PROFINET, 480
 - vista general, 27
 - DIV (dividir), 206
 - Documentación, 4
 - DPNRM_DG, 298
 - DPRD_DAT, 296
 - DPWR_DAT, 296
 - Drag & Drop entre editores, 37
- ## E
- E/S
 - Cargas inductivas, 65
 - Direccionamiento, 94
 - errores de acceso, PROFINET, 490
 - Forzar permanentemente, 725
 - Indicadores de estado analógicos, 710

- Indicadores de estado digitales, 710
- observar con una tabla de observación, 722
- observar estado en KOP, 720
- Operación de forzado permanente, 726
- Representación de entradas analógicas (intensidad), 814, 845
- Representación de entradas analógicas (tensión), 814, 845
- Representación de salidas analógicas (intensidad), 815, 846
- Representación de salidas analógicas (tensión), 815, 846
- Tiempos de respuesta a un escalón (CPU), 750, 760, 770, 780
- Tiempos de respuesta a un escalón (SB), 844
- tiempos de respuesta a un escalón del módulo de señales (SM), 813
- E/S analógicas
 - Configuración, 130
 - Conversión a unidades de ingeniería, 34, 94, 229
 - Indicadores de estado, 710
 - Representación de entradas (intensidad), 814, 845
 - Representación de entradas (tensión), 814, 845
 - Representación de salidas (intensidad), 815, 846
 - Representación de salidas (tensión), 815, 846
 - Tiempos de respuesta a un escalón (CPU), 750, 760, 770, 780
 - Tiempos de respuesta a un escalón (SB), 844
 - Tiempos de respuesta a un escalón (SM), 813
- E/S digitales
 - Configuración, 130
 - Indicadores de estado, 710
- Editor de programas
 - estado, 720
 - observar, 720
 - Obtener valores de un DB, 721
 - Restablecer los valores iniciales de un DB, 721
- Ejecución de eventos, 76
- Ejecución del programa, 67
- Ejemplo
 - Comunicación PtP, 626
 - Comunicación PtP, configuración, 627
 - Comunicación PtP, emulador de terminal, 633
 - Comunicación PtP, en proceso, 634
 - Comunicación PtP, programación de STEP 7, 632
 - Esclavo Modbus, 689
 - páginas web definidas por el usuario, 565
- Emulador de terminal para el programa de ejemplo PtP, 633
- EN y ENO (flujo de corriente), 167
- EN_AIRT (Habilitar tratamiento de eventos de alarma y errores asíncronos de mayor prioridad), 309
- ENCO (codificar), 249
- Enfriamiento, 42
- Enlazar a una CPU o Memory Card, 171
- Entradas y salidas
 - observar, 720
- Errores
 - Errores de diagnóstico, 80
 - Errores de tiempo, 79
- Errores de configuración de la recepción, 602
- Errores de configuración de la transmisión, 597
- Errores de procesamiento de señales, 609, 610
- Errores diversos de parámetros PtP, 593
- Errores en tiempo de ejecución de la transmisión, 604
- Escalar valores analógicos, 34, 229
- Espacio
 - Ventilación y refrigeración, 42
- Estado
 - Indicadores LED, 709
 - indicadores LED (interfaz de comunicación), 589
 - Instrucción LED, 312
- Estado operativo, 37, 38
 - cambiar STOP/RUN, 715
 - Estados operativos de la CPU, 70
- Estado operativo ARRANQUE
 - Operación de forzado permanente, 726
- Estado operativo RUN, 70, 73, 715
 - Botones de la barra de herramientas, 37
 - Operación de forzado permanente, 726
 - Panel de mando, 38
- Estado operativo STOP, 70, 715
 - Botones de la barra de herramientas, 37
 - desbloquear las salidas en modo STOP, 725
 - Operación de forzado permanente, 726
 - Panel de mando, 38
- Estructura de llamadas, 176
- Estructura del programa, 152
 - Llamar bloques lógicos, 152
- Ethernet
 - Conexión de red, 131
 - Dirección IP, 141
 - Dirección MAC, 141
 - DPNRM_DG, 298
 - DPRD_DAT, 296
 - DPWR_DAT, 296
 - GET, 509
 - ID de conexión, 445
 - Modo ad hoc, 449
 - Número de conexiones de comunicación, 442
 - PUT, 509
 - RALRM, 289
 - RDREC, 286
 - T_CONFIG, 470

TCON, 457
TDISCON, 457
Tipos de comunicación, 441
TRCV, 457
TRCV_C, 451
TSEND, 457
TSEND_C, 451
TURCV, 465
TUSEND, 465
Vista general, 448
WRREC, 286
Etiqueta, 238
EXP (exponente natural), 212
EXPT (exponente general), 212

F

FAQs, 4
FB (bloque de función)
 Números válidos de FB, 67
 Vista general, 67
FC (función), 67, 154
 Números válidos de FC, 67
Fecha
 SET_TIMEZONE (ajustar zona horaria), 263
FieldRead, 217
FieldWrite, 217
FILL_BLK, 218
Fin del mensaje, 619
FIND (Buscar caracteres en una cadena), 284
Flanco ascendente, 185
Flanco descendente, 185
FLOOR, 226
Forzado permanente, 725, 726
 Ciclo, 726
 Entradas de periferia, 725, 726
 Entradas y salidas, 726
 Memoria I, 725, 726
 Memory Card, 111
 tabla de observación, 722
FRAC (fracción), 212
Fragmentos (páginas web definidas por el usuario)
 crear a partir de comando AWP, 555
 importar con comando AWP, 556
Frecuencia, bits de reloj, 87
Función (FC)
 Llamar bloques lógicos en el programa de usuario, 152
 Números válidos de FC, 67
 Programas lineales y estructurados, 150
 Protección de know-how, 170
 Vista general, 67, 154

FUP (Diagrama de funciones), 160

G

Generar DB de páginas web definidas por el usuario, 561
GET, 509
 Configurar la conexión, 133
GET_DIAG, 316
GetError, 244
GetErrorID, 246

H

Herramientas online y diagnóstico
 Cargar en estado operativo RUN, 728
Homologación ATEX, 738
Homologación CE, 737
Homologación cULus, 738
Homologación FM, 738
Homologaciones
 Aprobación C-Tick, 739
 Aprobación marina, 739
 Homologación ATEX, 738
 Homologación CE, 737
 Homologación cULus, 738
 Homologación FM, 738
Hora
 configurar CPU online, 714
Hora local
 RD_LOC_T (leer hora local), 259
Hotline, 3
HSC (contador rápido)
 Configuración, 362
 Funcionamiento, 355
 Operación, 353
HTA (hexadecimal a ASCII), 277

I

Idiomas, páginas web definidas por el usuario, 576
Idle Line, 615, 617
Imagen de proceso
 estado, 720, 725
 Forzar permanentemente, 725
 observar, 720
Importar certificado de seguridad Siemens, 586
Impulso (TP), 187
 Operación, 191
IN_RANGE (dentro del rango), 203
INC (Incrementar), 209

- Indexación de matrices con variables, 218
- Indicador de primer ciclo, 86
- Indicadores LED
 - Estado de la CPU, 709
 - Instrucción LED, 312
 - interfaz de comunicación, 589, 709
- Índice de variable para una matriz, 218
- Información de contacto, 3
- Información del programa
 - En la estructura de llamadas, 176
- Inicializar temporizador (RT), 187
- Inicio del mensaje, 616
- Inicio y cierre de sesión
 - Páginas web estándar, 528
- INSERT (Insertar caracteres en una cadena), 282
- Insertar instrucciones
 - Arrastrar y soltar, 32
 - Drag & Drop entre editores, 37
 - Favoritos, 32
- Insertar la Memory Card en la CPU, 112
- Insertar un dispositivo
 - CPU sin especificar, 125
- Instalación
 - Cable de ampliación, 55
 - Cargas de lámparas, 64
 - Cargas inductivas, 65
 - Conector de bloque de terminales, 54
 - Corriente disponible, 42
 - CPU, 47
 - Dimensiones de montaje, 45
 - Directrices de aislamiento, 62
 - Espacio, 42
 - Módulo de comunicación (CM), 53
 - Módulo de señales (SM), 25, 51
 - Placa de comunicación (CB), 49
 - Puesta a tierra, 63
 - Refrigeración, 42
 - Reglas, 41
 - Reglas de cableado, 61, 63
 - Signal Board (SB), 49
 - Tarjeta SIM del TS Adapter, 58
 - TS Adapter en un perfil DIN, 60
 - TS Adapter en una pared, 60
 - TS Adapter y módulo TS, 57
 - Ventilación, 42
 - Vista general, 41, 46
 - Zona de disipación, 42, 45
- Instrucción Comprobar invalidez, 204
- Instrucción Comprobar validez, 204
- Instrucciones
 - ABS (Calcular valor absoluto), 209
 - ACOS (arcocoseno o coseno inverso), 212
 - Activar, 182
 - ADD (sumar), 206
 - Agregar entradas o salidas a instrucciones KOP o FUP, 35
 - ASIN (arcoseno o seno inverso), 212
 - ATAN (arcotangente o tangente inversa), 212
 - ATH (ASCII a hexadecimal), 276
 - ATTACH, 301
 - CALCULAR, 33, 205
 - Calendario, 257
 - CAN_DINT (Anular alarma de retardo), 307
 - CASE (SCL), 232
 - CEIL (redondear número en coma flotante a entero superior), 226
 - Códigos de estado USS, 646
 - Columnas y encabezados, 36, 671
 - Comparación, 202
 - CONCAT (agrupar), 279
 - Contador de horas de funcionamiento, 261
 - Contadores, 196
 - CONTINUE (SCL), 236
 - Control de bloques de datos, 348
 - Control del programa (SCL), 230
 - CONV (convertir), 221
 - Copiar área (MOVE_BLK), 214
 - Copiar área sin interrupciones (UMOVE_BLK), 214
 - COS (coseno), 212
 - CTD (decrementar contador), 196
 - CTRL_PWM, 323
 - CTU (incrementar contador), 196
 - CTUD (incrementar y decrementar contador), 196
 - DataLogClose (cerrar registro), 335
 - DataLogCreate (crear registro), 330
 - DataLogNewFile (crear un registro basándose en otro existente), 337
 - DataLogOpen (abrir registro), 333
 - DataLogWrite (escribir registro), 336
 - Date, 257
 - DEC (Decrementar), 209
 - DECO (descodificar), 249
 - DELETE (Borrar caracteres de una cadena), 281
 - Desactivar, 182
 - DETACH, 301
 - DeviceStates, 313
 - DIS_AIRT (Retardar tratamiento de eventos de alarma y errores asíncronos de mayor prioridad), 309
 - DIV (dividir), 206
 - DPNRM_DG, 298
 - DPRD_DAT, 296
 - DPWR_DAT, 296
 - Drag & Drop, 32

- Drag & Drop entre editores, 37
- E/S descentralizadas AS-i, 285
- E/S descentralizadas PROFIBUS, 285
- E/S descentralizadas PROFINET, 285
- EN_AIRT (Habilitar tratamiento de eventos de alarma y errores asíncronos de mayor prioridad), 309
- ENCO (codificar), 249
- Escalar valores analógicos, 34
- estado, 720
- Estado LED, 312
- Etiqueta, 238
- EXIT (SCL), 236
- EXP (exponente natural), 212
- EXPT (exponente general), 212
- Favoritos, 32
- FieldRead, 217
- FieldWrite, 217
- FILL_BLK, 218
- FIND (Buscar caracteres en una cadena), 284
- Flanco ascendente, 185
- Flanco descendente, 185
- FLOOR, 226
- FOR (SCL), 233
- Forzar permanentemente, 725
- FRAC (fracción), 212
- GET, 509
- GET_DIAG, 316
- GetError, 244
- GetErrorID, 246
- GOTO (SCL), 237
- HSC (contador rápido), 353, 355
- HTA (Hex a ASCII), 277
- IF-THEN (SCL), 231
- IN_RANGE (dentro del rango), 203
- INC (Incrementar), 209
- INSERT (Insertar caracteres en una cadena), 282
- Insertar, 32
- Instrucciones ampliables, 35
- Instrucciones de conversión de SCL, 222
- INV (Complemento a uno), 249
- JMP, 238
- JMP_LIST, 239
- LEFT (Leer los caracteres izquierdos de una cadena), 280
- LEN (longitud), 278
- Límite, 211
- LN (logaritmo natural), 212
- Lógica con bits, 179
- Matemáticas en coma flotante, 212
- MAX (Determinar máximo), 210
- MB_CLIENT, 654
- MC_ChangeDynamic, 412
- MC_CommandTable, 410
- MC_Halt, 399
- MC_Home, 397
- MC_MoveAbsolute, 401
- MC_MoveJog, 407
- MC_MoveRelative, 403
- MC_MoveVelocity, 405
- MC_Power, 393
- MC_Reset, 396
- MID (Leer los caracteres centrales de una cadena), 280
- MIN (Determinar mínimo), 210
- MOD (Obtener resto de división), 207
- ModuleStates, 314
- MOVE, 214
- MUL (multiplicar), 206
- MUX (multiplexar), 251
- N_TRIG, 186
- NEG (Generar complemento a dos), 208
- NORM_X (normalizar), 227
- NOT OK, 204
- O, 248
- observar, 720
- OK, 204
- Operación de forzado permanente, 726
- Operaciones de temporizador, 191
- OUT_RANGE (fuera de rango), 203
- P_TRIG, 186
- Parámetros comunes, 476
- PID_Compact, 367
- PORT_CFG (configuración de puerto), 594
- PUT, 509
- QRY_CINT (consultar parámetros de alarma cíclica), 306
- RALRM, 289
- RCV_CFG (recibir configuración), 598
- RCV_PtP (recepción punto a punto), 606
- RCV_RST (inicializar receptor), 607
- RD_LOC_T (leer hora local), 259
- RD_SYS_T (leer hora del sistema), 259
- RDREC, 286
- RE_TRIGR, 81, 243
- Reloj, 259
- REPEAT (SCL), 235
- REPLACE (Reemplazar caracteres de una cadena), 283
- Retroceder (RET), 242
- RETURN (SCL), 238
- RIGHT (Leer los caracteres derechos de una cadena), 280

- ROL y ROR (rotar a la izquierda y rotar a la derecha), 254
 ROUND, 225
 RT (inicializar temporizador), 187
 S_CONV (convertir cadena de caracteres), 265
 S_MOV (desplazar cadena de caracteres), 265
 SCALE_X (escalar), 227
 SEL (seleccionar), 251
 SEND_CFG (enviar configuración), 596
 SEND_PtP (enviar datos punto a punto), 602
 SET_CINT (ajustar parámetros de alarma cíclica), 304
 SET_TIMEZONE, 263
 SGN_GET (consultar señales RS232), 608
 SGN_SET (activar señales RS232), 609
 SHL y SHR (desplazar a la izquierda y desplazar a la derecha), 253
 SIN (seno), 212
 SQR (cuadrado), 212
 SQRT (raíz cuadrada), 212
 SRT_DINT (Iniciar alarma de retardo), 307
 STP (Parar ciclo del PLC), 244
 STRG_VAL (convertir cadena de caracteres en valor numérico), 265
 SUB (restar), 206
 SWAP, 220
 SWITCH, 240
 T_ADD (sumar tiempos), 258
 T_COMBINE (combinar tiempos), 259
 T_CONFIG, 470
 T_CONV (convertir y extraer tiempos), 257
 T_DIFF (diferencia de tiempo), 258
 T_SUB (restar tiempos), 258
 TAN (tangente), 212
 TCON, 457
 TDISCON, 457
 Temporizador, 187
 Time, 257
 TOF (retardo al desconectar), 187
 TON (retardo al conectar), 187
 TONR (acumulador de tiempo), 187
 TP (impulso), 187
 TRCV, 457
 TRCV_C, 451, 485
 TRUNC (truncar), 225
 TSEND, 457
 TSEND_C, 451, 484
 TURCV, 465
 TUSEND, 465
 UFILL_BLK (rellenar área sin interrupciones), 218
 USS_DRV, 639
 USS_PORT, 642
 USS_RPM, 643
 USS_WPM, 644
 VAL_STRG (Convertir valor numérico en cadena de caracteres), 265
 Versiones de instrucciones, 36, 671
 WHILE (SCL), 234
 WR_SYS_T (escribir hora del sistema), 259
 WRREC, 286
 WWW (activa páginas web definidas por el usuario), 561
 XOR (O-exclusiva), 248
 Y, 248
 Instrucciones ampliables, 35
 Interfaces de comunicación
 Agregar módulos, 126
 CB 1241 RS485, 867
 CM 1241 RS232, 870
 Configuración, 611
 Configuración de dispositivos, 123
 Indicadores LED, 709
 Programación, 624
 RS232 y RS485, 589
 Tabla de comparación de módulos, 22
 Interfaz de usuario
 Vista del portal, 31
 Vista del proyecto, 31
 ISO on TCP
 Configuración de la conexión, 132
 ID de conexión, 445
 Modo ad hoc, 449
 Parámetros, 134
- J**
- JMP, 238
 JMP_LIST, 239
 JMPN, 238
- K**
- KOP (esquema de contactos)
 editor de programas, 720
 estado, 720
 observar, 720
 KOP (Esquema de contactos)
 estado, 720, 725
 observar, 720
 Vista general, 159

L

- Latencia, 76
- LED (Obtención estado LED), 312
- Leer variables HTTP, 549
- LEFT (Leer los caracteres izquierdos de una cadena), 280
- LEN (longitud), 278
- Librería del protocolo USS
 - Códigos de estado, 646
 - Información general sobre la configuración del accionamiento, 648
 - Requisitos de uso, 636
 - USS_DRV, 639
 - USS_PORT, 642
 - USS_RPM, 643
 - USS_WPM, 644
 - Vista general, 634
- Librería global
 - Vista general del protocolo USS, 634
- Limitaciones
 - páginas web definidas por el usuario, 564
 - Servidor web, 583
- Límite, 211
- Listado HTML, páginas web definidas por el usuario, 570

LI

- Llamar bloques lógicos en el programa de usuario, 152

L

- LN (logaritmo natural), 212
- Lógica con bits, 179
- Longitud
 - Mensaje, 622
- Longitud del mensaje, 621
- Longitud m, 622
- Longitud máxima del mensaje, 621
- Longitud n, 622

M

- Maestro AS-i CM 1243-2, 499
 - Funciones del módulo, 499
- Manuales, 4
- Máscara de subred, 141
- Matemáticas, 33, 205, 206
- Matemáticas en coma flotante, 212
- Matrices

- acceder a los miembros con una variable, 218

- MAX (Determinar máximo), 210

- MB_CLIENT, 654

- MB_COMM_LOAD, 672

- MB_MASTER, 675

- MB_SERVER, 661

- MB_SLAVE, 681

- MC_ChangeDynamic, 412

- MC_CommandTable, 410

- MC_Halt, 399

- MC_Home, 397

- MC_MoveAbsolute, 401

- MC_MoveJog, 407

- MC_MoveRelative, 403

- MC_MoveVelocity, 405

- MC_Power, 393

- MC_Reset, 396

- Memoria

- Direcciones de entradas de periferia (tabla de forzado permanente), 725

- I (memoria imagen de proceso de las entradas), 91

- L (memoria local), 89

- M (área de marcas), 92

- Marcas de ciclo, 85

- Marcas de sistema, 85

- Memoria de carga, 83

- Memoria de trabajo, 83

- Memoria remanente, 84

- Memoria temporal, 93

- Q (memoria imagen de proceso de las salidas), 92

- vigilar la carga de la memoria, 716

- Memoria de carga, 20

- CPU 1211C, 744

- CPU 1212C, 754

- CPU 1214C, 764

- CPU 1215C, 774

- Memory Card, 111

- Tarjeta de programa, 111

- Tarjeta de transferencia, 111

- Memoria de carga, para páginas web definidas por el usuario, 564

- Memoria de trabajo, 20

- CPU 1211C, 744

- CPU 1212C, 754

- CPU 1214C, 764

- CPU 1215C, 774

- Memoria I

- Direcciones de entradas de periferia (tabla de forzado permanente), 725

- Forzar permanentemente, 725

- HSC (contador rápido), 355

- observar, 720

- observar KOP, 720
- Operación de forzado permanente, 726
- Tabla de forzado permanente, 725
- tabla de observación, 720
- Memoria imagen de proceso
 - Operación de forzado permanente, 726
- Memoria Q
 - Configurar canales de impulsos, 327
 - Salidas de impulsos, 325
- Memoria remanente, 20, 84
 - CPU 1211C, 744
 - CPU 1212C, 754
 - CPU 1214C, 764
 - CPU 1215C, 774
- Memory Card
 - Actualización de firmware, 119
 - Configurar los parámetros de arranque, 113
 - Contraseña olvidada, 122
 - Datos técnicos, 872
 - Funcionamiento, 111
 - Insertar en la CPU, 112
 - referencia, 872
 - Tarjeta de programa, 117
 - Tarjeta de transferencia, 114
 - Tarjeta de transferencia vacía para una contraseña perdida, 122
 - Vista general, 111
- MID (Leer los caracteres centrales de una cadena), 280
- MIN (Determinar mínimo), 210
- MOD (Obtener resto de división), 207
- MODBUS
 - Ejemplo de esclavo Modbus, 689
 - MB_CLIENT, 654
 - MB_COMM_LOAD, 672
 - MB_MASTER, 675
 - MB_SERVER, 661
 - MB_SLAVE, 681
 - Versiones, 36, 671
- Modificar
 - estado del editor de programas, 720
 - tabla de observación, 722
- Modificar variables desde un PC, 537
- Modo ad hoc
 - ISO on TCP, 449
 - TCP, 449
- ModuleStates, 314
- Módulo de comunicación
 - Datos técnicos del CM 1241 RS422/RS485, 870
- Módulo de comunicación (CM)
 - Agregar módulo CM 1243-5 (maestro DP), 496
 - Agregar módulo maestro AS-i CM1243-2, 500
- Agregar módulos, 126
- Configuración de dispositivos, 123
- Configuración de parámetros, 130
- configuración del programa de ejemplo PtP, 627
- Consumo de corriente, 875
- Datos técnicos del CM 1241 RS232, 870
- Desmontaje, 53
- Indicadores LED, 589, 709
- Instalación, 53
- Programación, 624
- Recepción de datos, 606
- RS232 y RS485, 589
- Tabla de comparación, 22
- Vista general, 25
- Módulo de señales (SM)
 - Agregar módulos, 126
 - Cable de ampliación, 55
 - Configuración de parámetros, 130
 - Consumo de corriente, 875
 - Datos técnicos SM 1223, 800
 - Desmontaje, 52
 - Instalación, 51
 - Representación de entradas analógicas (intensidad), 814, 845
 - Representación de entradas analógicas (tensión), 814, 845
 - Representación de salidas analógicas (intensidad), 815, 846
 - Representación de salidas analógicas (tensión), 815, 846
 - SM 1221, 786
 - SM 1222, 790, 791
 - SM 1222 DQ8 RLY inversor, 790
 - SM 1223, 797
 - SM 1231 AI 4 x 13 bits, 803
 - SM 1231 AI 4 x 16 bits TC, 816
 - SM 1231 AI 4 x RTD x 16 bits, 822
 - SM 1231 AI 8 x 16 bits TC, 816
 - SM 1231 AI 8 x RTD x 16 bits, 822
 - SM 1232 AQ 2 x 14 bits, 807
 - SM 1232 AQ 4 x 14 bits, 807
 - SM 1234 AI 4 x 13 bits / AQ 2 x 14 bits, 810
 - Tiempos de respuesta a un escalón, 813
 - Vista general, 25
- Módulo de señales (SM) analógicas
 - SM 1231 AI 4 x RTD x 16 bits, 822
 - SM 1231 AI 8 x 13 bits, 803
 - SM 1231 AI 8 x 16 bits TC, 816
 - SM 1231 AI 8 x RTD x 16 bits, 822
 - SM 1231 AI 4 x 16 bits TC, 816
 - SM 1232 AQ 2 x 14 bits, 807
 - SM 1234 AI 4 x 13 bits / AQ 2 x 14 bits, 810

- Módulo de señales (SM) digitales
 - SM 1221, 786
 - SM 1222, 790, 791
 - SM 1223, 797
 - Módulos
 - CB 1241 RS485, 867
 - CM 1241 RS232, 870
 - Configurar parámetros, 130
 - Datos técnicos de la CPU 1211C, 744
 - Datos técnicos de la CPU 1212C, 754
 - Datos técnicos de la CPU 1214C, 764
 - Datos técnicos de la CPU 1215C, 774
 - Módulo de comunicación (CM), 25
 - Módulo de señales (SM), 25
 - Placa de comunicación (CB), 25
 - Procesador de comunicaciones (CP), 25
 - SB 1221 DI 4, 200 kHz, 828
 - SB 1222 DQ 4, 200 kHz, 200 kHz, 831
 - SB 1223 DI 2 / DQ 2, 837
 - SB 1223 DI 2 / DQ 2, 200 kHz, 834
 - SB 1231 AI 1 x 12 bits, 840
 - SB 1231 AI 1 x 16 bits RTD, 851
 - SB 1232 AQ 1x12 bits, 842
 - Signal Board (SB), 25
 - Signal Board de termopar SB 1231 AI 1 x 16 bits, 847
 - SM 1221, 786
 - SM 1222, 790, 791
 - SM 1222 DQ8 RLY inversor, 790
 - SM 1223, 797
 - SM 1231 AI 4 x 16 bits TC, 816
 - SM 1231 AI 4 x RTD x 16 bits, 822
 - SM 1231 AI 8 x 13 bits, 803
 - SM 1231 AI 8 x 16 bits módulo de señales TC, 816
 - SM 1231 AI 8 x RTD x 16 bits, 822
 - SM 1232 AQ 2 x 14 bits, 807
 - SM 1232 AQ 4 x 14 bits, 807
 - SM 1234 AI 4 x 13 bits / AQ 2 x 14 bits, 810
 - Tabla de comparación, 22
 - Zona de disipación, 42, 45
 - Módulos de comunicación RS232 y RS485, 589
 - Módulos no enchufados, 40
 - Montaje
 - Aislamiento, 62
 - Cable de ampliación, 55
 - Cargas de lámparas, 64
 - Cargas inductivas, 65
 - Conector de bloque de terminales, 54
 - CPU, 47
 - Dimensiones, 45
 - Espacio, 42
 - Módulo de comunicación (CM), 53
 - Módulo de señales (SM), 51
 - Placa de comunicación (CB), 49
 - Puesta a tierra, 63
 - Refrigeración, 42
 - Reglas, 41
 - Reglas de cableado, 61, 63
 - Signal Board (SB), 49
 - Ventilación, 42
 - Vista general, 46
 - Zona de disipación, 42, 45
 - Motion Control
 - Configurar el eje, 387
 - Finales de carrera por software y hardware, 416
 - MC_ChangeDynamic, 412
 - MC_CommandTable, 410
 - MC_Halt, 399
 - MC_Home, 397
 - MC_MoveAbsolute, 401
 - MC_MoveJog, 407
 - MC_MoveRelative, 403
 - MC_MoveVelocity, 405
 - MC_Power, 393
 - MC_Reset, 396
 - Parámetros de configuración de referenciación, 420
 - Referenciar (secuencia de referenciación activa), 422
 - Referenciar el eje, 419
 - Vista general, 384
 - MOVE, 214
 - MRES
 - Panel de mando, 38
 - MUL (multiplicar), 206
 - MUX (multiplexar), 251
 - My Documentation Manager, 4
- ## N
- N_TRIG, 186
 - NEG (Generar complemento a dos), 208
 - Network Time Protocol (NTP), 147
 - Nivel de protección
 - Bloque lógico, 170
 - Contraseña olvidada, 122
 - CPU, 168
 - Enlazar a una CPU o Memory Card, 171
 - Nombres de dispositivo
 - PROFINET IO, 487
 - Nombres de dispositivo online
 - PROFINET IO, 712
 - NORM_X (normalizar), 227
 - Normalización de valores analógicos, 229
 - Número de puerto, 448

Números

- Binario, 96
- Entero, 97
- Real, 98

Números de puerto

- Restringidos, 478

O

- OB de alarma cíclica, 75
- OB de alarma de error de diagnóstico, 76
- OB de alarma de error de tiempo, 76
- OB de alarma de proceso, 75
- OB de arranque, 75
- Objetos tecnológicos
 - HSC (contador rápido), 355
- Observar
 - Estado KOP, 720
 - Instrucción LED, 312
 - Operación de forzado permanente, 726
 - Tabla de forzado permanente, 725
 - tabla de observación, 720, 722
- Observar el programa, 175
- Obtención estado LED, 312
- Obtener valores de un DB online, 721
- Online
 - Asignar una dirección IP, 139
 - Botones RUN/STOP, 37
 - búfer de diagnóstico, 717
 - carga de la memoria, 716
 - comparar y sincronizar, 718
 - Conectar online, 711
 - Dirección IP, 714
 - estado, 720
 - Forzar permanentemente, 725
 - herramientas, 719
 - hora, 714
 - observar, 720
 - Obtener valores de un DB, 721
 - Operación de forzado permanente, 726
 - Panel de control, 715
 - Panel de mando, 38
 - Restablecer los valores iniciales de un DB, 721
 - tabla de observación, 720, 722
 - Tiempo de ciclo, 716
- OPC, configuración, 697
- OR, 248
- OUT_RANGE (fuera de rango), 203

P

- P_TRIG, 186
- Página web estándar Data Logs, 539
- Página web estándar de arranque, 531
- Página web estándar de comunicación, 535
- Página web estándar de identificación, 532
- Página web estándar de identificación de la CPU, 532
- Página web estándar de información del módulo, 533
- Página web estándar de introducción, 530
- Página web estándar Diagnostic, 532
- Página web estándar Variable Status, 537
- Páginas HTML, definidas por el usuario, 543
 - acceder a datos del S7-1200, 545
 - actualizar, 545
 - desarrollar, 544
 - ubicaciones de idiomas, 579
 - ubicaciones de páginas, 560
- Páginas web
 - STEP 7, 4
- Páginas web de STEP 7, 4
- Páginas web definidas por el usuario, 523, 543
 - acceder desde un PC, 563
 - activar con instrucción WWW, 561
 - activar y desactivar desde el DB de control, 580
 - actualizar, 545
 - borrar bloques de programa, 561
 - cargar DBs correspondientes, 563
 - Comandos AWP para acceder a datos del S7-1200, 545
 - configuración de idiomas múltiples, 579
 - Configurar, 560
 - control manual de DB de fragmentos, 580
 - crear con editor HTML, 544
 - crear fragmentos, 555
 - Ejemplo, 565
 - escribir variables, 547
 - escribir variables especiales, 551
 - generar bloques de programa, 561
 - Idiomas múltiples, 576
 - importar fragmentos, 556
 - leer variables, 546
 - leer variables especiales, 549
 - limitaciones de memoria de carga, 564
 - Listado HTML, 570
 - Procesamiento de caracteres especiales, 557
 - programación en STEP 7, 561
- Páginas web estándar, 523
 - acceder desde un PC, 525
 - acceso seguro, 526
 - Arranque, 531
 - cambiar el modo de operación, 531
 - Comunicación, 535

- Diagnóstico, 532
- Identificación, 532
- Información del módulo, 533
- inicio y cierre de sesión, 528
- Introducción, 530
- Registros, 539
- representación, 527
- restricciones de cookies, 585
- Restricciones de JavaScript, 584
- Variable Status, 537
- Panel de mando, 38
 - Estados operativos de la CPU, 70
- Paneles (HMI), 27
- Paneles Basic (HMI), 27
- Paneles de operador, 27
- Parametrización, 155
- Parámetro BUFFER, SEND_PTP, 605
- Parámetro LENGTH, SEND_PTP, 605
- Parámetros de arranque, 113
- Parámetros de salida, 155
 - Configurar canales de impulsos, 327
 - Salidas de impulsos, 325
- Paridad, 612
- Pausa, 615, 616
- Perfil DIN, 46
- PID
 - Algoritmo PID_3Step, 364, 373
 - Algoritmo PID_Compact, 364, 367
 - PID_3STEP, 373
 - PID_Compact, 367
 - Vista general, 363
- Placa de comunicación (CB)
 - Agregar módulos, 126
 - CB 1241 RS485, 867
 - Configuración de dispositivos, 123
 - Configuración de parámetros, 130
 - Desmontaje, 49
 - Indicadores LED, 589, 709
 - Instalación, 49
 - Programación, 624
 - RS485, 589
 - Tabla de comparación, 22
 - Vista general, 25
- PLC
 - Agregar módulos, 126
 - Asignar una dirección IP a una CPU online, 139
 - Cable de ampliación, 55
 - Carga de comunicación, 83
 - Cargar, 172
 - comparar y sincronizar, 718
 - Conector de bloque de terminales, 54
 - Configuración de dispositivos, 123
 - Configuración del HSC, 362
 - Contador de horas de funcionamiento, 261
 - copiar bloques desde una CPU online, 172
 - Corriente disponible, 42
 - CPU 1211C, 744
 - CPU 1212C, 754
 - CPU 1214C, 764
 - CPU 1215C, 774
 - Diseño del sistema, 149
 - Estados operativos, 70
 - Forzar permanentemente, 725
 - Instalación, 46, 47
 - Memory Card, 111, 872
 - observar, 720
 - Operación de forzado permanente, 726
 - Procesamiento del arranque, 72
 - Propiedad de sincronización horaria, 147
 - Protección de know-how, 170
 - RD_LOC_T (leer hora local), 259
 - RD_SYS_T (leer hora del sistema), 259
 - tabla de observación, 722
 - Tiempo de ciclo, 83
 - Tiempo de ciclo, 83
 - Usar bloques, 150
 - Variables, 89
 - Vista general de la CPU, 19
 - WR_SYS_T (escribir hora del sistema), 259
- Podcasts, 4
- Pointer
 - Vista general de Pointer, 103
- PORT_CFG (configuración de puerto), 594
- Posición de carácter
 - Longitud del mensaje, 622
- Posiciones de memoria, 89, 91
- Prioridad
 - Clase de prioridad, 74
 - Prioridad de procesamiento, 76
- Procesador de comunicaciones (CP)
 - Agregar módulos, 126
 - Configuración de dispositivos, 123
 - Configuración de parámetros, 130
 - Tabla de comparación, 22
 - Vista general, 25
- PROFIBUS
 - Agregar esclavo DP, 496
 - Agregar módulo CM 1243-5 (maestro DP), 496
 - Conexión de red, 131, 497
 - Conexión S7, 513
 - Dirección PROFIBUS, 498
 - DPNRM_DG, 298
 - DPRD_DAT, 296
 - DPWR_DAT, 296

- Esclavo, 491
- GET, 509
- Instrucciones de E/S descentralizadas, 285
- Maestro, 491
- Módulo CM 1242-5 (esclavo DP), 491
- Módulo CM 1243-5 (maestro DP), 492
- Propiedades de direcciones PROFIBUS, 498
- PUT, 509
- RALRM, 289
- RDREC, 286
- WRREC, 286
- PROFINET
 - Asignación de dirección IP, 148
 - Asignación de nombre y dirección de dispositivo, 148
 - Comprobar una red, 144
 - Comunicación de CPU a CPU, 482
 - Comunicación entre PLCs, 482
 - Conexión de red, 131, 482, 483, 487
 - Conexión S7, 513
 - configurar la comunicación entre la CPU y el dispositivo HMI, 480
 - Configurar la dirección IP, 127
 - diagnóstico, 490
 - Dirección IP, 141
 - Dirección MAC, 141
 - DPRD_DAT, 296
 - DPWR_DAT, 296
 - GET, 509
 - ID de conexión, 445
 - Instrucciones de E/S descentralizadas, 285
 - Modo ad hoc, 449
 - Número de conexiones de comunicación, 442
 - Propiedad de sincronización horaria, 147
 - Propiedades de direcciones Ethernet, 142
 - PUT, 509
 - RALRM, 289
 - RDREC, 286
 - Sincronización horaria, 127
 - T_CONFIG, 470
 - TCON, 457
 - TDISCON, 457
 - Tiempo de arranque del sistema, 148
 - Tipos de comunicación, 441
 - TRCV, 457
 - TRCV_C, 451
 - TSEND, 457
 - TSEND_C, 451
 - TURCV, 465
 - TUSEND, 465
 - Vista general, 448
 - WRREC, 286
- PROFINET IO
 - Agregar un dispositivo, 486
 - Asignar nombres de dispositivo, 487
 - Asignar nombres de dispositivo online, 712
 - Asignar una CPU, 487
 - Dispositivos, 486
 - Nombres de dispositivo, 487
 - Nombres de dispositivo online, 712
- PROFINET RT, 448
- Profundidad de anidamiento, 67
- Programa
 - Bloque de organización (OB), 153
 - Cargar, 172
 - Clase de prioridad, 74
 - copiar bloques desde una CPU online, 172
 - Enlazar a una CPU o Memory Card, 171
 - Llamar bloques lógicos en el programa de usuario, 152
 - Memory Card, 111
 - Obtener valores de un DB, 721
 - Programas lineales y estructurados, 150
 - Protección por contraseña, 170
 - Restablecer los valores iniciales de un DB, 721
- Programa de usuario
 - Agregar entradas o salidas a instrucciones KOP o FUP, 35
 - Bloque de organización (OB), 153
 - Cargar, 172
 - copiar bloques desde una CPU online, 172
 - Drag & Drop entre editores, 37
 - Enlazar a una CPU o Memory Card, 171
 - Favoritos, 32
 - Insertar instrucciones, 32
 - Instrucciones ampliables, 35
 - Llamar bloques lógicos en el programa de usuario, 152
 - Memory Card, 111
 - Programas lineales y estructurados, 150
 - Protección por contraseña, 170
 - Tarjeta de programa, 111
 - Tarjeta de transferencia, 111
- Programación
 - Agregar entradas o salidas a instrucciones KOP o FUP, 35
 - Algoritmo PID_3Step, 364, 373
 - Algoritmo PID_Compact, 364, 367
 - Bloque de datos (DB), 67
 - Bloque de datos (DB) de instancia, 155
 - Bloque de función (FB), 67, 155
 - Bloque de organización (OB), 153
 - Clase de prioridad, 74
 - comparar y sincronizar bloques lógicos, 718

- Contador de horas de funcionamiento, 261
- Contadores, 196
- CPU sin especificar, 125
- Drag & Drop entre editores, 37
- Enlazar a una CPU o Memory Card, 171
- Estados operativos de la CPU, 70
- Favoritos, 32
- Flujo de corriente (EN y ENO), 167
- Función (FC), 154
- FUP (Diagrama de funciones), 160
- Insertar instrucciones, 32
- Instrucciones ampliables, 35
- Instrucciones PtP, 624
- KOP (Esquema de contactos), 159
- Llamadas de bloque, 67
- Llamar bloques lógicos en el programa de usuario, 152
- Módulos no enchufados, 40
- Números válidos de FC, FB y DB, 67
- PID_3STEP, 373
- PID_Compact, 367
- Programa estructurado, 150
- Programa lineal, 150
- RD_LOC_T (leer hora local), 259
- RD_SYS_T (leer hora del sistema), 259
- SCL (Structured Control Language), 160, 161
- Tipos de bloques lógicos, 67
- Valor inicial de un FB, 155
- Vista general de PID, 363
- WR_SYS_T (escribir hora del sistema), 259
- Programación de STEP 7
 - páginas web definidas por el usuario, 561
 - programa de ejemplo PtP, 632
- Programación estructurada, 150, 152
 - Llamar bloques, 152
- Programación lineal, 150
- Programación punto a punto, 624
- Programar un cambio de idioma para la página web definida por el usuario, 577
- Propiedad de sincronización horaria, 147
- Propiedades de la CPU, para páginas web definidas por el usuario, 560
- Propiedades de la CPU, para páginas web definidas por el usuario (idiomas múltiples), 579
- Protección anticopia
 - Enlazar a una CPU o Memory Card, 171
- Protección de acceso
 - CPU, 168
- Protección de know-how
 - Protección por contraseña, 170
- Protección por contraseña
 - Acceso a la CPU, 168
 - Bloque lógico, 170
 - Contraseña olvidada, 122
 - CPU, 168
 - Enlazar a una CPU o Memory Card, 171
 - Protección anticopia, 171
 - Tarjeta de transferencia vacía, 122
- Protocolo
 - Comunicación, 591
 - Freeport, 591
 - ISO on TCP, 448
 - Modbus, 591
 - PROFINET RT, 448
 - TCP, 448
 - UDP, 448
 - USS, 591
- Protocolo Freeport, 591
- Protocolo ISO on TCP, 448
- Protocolo UDP, 448
- Protocolos Ethernet, 448
 - conexiones multinodo, 513
- Proyecto
 - Cargar, 172
 - comparar y sincronizar, 718
 - Contraseña olvidada, 122
 - Enlazar a una CPU o Memory Card, 171
 - Protección de acceso, 168
 - Proteger un bloque lógico, 170
 - Restringir el acceso a la CPU, 168
 - Tarjeta de programa, 117
 - Tarjeta de transferencia, 114
 - Tarjeta de transferencia vacía, 122
- PTO (tren de impulsos)
 - Configurar canales de impulsos, 327
 - CTRL_PWM, 323
 - Funcionamiento, 325
 - No se puede forzar permanentemente, 727
- Punteros
 - Tipo de datos Any, 105
 - Tipo de datos Pointer, 104
 - Tipo de datos Variant, 106
- PUT, 509
 - Configurar la conexión, 133
- PWM (modulación del ancho de impulso)
 - Configurar canales de impulsos, 327
- PWM (Modulación del ancho de impulso)
 - CTRL_PWM, 323
 - Funcionamiento, 325
 - No se puede forzar permanentemente, 727

Q

QRY_CINT (consultar parámetros de alarma cíclica), 306

R

RALRM, 289, 292

RCV_CFG (recibir configuración), 598

RCV_PTP (recepción punto a punto), 606

RCV_RST (inicializar receptor), 607

RD_LOC_T (leer hora local), 259

RD_SYS_T (leer hora del sistema), 259

RDREC, 286, 292

RE_TRIGR, 243

READ_DBL, 348

Recursos de información, 4

Referenciar tipos de enumeración, páginas web definidas por el usuario, 554

Referencias

Bloques de conectores, 883

Cables de ampliación, 883

Conectores y conexiones en borne, 881

CPUs, 879

Documentación, 884

Fuente de alimentación PM 1207, 882

Interfaces de comunicación (CM, CB y CP), 881

Memory Cards, 882

Módulos de señales (SM), 879

Paneles HMI Basic, 882

Signal Boards (SB), 880

Simuladores, 883

software de programación, 883

Software de visualización, 883

STEP 7, 883

Switch Ethernet CSM 1277, 882

WinCC, 883

Referencias cruzadas, 176

Introducción, 176

Usos, 176

Registro

DataLogClose (cerrar registro), 335

DataLogCreate (crear registro), 330

DataLogNewFile (crear un registro basándose en otro existente), 337

DataLogOpen (abrir registro), 333

DataLogWrite (escribir registro), 336

Estructura del registro, 328

Límites de tamaño del registro, 341

Ver registros de datos, 339

Vista general de los registros, 328

Reloj

Byte de memoria, 87

RD_LOC_T (leer hora local), 259

RD_SYS_T (leer hora del sistema), 259

Reloj en tiempo real, 88

WR_SYS_T (escribir hora del sistema), 259

Reloj del sistema

RD_LOC_T (leer hora local), 259

RD_SYS_T (leer hora del sistema), 259

WR_SYS_T (escribir hora del sistema), 259

REPLACE (Reemplazar caracteres de una cadena), 283

Requisitos de la instalación, 30

Requisitos del ordenador, 30

Requisitos del sistema, 30

Requisitos, instalación, 30

Restablecer los valores iniciales de un DB, 721

Restablecimiento de los ajustes de fábrica, 714

Restricciones de cookies en páginas web estándar, 585

Restricciones de JavaScript en páginas web estándar, 584

Retardo al conectar (TON), 187

Operación, 191

Retardo al desconectar (TOF), 187

Operación, 191

Retardo RTS OFF, 615

Retardo RTS ON, 615

Retroceder (RET), 242

RIGHT (Leer los caracteres derechos de una cadena), 280

ROL y ROR (rotar a la izquierda y rotar a la derecha), 254

ROUND, 225

Router IP, 141

RT (inicializar temporizador), 187

RTS, 613

RTS conmutado, 613

RTS siempre ON, 614

S

S_CONV (Convertir cadena de caracteres), 265

S_MOV (desplazar cadena de caracteres), 265

S7-1200

Agregar dispositivo, 124

Agregar módulos, 126

AS-i, 502

Botones RUN/STOP, 37

Cable de ampliación, 55

Carga de comunicación, 83

Cargas de lámparas, 64

Cargas inductivas, 65

- comparar bloques lógicos, 718
- Conector de bloque de terminales, 54
- Conexión de red, 131
- Configuración de dispositivos, 123
- Configuración del HSC, 362
- Configurar los módulos, 130
- Configurar los parámetros de la CPU, 127
- Contraseña olvidada, 122
- Corriente disponible, 42
- Dimensiones de montaje, 45
- Dirección AS-i, 502
- Dirección IP, 141
- Dirección MAC, 141
- Dirección PROFIBUS, 498
- Directrices de aislamiento, 62
- Dispositivos HMI, 27
- Espacio, 42
- Estados operativos, 70
- Forzar permanentemente, 725
- Funcionamiento, 722
- Instalación, 46
- Instalación de la CPU, 47
- Instalación de una CB, 49
- Memory Card, 872
- Módulo de comunicación (CM), 25
- Módulo de señales (SM), 25
- Módulos, 22
- Montaje de un CM, 53
- Montaje de un SM, 51
- Montaje de una SB, 49
- observar, 720
- Obtener valores de un DB, 721
- Operación de forzado permanente, 726
- Panel de mando, 38
- Parámetros de arranque, 113
- Placa de comunicación (CB), 25
- Procesador de comunicaciones (CP), 25
- Procesamiento del arranque, 72
- PROFIBUS, 497
- PROFINET, 141
- Protección de acceso, 168
- Protección de know-how, 170
- Protección por contraseña, 168
- Puerto AS-i, 501
- Puerto Ethernet, 141
- Puerto PROFIBUS, 497
- Puesta a tierra, 63
- Refrigeración, 42
- Reglas de cableado, 61, 63
- Restablecer los valores iniciales de un DB, 721
- Salidas de impulsos, 325
- Signal Board (SB), 25
- Tabla de comparación de los modelos de CPU, 20
- Tarjeta de programa, 117
- Tarjeta de transferencia, 114
- Tarjeta de transferencia vacía para una contraseña perdida, 122
- Tiempo de ciclo, 83
- TS Adapter, 22
- Ventilación, 42
- Vista general de la CPU, 19
- Zona de disipación, 42, 45
- Salidas de impulsos, 325
- SB 1221
 - diagrama de cableado de la SB 1221 DI 4, 200 kHz, 830
- SB 1222
 - Diagrama de cableado de la SB 1222 DQ 4 x 24 V DC, 200 kHz, 833
- SB 1223
 - diagrama de cableado de la SB 1223 DI 2 / DQ 2, 839
 - diagrama de cableado de la SB 1223 DI 2 / DQ, 200 kHz, 836
- SB 1231 AI 1 x 16 bits de termopar
 - Tabla de selección de filtros, 849
- SB 1232
 - Diagrama de cableado de la SB 1232 AQ 1 x 12 bits, 843
- SCALE_X (escalar), 227
- SCL (Structured Control Language)
 - ABS (Calcular valor absoluto), 209
 - ACOS (arcocoseno o coseno inverso), 212
 - Activar y desactivar, 182
 - Algoritmo PID_3Step, 364, 373
 - Algoritmo PID_Compact, 364, 367
 - ASIN (arcoseno o seno inverso), 212
 - ATAN (arcotangente o tangente inversa), 212
 - ATH (ASCII a hexadecimal), 276
 - ATTACH y DETACH, 301
 - Cambiar disposición, 220
 - CAN_DINT (Anular alarma de retardo), 307
 - CASE, 232
 - CEIL (redondear número en coma flotante a entero superior), 226
 - Comparación, 202
 - CONCAT (agrupar), 279
 - Condiciones, 161
 - Contador de horas de funcionamiento, 261
 - Contadores, 196
 - CONTINUE, 236
 - Control del programa, 230
 - CONV (convertir), 221
 - COS (coseno), 212

- CTD (decrementar contador), 196
 CTU (incrementar contador), 196
 CTUD (incrementar y decrementar contador), 196
 DataLogClose (cerrar registro), 335
 DataLogCreate (crear registro), 330
 DataLogNewFile (crear un registro basándose en otro existente), 337
 DataLogOpen (abrir registro), 333
 DataLogWrite (escribir registro), 336
 DEC (Decrementar), 209
 DECO (descodificar), 249
 DELETE (Borrar caracteres de una cadena), 281
 Desplazamiento, 214
 DeviceStates, 313
 Direccionamiento, 161
 DIS_AIRT (Retardar tratamiento de eventos de alarma y errores asíncronos de mayor prioridad), 309
 Editor de programas, 160
 EN y ENO (flujo de corriente), 167
 EN_AIRT (Habilitar tratamiento de eventos de alarma y errores asíncronos de mayor prioridad), 309
 ENCO (codificar), 249
 Estado LED, 312
 EXIT, 236
 EXP (exponente natural), 212
 Expresiones, 161
 EXPT (exponente general), 212
 FILL_BLK, 218
 FIND (Buscar caracteres en una cadena), 284
 FLOOR, 226
 FOR, 233
 FRAC (fracción), 212
 GET_DIAG, 316
 GOTO, 237
 HTA (hexadecimal a ASCII), 277
 IF-THEN, 231
 IN_RANGE (dentro del rango), 203
 INC (Incrementar), 209
 INSERT (Insertar caracteres en una cadena), 282
 Instrucciones de control, 161, 230, 231, 232, 233, 234, 235, 236, 237, 238
 Instrucciones de conversión, 222
 INV (Complemento a uno), 249
 JMP_LIST, 239
 LEFT (Leer los caracteres izquierdos de una cadena), 280
 LEN (longitud), 278
 LIMIT, 211
 Llamar bloques, 152
 Llamar un FB o una FC, 161
 LN (logaritmo natural), 212
 Lógica con bits, 179
 Matemáticas, 206
 Matemáticas (en coma flotante), 212
 Matemáticas en coma flotante, 212
 MAX (Determinar máximo), 210
 MC_ChangeDynamic, 412
 MC_CommandTable, 410
 MC_Halt, 399
 MC_Home, 397
 MC_MoveAbsolute, 401
 MC_MoveJog, 407
 MC_MoveRelative, 403
 MC_MoveVelocity, 405
 MC_Power, 393
 MC_Reset, 396
 MID (Leer los caracteres centrales de una cadena), 280
 MIN (Determinar mínimo), 210
 MOD (Obtener resto de división), 207
 ModuleStates, 314
 MUX (multiplexar), 251
 N_TRIG, 186
 NEG (Generar complemento a dos), 208
 NORM_X (normalizar), 227
 O, 248
 OK, 204
 Operaciones de temporizador, 191
 Operadores, 161
 OUT_RANGE (fuera de rango), 203
 P_TRIG, 186
 PID_3STEP, 373
 PID_Compact, 367
 Prioridad de operadores, 161
 QRY_CINT (consultar parámetros de alarma cíclica), 306
 RD_LOC_T (leer hora local), 259
 RD_SYS_T (leer hora del sistema), 259
 Redondear, 225
 REPEAT, 235
 REPLACE (Reemplazar caracteres de una cadena), 283
 RETURN, 238
 RIGHT (Leer los caracteres derechos de una cadena), 280
 ROL y ROR (rotar a la izquierda y rotar a la derecha), 254
 S_CONV (convertir cadena de caracteres), 265
 S_MOV (desplazar cadena de caracteres), 265
 SCALE_X (escalar), 227
 Sección Var, 160
 SEL (seleccionar), 251

- SET_CINT (ajustar parámetros de alarma cíclica), 304
- SET_TIMEZONE (ajustar zona horaria), 263
- SHL y SHR (desplazar a la izquierda y desplazar a la derecha), 253
- SIN (seno), 212
- SQR (cuadrado), 212
- SQRT (raíz cuadrada), 212
- SRT_DINT (Iniciar alarma de retardo), 307
- STRG_VAL (convertir cadena de caracteres en valor numérico), 265
- SWITCH, 240
- T_ADD (sumar tiempos), 258
- T_COMBINE (combinar tiempos), 259
- T_CONV (convertir y extraer tiempos), 257
- T_DIFF (diferencia de tiempo), 258
- T_SUB (restar tiempos), 258
- TAN (tangente), 212
- Temporizadores, 187
- Truncar, 225
- UFILL_BLK (rellenar área sin interrupciones), 218
- VAL_STRG (Convertir valor numérico en cadena de caracteres), 265
- Vista general, 160
- Vista general de PID, 363
- WHILE, 234
- WR_SYS_T (escribir hora del sistema), 259
- XOR (O-exclusiva), 248
- Y, 248
- Secuencia de caracteres
 - Fin del mensaje, 621
 - Inicio del mensaje, 617
- Seguridad
 - Contraseña olvidada, 122
 - CPU, 168
 - Enlazar a una CPU o Memory Card, 171
 - Protección anticopia, 171
 - Protección de acceso, 168
 - Protección de know-how de un bloque lógico, 170
- SEL (seleccionar), 251
- SEND_CFG (enviar configuración), 596
- SEND_PtP (enviar datos punto a punto), 602
 - Parámetros LENGH y BUFFER, 605
- Señal analógica (SM)
 - SM 1232 AQ 4 x 14 bits, 807
- Servicio y asistencia, 3
- Servidor web, 523
 - Conexiones HTTP máximas, 584
 - Convenciones para comillas, 557
 - Frecuencia de actualización, 524
 - Habilitar, 524
 - limitaciones, 583
 - Páginas web estándar, 525
- Servidor web, páginas web definidas por el usuario, 543
- SET_CINT (ajustar parámetros de alarma cíclica), 304
- SET_TIMEZONE (ajustar zona horaria), 263
- SGN_GET (consultar señales RS232), 608
- SGN_SET (activar señales RS232), 609
- SHL y SHR (desplazar a la izquierda y desplazar a la derecha), 253
- siemens_automation_language cookie, 577
- Signal Board (SB)
 - Agregar módulos, 126
 - Configuración de parámetros, 130
 - Consumo de corriente, 875
 - Desmontaje, 49
 - Instalación, 49
 - Representación de entradas (intensidad), 814, 845
 - Representación de entradas (tensión), 814, 845
 - Representación de salidas analógicas (intensidad), 815, 846
 - Representación de salidas analógicas (tensión), 815, 846
 - SB 1221 DI 4, 200 kHz, 828
 - SB 1222 DQ, 200 kHz, 831
 - SB 1223 DI 2 / DQ 2, 837
 - SB 1223 DI 2 / DQ 2, 200 kHz, 834
 - SB 1231 AI 1 x 12 bits, 840
 - SB 1231 AI 1 x 16 bits de termopar, 847
 - SB 1231 AI 1 x 16 bits RTD, 851
 - SB 1232 AQ 1x12 bits, 842
 - Tiempos de respuesta a un escalón, 844
 - Vista general, 25
- Signal Board analógica (SB)
 - SB 1231 AI 1 x 12 bits, 840
 - SB 1231 AI 1 x 16 bits de termopar, 847
 - SB 1231 AI 1 x 16 bits RTD, 851
 - SB 1232 AQ 1x12 bits, 842
- Simuladores, 873
- Simuladores de entradas, 873
- SIN (seno), 212
- Sin rearranque, 70
- Sincronización
 - Propiedad de sincronización horaria (PROFINET), 147
- SM 1231 RTD
 - tablas de selección, 827, 855
- SM y SB
 - Configuración de dispositivos, 123
 - Tabla de comparación, 22
- SMS, 696
- Solución de problemas
 - búfer de diagnóstico, 717

- Indicadores LED, 709
 - Soporte técnico de Siemens, 3
 - SQR (cuadrado), 212
 - SQRT (raíz cuadrada), 212
 - SRT_DINT (Iniciar alarma de retardo), 307
 - STEP 7
 - Agregar dispositivo, 124
 - Agregar entradas o salidas a una instrucción KOP o FUP, 35
 - Agregar módulos, 126
 - Agregar un dispositivo PROFINET IO, 486
 - AS-i, 502
 - Asignar una dirección IP a una CPU online, 139
 - Bloque de datos (DB), 67
 - Bloque de datos (DB) de instancia, 155
 - Bloque de función (FB), 67, 155
 - Botones RUN/STOP, 37
 - Cambiar la configuración, 36
 - Carga de comunicación, 83
 - Cargar, 172
 - Clase de prioridad (OB), 74
 - comparar y sincronizar, 718
 - Conexión de red, 131
 - Configuración de dispositivos, 123
 - Configuración del HSC, 362
 - Configurar la CPU, 127
 - Configurar los módulos, 130
 - Contador de horas de funcionamiento, 261
 - Contadores, 196
 - copiar bloques desde una CPU online, 172
 - Drag & Drop entre editores, 37
 - Entradas o salidas ampliables, 35
 - Estados operativos, 70
 - Favoritos, 32
 - Forzar permanentemente, 725
 - Función (FC), 154
 - Funcionamiento, 722
 - Insertar instrucciones, 32
 - Llamadas de bloque, 67
 - Llamar bloques lógicos en el programa de usuario, 152
 - Memory Card, 111, 872
 - Módulos no enchufados, 40
 - Números válidos de FC, FB y DB, 67
 - observar, 720
 - Obtener valores de un DB, 721
 - Operación de forzado permanente, 726
 - Panel de mando, 38
 - Procesamiento del arranque, 72
 - PROFIBUS, 497
 - PROFINET, 141
 - Programas lineales y estructurados, 150
 - Propiedad de sincronización horaria (PROFINET), 147
 - Protección por contraseña, 170
 - Puerto AS-i, 501
 - Puerto Ethernet, 141
 - Puerto PROFIBUS, 497
 - RD_LOC_T (leer hora local), 259
 - RD_SYS_T (leer hora del sistema), 259
 - Restablecer los valores iniciales de un DB, 721
 - Tarjeta de programa, 111
 - Tarjeta de transferencia, 111
 - Tiempo de ciclo, 83
 - Tiempo de ciclo, 83
 - Tipos de bloques lógicos, 67
 - Valor inicial de un FB, 155
 - Vista del portal, 31
 - Vista del proyecto, 31
 - WR_SYS_T (escribir hora del sistema), 259
 - STP (Parar ciclo del PLC), 244
 - STRG_VAL (convertir cadena de caracteres en valor numérico), 265
 - String
 - S_MOVE (desplazar cadena de caracteres), 265
 - Sinopsis del tipo de datos String, 264
 - Tipo de datos STRING, 100
 - Vista general de las operaciones con cadenas, 278
 - SUB (restar), 206
 - Support, 3
 - Sustitución de módulos, 40
 - SWAP, 220
 - SWITCH, 240
- T**
- T_ADD (sumar tiempos), 258
 - T_COMBINE (combinar tiempos), 259
 - T_CONFIG, 470
 - T_CONV (convertir y extraer tiempos), 257
 - T_DIFF (diferencia de tiempo), 258
 - T_SUB (restar tiempos), 258
 - Tabla de comparación
 - Dispositivos HMI, 27
 - Modelos de CPU, 20
 - Módulos, 22
 - Tabla de forzado permanente
 - direccionar entradas de periferia, 725
 - Forzar permanentemente, 725
 - Operación de forzado permanente, 726
 - Tabla de observación
 - desbloquear las salidas en modo STOP, 725
 - disparar valores, 724
 - Forzar permanentemente, 175

- Funcionamiento, 722
- Memory Card, 111
 - observar, 720
- TAN (tangente), 212
- Tarjeta de programa
 - Configurar los parámetros de arranque, 113
 - Crear, 117
 - Funcionamiento, 111
 - Insertar en la CPU, 112
 - referencia, 872
 - Vista general, 111
- Tarjeta de transferencia, 114
 - Configurar los parámetros de arranque, 113
 - Contraseña olvidada, 122
 - Funcionamiento, 111
 - Insertar en la CPU, 112
 - referencia, 872
 - Tarjeta de transferencia vacía para una contraseña olvidada, 122
 - Vista general, 111
- Task Cards
 - Columnas y encabezados, 36, 671
- TCON, 457
 - Configuración, 132
 - ID de conexión, 445
 - Parámetros de la conexión, 134
- TCON_Param, 134
- TCP
 - Configuración de la conexión, 132
 - ID de conexión, 445
 - Modo ad hoc, 449
 - Parámetros, 134
 - Protocolo, 448
- TDISCON, 457
- Technical support, 3
- Telecontrol, 692
- TeleService vía GPRS, 692
- Temporizadores
 - Cantidad, 21, 746, 756, 766, 776
 - Operación, 191
 - RT (inicializar temporizador), 187
 - Tamaño, 21, 746, 756, 766, 776
 - TOF (retardo al desconectar), 187
 - TON (retardo al conectar), 187
 - TONR (acumulador de tiempo), 187
 - TP (impulso), 187
- Tensiones nominales, 742
- Termopar
 - Compensación de unión fría, 820, 849
 - funcionamiento básico, 820, 849
 - Tabla de selección de filtros de termopar para el SM 1231, 820
 - Tabla de selección de filtros de termopar para la SB 1231, 849
 - Tabla de selección de termopares de la SB 1231, 849
 - Tabla de selección de termopares del SM 1231, 820
- TIA Portal
 - Vista del portal, 31
 - Vista del proyecto, 31
- Tiempo
 - RD_LOC_T (leer hora local), 259
 - RD_SYS_T (leer hora del sistema), 259
 - SET_TIMEZONE (ajustar zona horaria), 263
 - WR_SYS_T (escribir hora del sistema), 259
- Tiempo de ciclo
 - Configuración, 83
 - vigilar, 716
 - Vista general, 81
- Tiempo de espera, 613
- Tiempo excedido entre caracteres, 621
- Time
 - DTL (tipo de datos Data and Time Long), 99
 - T_ADD (sumar tiempos), 258
 - T_COMBINE (combinar tiempos), 259
 - T_CONV (convertir y extraer tiempos), 257
 - T_DIFF (diferencia de tiempo), 258
 - T_SUB (restar tiempos), 258
 - Tipo de datos Time, 99
 - TOD (tipo de datos Time of Day), 99
- Tipo de datos DTL
 - Instrucciones para el reloj del sistema, 259
- Tipos de datos, 95
 - Any (puntero), 105
 - Bool, byte, palabra o palabra doble, 96
 - Char (carácter) y string, 100
 - Editor de tipo de datos PLC, 103
 - Matrices, 101
 - Pointer (puntero), 104
 - Real, LReal (real en coma flotante), 98
 - Struc, 103
 - Time, Date, TOD (hora del día), DTL (fecha y hora largo), 99
 - USInt, SInt, UInt, Int, UDInt, Dint (entero), 97
 - Variant (puntero), 106
 - Vista general del tipo de datos Pointer, 103
- Tipos de enumeración en páginas web definidas por el usuario, 553, 554
- TM_MAIL, 701
- Transición de RUN a STOP, 88
- Transmisión de datos, iniciar, 602
- TRCV, 457
 - ID de conexión, 445

Modo ad hoc, 449
 TRCV_C, 451, 485
 Configuración, 132
 ID de conexión, 445
 Modo ad hoc, 449
 Parámetros de la conexión, 134
 TRUNC (truncar), 225
 TS Adapter, 22
 Instalar en un perfil DIN, 60
 Instalar en una pared, 60
 Instalar un módulo TS, 57
 Tarjeta SIM, 58
 TSAP, 448
 TSAP (Transport Service Access Points), 134, 450, 483, 513
 TSAPs
 Restringidos, 478
 TSAPs y números de puerto restringidos, 478
 TSEND, 457
 ID de conexión, 445
 TSEND_C, 484
 Configuración, 132
 ID de conexión, 445
 Parámetros de la conexión, 134
 TSEND_C, 484
 TURCV, 465
 Configuración, 132
 Parámetros de la conexión, 134
 TUSEND, 465
 Configuración, 132
 Parámetros, 134

U

UDP
 Configuración de la conexión, 132
 Parámetros, 134
 UFILL_BLK (rellenar área sin interrupciones), 218

V

VAL_STRG (Convertir valor numérico en cadena de caracteres), 265
 Valores booleanos o de bit, 90
 Valores de retorno
 Instrucciones de la comunicación abierta, 477
 Instrucciones PtP, 592
 Valores de retorno de instrucciones PtP, 592
 Valores de retorno de las instrucciones de la comunicación abierta, 477

Valores de retorno en el tiempo de ejecución de la recepción, 606
 Valores iniciales
 Obtener y restablecer los valores iniciales de un DB, 721
 Variables
 estado, 720
 Forzar permanentemente, 725
 observar, 720
 Operación de forzado permanente, 726
 vigilar y modificar desde un PC, 537
 Velocidad de transferencia, 612
 Ventilación, 42
 Versiones de instrucciones, 36, 671
 Vida útil de los relés, 743
 Vigilar
 carga de la memoria, 716
 Obtener valores de un DB, 721
 Restablecer los valores iniciales de un DB, 721
 Tiempo de ciclo, 716
 Vigilar variables desde un PC, 537
 Vista del portal, 31
 Vista del proyecto, 31
 Visualización
 Dispositivos HMI, 27
 Visualizar las direcciones MAC e IP, 145

W

Watchdog, 243
 WR_SYS_T (escribir hora del sistema), 259
 WRIT_DBL, 348
 WRREC, 286, 292
 WWW (activa páginas web definidas por el usuario), 561

X

XON/XOFF, 614
 XOR (O exclusiva), 248

Y

Y, 248

Z

Zona de disipación, 42, 45

