SIEMENS

SIMATIC

S7-1200 Easy Book

Manual de producto

Prólogo

Presentamos el potente y flexible S7-1200 1

STEP 7 facilita el trabajo 2

Getting Started 3

Conceptos básicos de los PLC de la forma más sencilla 4

La configuración de dispositivos se crea fácilmente 5

Programación cada vez más sencilla 6

Fácil comunicación entre dispositivos 7

Instrucción PID sencilla 8

Servidor web para una conexión a Internet sencilla 9

El control de movimiento es muy fácil 10

Facilidad de uso de las herramientas online 11

IO-Link es fácil 12

Datos técnicos A

Reemplazar una CPU V3.0 por una CPU V4.1 B

01/2015

A5E02486778-AG

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

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

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.

Siemens AG
Division Digital Factory
Postfach 48 48
90026 NÜRNBERG
ALEMANIA

A5E02486778-AG
℗ 12/2014 Sujeto a cambios sin previo aviso

Copyright © Siemens AG 2015.
Reservados todos los derechos

Prólogo

Bienvenido al mundo del S7-1200. El controlador compacto SIMATIC S7-1200 es el modelo modular y de tamaño reducido para pequeños sistemas de automatización que requieran funciones simples o avanzadas para lógica, HMI o redes. Gracias a su diseño compacto, su bajo coste y sus potentes funciones, los sistemas de automatización S7-1200 son idóneos para controlar tareas sencillas.

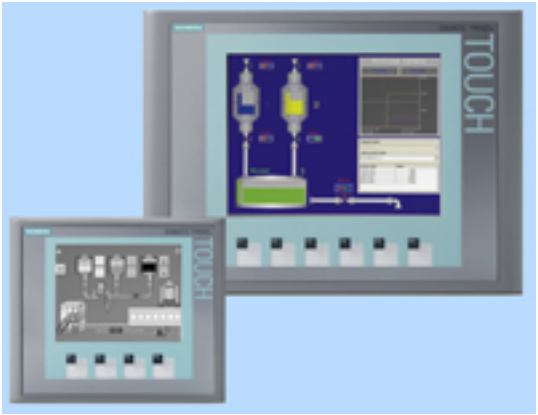
Conforme al compromiso de SIMATIC de ofrecer una automatización plenamente integrada (TIA: Totally Integrated Automation), la familia de productos S7-1200 y el software de programación TIA Portal proporcionan la flexibilidad necesaria para cubrir las diferentes necesidades de automatización en cada caso.

El S7-1200 ayuda a facilitar las tareas más complejas

La solución basada en el controlador SIMATIC S7-1200, diseñado dentro de la categoría de "compactos", se compone del controlador SIMATIC S7-1200 y los paneles SIMATIC HMI Basic, ambos programables con el software de ingeniería TIA Portal. La posibilidad de programar ambos dispositivos con el mismo software reduce significativamente los costes de desarrollo. El TIA Portal incluye STEP 7 para la programación del S7-1200 y WinCC para diseñar los proyectos del Basic Panel.

El controlador S7-1200 compacto incluye:

- PROFINET incorporado
- Entradas y salidas rápidas aptas para el control de movimiento (Motion Control), entradas analógicas integradas para minimizar el espacio requerido y la necesidad de E/S adicionales, 4 generadores de impulsos para aplicaciones de tren de impulsos y de ancho de impulso (Página 75), y hasta 6 contadores rápidos (Página 136)
- E/S integradas en los módulos CPU que ofrecen entre 6 y 14 entradas y entre 4 y 10 salidas



Módulos de señales para DC, relé o E/S analógicas amplían el número de E/S, mientras que las innovadoras Signal Boards integradas en el frontal de la CPU proporcionan entradas y salidas adicionales (Página 19).

Los paneles SIMATIC HMI Basic (Página 21) han sido diseñados específicamente para el S7-1200.

Este Easy Book es una introducción al PLC S7-1200. Las siguientes páginas ofrecen una visión general de las numerosas funciones y prestaciones de los dispositivos.

Para más información, consulte el *Manual de sistema del controlador programable S7-1200*. Para más información sobre homologaciones UL y FM, marcado CE, C-Tick y otros estándares, consulte los Datos técnicos (Página 371).

Este manual describe los productos siguientes:

- STEP 7 V13 SP1 Basic y Professional
- Versión de firmware V4.1 de la CPU S7-1200

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 del controlador programable 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). Los manuales electrónicos se pueden descargar de la página web de atención al cliente y también se incluyen en el disco de documentación 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 CPU 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 y soltar 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, FAQ 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 eficiencia 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)

Service & Support

Además de la documentación, Siemens ofrece conocimientos técnicos en Internet, en la página web del Customer Support (http://www.siemens.com/tiaportal).

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.

Información de seguridad

Siemens suministra productos y soluciones con funciones de seguridad industrial que contribuyen al funcionamiento seguro de instalaciones, soluciones, máquinas, equipos y redes. Dichas funciones son un componente importante de un sistema global de seguridad industrial. En consideración de lo anterior, los productos y soluciones de Siemens son objeto de mejoras continuas. Por ello, le recomendamos que se informe periódicamente sobre las actualizaciones de nuestros productos

Para el funcionamiento seguro de los productos y soluciones de Siemens, es preciso tomar medidas de protección adecuadas (como el concepto de protección de células) e integrar cada componente en un sistema de seguridad industrial integral que incorpore los últimos avances tecnológicos. También deben tenerse en cuenta los productos de otros fabricantes que se estén utilizando. Encontrará más información sobre seguridad industrial en (http://www.siemens.com/industrialsecurity).

Si desea mantenerse al día de las actualizaciones de nuestros productos, regístrese para recibir un boletín de noticias específico del producto que desee. Encontrará más información en (http://support.automation.siemens.com).

Índice

Prólogo 3

1 Presentamos el potente y flexible S7-1200 15

1.1 Introducción al PLC S7-1200 15

1.2 Capacidad de expansión de la CPU 19

1.3 Módulos S7-1200 20

1.4 Paneles HMI Basic 21

1.5 Dimensiones de montaje y espacio libre necesario 22

1.6 Nuevas funciones 27

2 STEP 7 facilita el trabajo 31

2.1 Fácil entrada de instrucciones en el programa de usuario 32

2.2 Fácil acceso a las instrucciones más utilizadas desde la barra de herramientas 33

2.3 Facilidad para agregar entradas o salidas a instrucciones KOP y FUP 33

2.4 Instrucciones ampliables 34

2.5 Facilidad de cambio del modo de operación de la CPU 35

2.6 Facilidad de modificación de la apariencia y configuración de STEP 7 36

2.7 Librerías de proyecto y globales para un acceso fácil 36

2.8 Facilidad para seleccionar la versión de una instrucción 37

2.9 La función de Drag & Drop puede utilizarse fácilmente entre editores 38

2.10 Modificar el tipo de llamada de un DB 39

2.11 Desconectar temporalmente dispositivos de una red 40

2.12 Facilidad de "desenchufar" virtualmente módulos sin perder la configuración 41

3 Getting Started 43

3.1 Crear un proyecto 43

3.2 Crear variables para las E/S de la CPU 44

3.3 Crear un segmento simple en el programa de usuario 46

3.4 Utilice las variables PLC de la tabla de variables para direccionar las instrucciones 48

3.5 Adición de una instrucción "cuadro" 49

3.6 Uso de la instrucción CALCULAR para una ecuación matemática compleja 50

3.7 Agregar un dispositivo HMI al proyecto 52

3.8 Creación de una conexión de red entre la CPU y el dispositivo HMI 53

3.9 Creación de una conexión HMI para compartir variables 53

3.10 Creación de una pantalla HMI ... 54
3.11 Selección de una variable PLC para el elemento HMI ... 55
4 Conceptos básicos de los PLC de la forma más sencilla ... 57
4.1 Las tareas se realizan en cada ciclo ... 57
4.2 Estados operativos de la CPU ... 59
4.3 Ejecución del programa de usuario ... 60
4.3.1 Procesamiento del ciclo en estado operativo RUN ... 61
4.3.2 Los OB ayudan a estructurar el programa de usuario ... 61
4.3.3 Prioridades y colas de espera para la ejecución de eventos ... 62
4.4 Áreas de memoria, direccionamiento y tipos de datos ... 65
4.4.1 Tipos de datos soportados por el S7-1200 ... 66
4.4.2 Direccionamiento de áreas de memoria ... 69
4.4.3 Acceder a un "segmento" de un tipo de datos de variable ... 71
4.4.4 Acceso a una variable con una superposición de AT ... 73
4.5 Generadores de impulsos ... 75
5 La configuración de dispositivos se crea fácilmente ... 77
5.1 Cargar la configuración de una CPU conectada ... 78
5.2 Agregar una CPU a la configuración ... 80
5.3 Cambiar un dispositivo ... 81
5.4 Agregar módulos a la configuración ... 81
5.5 Control de configuración ... 83
5.6 Configurar el funcionamiento de la CPU y sus módulos ... 84
5.6.1 La memoria de sistema y la marca de ciclo proporcionan funciones estándar ... 86
5.7 Configurar la dirección IP de la CPU ... 89
5.8 Proteger el acceso a la CPU o al bloque lógico es fácil ... 91
5.8.1 Protección de know-how ... 94
5.8.2 Protección anticopia ... 95
6 Programación cada vez más sencilla ... 97
6.1 Fácil diseño del programa de usuario ... 97
6.1.1 Uso de OBs para organizar el programa de usuario ... 99
6.1.2 Con los FBs y las FCs, la programación de tareas modulares es muy fácil ... 101
6.1.3 Los bloques de datos permiten almacenar fácilmente los datos del programa ... 102
6.1.4 Crear un nuevo bloque lógico ... 103
6.1.5 Crear bloques lógicos reutilizables ... 104
6.1.6 Llamar un bloque lógico desde otro bloque lógico ... 105
6.2 Lenguajes de programación fáciles de usar ... 106
6.2.1 Esquema de contactos (KOP) ... 106
6.2.2 Diagrama de funciones (FUP) ... 107
6.2.3 Vista general SCL ... 108
6.2.4 Editor de programas SCL ... 109
6.3 Instrucciones eficaces que facilitan la programación ... 111
6.3.1 Disponer de las instrucciones básicas esperadas ... 111

6.3.2 Instrucciones de comparación y desplazamiento 114
6.3.3 Operaciones de conversión 116
6.3.4 Cálculos matemáticos sencillos con la instrucción Calcular 118
6.3.5 Operaciones de temporizadores 119
6.3.6 Operaciones de contadores 125
6.3.7 Modulación del ancho de pulso (PWM) 128

6.4 Registros de datos fáciles de crear 129

6.5 Programa de usuario fácil de controlar y comprobar 131
6.5.1 Tablas de observación y tablas de forzado 131
6.5.2 Referencia cruzada para mostrar la utilización 131
6.5.3 Estructura de llamadas para ver la jerarquía de llamadas 133
6.5.4 Instrucciones de diagnóstico para controlar el hardware 134
6.5.4.1 Lectura de los estados de los LEDs de la CPU 134
6.5.4.2 Instrucciones para leer el estado de diagnóstico de los dispositivos 134

6.6 Contador rápido (HSC) 136
6.6.1 Funcionamiento del contador rápido 137
6.6.2 Configuración del HSC 145

7 Fácil comunicación entre dispositivos 147

7.1 Crear una conexión de red 148

7.2 Opciones de comunicación 149

7.3 Conexiones de comunicación asíncronas de V4.1 151

7.4 Instrucciones PROFINET y PROFIBUS 154

7.5 PROFINET 156
7.5.1 Open User Communication 156
7.5.1.1 Modo ad hoc 157
7.5.1.2 ID de conexión para las instrucciones de Open User Communication 157
7.5.1.3 Parámetros de la conexión PROFINET 161
7.5.2 Configurar la vía de conexión local/de interlocutor 163

7.6 PROFIBUS 166
7.6.1 Servicios de comunicaciones de CM PROFIBUS 167
7.6.2 Referencia a los manuales de usuario de CM PROFIBUS 168
7.6.3 Agregar el módulo CM 1243-5 (maestro DP) y un esclavo DP 169
7.6.4 Asignar direcciones PROFIBUS al módulo CM 1243-5 y al esclavo DP 170

7.7 AS-i 172
7.7.1 Agregar el módulo maestro AS-i CM 1243-2 y esclavo AS-i 172
7.7.2 Asignar una dirección AS-i a un esclavo AS-i 173

7.8 Comunicación S7 176
7.8.1 Instrucciones GET y PUT 176
7.8.2 Crear una conexión S7 177
7.8.3 Asignación de parámetros de conexión GET/PUT 178

7.9 GPRS 179
7.9.1 Conexión a una red GSM 179
7.9.2 Aplicaciones del CP 1242-7 180
7.9.3 Otras propiedades del CP 1242-7 182
7.9.4 Configuración y conexiones eléctricas 182
7.9.5 Más información 183

7.9.6 Accesorios ... 184
7.9.7 Consulte el manual de la antena GSM ... 185
7.9.8 Ejemplos de configuración de telecontrol ... 185

7.10 Protocolos de comunicación PtP, USS y Modbus ... 190
7.10.1 Comunicación punto a punto ... 190
7.10.2 Utilizar las interfaces de comunicación serie ... 192
7.10.3 Instrucciones PtP ... 193
7.10.4 Instrucciones USS ... 194
7.10.5 Instrucciones Modbus ... 196

8 Instrucción PID sencilla ... 199

8.1 Insertar la instrucción PID y un objeto tecnológico ... 201

8.2 Instrucción PID_Compact ... 203

8.3 Parámetros de la instrucción ErrorBit de PID_Compact ... 207

8.4 Instrucción PID_3Step ... 209

8.5 Parámetros ErrorBit de la instrucción PID_3Step ... 216

8.6 Instrucción PID_Temp ... 218
8.6.1 Vista general ... 218
8.6.2 Operación del regulador PID_Temp ... 222
8.6.3 Reguladores en cascada ... 225

8.7 Parámetros ErrorBit de la instrucción PID_Temp ... 229

8.8 Configuración de los reguladores PID_Compact y PID_3Step ... 231

8.9 Configuración del regulador PID_Temp ... 234

8.10 Puesta en servicio de los reguladores PID_Compact y PID_3Step ... 248

8.11 Puesta en servicio del regulador PID_Temp ... 250

9 Servidor web para una conexión a Internet sencilla ... 261

9.1 Páginas web estándar fáciles de usar ... 262

9.2 Limitaciones que pueden afectar al uso del servidor web ... 265

9.3 Páginas web definidas por el usuario fáciles de crear ... 266
9.3.1 Páginas web definidas por el usuario personalizadas fáciles de crear ... 266
9.3.2 Limitaciones específicas de las páginas web definidas por el usuario ... 268
9.3.3 Configuración de una página web definida por el usuario ... 269
9.3.4 Uso de la instrucción WWW ... 270

10 El control de movimiento es muy fácil ... 271

10.1 Escalonamiento ... 277

10.2 Configurar un generador de impulsos ... 279

10.3 Control de movimiento en lazo abierto ... 280
10.3.1 Configurar el eje ... 280
10.3.2 Puesta en servicio ... 284

10.4 Control de movimiento en lazo cerrado ... 290
10.4.1 Configurar el eje ... 290
10.4.2 Puesta en servicio ... 297

10.5 Configurar TO_CommandTable_PTO 303

10.6 Operación de Motion Control para S7-1200 306
10.6.1 Salidas de la CPU utilizadas para control de movimiento 306
10.6.2 Finales de carrera por hardware y software para Motion Control 308
10.6.3 Referenciación 311
10.6.3.1 Referenciar el eje 311
10.6.3.2 Configuración de parámetros de referenciación 313
10.6.3.3 Secuencia de referenciación activa 315

10.7 Instrucciones de Motion Control 317
10.7.1 Vista general de instrucciones de MC 317
10.7.2 Instrucción MC_Power (Habilitar/bloquear eje) 318
10.7.3 Instrucción MC_Reset (Confirmar error) 321
10.7.4 Instrucción MC_Home (Referenciar eje) 322
10.7.5 Instrucción MC_Halt (Detener eje) 325
10.7.6 Instrucción MC_MoveAbsolute (Posicionamiento absoluto del eje) 327
10.7.7 Instrucción MC_MoveRelative (Posicionamiento relativo del eje) 329
10.7.8 Instrucción MC_MoveVelocity (Mover el eje a la velocidad predefinida) 331
10.7.9 Instrucción MC_MoveJog (Desplazamiento del eje en modo Jog) 334
10.7.10 Instrucción MC_CommandTable (Ejecutar comandos de eje como secuencia de movimientos) 336
10.7.11 Instrucción MC_ChangeDynamic (Cambiar la configuración de la dinámica del eje) 339
10.7.12 Instrucción MC_WriteParam (Escribir parámetros de un objeto tecnológico) 341
10.7.13 Instrucción MC_ReadParam (leer parámetros de un objeto tecnológico) 343

11 Facilidad de uso de las herramientas online 345

11.1 Establecer una conexión online con una CPU 345

11.2 Interactuar con la CPU online 346

11.3 Conexión online para observar los valores en la CPU 347

11.4 Visualización sencilla del estado del programa de usuario 348

11.5 Utilizar una tabla de observación para vigilar la CPU 348

11.6 Utilizar la tabla de forzado permanente 350

11.7 Obtener los valores online de un DB para restablecer los valores iniciales 353

11.8 Carga de elementos del proyecto 354

11.9 Comparar CPUs online y offline 355

11.10 Visualizar los eventos de diagnóstico 356

11.11 Ajustar la dirección IP y la hora 356

11.12 Restablecimiento de los ajustes de fábrica 357

11.13 Actualización del firmware 358

11.14 Descargar una dirección IP permanente en una CPU online 359

11.15 Utilizar la "CPU sin especificar" para cargar la configuración hardware 360

11.16 Cargar en estado operativo RUN 361
11.16.1 Modificar el programa en el estado operativo RUN 363

11.17 Trazas y registros de datos de la CPU en condiciones de disparo 364

12 IO-Link es fácil 365

12.1 Sinopsis de la tecnología IO-Link 365

12.2 Componentes de un sistema IO-Link 365

12.3 Después del arranque 366

12.4 Protocolo IO-Link 366

12.5 Configuración en el bus de campo 367

12.6 IO-Link y el programa STEP 7 367

12.7 El SM 1278 4 maestros IO-Link 368

A Datos técnicos 371

A.1 Datos técnicos generales 371

A.2 Módulos CPU 382

A.3 Módulos de E/S digitales 387
A.3.1 SB 1221, SB 1222 y SB 1223 entrada/salida digital (DI, DQ y DI/DQ) 387
A.3.2 SM 1221 con entrada digital (DI) 390
A.3.3 SM 1222 con salida digital (DQ) 391
A.3.4 SM 1223 V DC entrada/salida digital (DI / DQ) 393
A.3.5 SM 1223 120/230 V AC entradas / salidas de relé 394

A.4 Datos técnicos de las entradas y salidas digitales 395
A.4.1 Entradas digitales de 24 V DC (DI) 395
A.4.2 Entradas digitales 120/230 V AC 397
A.4.3 Salidas digitales (DQ) 398

A.5 Módulos de E/S analógicas 401
A.5.1 SB 1231 y SB 1232 con entrada analógica (AI) y salida analógica (AQ) 401
A.5.2 SM 1231 con entrada analógica (AI) 402
A.5.3 SM 1232 con salida analógica (AQ) 402
A.5.4 SM 1234 con entrada/salida analógica (AI/AQ) 402
A.5.5 Diagramas de cableado de SM 1231 (AI), SM 1232 (AQ) y SM 1234 (AI/AQ) 403

A.6 Battery Board BB 1297 404

A.7 Datos técnicos de E/S analógicas 405
A.7.1 Datos técnicos de entradas analógicas (CPU, SM y SB) 405
A.7.2 Rangos de medida de entradas (AI) de tensión e intensidad 406
A.7.3 Respuesta a un escalón de las entradas analógicas (AI) 408
A.7.4 Tiempo de muestreo y tiempos de actualización de entradas analógicas 408
A.7.5 Datos técnicos de las salidas analógicas 409
A.7.6 Rangos de medición de salidas (AQ) de tensión e intensidad 410

A.8 Módulos RTD y termopar 412
A.8.1 Datos técnicos de SB 1231 RTD y SB 1231 TC 413
A.8.2 Datos técnicos de SM 1231 RTD 415
A.8.3 Datos técnicos de SM 1231 TC 416
A.8.4 Datos técnicos de entradas analógicas para RTD y TC (SM y SB) 418
A.8.5 Tipo de termopar 419
A.8.6 Selección de filtros de termopares y tiempos de actualización 420
A.8.7 Tabla de selección de tipos de sensor RTD 420
A.8.8 Selección de filtros RTD y tiempos de actualización 422

A.9 Interfaces de comunicación 423
A.9.1 Maestro/esclavo PROFIBUS 423
A.9.1.1 CM 1242-5 ESCLAVO PROFIBUS DP....423
A.9.1.2 Asignación de pines del conector sub-D del CM 1242-5....424
A.9.1.3 CM 1243-5 maestro PROFIBUS DP....425
A.9.1.4 El maestro PROFIBUS (CM 1243-5) requiere alimentación de 24 V DC de la CPU 426
A.9.1.5 Asignación de pines del conector sub-D del CM 1243-5....427
A.9.2 CP GPRS....428
A.9.2.1 CP 1242-7 GPRS....428
A.9.2.2 Antena GSM/GPRS ANT794-4MR 429
A.9.2.3 Antena plana ANT794-3M 430
A.9.3 Teleservice (TS)....431
A.9.4 Comunicación RS485, RS232 y RS422 431
A.9.4.1 Datos técnicos de CB 1241 RS485 431
A.9.4.2 Datos técnicos del CM 1241 RS422/485....434
A.9.4.3 Datos técnicos del CM 1241 RS232....435

A.10 Módulos tecnológicos 437
A.10.1 SM 1278 4 maestros IO-Link SM....437
A.10.1.1 Especificaciones del módulo de señales SM 1278 4 maestros IO-Link....437
A.10.1.2 Diagramas de cableado del maestro SM 1278 4xIO-Link....439

A.11 Productos adicionales....441
A.11.1 Módulo de alimentación PM 1207 441
A.11.2 Módulo de conmutación compacto CSM 1277....441
A.11.3 Módulo CM CANopen....442

B Reemplazar una CPU V3.0 por una CPU V4.1 443

B.1 Reemplazar una CPU V3.0 por una CPU V4.1 443

Índice alfabético 449

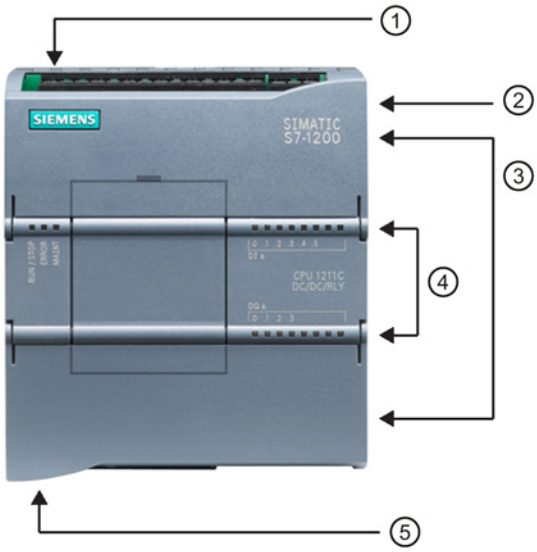
Presentamos el potente y flexible S7-1200 1

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, esta 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, RS232, IEC, DNP3 y WDC.



① Conector de corriente

② Ranura para Memory Card (debajo de la tapa superior)

③ Conectores extraíbles para el cableado de usuario (detrás de las tapas)

④ LEDs de estado para las E/S integradas

⑤ Conector PROFINET (en el lado inferior de la CPU)

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

- Todas las CPU ofrecen protección por contraseña (Página 91) que permite configurar el acceso a sus funciones.
- Es posible utilizar la "protección de know-how" (Página 94) para ocultar el código de un bloque específico.
- Además, la protección anticopia (Página 95) 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	CPU 1217C
Dimensiones físicas (mm)		90 x 100 x 75		110 x 100 x 75	130 x 100 x 75	150 x 100 x 75
Memoria de usuario	Trabajo	50 KB	75 KB	100 KB	125 KB	150 KB
	Carga	1 MB		4 MB		
	Remanente	10 KB				
E/S integradas locales	Digital	6 entradas/4 salidas	8 entradas/6 salidas	14 entradas/10 salidas		
	Analógico	2 entradas			2 entradas/2 salidas	
Tamaño de la memoria imagen de proceso	Entradas (I)	1024 bytes				
	Salidas (Q)	1024 bytes				
Área de marcas (M)		4096 bytes		8192 bytes		
Ampliación con módulo de señales (SM)		Ninguna	2	8		
Signal board (SB), Battery Board (BB) o Communication Board (CB)		1				
Módulo de comunicación (CM) (ampliación en el lado izquierdo)		3				
Contadores rápidos	Total	Se han configurado un máximo de 6 para usar cualquier entrada integrada o de SB.				
	1 MHz	-				De Ib.2 a Ib.5
	100/[1]80 kHz	De Ia.0 a Ia.5				
	30/[1]20 kHz	--	De Ia.6 a Ia.7	De Ia.6 a Ib.5		De Ia.6 a Ib.1
	200 kHz[3]					
Salidas de impulso [2]	Total	Se han configurado un máximo de 4 para usar cualquier salida integrada o de SB				
	1 MHz	--				De Qa.0 a Qa.3
	100 kHz	De Qa.0 a Qa.3				De Qa.4 a Qb.1
	20 kHz	--	De Qa.4 a Qa.5	De Qa.4 a Qb.		--
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)				
Puerto de comunicación Ethernet PROFINET		1			2	
Velocidad de ejecución de funciones matemáticas con números reales		2,3 µs/instrucción				
Velocidad de ejecución booleana		0,08 µs/instrucción				

1 La velocidad más lenta se aplica cuando el HSC está configurado para para el modo de operación en cuadratura.

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

3 Se dispone de hasta 200 kHz con la SB 1221 DI x 24 VDC 200 kHz y la SB 1221 DI 4 x 5 VDC 200 kHz.

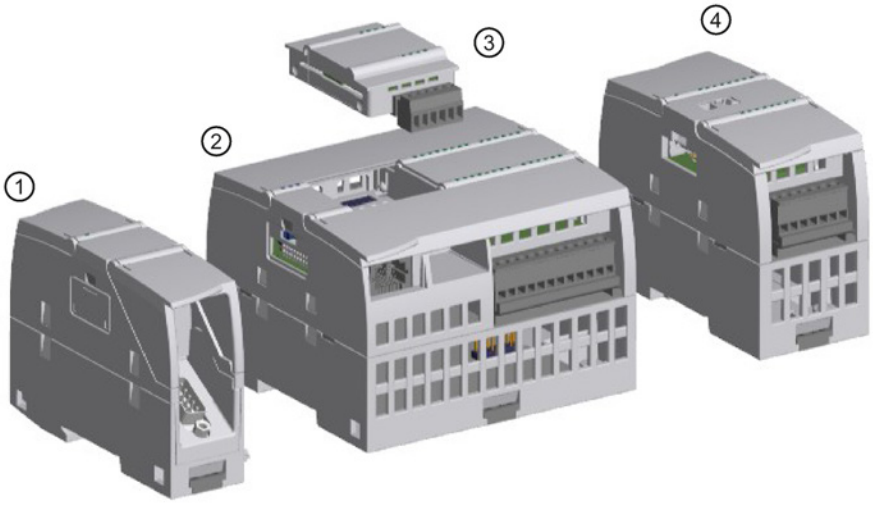
Los diferentes modelos de CPU ofrecen una gran variedad de funciones y prestaciones que permiten crear soluciones efectivas destinadas a numerosas aplicaciones. Para más información sobre una CPU en particular, consulte los datos técnicos (Página 371).

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

Elemento		Descripción
Bloques	Tipo	OB, FB, FC, DB
	Tamaño	50 KB (CPU 1211C) 75 KB (CPU 1212C) 100 KB (CPU 1214C) 125 KB (CPU 1215C) 150 KB (CPU 1217C)
	Cantidad	Un total de hasta 1024 bloques (OB + FB + FC + DB)
	Profundidad de anidamiento	16 desde el OB de ciclo o arranque; 6 desde cualquier OB de evento de alarma
	Observar	Se puede observar a la vez el estado de 2 bloques lógicos
OB	Ciclo del programa	Múltiple
	Arranque	Múltiple
	Alarmas de retardo	4 (1 por evento)
	Alarmas cíclicas	4 (1 por evento)
	Alarmas de proceso	50 (1 por evento)
	Alarmas de error de tiempo	1
	Alarmas de error de diagnóstico	1
	Inserción o extracción de módulos	1
	Fallo de rack o estación	1
	Hora	Múltiple
	Estado	1
	Actualizar	1
	Perfil	1
Temporizadores	Tipo	CEI
	Cantidad	Solo limitada por el tamaño de la memoria
	Almacenamiento	Estructura en DB, 16 bytes por temporizador
Contadores	Tipo	CEI
	Cantidad	Solo limitada por el tamaño de la memoria
	Almacenamiento	Estructura en DB, tamaño dependiente del tipo de contaje • SInt, USInt: 3 bytes • Int, UInt: 6 bytes • DInt, UDInt: 12 bytes

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 371).



① Módulo de comunicación (CM) o procesador de comunicaciones (CP)

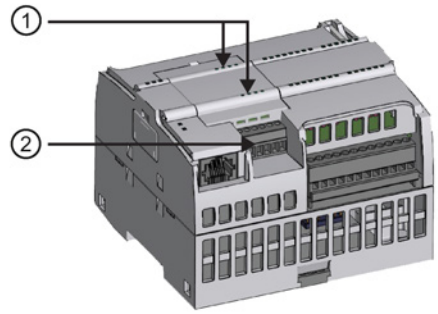
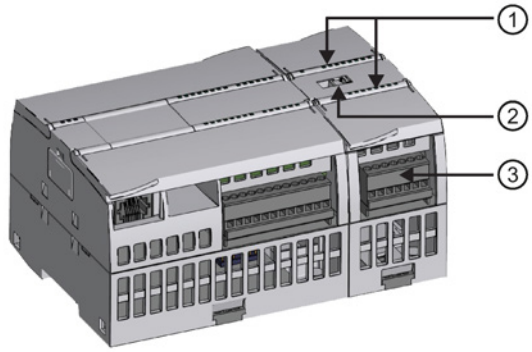
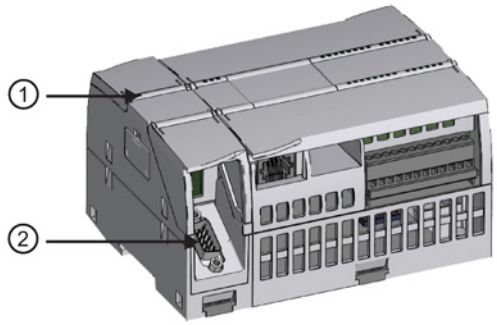
② CPU (CPU 1211C, CPU 1212C, CPU 1214C, CPU 1215C, CPU 1217C)

③ Signal Board (SB) (SB digital, SB analógica), Communication Board (CB) o Battery Board (BB)CPU (CPU 1211C, CPU 1212C, CPU 1214C, CPU 1215C, CPU 1217C)

④ Módulo de señales (SM) (SM digital, SM analógico, SM termopar, SM RTD, SM tecnológico)

1.3 Módulos S7-1200

Tabla 1- 3 Módulos de ampliación S7-1200

Tipo de módulo	Descripción
La CPU soporta una placa de ampliación tipo plug-in: • 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
	② Conector extraíble para el cableado de usuario
Los módulos de señales (SM) agregan funciones a la CPU. Los SM se conectan en el lado derecho de la CPU. • E/S digitales • E/S analógicas • RTD y termopar • Maestro SM 1278 IO-Link	
	① LEDs de estado
	② Lengüeta de desplazamiento del conector de bus
	③ Conector extraíble para el cableado de usuario
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, IEC, DNP3, o WDC. • La CPU soporta hasta tres CMs o CPs • Cada CM o CP se conecta en el lado	

Tipo de módulo	Descripción
izquierdo de la CPU (o en el lado izquierdo de otro CM o CP)	① LEDs de estado
	② Conector de comunicación

1.4 Paneles HMI Basic

Los SIMATIC HMI Basic Panels incorporan pantalla táctil para el control básico por parte del operador y tareas de control. Todos los paneles ofrecen el grado de protección IP65 y certificación CE, UL, cULus y NEMA 4x.

Los paneles HMI Basic disponibles se describen a continuación:

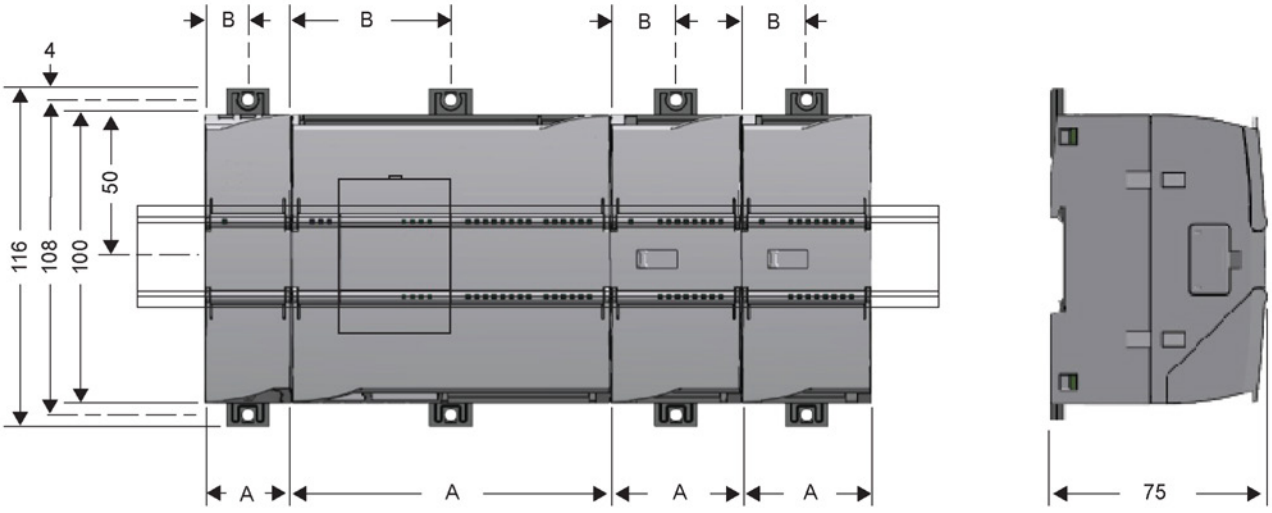
- KTP400 Basic: pantalla táctil de 4 pulgadas con 4 teclas configurables, resolución de 480 x 272 y 800 variables
- KTP700 Basic: pantalla táctil de 7 pulgadas con 8 teclas configurables, resolución de 800 x 480 y 800 variables
- KTP700 Basic DP: pantalla táctil de 7 pulgadas con 8 teclas configurables, resolución de 800 x 480 y 800 variables
- KTP900 Basic: pantalla táctil de 9 pulgadas con 8 teclas configurables, resolución de 800 x 480 y 800 variables
- KTP1200 Basic: pantalla táctil de 12 pulgadas con 10 teclas configurables, resolución de 800 x 480 y 800 variables
- KTP 1200 Basic DP: pantalla táctil de 12 pulgadas con 10 teclas configurables, resolución de 800 x 400 y 800 variables

1.5 Dimensiones de montaje y espacio libre necesario

El PLC S7-1200 ha sido diseñado para un fácil montaje. Tanto montado sobre un panel como sobre un perfil DIN normalizado, su tamaño compacto permite optimizar el espacio.

Consulte el *Manual de sistema del controlador programable S7-1200* para conocer los requisitos específicos y las instrucciones de montaje.

CPU 1211C, CPU 1212C, CPU 1214C
(measurements in mm)



CPU 1215C, CPU 1217C

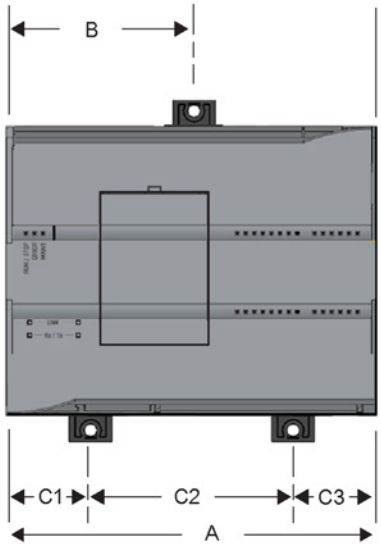


Tabla 1- 4 Dimensiones de montaje (mm)

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
	CPU 1217C	150	75	Parte inferior: C1: 37.5 C2: 75 C3: 37.5
Módulos de señales	Digitales 8 y 16 puntos Analógicas 2, 4 y 8 puntos Termopar 4 y 8 puntos RTD 4 puntos Maestro SM 1278 IO-Link	45	22.5	--
	Digital DQ 8 x Relay (inversor)	70	35	--
	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 V2 CP 1243-7 LTE-EU CP 1243-1 DNP3 CP 1243-1 IEC CP 1243-1 CP1243-1 PCC CP 1243-8 ST7 RF120C	30	15	--
	TS (TeleService) Adapter IE Advanced [1] TS (TeleService) Adapter IE Basic [1]			
	TS Adapter	30	15	--
	Módulo TS	30	15	--

[1] Antes de instalar el TS (TeleService) Adapter IE Advanced o IE Basic hay que conectar el TS Adapter y un módulo TS. 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.

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

Las CPU S7-1200 de seguridad no soportan E/S de seguridad descentralizadas PROFIBUS ni PROFINET.

Los estándares de equipos eléctricos clasifican el sistema SIMATIC S7-1200 como Equipo abierto. El S7-1200 debe montarse en una carcasa, un armario eléctrico o una sala de control. El acceso a la carcasa, el armario eléctrico o la sala de control debe limitarse a personal autorizado.

El S7-1200 debe instalarse en un entorno seco. Los circuitos SELV/PELV están diseñados para proporcionar protección contra choque eléctrico en entornos secos.

La instalación debe proporcionar protección mecánica y ambiental homologada para equipos abiertos en la categoría específica del lugar de instalación conforme a los reglamentos eléctricos y de edificaciones vigentes.

La contaminación conductora debida a polvo, humedad y polución aérea puede causar fallos funcionales y eléctricos en el PLC.

Si el PLC se ubica en una zona en la que puede haber contaminación conductora, habrá que proteger el PLC con una envolvente con grado de protección adecuado. IP54 es un grado utilizado generalmente para envolventes de equipos eléctricos en entornos sucios, y puede ser adecuado en su caso.

⚠ ADVERTENCIA

El montaje incorrecto del S7-1200 puede provocar fallos eléctricos o un funcionamiento inesperado de las máquinas.

Los fallos eléctricos y el funcionamiento inesperado pueden derivar en muerte, lesiones corporales graves y/o daños materiales.

Para garantizar que el equipo funciona de forma segura hay que seguir todas las instrucciones de montaje, así como del mantenimiento de un entorno de funcionamiento correcto.

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.

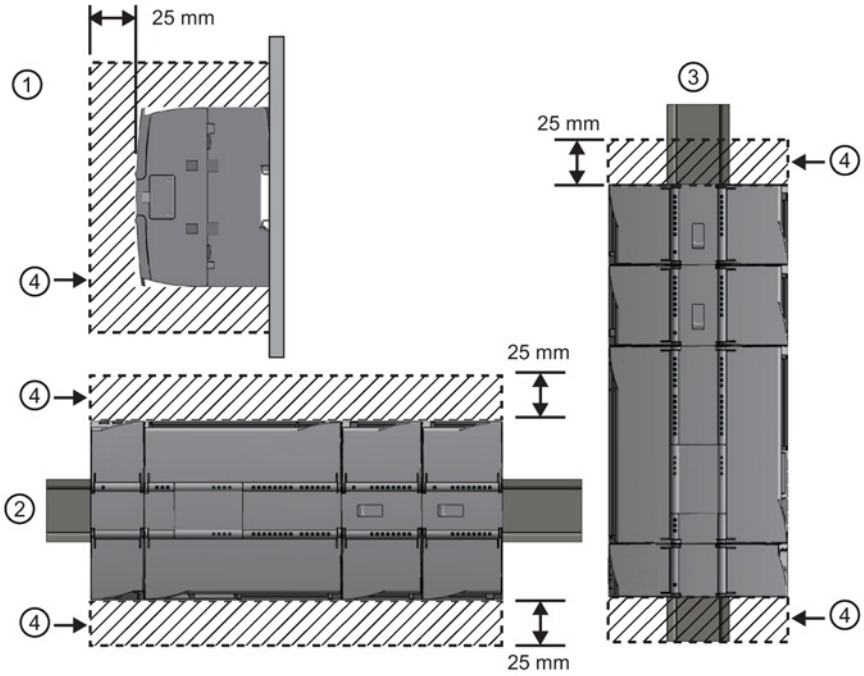
También se debe considerar la ruta del cableado de los dispositivos montados en el panel. Evite tender las líneas de señales de baja tensión y los cables de comunicación en un mismo canal junto con los cables AC y DC de alta energía y conmutación rápida.

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
En el montaje vertical, la temperatura ambiente máxima admisible se reduce en 10 grados Celsius. Las S7-1200 montadas en vertical deben orientarse de la forma mostrada en la figura. Asegúrese de que el sistema S7-1200 se ha montado correctamente.

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



① Vista lateral
② Montaje horizontal
③ Montaje vertical
④ Espacio libre

⚠ 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 o daños materiales debidos a choques eléctricos o al funcionamiento inesperado de los equipos.

Respete siempre las medidas de seguridad necesarias y asegúrese de 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 un 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 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.

1.6 Nuevas funciones

La presente versión ofrece las siguientes nuevas funciones:

- Ahora es posible implementar la seguridad funcional empleando el hardware y firmware de las CPU de seguridad positiva S7-1200 y los módulos de señales (SM) en combinación con el programa de seguridad descargado por el software (ES). Para más información, consulte el manual S7-1200 Functional Safety (http://support.automation.siemens.com/WW/view/en/104547552).
- Simulación de CPU S7-1200 con versión de firmware V4.0 y superior: S7-PLCSIM V13 SP1 permite probar los programas del PLC en un PLC simulado sin necesidad de hardware real. S7-PLCSIM es una aplicación instalada por separado que funciona en combinación con STEP 7 en el TIA Portal. Permite configurar el PLC y cualquier módulo conectado en STEP 7, programar la lógica de la aplicación y, a continuación, descargar la configuración hardware y el programa al S7-PLCSIM. Seguidamente, las herramientas de S7-PLCSIM permiten simular y probar el programa. Para más información consulte la Ayuda en pantalla de S7-PLCSIM. Tenga en cuenta que no es posible simular CPU de seguridad.
- Control de la configuración (configuración futura) (Página 83): es posible configurar el hardware para una configuración máxima de la máquina, incluyendo módulos que posiblemente no se utilicen durante la operación. La configuración y designación de estos módulos flexibles es nueva en esta versión de STEP 7 y S7-1200.Los módulos que se designen de este modo no crearán condiciones de error en caso de estar ausentes.
- El servidor web (Página 261) soporta ahora el acceso a través de la dirección IP de los módulos (procesador de comunicaciones) seleccionados en el rack local así como a través de la dirección IP de la CPU S7-1200.
- Funciones de movimiento mejoradas:
  - Conexiones analógicas y PROFIdrive
  - Parámetros ampliados del módulo y del lazo de regulación
- Medición de períodos mediante Contadores rápidos (HSC) (Página 136)
- Mejoras de rendimiento del compilador SLC

- Protección anticopia (Página 95) dinámica enlazada a bloques de programa con una contraseña obligatoria
- Funcionalidad PROFINET mejorada, incluido el soporte de shared devices (dispositivos compartidos).
- Nuevas instrucciones de programación:
  - EQ_Type, NE_Type, EQ_ElemType, NE_ElemType
  - IS_NULL, NOT_NULL
  - IS_ARRAY
  - Deserializar, Serializar
  - VariantGet, VariantPut, CountOfElements
  - Variant_to_DB_Any, DB_Any_To_Variant
  - GET_IM_DATA
  - RUNTIME
  - GEO2LOG, IO2MOD
  - ReadLittle, WriteLittle, ReadBig, WriteBig (solo SCL)
  - T_RESET, T_DIAG y TMAIL_C
  - PID_Temp
  - Instrucciones Modbus (Página 196) nuevas
  - Instrucciones punto a punto (PtP) (Página 193) nuevas
  - Instrucciones USS (Página 194) nuevas

Nuevos módulos para S7-1200

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

- Módulos de comunicación industriales para control remoto: estos CP se pueden utilizar como módulos de comunicaciones con la CPU S7-1200 V4.1.
- CPU y E/S de seguridad: hay cuatro CPU de seguridad y tres módulos de señales (SM) de seguridad en combinación con S7-1200 V4.1 o versiones posteriores:
  - CPU 1214FC DC/DC/DC (6ES7 214-1AF40-0XB0)
  - CPU 1214FC DC/DC/RLY (6ES7 214-1HF40-0XB0)
  - CPU 1215FC DC/DC/DC (6ES7 215-1AF40-0XB0)
  - CPU 1215FC DC/DC/RLY (6ES7 215-1HF40-0XB0)
  - SM 1226 F-DI 16 x 24 VDC (6ES7 226-6BA32-0XB0)
  - SM 1226 F-DQ 4 x 24 VDC (6ES7 226-6DA32-0XB0)
  - SM 1226 F-DQ 2 x Relay (6ES7 226-6RA32-0XB0)

  Existe la posibilidad de emplear los módulos de señales (SM), módulos de comunicación (CM) y Signal Boards (SB) estándar de S7-1200 en el mismo sistema con SMs de seguridad positiva para completar las funciones de control de aplicaciones que no requieran una clasificación de seguridad funcional. Los SMs estándar aptos para utilizar con SMs de seguridad positiva tienen las referencias (6ES7 --- ---32 0XB0) o posteriores.

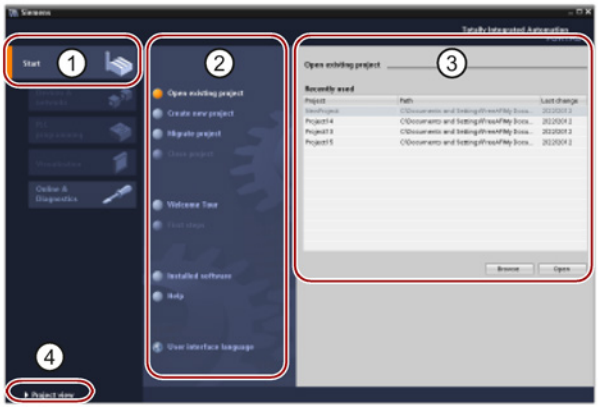
Reemplazar una CPU V3.0 por una CPU V4.1

Si se reemplaza una CPU S7-1200 V3.0 por una CPU S7-1200 V4.1, tenga en cuenta las diferencias (Página 443) documentadas entre ambas versiones y las acciones que debe realizar el usuario.

STEP 7 facilita el trabajo

2

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.



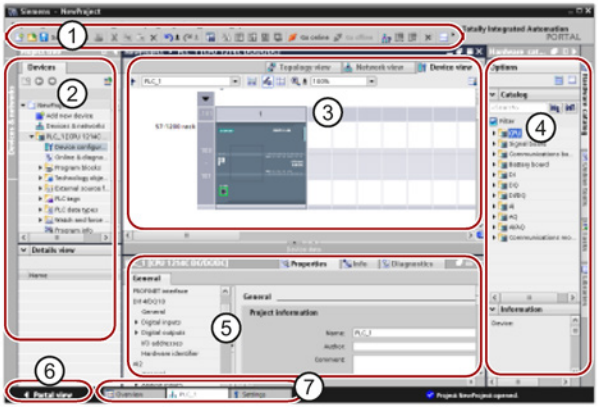
Vista del portal

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

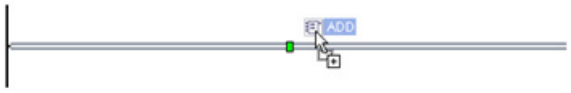
⑦ Barra del editor

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 muestras 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.

2.1 Fácil entrada de instrucciones en el programa de usuario

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

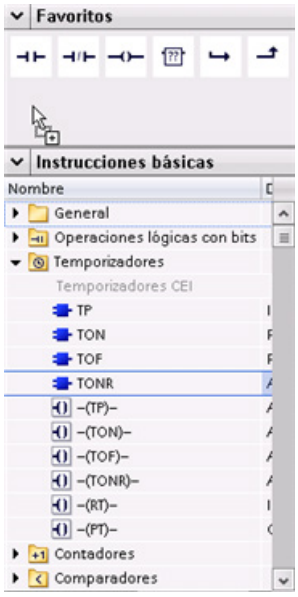


Para crear el programa, arrastre las instrucciones desde las Task Cards a los diferentes segmentos mediante Drag & Drop.

2.2 Fácil acceso a las instrucciones más utilizadas desde la barra de herramientas

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.)



Los "Favoritos" pueden personalizarse fácilmente agregando nuevas instrucciones.

Para ello sólo hay que mover la instrucción a "Favoritos" mediante Drag & Drop.

La instrucción ya está al alcance de un clic.

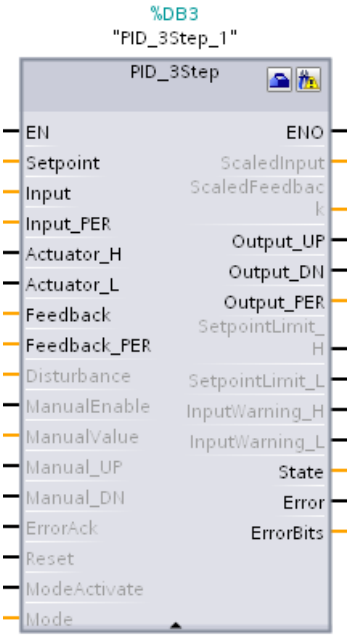
2.3 Facilidad para agregar entradas o salidas a instrucciones KOP y FUP

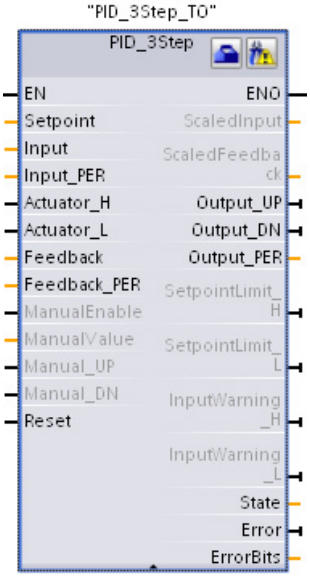
Algunas de las instrucciones permiten crear entradas o salidas adicionales.

- 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".

2.4 Instrucciones ampliables

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





2.5 Facilidad de cambio del modo de operación de la CPU

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 (Página 84).

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 estado operativo de la CPU. El panel también permite cambiar el estado operativo de la CPU.

Utilice el botón del panel de operador para cambiar el estado operativo (STOP o RUN). El panel de operador también dispone de un botón MRES para resetear la memoria.

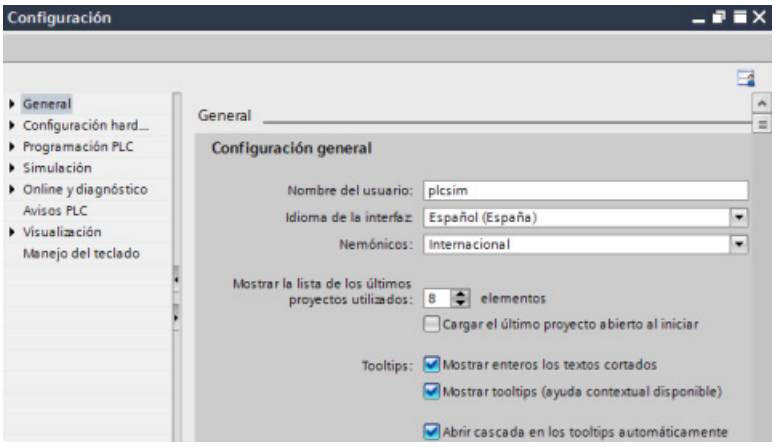
El color del indicador RUN/STOP muestra el estado operativo actual de la CPU. El amarillo indica el estado operativo STOP y el verde RUN.

Desde la configuración de dispositivos en STEP 7 también se puede configurar el modo de operación predeterminado al poner en marcha la CPU.

2.6 Facilidad de modificación de la apariencia y configuración de STEP 7

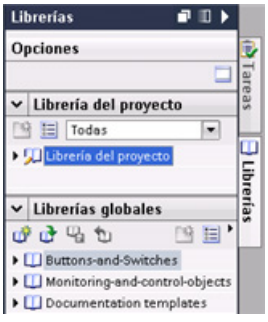
Es posible seleccionar una serie de ajustes, como p. ej. la apariencia de la interfaz, el idioma o la carpeta en la que guardar el trabajo.

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



2.7 Librerías de proyecto y globales para un acceso fácil

Las librerías globales y de proyecto permiten reutilizar los objetos guardados dentro de un proyecto o en varios proyectos. Así, es posible crear, por ejemplo plantillas de bloques para utilizar en diferentes proyectos y adaptarlas a las necesidades particulares de la tarea de automatización. En las librerías pueden guardarse diferentes objetos, como FCs, FBs, DBs, configuración de dispositivos, tipos de datos, tablas de monitorización, sinópticos de proceso y faceplates. También es posible guardar los componentes de los dispositivos HMI en el proyecto.

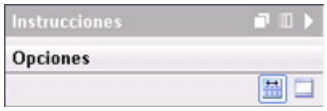


Cada proyecto dispone de una librería de proyecto para guardar los objetos que se utilizan en él más de una vez. La librería del proyecto forma parte del proyecto. Al abrir o cerrar el proyecto se abre o cierra la librería, y al guardar el proyecto se guardan todos los cambios en la librería del proyecto.

Es posible crear una librería global personalizada para guardar los objetos que se desee para que estén disponibles para otros proyectos. Al crear una librería global nueva, esta se guarda en una ubicación del equipo o en una unidad de red.

2.8 Facilidad para seleccionar la versión de una instrucción

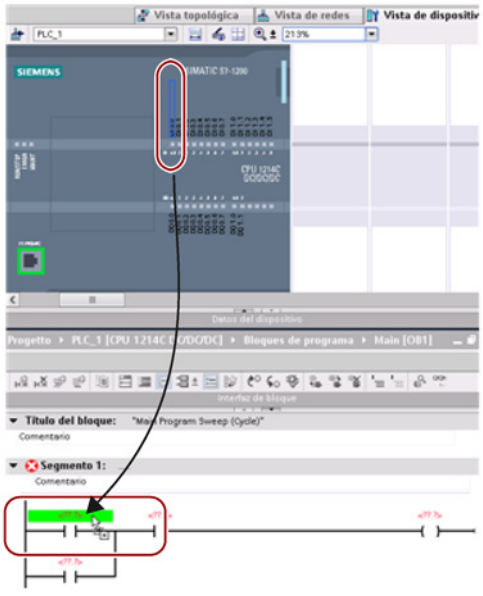
Los ciclos de desarrollo y lanzamiento de determinados conjuntos de instrucciones (como Modbus, PID y de movimiento) han dado lugar a múltiples versiones de dichas instrucciones. Para ayudar a garantizar la compatibilidad y la migración de proyectos más antiguos, STEP 7 permite seleccionar la versión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

2.9 La función de Drag & Drop puede utilizarse fácilmente entre editores

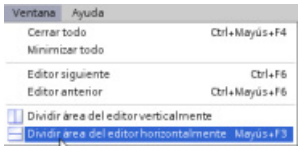


Para ayudar a realizar las tareas rápida y fácilmente, STEP 7 permite arrastrar y soltar elementos de un editor a otro. Así, es posible, por ejemplo, arrastrar una entrada de la CPU a la dirección de una instrucción del programa de usuario.

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

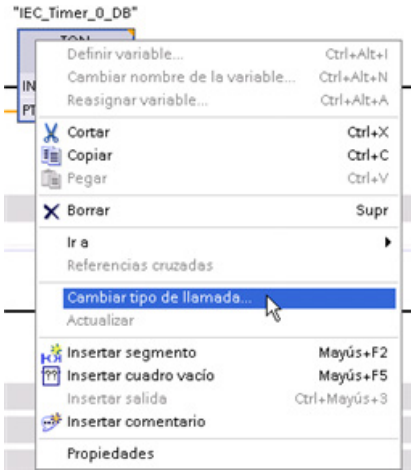
Recuerde que los nombres de las variables no solo se muestran en la tabla de variables PLC, sino también en la CPU.

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.

2.10 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.

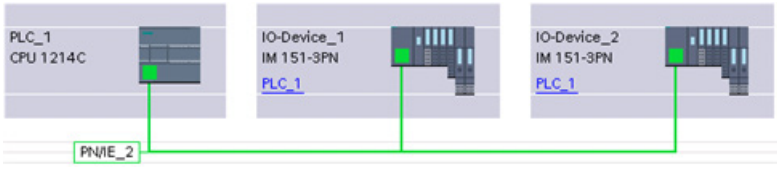
- Es posible conmutar la asignación entre diferentes DBs.
- Es posible conmutar la asignación entre un DB monoinstancia y un DB multiinstancia.
- Es posible crear un DB de instancia (si falta o no está disponible).

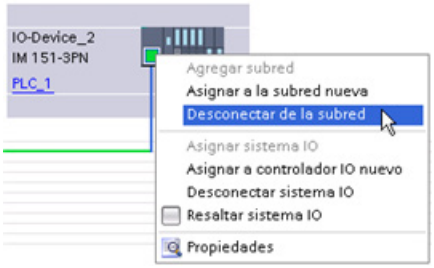
Al comando "Modificar tipo de llamada" se accede haciendo clic con el botón derecho del ratón en la instrucción o FB en el editor de programas o bien seleccionando el comando "Llamada de bloque" del menú "Opciones".

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.

2.11 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.

2.12 Facilidad de "desenchufar" virtualmente módulos sin perder 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.

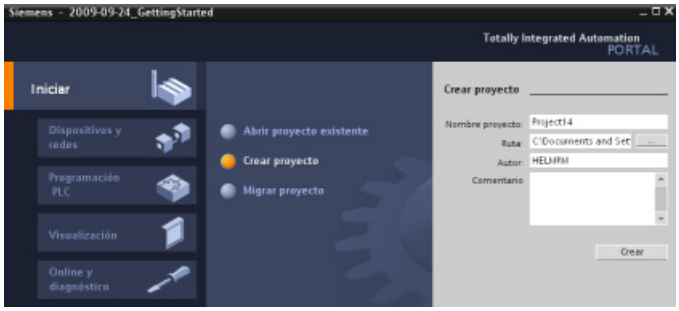
Uno de los usos de esta función corresponde al mantenimiento temporal. Consideremos una situación en la que se está esperando un módulo de recambio y se planifica un uso temporal de un módulo diferente a modo de sustituto provisional. Es posible arrastrar el módulo configurado desde el rack a los "módulos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

Getting Started

3

3.1 Crear un proyecto

¡Trabajar con STEP 7 es muy fácil! Aprender a crear un proyecto resulta rápido y sencillo.

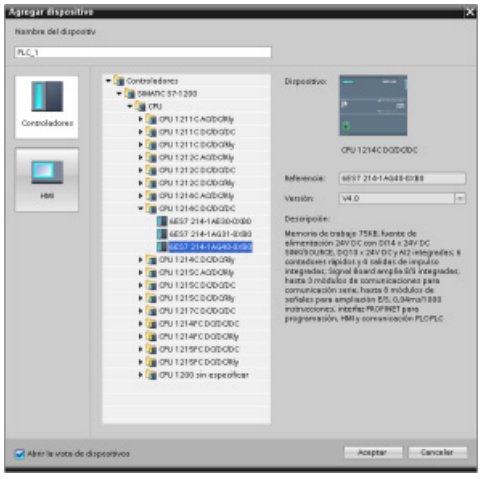


En el portal Inicio, haga clic en "Crear proyecto nuevo".

Introduzca el nombre del proyecto y haga clic en el botón "Crear".

Una vez creado el proyecto, seleccione el portal Dispositivos y redes.

Haga clic en el botón "Agregar nuevo dispositivo".



Seleccione la CPU que desea agregar al proyecto:

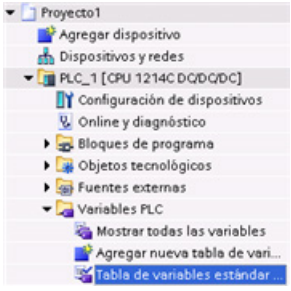
1. En el cuadro de diálogo "Agregar nuevo dispositivo", haga clic en el botón "SIMATIC PLC".
2. Seleccione una CPU de la lista.
3. Para agregar la CPU seleccionada al proyecto, haga clic en el botón "Agregar".

Observe que la opción "Abrir la vista de dispositivos" está seleccionada. Al hacer clic en "Agregar" con esta opción seleccionada se abre la "Configuración de dispositivos" de la vista del proyecto.

La vista de dispositivos muestra la CPU agregada.

3.2 Crear variables para las E/S de la CPU

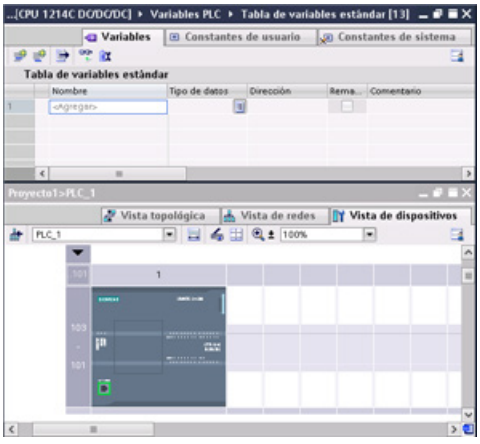
Las "variables PLC" son nombres simbólicos para entradas y salidas y para direcciones. Tras crear una variable PLC, STEP 7 la guarda en una tabla de variables. Todos los editores del proyecto (ya sea el editor de programación, el de dispositivos, el de visualización o el de la tabla de observación) pueden acceder a la tabla de variables.



Con el editor de dispositivos abierto, abra una tabla de variables.

Los editores abiertos pueden verse en la barra de editores.

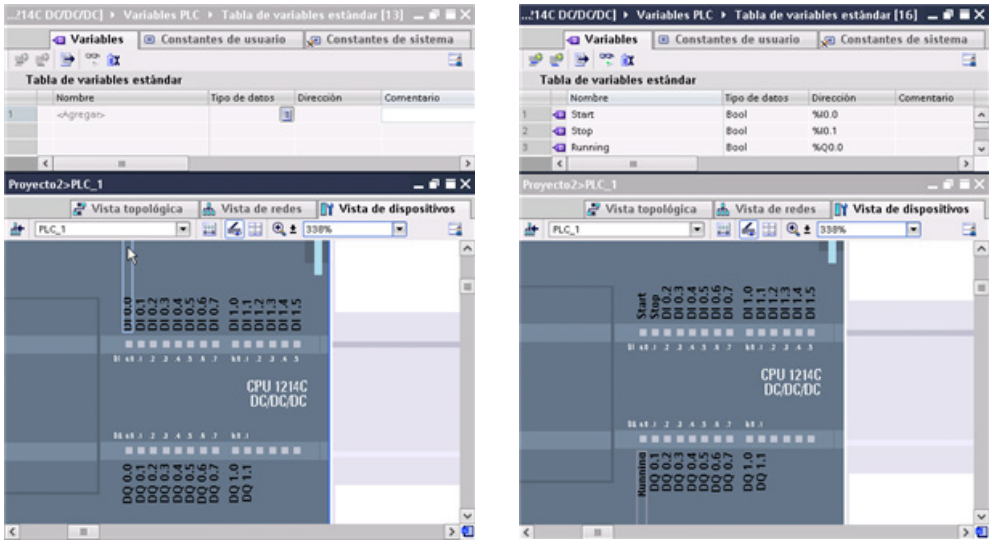
En la barra de herramientas, haga clic en el botón "Dividir el área del editor horizontalmente".



STEP 7 muestra simultáneamente la tabla de variables y el editor de dispositivos.

Aumente el zoom de la configuración de dispositivos al 200% para poder leer y seleccionar las E/S. Arrastre las entradas y salidas de la CPU a la tabla de variables:

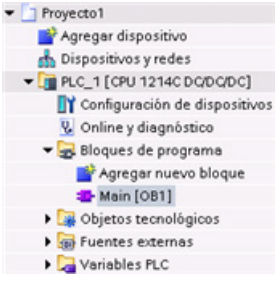
1. Seleccione la entrada I0.0 y arrástrela a la primera fila de la tabla de variables.
2. Cambie el nombre de la variable de "I0.0" a "Start".
3. Arrastre la entrada I0.1 a la tabla de variables y cambie el nombre a "Stop".
4. Arrastre Q0.0 (en la parte inferior de la CPU) a la tabla de variables y cambie el nombre a "Running".



Una vez que las variables se han introducido en la tabla de variables PLC, están disponibles para el programa de usuario.

3.3 Crear un segmento simple en el programa de usuario

El código del programa consiste en instrucciones que ejecuta la CPU siguiendo una secuencia. Para este ejemplo utilizamos la lógica de Esquema de contactos (KOP) para crear la lógica del programa. El programa KOP es una secuencia de segmentos semejantes a los peldaños de una escalera.

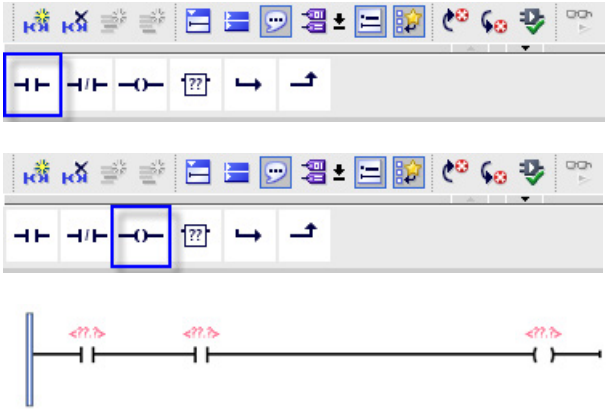


Para abrir el editor de programación, proceda del siguiente modo:

1. Abra la carpeta "Bloques de programa" en el árbol del proyecto para ver el bloque "Principal [OB1]".
2. Haga doble clic en el bloque "Principal [OB1]".

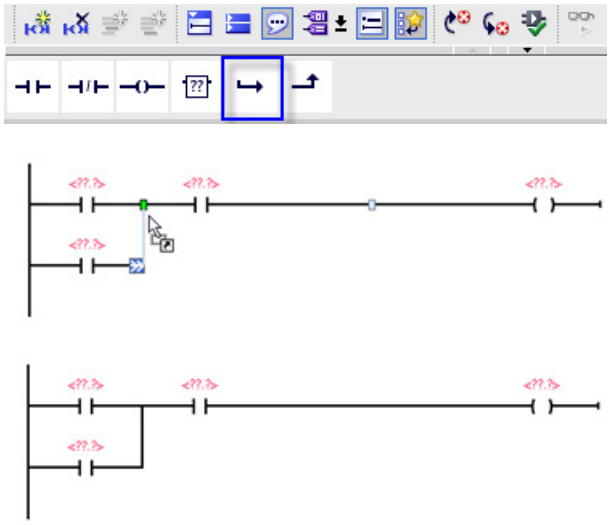
El editor de programación abre el bloque de programa (OB1).

Utilice los botones de "Favoritos" para insertar contactos y bobinas al segmento.



1. Haga clic en el botón "Contacto normalmente abierto" de "Favoritos" para agregar un contacto al segmento.
2. Para este ejemplo, agregue un segundo contacto.
3. Haga clic en el cotón "Bobina de relé" para insertar una bobina.

Los "Favoritos" también incluyen un botón para crear una rama.

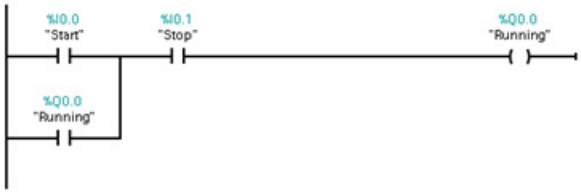


1. Seleccione el perfil izquierdo para seleccionar el perfil de la rama.
2. Haga clic en el botón "Abrir rama" para agregar una rama al perfil del segmento.
3. Inserte otro contacto normalmente abierto en la rama abierta.
4. Arrastre la flecha de doble punta hasta un punto de conexión (el cuadrado verde del circuito) entre los dos contactos del primer circuito.

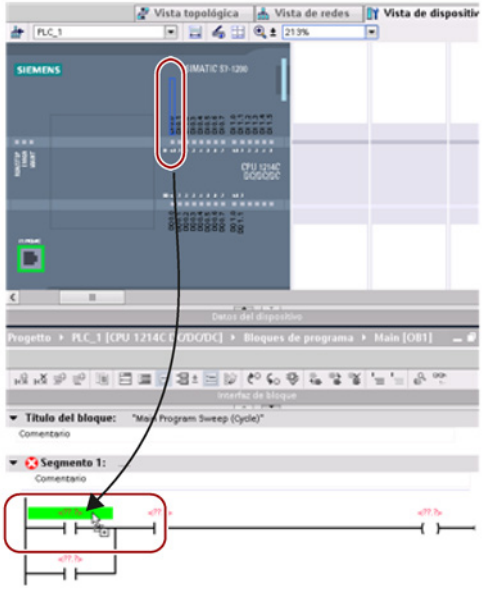
Para guardar el proyecto, haga clic en el botón "Guardar proyecto" de la barra de herramientas. Recuerde que no es necesario terminar de editar el circuito antes de guardarlo. Ahora se pueden asignar los nombres de variables a las diferentes instrucciones.

3.4 Utilice las variables PLC de la tabla de variables para direccionar las instrucciones

La tabla de variables permite introducir las variables PLC para las direcciones de los contactos y bobinas.



1. Haga doble clic en la dirección predeterminada <*??.?*> situada sobre el primer contacto normalmente abierto.
2. Haga clic en el botón selector situado a la derecha de la dirección para abrir las variables de la tabla.
3. En la lista desplegable, seleccione "Start" para el primer contacto.
4. Para el segundo contacto, repita los pasos anteriores y seleccione la variable "Stop".
5. Para la bobina y el contacto de enclavamiento, seleccione la variable "Running".



También puede arrastrar las direcciones de E/S directamente desde la CPU. Sencillamente, divida el área de trabajo de la vista del proyecto (Página 38).

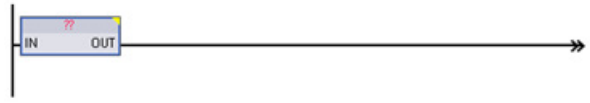
Es necesario ampliar el zoom de la CPU como mínimo al 200% para poder seleccionar las E/S.

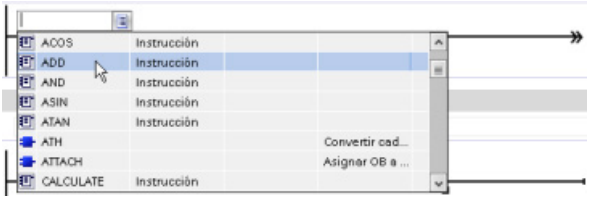
Es posible arrastrar las E/S de la CPU de la "Configuración de dispositivos" a la instrucción KOP del editor de programación, no solo para crear la dirección de la instrucción, sino también la entrada en la tabla de variables PLC.

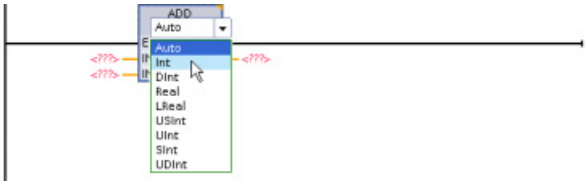
3.5 Adición de una instrucción "cuadro"

El editor de programación dispone de una instrucción "cuadro" genérica. Tras insertar dicha instrucción, se selecciona el tipo de instrucción, como p. e. una instrucción ADD, desde una lista desplegable.

Haga clic en la instrucción genérica "cuadro" en la barra de herramientas "Favoritos".

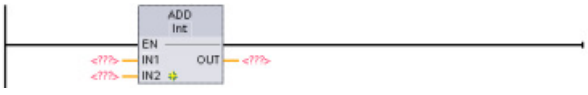




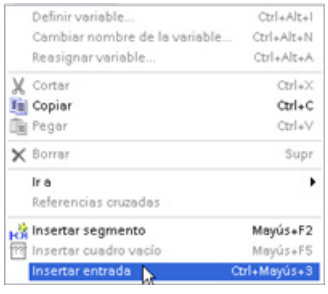


La instrucción genérica "cuadro" soporta varias instrucciones. Para este ejemplo se crea una instrucción ADD:

1. Haga clic en la esquina amarilla de la instrucción cuadro para visualizar la lista desplegable correspondiente.
2. Avance por la lista hacia abajo y seleccione la instrucción ADD.
3. Haga clic en la esquina amarilla junto a "?" para seleccionar el tipo de datos para las salidas y la entrada.



Ahora puede introducir las variables (o direcciones de memoria) que deben utilizar los valores con la instrucción ADD.



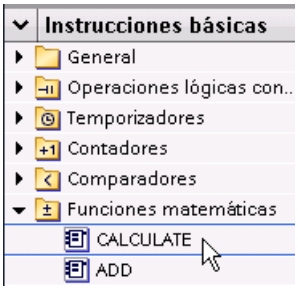
También puede crear entradas adicionales para determinadas instrucciones:

1. Haga clic en una de las entradas del cuadro.
2. Haga clic con el botón derecho para visualizar el menú contextual y seleccionar el comando "Insertar entrada".

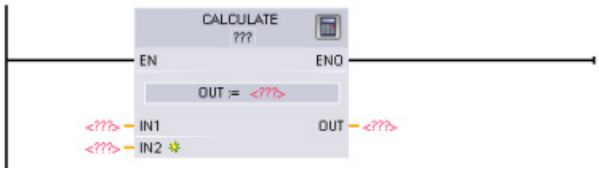
Ahora la instrucción ADD utiliza tres entradas.

3.6 Uso de la instrucción CALCULAR para una ecuación matemática compleja

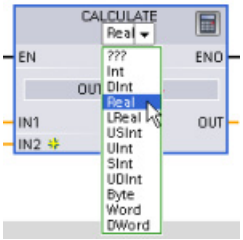
La instrucción Calcular (Página 118) permite crear una función matemática que se ejecuta con múltiples parámetros de entrada para obtener el resultado en función de la ecuación definida.



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.



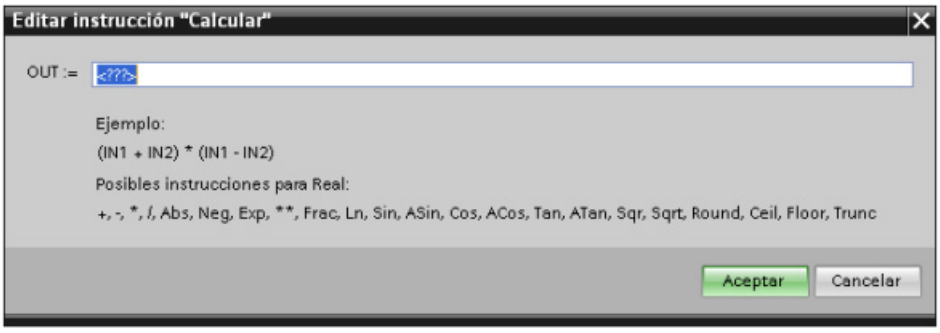
La instrucción Calculate no configurada ofrece dos parámetros de entrada y uno de salida.



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.)

$Out_{value} = ((Out_{high} - Out_{low}) / (In_{high} - In_{low})) * (In_{value} - In_{low}) + Out_{low}$

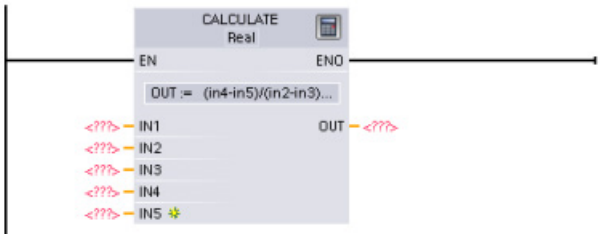
Out = ((in4 - in5) / (in2 - in3)) * (in1 - in3) + in5

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

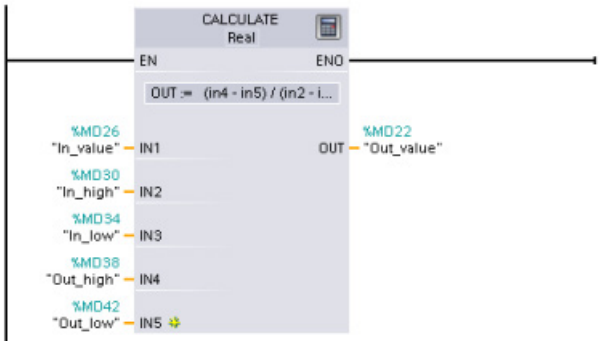
En el cuadro "Editar cálculo", introduzca la ecuación con los nombres de los parámetros:

OUT = ((in4 - in5) / (in2 - in3)) * (in1 - in3) + in5

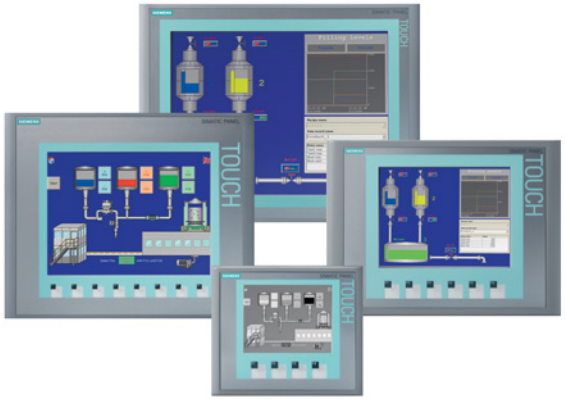
Al hacer clic en "Aceptar", la instrucción Calculate crea las entradas necesarias para la instrucción.



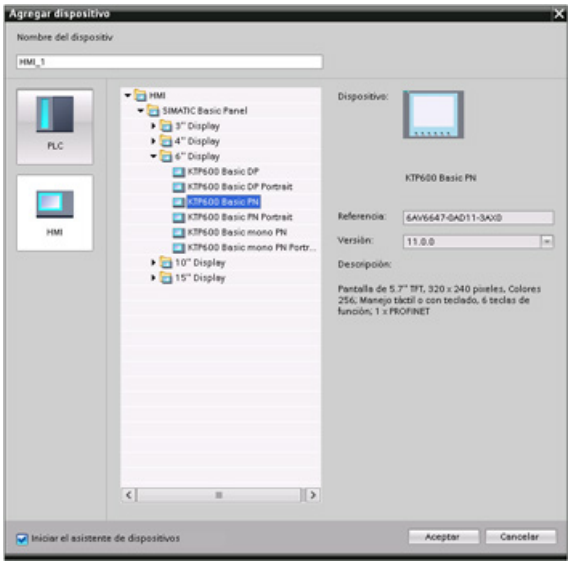
Introduzca los nombres de las variables de los valores que corresponden a los parámetros.



3.7 Agregar un dispositivo HMI al proyecto

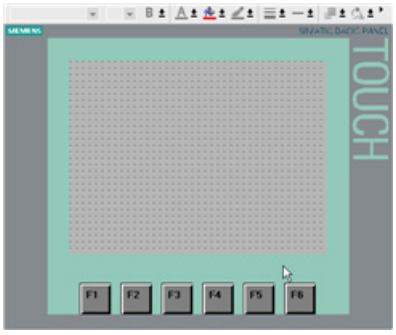


Agregar un dispositivo HMI al proyecto es muy fácil



1. Haga doble clic en el botón "Agregar nuevo dispositivo".
2. Haga clic en el botón "SIMATIC HMI" del cuadro de diálogo "Agregar nuevo dispositivo".
3. Seleccione el dispositivo HMI específico de la lista.

   Se puede optar por abrir el asistente para HMI para configurar los sinópticos del dispositivo HMI.
4. Haga clic en "Aceptar" para agregar el dispositivo HMI al proyecto.

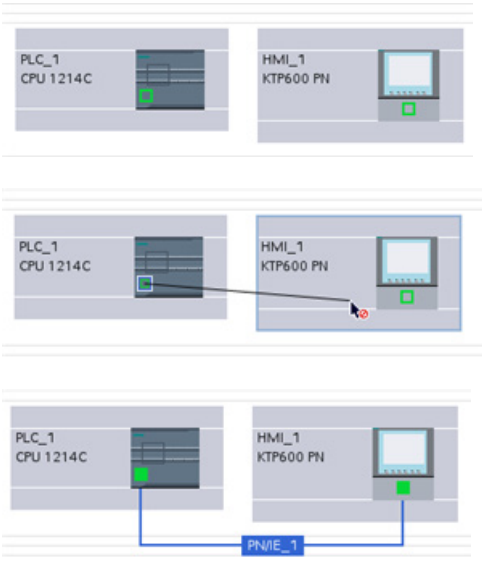


El TIA Portal agrega el dispositivo HMI al proyecto.

El TIA Portal proporciona un asistente para HMI que ayuda a configurar todas las pantallas y la estructura del dispositivo HMI.

Si no se ejecuta el asistente para HMI, el TIA Portal crea una sencilla pantalla HMI predeterminada. Se pueden agregar posteriormente pantallas u objetos adicionales a las pantallas.

3.8 Creación de una conexión de red entre la CPU y el dispositivo HMI

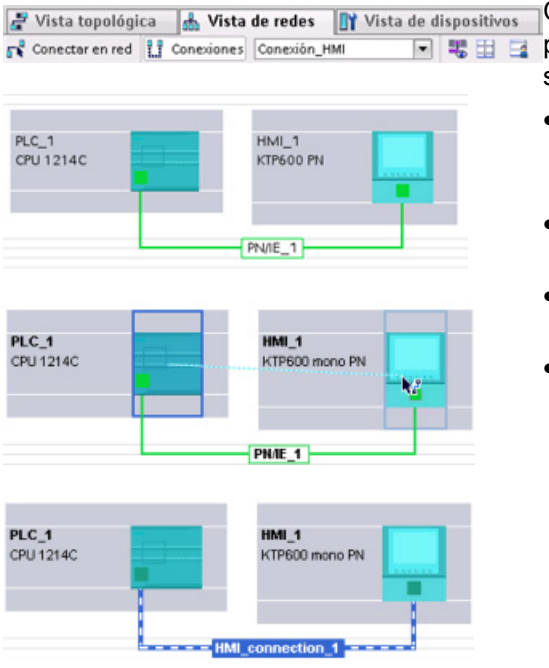


Crear una conexión de red es fácil

- Vaya a "Dispositivos y redes" y seleccione la vista de redes para visualizar la CPU y el dispositivo HMI.
- Para crear una red PROFINET, trace una línea desde el cuadro verde (puerto Ethernet) de un dispositivo hasta el cuadro verde del otro.

Se ha creado una conexión de red entre ambos dispositivos.

3.9 Creación de una conexión HMI para compartir variables



Creando una conexión HMI entre ambos dispositivos, estos pueden compartir fácilmente sus variables.

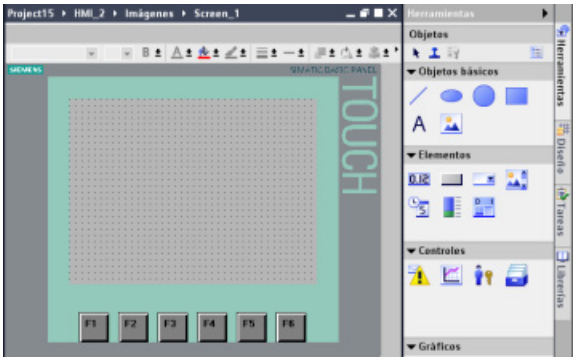
- Con la conexión de red seleccionada, haga clic en el botón "Conexiones" y seleccione "Conexión HMI" en la lista desplegable.
- La conexión HMI hace que ambos dispositivos se vuelvan azules.
- Seleccione la CPU y trace la línea hasta el dispositivo HMI.
- La conexión HMI permite configurar las variables HMI seleccionando una lista de variables PLC.

Hay otras opciones disponibles para crear una conexión HMI:

- Arrastrando una variable PLC desde la tabla de variables PLC, el editor de programas o el editor de configuración de dispositivos hasta el editor de la pantalla HMI se crea automáticamente una conexión HMI.
- Utilizando el asistente HMI para explorar el PLC se crea automáticamente una conexión HMI.

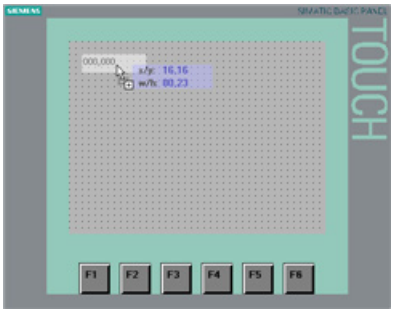
3.10 Creación de una pantalla HMI

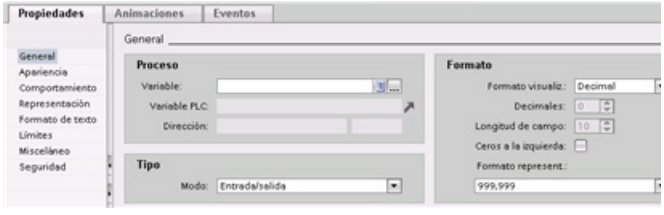
Aunque no se utilice el asistente HMI, configurar una pantalla HMI es muy fácil.



STEP 7 incorpora una serie predeterminada de librerías para la inserción de formas básicas, elementos interactivos e incluso gráficos estándar.

Para agregar un elemento basta con arrastrarlo a la pantalla mediante Drag & Drop. Utilice las propiedades del elemento (en la ventana de inspección) para configurar su apariencia y comportamiento.

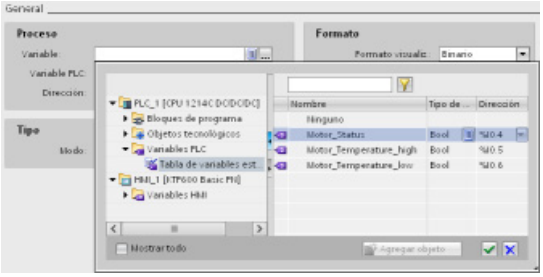




También es posible crear elementos en la pantalla arrastrando variables PLC desde el árbol del proyecto o el editor de programas hasta la pantalla HMI. La variable PLC se convierte en un elemento de la pantalla. Las propiedades pueden utilizarse entonces para cambiar los parámetros de este elemento.

3.11 Selección de una variable PLC para el elemento HMI

Tras haber creado el elemento en la pantalla, utilice las propiedades del elemento para asignarle una variable PLC. Haciendo clic en el botón de selección del campo de la variable para ver las variables PLC de la CPU.



También es posible arrastrar variables PLC desde el árbol del proyecto hasta la pantalla HMI. Muestre las variables PLC en las vista "Detalles" del árbol del proyecto y arrastre la variable deseada hasta la pantalla HMI.

Conceptos básicos de los PLC de la forma más sencilla

4

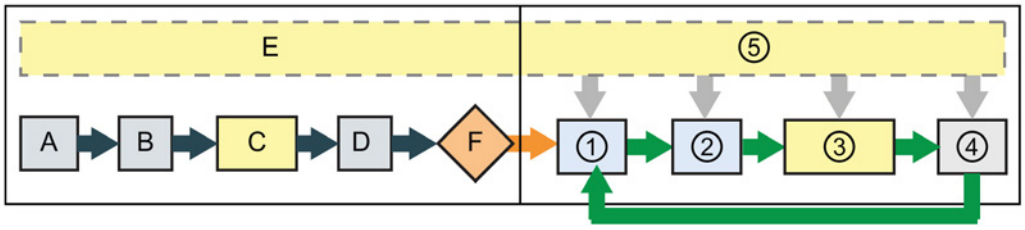
4.1 Las tareas se realizan en cada ciclo

En cada ciclo se escribe en las salidas, se leen las entradas, se ejecutan las instrucciones del programa de usuario y se realiza el mantenimiento del sistema o procesamiento en segundo plano.

En inglés, el ciclo también se llama "scan cycle" o "scan". En condiciones estándar, todas las E/S digitales y analógicas se actualizan de forma síncrona 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 de la CPU, de la Signal Board y de los módulos de señales.

- 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.
- Tras ejecutar el programa de usuario, la CPU escribe las salidas resultantes de la memoria imagen de proceso de las salidas en las salidas físicas.

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.



ARRANQUE

- A Borra el área de memoria (imagen) I
- B Inicializa el área de memoria (imagen) Q con cero, con el último valor o con el valor sustitutivo según la configuración, y pone a cero las salidas PB, PN y AS-i
- C Inicializa la memoria M no remanente y los bloques de datos a su valor inicial y habilita eventos configurados de alarma cíclica y de hora.

  Ejecuta los OB de arranque.
- D Copia el estado de las entradas físicas en la memoria I
- E Almacena los eventos de alarma en la cola de espera para procesarlos tras pasar 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 OB de ciclo
- ④ Ejecuta autodiagnósticos
- ⑤ Procesa alarmas y comunicaciones en cualquier parte del ciclo

La reacción predeterminada de un módulo se puede cambiar excluyéndolo de la actualización automática de E/S. También es posible leer y escribir inmediatamente valores de E/S digitales y analógicas en los módulos cuando se ejecuta una instrucción. La lectura inmediata de las entradas físicas no actualiza la memoria imagen de proceso de las entradas. La escritura inmediata en las salidas físicas actualiza tanto la memoria imagen de proceso de las salidas como las salidas físicas.

4.2 Estados operativos de la CPU

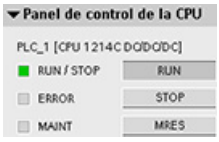
La CPU tiene tres estados operativos, a saber: STOP, ARRANQUE y RUN. Los LEDs de estado en el frente de la CPU indican el estado operativo actual.

- En estado operativo STOP, la CPU no ejecuta el programa. Entonces es posible cargar un proyecto en la CPU. El LED RUN/STOP permanece encendido en amarillo.
- En el modo ARRANQUE, la CPU ejecuta cualquier lógica de arranque existente. La CPU no procesa eventos de alarma en el modo de arranque. El LED RUN/STOP parpadea alternando entre verde y amarillo.
- En el estado operativo RUN, el ciclo se ejecuta repetidamente. Es posible que se produzcan eventos de alarma que la CPU procesa en cualquier fase del ciclo del programa. Algunas partes de un proyecto pueden descargarse en estado RUN. El LED RUN/STOP permanece encendido en verde.

La CPU soporta el arranque en caliente para pasar a estado operativo RUN. El rearranque normal no incluye un borrado total, pero se puede ordenar desde STEP 7. Un borrado total borra toda la memoria de trabajo, las áreas de memoria remanentes y no remanentes, copia la memoria de carga en la memoria de trabajo y pone las salidas en la "Reacción a STOP de la CPU" configurada. El borrado total no borra el búfer de diagnóstico ni tampoco la dirección IP almacenada permanentemente. Un rearranque normal inicializa todos los datos de usuario y de sistema no remanentes.

Se puede configurar el ajuste "Arranque tras conexión (POWER ON)" de toda la CPU con método de rearranque mediante STEP 7. Este elemento de configuración aparece en la Configuración de dispositivos para la CPU, en Arranque. Con el arranque, la CPU ejecuta una secuencia de tests de diagnóstico de arranque e inicialización del sistema. Durante la inicialización del sistema, la CPU borra toda el área de marcas no remanente e inicializa todos los contenidos DB no remanentes a valores iniciales. A continuación, la CPU conmuta al tipo de arranque configurado. Determinados errores impiden que la CPU pase al estado operativo RUN. La CPU soporta los tipos de arranque siguientes: Estado operativo STOP, "Cambio a estado operativo RUN después del arranque en caliente", y "Cambio a estado operativo anterior después del arranque en caliente".

ATENCIÓN
Configuración del modo Rearranque normal La CPU puede pasar al estado STOP a causa de fallos reparables, como p. ej. el fallo de un módulo de señales sustituible, o bien fallos temporales, como una interrupción de la línea de alimentación o un evento de arranque errático. Si la CPU ha sido configurada a "Arranque en caliente previo a POWER OFF", no retornará al modo RUN después de reparado o eliminado el fallo hasta que no reciba otro comando de STEP 7 para pasar a RUN. Sin un nuevo comando, el modo STOP se conserva como el modo anterior a la desconexión (POWER OFF). Las CPU que deben funcionar independientemente de una conexión STEP 7 deberán configurarse con "Arranque en caliente - RUN", de manera que la CPU puede volver al modo RUN por un ciclo de arranque debido a la eliminación de condiciones de fallo.



La CPU no dispone de un interruptor físico para cambiar el estado operativo. STEP 7 ofrece las siguientes herramientas para cambiar el estado operativo de la CPU:

- Botones "Stop" y "Run" en la barra de herramientas de STEP 7
- Panel de mando de la CPU en las herramientas online

También se puede insertar una instrucción STP en el programa para cambiar la CPU al estado operativo STOP. Esto permite detener la ejecución del programa en función de la lógica. El servidor web (Página 262) también ofrece una página para cambiar el estado operativo.

4.3 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 (Página 62).
- Las funciones (FCs) y los bloques de función (FBs) contienen el código del programa correspondiente a tareas específicas o combinaciones de parámetros. Cada FC o FB suministra un conjunto de parámetros de entrada y salida para compartir datos con el bloque que llama. 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 bloques de datos (DBs) almacenan datos que pueden ser utilizados por los bloques de programa.

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 (Página 15). 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.

4.3.1 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. En el caso de los eventos de alarma, la CPU lee las entradas, ejecuta el OB y, a continuación, escribe las salidas mediante la memoria imagen parcial del proceso asociada (IPP), si corresponde.

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.

4.3.2 Los OB ayudan a estructurar el programa de usuario

Los OB controlan la ejecución del programa de usuario. 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. Solo un evento, 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 de programa principal) y la más alta es 26.

4.3.3 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 el hardware dispara un evento, como p. ej. un flanco, en una entrada o un evento de contador rápido. Los eventos como errores de diagnóstico o de tiempo solo se producen cuando ocurre un error. Las prioridades de eventos y las colas de espera sirven para determinar el orden de procesamiento de los OB de alarma.

La CPU procesa los eventos en orden de prioridad, siendo 1 la prioridad más baja y 26 la más alta. Antes de la V4.0 de la CPU S7-1200, cada tipo de OB pertenecía a una clase de prioridad fija (1 a 26). A partir de la versión 4.0 se puede asignar una clase de prioridad a cada OB que se configure. El número de prioridad se configura en los atributos de las propiedades del OB.

Modos de ejecución con y sin interrupciones

Los OB (Página 61) se ejecutan en orden de prioridad de los eventos que los disparan. A partir de V4.0 es posible configurar la ejecución de los OB con o sin interrupciones. Recuerde que los OB de ciclo siempre son con interrupciones, pero todos los demás OB se pueden configurar para que sean con o sin interrupciones.

Con el modo con interrupciones activado, si se está ejecutando un OB y se produce un evento de mayor prioridad antes de que se complete la ejecución del OB, esta se interrumpe para permitir la ejecución del OB de evento de prioridad superior. El OB de evento de prioridad superior se ejecuta y cuando la ejecución se ha completado continúa el OB que se había interrumpido. Si se producen varios eventos mientras se está ejecutando un OB con interrupciones, la CPU procesará los eventos en orden de prioridad.

Si no activa el modo con interrupciones, un OB se ejecutará hasta el final una vez disparado, independientemente de cualquier otro evento que se dispare durante su ejecución.

Observe los dos siguientes casos, en los que unos eventos de alarma disparan un OB de ciclo y un OB de retardo de tiempo. En ambos casos, el OB de retardo de tiempo (OB201) carece de asignación de memoria imagen parcial de proceso y se ejecuta con prioridad 4. El OB de ciclo (OB200) tiene una asignación de memoria imagen parcial de proceso IPP1 y se ejecuta con prioridad 2. Las siguientes ilustraciones muestran la diferencia en la ejecución entre los modos de ejecución con y sin interrupciones:

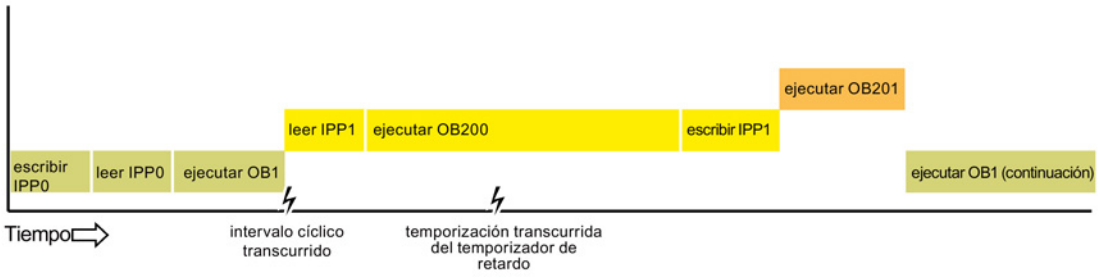


Figura 4-1 Caso 1: ejecución de OB sin interrupciones

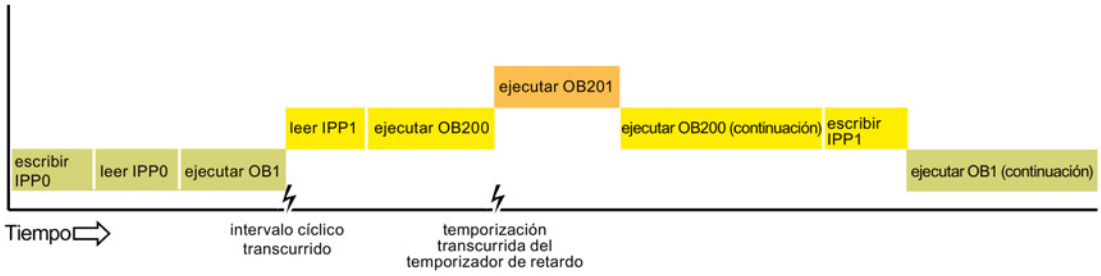


Figura 4-2 Caso 2: ejecución de OB con interrupciones

Nota

Si configura el modo de ejecución del OB sin interrupciones, un OB de error de tiempo no puede interrumpir OB que no sean OB de ciclo. Antes de la V4.0 de la CPU S7-1200, un OB de error de tiempo podía interrumpir cualquier OB en ejecución. A partir de la versión V4.0 hay que configurar la ejecución del OB con interrupciones para que un OB de error de tiempo (o cualquier otro OB de prioridad superior) pueda interrumpir los OB que se estén ejecutando y no sean OB de ciclo.

Prioridades y colas de espera para la ejecución de eventos

La CPU limita el número de eventos pendientes (en cola de espera) de una sola fuente 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. Se puede utilizar un OB de error de tiempo como respuesta a desbordamientos de la cola.

Todo evento de la CPU tiene asignada una prioridad. Generalmente, la CPU procesa los eventos según su prioridad (primero los de mayor prioridad). La CPU procesa los eventos de igual prioridad según su orden de aparición.

Tabla 4- 1 Eventos de OB

Evento	Cantidad permitida	Prioridad de OB pre-determinada
Ciclo del programa	1 evento de ciclo de programa Se admiten varios OB	1[4]
Arranque	1 evento de arranque [1] Se admiten varios OB	1[4]
Retardo	Hasta 4 eventos de tiempo 1 OB por evento	3
Alarma cíclica	Hasta 4 eventos 1 OB por evento	8
Alarma de hardware	Hasta 50 eventos de alarma de hardware[2]	18
	1 OB por evento, pero se puede utilizar el mismo OB para varios eventos	18
Error de tiempo	1 evento (solo si está configurado)[3]	22 o 26[4]
Error de diagnóstico	1 evento (solo si está configurado)	5

Evento	Cantidad permitida	Prioridad de OB predeterminada
Inserción o extracción de módulos	1 evento	6
Fallo de rack o estación	1 evento	6
Hora	Hasta 2 eventos	2
Estado	1 evento	4
Actualizar	1 evento	4
Perfil	1 evento	4

[1] Los eventos de arranque y de ciclo no ocurren nunca simultáneamente, ya que el evento de arranque debe haber finalizado antes de poder iniciar el evento de ciclo.

[2] Se puede contar con más de 50 OB de evento de alarma de hardware si se utilizan las instrucciones DETACH y ATTACH.

[3] 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 inicializar 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.

[4] La prioridad de una CPU nueva V4.0 o V4.1 es 22. Si se sustituye una CPU V3.0 por una CPU V4.0 o V4.1, la prioridad será 26: la prioridad que regía para V3.0. En ambos casos, el campo de prioridad es editable, de modo que la prioridad se puede ajustar a cualquier valor entre 22 y 26.

Consulte el tema "Reemplazar una CPU V3.0 por una CPU V4.1 (Página 443)" para obtener más información.

Además, la CPU reconoce otros eventos que carecen de OB asociados. La tabla siguiente describe esos eventos y las correspondientes acciones de la CPU:

Tabla 4- 2 Eventos adicionales

Evento	Descripción	Acción de la CPU
Error de acceso a E/S	Error de escritura/lectura directa de E/S	La CPU registra la primera aparición en el búfer de diagnóstico y permanece en estado RUN.
Error de tiempo de ciclo máximo	La CPU excede dos veces el tiempo de ciclo configurado	La CPU registra el error en el búfer de diagnóstico y pasa al estado STOP.
Error de acceso a la periferia	Error de E/S durante una actualización de la memoria imagen de proceso	La CPU registra la primera aparición en el búfer de diagnóstico y permanece en estado RUN.
Error de programación	Error de ejecución del programa	Si el bloque que tiene el error dispone de tratamiento de errores, actualiza la estructura errónea; si no, la CPU registra el error en el búfer de diagnóstico y permanece en estado RUN.

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.

4.4 Áreas de memoria, direccionamiento y tipos de datos

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 un corte de alimentación. Es posible aumentar la cantidad de memoria de carga disponible para registros con la instalación de una Memory Card.
- 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 un corte de alimentación. Si ocurre un corte de alimentación o una pérdida de potencia, la CPU restablece estos valores remanentes al retornar la alimentación.

Una Memory Card SIMATIC opcional proporciona una memoria alternativa para almacenar el programa de usuario, así como un medio para transferir el programa. Si se utiliza una Memory Card, la CPU ejecutará el programa desde allí y no desde la memoria de la CPU.

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

Utilice la Memory Card SIMATIC opcional como tarjeta de programa, como tarjeta de transferencia, para almacenar archivos de registro de datos o para actualizar el firmware.

- Utilice la tarjeta de transferencia para copiar el proyecto en varias CPU sin utilizar STEP 7. La tarjeta de transferencia copia un proyecto almacenado de la tarjeta a la memoria de la CPU. La tarjeta de transferencia debe extraerse tras copiar el programa en la CPU.
- La tarjeta de programa sustituye la memoria de la CPU. Todas las funciones de la CPU son controladas por la tarjeta de programa. Cuando se inserta una tarjeta de programa, se borra toda la memoria de carga interna de la CPU (incluyendo el programa de usuario y las E/S forzadas permanentemente). La CPU ejecuta entonces el programa de usuario desde la tarjeta de programa.

- También es posible usar la tarjeta de programa para recopilar archivos de registro (Página 129). La tarjeta de programa ofrece más capacidad de memoria que la memoria interna de la CPU. La función de servidor WEB (Página 261) de la CPU permite descargar los archivos de registro a un ordenador.
- También se puede utilizar una Memory Card para actualizar el firmware. Consulte el *Manual de sistema del controlador programable S7-1200* para más información.

Nota

La tarjeta de programa **debe** permanecer en la CPU. Si se extrae la tarjeta de programa, la CPU pasará a estado operativo STOP.

4.4.1 Tipos de datos soportados por el S7-1200

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.

Tabla 4- 3 Tipos de datos soportados por el S7-1200

Tipos de datos	Descripción
Tipos de datos de bits y secuencias de bits	• Bool es un valor de bit o booleano. • Byte es un valor de byte (8 bits). • Word es un valor de 16 bits. • DWord es un valor de palabra doble (32 bits).
Tipos de datos enteros	• USInt (entero sin signo de 8 bits) y SInt (entero con signo de 8 bits) son enteros "cortos" (8 bits o 1 byte de memoria) que pueden tener o no signo. • UInt (entero sin signo de 16 bits) e Int (entero con signo de 16 bits) son enteros (16 bits o 1 palabra de memoria) que pueden tener o no signo. • UDInt (entero de 32 bits sin signo) y DInt (entero de 32 bits con signo) son enteros dobles (32 bits o 1 palabra doble de memoria) que pueden tener o no signo.
Tipos de datos de números reales	• Real es un valor de número real de 32 bits o en coma flotante. • LReal es un valor de número real de 64 bits o en coma flotante.

Tipos de datos	Descripción
Tipos de datos de fecha y hora	• Date es un valor de fecha de 16 bits (similar a un valor UInt) que contiene el número de días desde el 1 de enero de 1990. El valor de fecha máximo es 65378 (16#FF62), que corresponde al 31 de diciembre de 2168. Todos los posibles valores de Date son válidos. • DTL (Date and Time Long) es una estructura de 12 bytes que almacena información de fecha y hora en una estructura predefinida. – Año (UInt): de 1970 a 2554 – Mes (USInt): de 1 a 12 – Día (USInt): de 1 a 31 – Día de la semana (USInt): de 1 (domingo) a 7 (sábado) – Horas (USInt): de 0 a 23 – Minutos (USInt): de 0 a 59 – Segundos (USInt): de 0 a 59 – Nanosegundos (UDInt): de 0 a 999999999 • Time es un valor de tiempo CEI de 32 bits (parecido al valor Dint) que almacena el número de milisegundos (de 0 a 24 días, 20 horas, 31 minutos, 23 segundos y 647 ms). Todos los posibles valores de Time son válidos. Los valores de Time se pueden usar para cálculos, y se pueden obtener tiempos negativos. • TOD (Time of Day) es un valor de hora de 32 bits (parecido al valor Dint) que contiene el número de milisegundos desde medianoche (de 0 a 86399999).
Tipos de datos de caracteres y cadenas	• Char es un carácter simple de 8 bits. • String es una cadena de longitud variable de hasta 254 caracteres.
Tipos de datos de matriz y estructura	• Array contiene varios elementos del mismo tipo de datos. Las matrices pueden crearse en los editores de interfaces de bloque para OB, FC, FB y DB. En el editor de variables PLC, no se pueden crear matrices. • Struct define una estructura de datos que consta de 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 indica el nombre y la estructura de datos interna del tipo de datos Struct en el editor de bloques de datos o en un editor de interfaces de bloque. Las matrices y estructuras también se pueden cablear 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.

Tipos de datos	Descripción
Tipos de datos PLC	Los tipos de datos PLC corresponden a una estructura de datos definida por el usuario que define una estructura de datos personalizados que se puede emplear varias veces en el programa. Al crear un tipo de datos PLC, el nuevo tipo de datos PLC aparece en las listas desplegables del selector de tipos de datos del editor DB y del editor de interfaces de bloques. Los tipos de datos PLC pueden usarse directamente como tipo de datos en una interfaz de bloques de códigos 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.
Tipos de datos de puntero	• Pointer ofrece una referencia indirecta a la dirección de una variable. Ocupa 6 bytes (48 bits) en la memoria y puede incluir la información siguiente para una variable: Número de DB (o 0 si los datos no se almacenan en un DB), área de memoria en la CPU y dirección de la memoria. • Any ofrece una referencia indirecta al inicio de un área de datos e identifica su longitud. El puntero de Any usa 10 bytes de la memoria y puede incluir la información siguiente: Tipo de datos de los elementos de datos, número de elementos de datos, área de memoria o número de DB, así como dirección de inicio "Byte.Bit" de los datos. • Variant ofrece una referencia indirecta a variables de diversos parámetros o tipos de datos. El puntero de Variant reconoce estructuras y componentes estructurales individuales. Variant no ocupa ningún espacio en la memoria.

Aunque no están disponibles como tipos de datos, las operaciones de conversión soportan los siguientes formatos numéricos BCD (Binary Coded Decimal, decimal codificado binariamente).

- BCD16 es un valor de 16 bits (de -999 a 999).
- BCD32 es un valor de 32 bits (de -9999999 a 9999999).

4.4.2 Direccionamiento de áreas de memoria

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.
- 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) especifico 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").

El forzado permanente solo aplica un valor fijo en una entrada física (Ix.y:P) o en una salida física (Qx.y:P). Para forzar permanentemente una entrada o salida, agregue una ":P" a la variable PLC o dirección. Para más información, consulte "Forzar valores permanentemente en la CPU" (Página 350).

Tabla 4- 4 Áreas de memoria

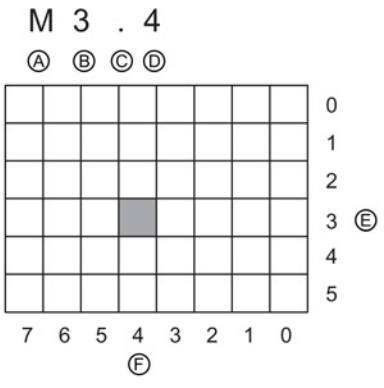
Área de memoria	Descripción	Forzado permanente	Remanente
I Memoria imagen de proceso de las entradas I_:P[1] (entrada física)	Se copia de las entradas físicas al inicio del ciclo	No	No
	Lectura inmediata de las entradas físicas de la CPU, SB y SM	Sí	No
Q Memoria imagen de proceso de las salidas Q_:P[1]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

- Área de memoria (como I, Q o M)
- Tamaño de los datos a los que se va a acceder (como "B" para Byte o "W" para Word)
- Dirección de los datos (como Byte 3 o Word 3)

Al acceder a un bit en la dirección para un valor booleano, no se introduce ningún nemónico para el tamaño. Solo 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).



Dirección absoluta de un área de memoria:

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

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).

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

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 (H	
HSC_2	1.17			Contador rápido (H	
HSC_3	1.18			Contador rápido (H	
HSC_4	1.19			Contador rápido (H	
HSC_5	1.20			Contador rápido (H	
HSC_6	1.21			Contador rápido (H	
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

Al agregar una CPU y módulos E/S a la configuración del dispositivo, STEP 7 asigna automáticamente direcciones de entrada y salida. El direccionamiento predeterminado puede cambiarse seleccionando el campo de dirección en la configuración de dispositivos e introduciendo números nuevos.

- STEP 7 asigna las entradas y salidas digitales en grupos de 8 E/S (1 byte), sin importar si el módulo utiliza todas las E/S o no.
- STEP 7 asigna entradas y salidas analógicas en grupos de 2, donde cada E/S analógica ocupa 2 bytes (16 bits).

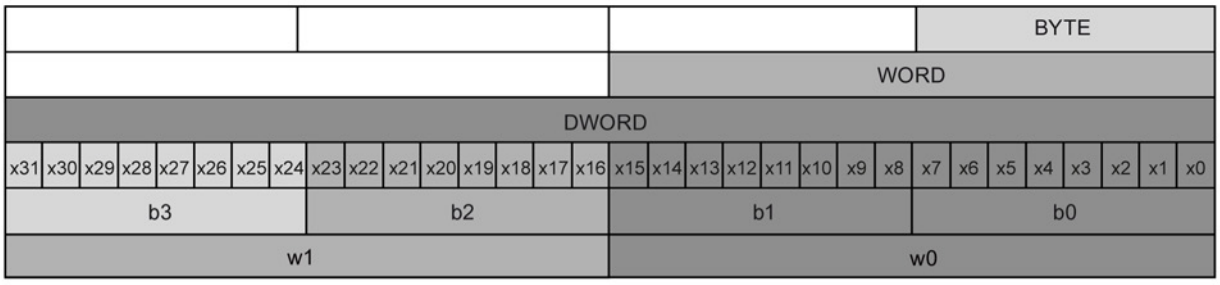
La figura muestra un ejemplo de una CPU 1214C con dos SM y una SB. En este ejemplo, la dirección del módulo DI8 se podría cambiar a 2 en vez de 8. La herramienta le ayuda cambiando los rangos de direcciones cuyo tamaño sea incorrecto o que causen conflictos con otras direcciones.

4.4.3 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 palabras 0 - 1. A una variable del tamaño de palabra se accede a través de los bits 0 - 15, bytes 0 - 1 o palabra 0. A una variable del tamaño de byte se accede a través de los bits 0 - 7 o byte 0. Los segmentos bit, byte y palabra se pueden utilizar dondequiera que bits, bytes y palabras estén previstos como operandos.



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	"DW".x11	"DW".x11 — &	`IF "DW".x11 THEN`
			`...`
			`END_IF;`

Acceso a byte	"DW".b2 == Byte "DW".b3	== Byte
		"DW".b2 — IN1
		"DW".b3 — IN2
			`IF "DW".b2 = "DW".b3`
			`THEN`
			`...`
			`END_IF;`

Acceso a palabra	AND Word EN ENO "DW".w0 — IN1 OUT "DW".w1 — IN2	AND Word ... — EN "DW".w0 — IN1 OUT "DW".w1 — IN2 ENO	`out:= "DW".w0 AND` `"DW".w1;`

4.4.4 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	0.0
OV	AT "B1"	Array[0..7] of Bool	0.0
OV[0]		Bool	0.0
OV[1]		Bool	0.1
OV[2]		Bool	0.2
OV[3]		Bool	0.3
OV[4]		Bool	0.4
OV[5]		Bool	0.5
OV[6]		Bool	0.6
OV[7]		Bool	0.7

Otro ejemplo es una variable DWord superpuesta con un Struct que incluye una palabra, un byte y dos booleanos:

DW1		DWord	2.0
DW1_Struct	AT "DW1"	Struct	2.0
W1		Word	0.0
B1		Byte	2.0
BO1		Bool	3.0
BO2		Bool	3.1

La columna Offset de la interfaz de bloque muestra la ubicación de los tipos de datos superpuestos en relación con la variable original.

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

KOP	FUP	SCL
#OV[1]	& #OV[1]	`IF #OV[1] THEN ... END_IF;`
#DW1_Struct.W1 == Word W#16#000C	== Word #DW1_Struct.W1 IN1 W#16#000C IN2	`IF #DW1_Struct.W1 = W#16#000C THEN ... END_IF;`
MOVE EN ENO #DW1_Struct.B1 IN OUT1	MOVE ... EN OUT1 <???> #DW1_Struct.B1 IN ENO	`out1 := #DW1_Struct.B1;`
#OV[4] #DW1_Struct.BO2	& #OV[4] #DW1_Struct.BO2	`IF #OV[4] AND #DW1_Struct.BO2 THEN ... END_IF;`

Reglas

- La superposición de variables solo es posible en bloques FB y FC con acceso estándar (no optimizado).
- 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.
- Debe declarar la variable que se superpone inmediatamente después de la variable que queda superpuesta y seleccionar la palabra clave "AT" como selección de tipo de datos inicial.

4.5 Generadores de impulsos

La CPU o la Signal Board (SB) se puede configurar de modo que ofrezca cuatro generadores de impulsos para controlar funciones de salida de impulsos rápidos, bien como modulación del ancho de pulso (PWM) bien como salida de cadena de impulsos (PTO). Las instrucciones básicas de movimiento emplean salidas PTO. Cada generador de impulsos puede asignarse a PWM o PTO, pero no a ambos simultáneamente.

Los generadores de impulsos no pueden ser utilizados por otras instrucciones dentro del programa de usuario. Al configurar las salidas de la CPU o SB como generadores de impulsos, las direcciones correspondientes de las salidas se eliminan de la memoria Q y no pueden utilizarse para otros fines dentro del 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.

Nota

No debe excederse la frecuencia de pulsos máxima.

La frecuencia de impulsos máxima de los generadores de salida de impulsos es 1 MHz para la CPU 1217C y 100 kHz para las CPU 1211C, 1212C, 1214C y 1215C; 20 kHz (para una SB estándar) o 200 kHz (para una SB rápida).

Los cuatro generadores de impulsos tienen asignaciones de E/S predeterminadas, pero pueden configurarse para cualquier salida digital de la CPU o SB. Los generadores de impulsos de la CPU no pueden asignarse a la periferia descentralizada.

Cuando configure instrucciones básicas de movimiento, recuerde que STEP 7 **no** emite ninguna alerta si configura un eje con una velocidad o frecuencia máxima superior a la citada limitación del hardware. Ello podría ocasionar problemas en la aplicación. Por tanto, vigile que no se exceda la frecuencia de pulsos máxima del hardware.

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. Tenga en cuenta que PWM solo 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.

Los cuatro generadores de impulsos tienen asignaciones de E/S predeterminadas, pero pueden configurarse para cualquier salida digital de la CPU o SB. Los generadores de impulsos de la CPU no pueden asignarse a SMs o la periferia descentralizada.

Tabla 4- 5 Asignaciones de salidas predeterminadas para los generadores de impulsos

Descripción	Impulso	Sentido
PTO1		
E/S integradas	Q0.0	Q0.1
E/S de la SB	Q4.0	Q4.1
PWM1		
Salidas integradas	Q0.0	-
Salidas de la SB	Q4.0	-
PTO2		
E/S integradas	Q0.2	Q0.3
E/S de la SB	Q4.2	Q4.3
PWM2		
Salidas integradas	Q0.2	-
Salidas de la SB	Q4.2	-
PTO3		
E/S integradas	Q0.4[1]	Q0.5[1]
E/S de la SB	Q4.0	Q4.1
PWM3		
Salidas integradas	Q0.4[1]	-
Salidas de la SB	Q4.1	-
PTO4		
E/S integradas	Q0.6[2]	Q0.7[2]
E/S de la SB	Q4.2	Q4.3
PWM4		
Salidas integradas	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.

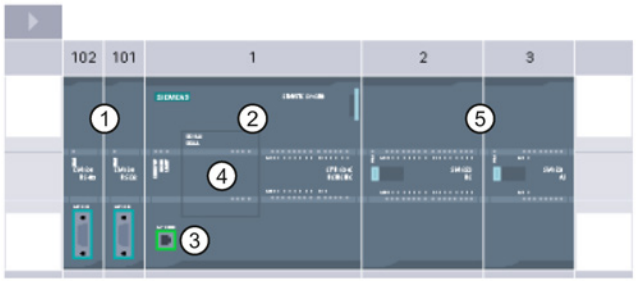
[2] La CPU 1212C no tiene las salidas Q0.6 ni Q0.7. Por lo tanto, dichas salidas no pueden usarse en la CPU 1212C.

[3] Esta tabla es válida para las funciones de PTO/PWM de la CPU 1211C, CPU 1212C, CPU 1214C y CPU 1215C y de la CPU 1217C.

La configuración de dispositivos se crea fácilmente

5

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 (las CPU 1214C, CPU 1215C y 1217C admiten 8, la CPU 1212C admite 2, la CPU 1211C no admite ninguno)

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

- En la vista del 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 Cargar la configuración de una CPU conectada

STEP 7 ofrece dos métodos para cargar la configuración hardware de una CPU conectada:

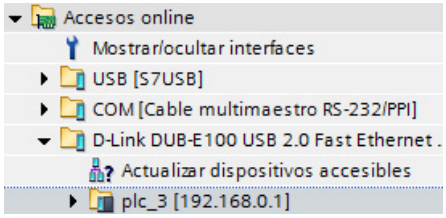
- Cargar el dispositivo conectado como una estación nueva
- Configurar una CPU no especificada y detectar la configuración hardware de la CPU conectada

Sin embargo, tenga en cuenta que el primer método carga tanto la configuración hardware como el software de la CPU conectada.

Cargar un dispositivo como una estación nueva

Para cargar un dispositivo conectado como estación nueva, proceda del siguiente modo:

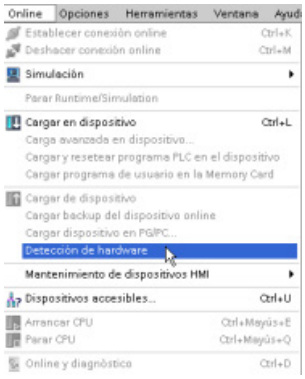
1. Amplíe su interfaz de comunicación desde el nodo "Acceso online" del árbol de proyecto.
2. Haga doble clic en "Actualizar dispositivos accesibles".
3. Seleccione el PLC de entre los dispositivos detectados.



4. Desde el menú "Online" de STEP 7 seleccione el comando "Carga del dispositivo como estación nueva (hardware y software)".

STEP 7 carga tanto la configuración hardware como los bloques de programa.

Detectar la configuración hardware de una CPU sin especificar

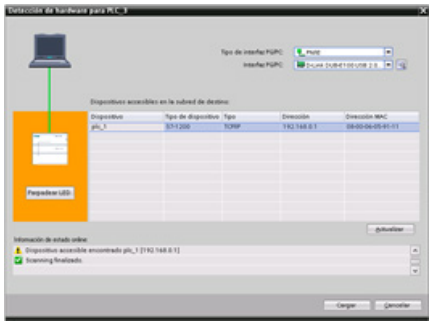


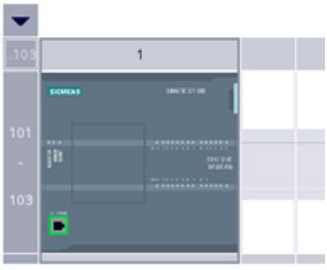
Si existe una conexión con una CPU, es posible cargar su configuración en el módulo (incluidos los módulos). Tan solo hay que crear un proyecto nuevo y seleccionar la "CPU sin especificar" en lugar de una específica. (También es posible omitir la configuración de dispositivo por completo seleccionando "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.

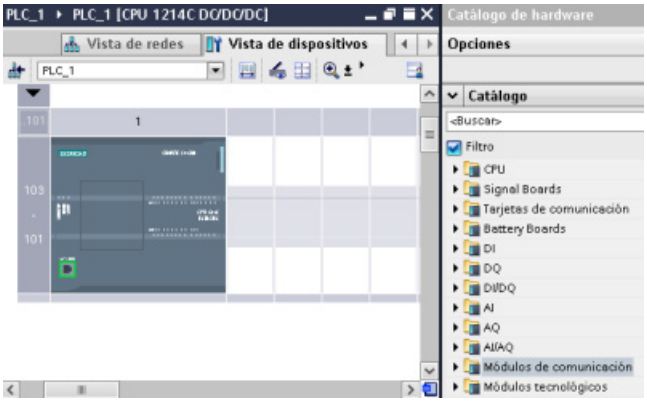
Tras seleccionar la CPU en el cuadro de diálogo online y pulsar el botón de carga, STEP 7 carga la configuración de hardware de la CPU, incluidos todos los módulos (SM, SB o CM). Entonces pueden configurarse los parámetros de la CPU y de los módulos (Página 84).



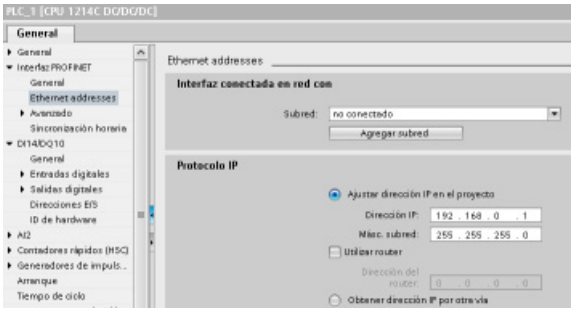


5.2 Agregar una CPU a la configuración

La configuración de dispositivos se crea insertando una CPU en el proyecto. Seleccione la CPU en el cuadro de diálogo "Agregar nuevo dispositivo" y haga clic en "Aceptar" para agregar la CPU al proyecto.



La vista de dispositivos muestra la CPU y el rack.



Al seleccionar la CPU en la vista de dispositivos se visualizan sus propiedades en la ventana de inspección. Utilice las propiedades para configurar los parámetros operativos de la CPU (Página 84).

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.

5.3 Cambiar un dispositivo

Existe la posibilidad de cambiar el tipo de dispositivo de una CPU o un módulo configurado. En la configuración de dispositivos, haga clic con el botón derecho del ratón en el dispositivo y elija el comando "Cambiar dispositivo" del menú contextual. En el cuado de diálogo que aparece, navegue hasta la CPU o el módulo que desee reemplazar y selecciónelo. El cuadro de diálogo Cambiar dispositivo muestra información sobre la compatibilidad entre ambos dispositivos.

Nota

Cambio de dispositivo: reemplazar una CPU V3.0 por una CPU V4.1

Un proyecto de STEP 7 V12 puede abrirse en STEP 7 V13, donde pueden reemplazarse las CPU V3.0 por CPU V4.1. No es posible reemplazar CPU de versiones anteriores a la V3.0. Al reemplazar una CPU V3.0 por una CPU V4.1, tenga en cuenta las diferencias (Página 443) existentes cuanto a las características y el comportamiento de ambas versiones y las acciones que deben llevarse cabo.

Si tiene un proyecto para una versión de CPU anterior a la V3.0, primero deberá actualizar dicho proyecto a la V3.0 y después a la V4.1.

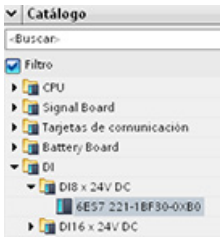
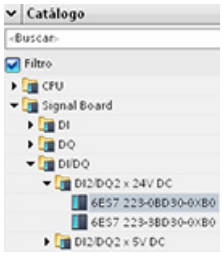
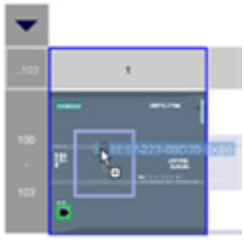
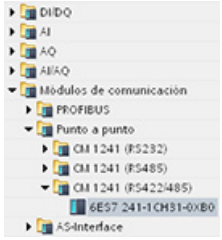
5.4 Agregar módulos a la configuración

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			

Con la función "control de configuración" (Página 83) es posible agregar módulos de señales y Signal Boards a la configuración del dispositivo que no se corresponden con el hardware real para una aplicación en concreto, pero que serán usados en aplicaciones relacionadas que comparten un programa de usuario o modelo de CPU común y, posiblemente, algunos de los módulos configurados.

5.5 Control de configuración

El control de configuración puede ser útil al crear una solución de automatización (máquina) que se pretende utilizar en múltiples instalaciones con diferentes variaciones.

El control de configuración con STEP 7 y el S7-1200 permite configurar una configuración máxima para una máquina estándar y utilizar distintas versiones (opciones) que usen parte de dicha configuración. El manual PROFINET con STEP 7 (http://support.automation.siemens.com/WW/view/en/49948856) hace referencia a este tipo de proyectos como "Proyectos de maquinaria de serie".

Se puede cargar una configuración de dispositivos STEP 7 y un programa de usuario a configuraciones de PLC con diferentes instalaciones. Tan solo hay que realizar unas pocas adaptaciones sencillas para que el proyecto STEP 7 se corresponda con la instalación real.

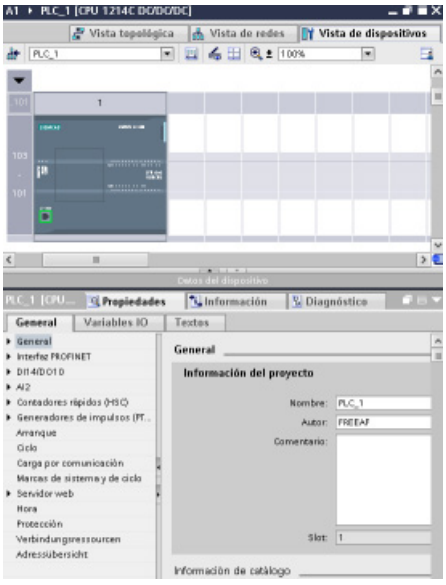
Un juego de datos de control programado en el bloque de programa de arranque notifica a la CPU qué módulos faltan en la instalación real en comparación con la configuración o qué módulos están situados en slots diferentes a los de la configuración. El control de configuración no influye en la asignación de parámetros de los módulos.

El control de configuración proporciona la flexibilidad de variar la instalación mientras que la configuración real pueda derivarse de la configuración de dispositivos máxima en STEP 7.

Encontrará instrucciones y ejemplos del control de configuración en el *Manual de sistema del controlador programable S7-1200*.

5.6 Configurar el funcionamiento de la CPU y sus módulos

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



Se pueden configurar las siguientes propiedades de la CPU:

- Dirección IP PROFINET y sincronización horaria para la CPU
- Comportamiento de arranque de la CPU tras una transición de OFF a ON
- Entradas y salidas digitales y analógicas locales (incorporadas), contadores rápidos (HSC) y generadores de impulsos
- Reloj del sistema (hora, zona horaria y horario de verano/invierno)
- Protección de lectura/escritura y contraseña para acceder a la CPU
- Tiempo máximo de ciclo o un tiempo de ciclo mínimo fijo y carga de comunicaciones
- Propiedades del servidor web

Configurar la transición de STOP a RUN de la CPU

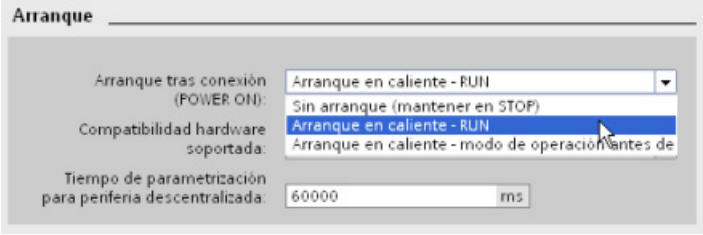
Cada vez que el estado operativo cambia de STOP a RUN, la CPU borra las entradas de la memoria imagen de proceso, inicializa las salidas de la memoria imagen de proceso y procesa los OBs de arranque. (Por este motivo, en los accesos de lectura a las entradas de la memoria imagen de proceso realizados por instrucciones en los OBs de arranque se leerá cero, en vez del valor actual de la entrada física.) Para leer el estado actual de una entrada física durante el 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.

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 HSCs, la PWM y los módulos de comunicación punto a punto sólo funcionan en el 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.

Utilice las propiedades de la CPU para configurar cómo debe arrancar tras desconectar y conectar la alimentación.



- En estado operativo STOP
- En estado operativo RUN
- En el estado operativo anterior (antes de desconectar y conectar la alimentación)

La CPU ejecuta un arranque en caliente antes de pasar a estado operativo RUN. El arranque en caliente inicializa toda la memoria no remanente a los valores iniciales predeterminados. No obstante, la CPU conserva los valores actuales almacenados en la memoria remanente.

Nota

La CPU ejecuta siempre un rearranque tras una carga

Cada vez que se carga un elemento del proyecto (p. ej. bloque de programa, bloque de datos o configuración hardware), la CPU ejecuta un rearranque en la próxima transición al estado operativo RUN. Además de borrar las entradas e inicializar las salidas y la memoria no remanente, el rearranque inicializa asimismo las áreas de memoria remanente.

Tras el rearranque posterior a la carga, todas las transiciones de STOP a RUN subsiguientes ejecutan un arranque en caliente (que no inicializa la memoria remanente).

5.6.1 La memoria de sistema y la marca de ciclo proporcionan funciones estándar

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.)
  - Ha cambiado el estado de diagnóstico: (nombre de la 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 "DiagStatusUpdate" 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.

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

Riesgos asociados con la sobrescritura de los bits de marca de sistema o de ciclo

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.

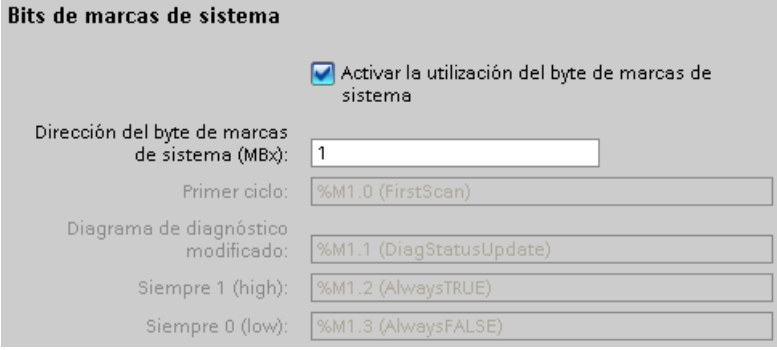


Tabla 5- 2 Marcas de sistema

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.

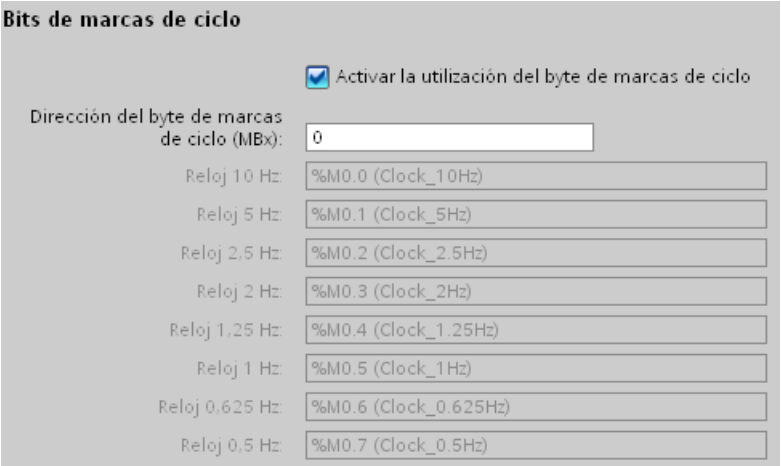


Tabla 5- 3 Marcas de ciclo

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.

Configurar el funcionamiento de las E/S y los módulos de comunicación

Para configurar los parámetros operativos de un módulo de señales (SM), una Signal Board (SB) o un módulo de comunicación (CM), selecciónelo en la vista de dispositivos y utilice la ficha "Propiedades" de la ventana de inspección.

Módulo de señales (SM) y Signal Board (SB)

- E/S digitales: Configurar las distintas entradas, p. ej. para la detección de flancos y "captura de impulsos" (para que permanezcan activadas o desactivadas durante un ciclo tras un impulso momentáneo alto o bajo). Configurar las salidas para que se congelen o utilicen un valor sustitutivo en una transición de RUN a STOP.
- E/S analógicas: Configurar los parámetros de las distintas entradas (tales como tensión o intensidad, rango y alisamiento) y habilitar el diagnóstico de rebase por defecto o por exceso. Configurar los parámetros de las distintas salidas analógicas y habilitar el diagnóstico, p. ej. de cortocircuito (para salidas de tensión) o valores de rebase por exceso.
- Direcciones de E/S: Configurar la dirección inicial de las entradas y salidas del módulo.

Módulo de comunicación (CM) y tarjeta de comunicación (CB)

- Configuración del puerto: Configurar los parámetros de comunicación, p. ej. la velocidad de transferencia, paridad, bits de datos, bits de parada y tiempo de espera.
- Mensaje de transmisión y recepción: Configurar las opciones relativas a la transmisión y recepción de datos (p. ej. los parámetros de inicio y fin de mensajes).

También es posible cambiar esos parámetros de configuración con el programa de usuario.

5.7 Configurar la dirección IP de la CPU

Dado que la CPU no dispone de una dirección IP preconfigurada, es necesario asignarla manualmente. La dirección IP, junto con otros parámetros de la interfaz PROFINET, se fija al configurar las propiedades de la CPU.

- Cada dispositivo de una red PROFINET obtiene del fabricante una dirección MAC (Media Access Control o control de acceso al medio) unívoca para su identificación. Cada dispositivo debe tener además una dirección IP.
- Una subred es una agrupación lógica de dispositivos de red conectados. Una máscara (denominada también "máscara de subred" o "máscara de red") define los límites de una subred. La única conexión entre las diferentes subredes se realiza a través de un router. Los routers son el enlace entre LANs y necesitan direcciones IP para poder transferir y recibir paquetes de datos.

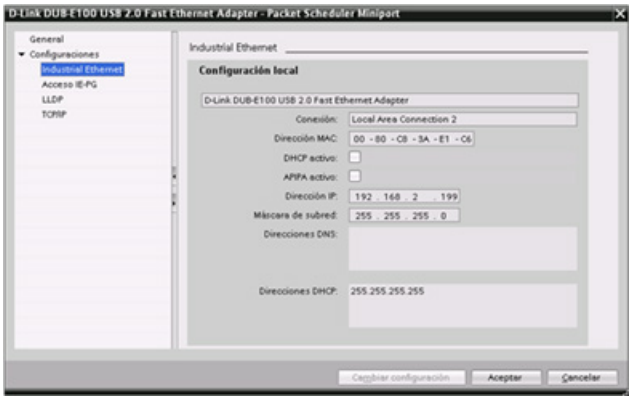
Antes de poder cargar una dirección IP en la CPU, es necesario asegurarse de que la dirección IP de la CPU es compatible con la dirección IP de la programadora.

La dirección IP de la programadora se puede determinar con STEP 7:

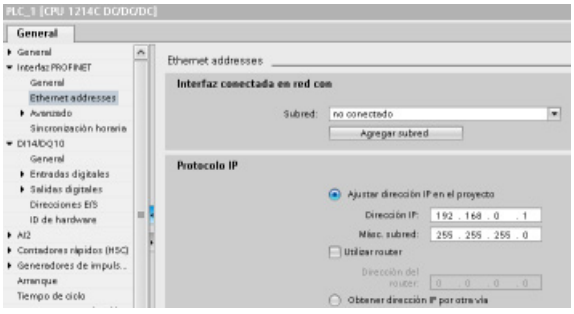
1. Expanda la carpeta "Acceso online" del árbol del proyecto para ver las redes.
2. Seleccione la red que se conecta a la CPU.
3. Haga clic con el botón derecho del ratón en la red correspondiente para abrir el menú contextual.
4. Elija el comando "Propiedades".

Nota

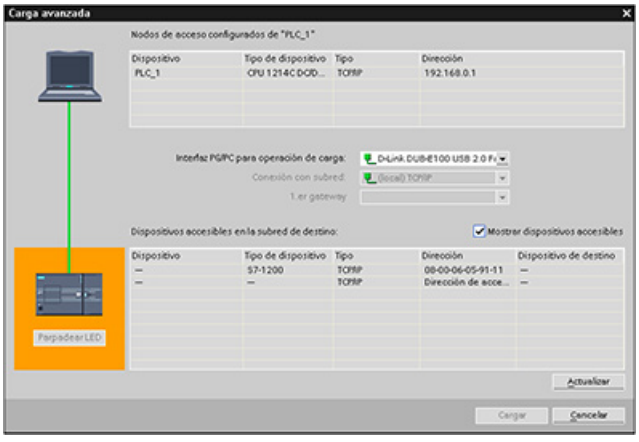
La dirección IP de la CPU debe ser compatible con la dirección IP y la máscara de subred de la programadora. Consulte a su especialista en redes para obtener la dirección IP y la máscara de subred adecuadas para la CPU.



La ventana "Propiedades" muestra la configuración de la programadora.



Una vez determinadas la dirección IP y la máscara de subred de la CPU, introduzca la dirección IP de la CPU y del router (si procede). Consulte el *Manual de sistema del controlador programable S7-1200* para más información.



Tras finalizar la configuración, cargue el proyecto en la CPU.

Las direcciones IP de la CPU y del router (si procede) quedan configuradas al descargar el proyecto.

5.8 Proteger el acceso a la CPU o al bloque lógico es fácil

La CPU ofrece cuatro 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.

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 se puede acceder a todas las funciones del nivel.

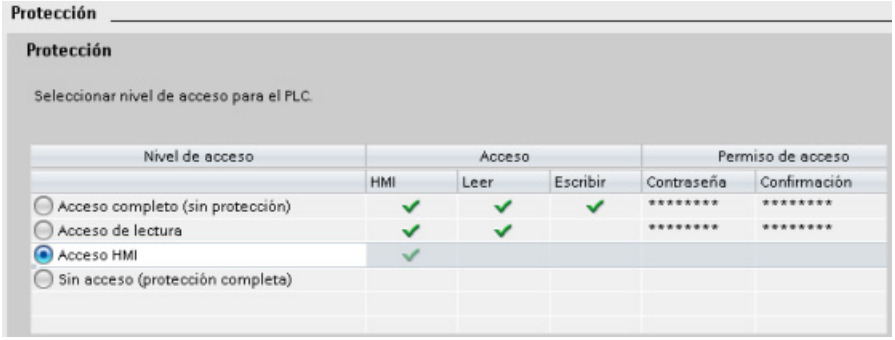
El nivel de protección de la CPU no restringe la comunicación entre PLCs (mediante instrucciones de comunicación en los bloques lógicos).

Tabla 5- 4 Niveles de protección de la CPU

Nivel de protección	Restricciones de acceso
Acceso completo (sin protección)	Permite el acceso completo sin protección por contraseña.
Acceso de lectura	Permite el acceso HMI y 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).
Acceso HMI	Permite el acceso HMI y 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).
Sin acceso (protección total)	No permite el acceso sin protección por contraseña. La contraseña se requiere para el acceso HMI, para leer los datos de la CPU y para modificar (escribir en) la CPU.

Las contraseñas distinguen entre mayúsculas y minúsculas. Para configurar el nivel de protección y las contraseñas, 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 contraseñas.



Cuando se descarga esta configuración a la CPU, el usuario dispone de acceso HMI y puede acceder a funciones HMI sin una contraseña. Para leer datos, el usuario debe introducir la contraseña configurada para "Acceso de lectura" o la contraseña para "Acceso completo (sin protección)". Para escribir datos, el usuario debe introducir la contraseña configurada para "Acceso completo (sin protección)".

⚠ ADVERTENCIA

Acceso no autorizado a una CPU protegida

Los usuarios con derechos de pleno acceso a la CPU pueden leer y escribir variables PLC. Independientemente del nivel de acceso a la CPU, los usuarios de servidor web pueden tener derechos para leer y escribir variables PLC. El acceso no autorizado a la CPU o el cambio de variables PLC a valores no válidos puede afectar al funcionamiento del proceso y puede producir la muerte, lesiones corporales graves o daños materiales.

Los usuarios autorizados pueden realizar cambios en el estado operativo, escribir datos de PLC y actualizar el firmware. Siemens recomienda observar las siguientes consignas de seguridad:

- Niveles de acceso a la CPU protegidos por contraseña e IDs de usuario de servidor web (Página 262) con contraseñas seguras. Las contraseñas seguras tienen diez 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. La contraseña debe mantenerse en secreto y cambiarse con frecuencia.
- Permitir el acceso al servidor web solo con el protocolo HTTPS.
- No amplíe los derechos mínimos predeterminados del usuario "Todos" del servidor web.
- Realice comprobaciones de errores y de rango de las variables usadas en la lógica del programa, puesto que los usuarios de páginas web pueden cambiar variables PLC a valores no válidos.

Mecanismos de conexión

Para acceder a interlocutores remotos con instrucciones PUT/GET, el usuario también debe disponer de permisos.

Por defecto, la opción "Permitir acceso vía comunicación PUT/GET" está desactivada. En ese caso, el acceso de lectura y escritura a los datos de la CPU solo es posible para conexiones de comunicación que requieren configuración o programación tanto para la CPU local como para el interlocutor. El acceso mediante instrucciones BSEND/BRCV es posible, por ejemplo.

En consecuencia, las conexiones en las que la CPU local tan solo hace de servidor (es decir, en la CPU local no se ha configurado ni programado la comunicación con el interlocutor) no son posibles mientras la CPU está en funcionamiento, por ejemplo:

- Acceso PUT/GET, FETCH/WRITE o FTP mediante módulos de comunicaciones
- Acceso PUT/GET desde otras CPU S7
- Acceso HMI mediante comunicación PUT/GET

Si desea permitir el acceso a datos de la CPU desde el lado del cliente, es decir, que no desea restringir los servicios de comunicación de la CPU, siga los siguientes pasos:

1. Configure un nivel de acceso protegido cualquiera excepto "Sin acceso (protección total)".
2. Active la casilla de verificación "Permitir acceso vía comunicación PUT/GET".

Al descargar la configuración en la CPU, esta permitirá la comunicación PUT/GET desde interlocutores remotos

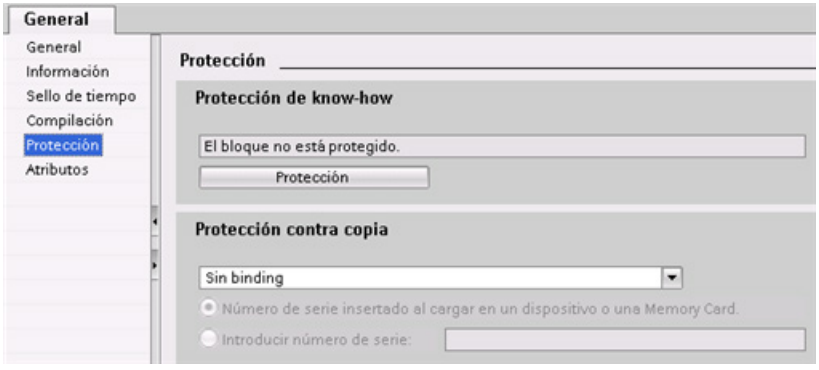
5.8.1 Protección de know-how

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, solo 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".

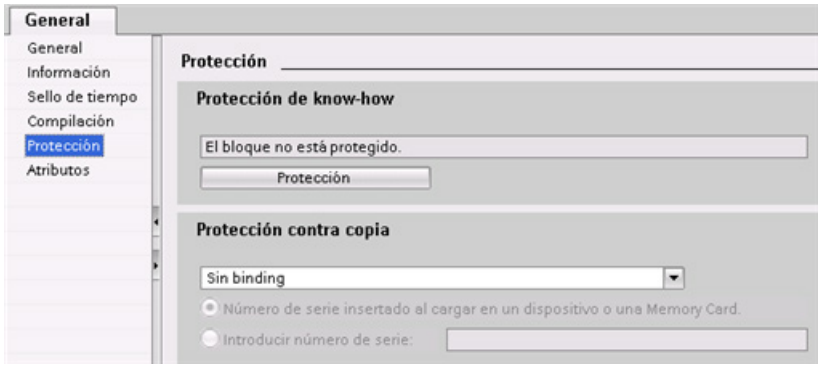
5.8.2 Protección anticopia

Una función de protección adicional permite enlazar bloques de programa para el uso con una Memory Card o CPU específica. Esta función se usa especialmente para proteger la propiedad intelectual. Al enlazar un bloque de programa a un dispositivo específico, solo 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. La protección contra copia está disponible para OBs (Página 99), FBs (Página 101) y FCs (Página 101). La CPU S7-1200 soporta tres tipos de protección de bloques:

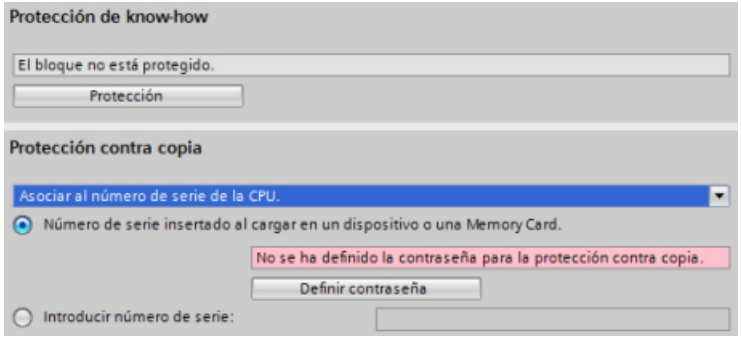
- Enlace al número de serie de una CPU
- Enlace al número de serie de una Memory Card
- Enlace dinámico con una contraseña obligatoria

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 el tipo de protección contra copia que desea usar.



3. Para enlazar al número de serie de una CPU o Memory Card, elija la opción de insertar el número de serie cuando realice la descarga o introduzca el número de serie para la Memory Card o CPU.

Nota

El número de serie distingue entre mayúsculas y minúsculas.

Para un enlace dinámico con contraseña obligatoria, defina la contraseña que debe usar para descargar o copiar el bloque.

Si, posteriormente, descarga un bloque con un enlace dinámico, debe introducir la contraseña para poder descargar el bloque. Tenga en cuenta que la contraseña de protección contra copia y la contraseña de protección de know-how (Página 94) son dos contraseñas diferentes.

Programación cada vez más sencilla 6

6.1 Fácil diseño del programa de usuario

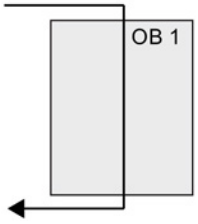
Al crear el programa de usuario para las tareas de automatización, las instrucciones del programa se insertan en bloques lógicos (OB, FB o FC).

Seleccionar el tipo de estructura del programa de usuario

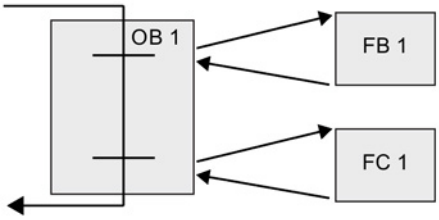
Según los requisitos de la aplicación, es posible seleccionar una estructura lineal o modular para crear el programa de usuario.

- 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 un OB de ciclo (como OB 1), encargado de la ejecución cíclica del programa.
- 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 tareas funcionales que se realizan en el 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:

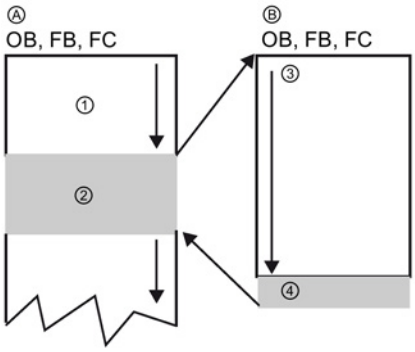


Estructura modular:



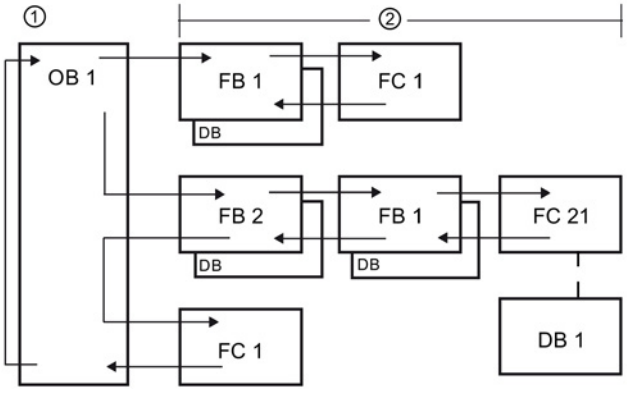
Diseñando FBs y FCs que ejecuten tareas genéricas, se crean bloques lógicos modulares. El programa de usuario se estructura luego, de manera que otros bloques lógicos llamen estos bloques modulares reutilizables. El bloque que efectúa la llamada transfiere los parámetros específicos del dispositivo al bloque llamado. Cuando un bloque lógico llama otro bloque lógico, la CPU ejecuta la lógica de programa contenida 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.

También se puede asignar un OB a un evento de alarma. Cuando se produce el evento, la CPU ejecuta el código del programa en el OB asociado. Tras finalizar la ejecución del OB, la CPU reanuda la ejecución en el punto del programa de usuario correspondiente al momento en que se produjo el evento de alarma, que podría ser cualquier punto del ciclo.



A Bloque que llama (o bloque interrumpido)
B FB o BC llamado (u OB que interrumpe)
① Ejecución del programa
② Instrucción (o evento que interrumpe) que inicia la ejecución de otro bloque
③ Ejecución del programa
④ Fin del bloque (regresa al bloque que llama)

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



① Inicio del ciclo
② Profundidad de anidamiento

Creando bloques lógicos genéricos que pueden reutilizarse en el programa de usuario, es posible simplificar el diseño y la implementación del programa de usuario.

- Se pueden 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 no sólo ayudan a estandarizar el diseño del programa, sino que también pueden facilitar y agilizar la actualización o modificación de la lógica del programa.
- La creación de componentes modulares 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.
- Utilizando un diseño modular basado en tareas funcionales específicas se puede reducir el tiempo necesario para la ejecución de toda la aplicación.

6.1.1 Uso de OBs para organizar el programa de usuario

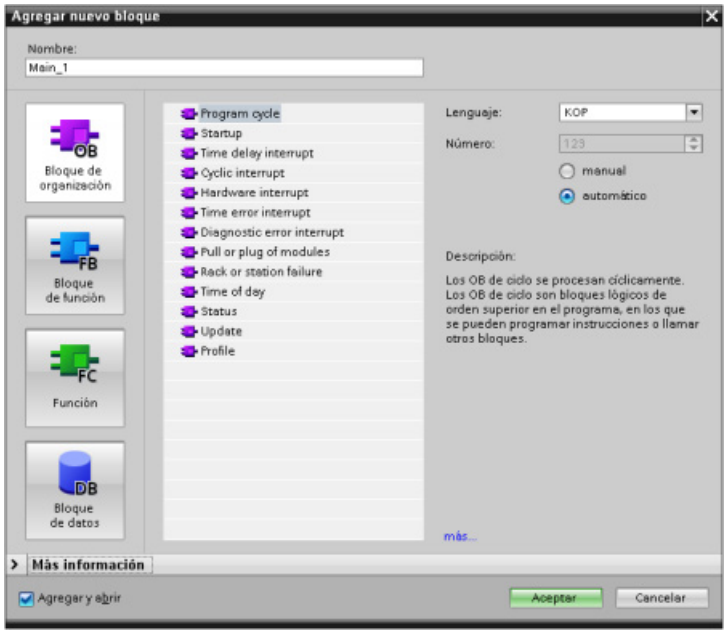
Los bloques de organización permiten estructurar el programa. Estos bloques sirven de interfaz entre el sistema operativo y el programa de usuario. Los OBs son controlados por eventos. Un evento, p. ej. una alarma de diagnóstico o un intervalo, hace que la CPU ejecute un OB. Algunos OB tienen eventos 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 el estado operativo RUN, los OB de ciclo se ejecutan en el nivel de prioridad más bajo y pueden ser interrumpidos por todos los demás tipos de eventos. 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 OB 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 OB 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 OB interrumpen la ejecución de los OB de ciclo.

Utilice el diálogo "Agregar nuevo bloque" para crear OB 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 los eventos de alarma por prioridad. Se pueden asignar múltiples eventos de alarma a la misma clase de prioridad. Para más información, consulte los temas relativos a bloques de organización (Página 61) y a la ejecución del programa de usuario (Página 60).

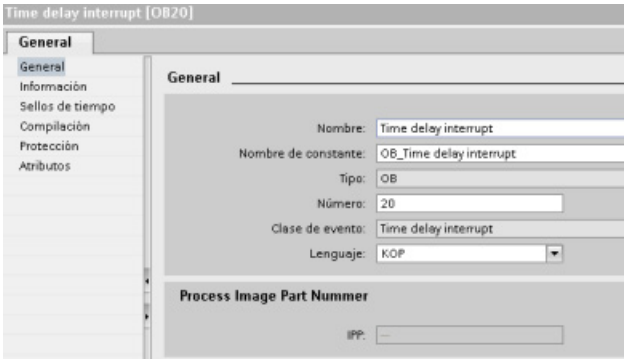
Creación de OB adicionales

Puede crear varios OB para el programa de usuario, incluso para los eventos de OB correspondientes a los OB de ciclo y de arranque. Use el cuadro de diálogo "Agregar nuevo bloque" para crear un OB e introduzca un nombre para el OB.

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 (p. ej. OB 1), la CPU ejecuta el OB de ciclo con el siguiente número más alto.

Configuración de las propiedades de un OB

Las propiedades de un OB pueden modificarse. Por ejemplo, se puede configurar el número de OB o el lenguaje de programación.



Nota

Recuerde que se puede asignar un número de memoria imagen parcial de proceso a un OB que se corresponda con IPP0, IPP1, IPP2, IPP3 o IPP4. Si introduce un número para la memoria imagen parcial de proceso, la CPU crea la memoria imagen parcial en cuestión. Consulte el tema "Ejecución del programa de usuario (Página 60)" para obtener una explicación sobre memorias imagen parciales de proceso.

6.1.2 Con los FBs y las FCs, la programación de tareas modulares es muy fácil

Una función (FC) es como una subrutina. Una FC es un bloque lógico que generalmente realiza una operación específica con una serie de valores de entrada. La FC almacena los resultados de esta operación en posiciones de memoria. Las FCs se utilizan para realizar las tareas siguientes:

- Para ejecutar operaciones estándar y reutilizables, p. ej. en cálculos matemáticos.
- Para ejecutar tareas funcionales, p. ej. controles individuales con operaciones lógicas con 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.

Al contrario que un FB, una FC no tiene un DB de instancia asociado. La FC usa su memoria temporal (L) para los datos utilizados para calcular la operación. Los datos temporales no se almacenan. Para almacenar los datos a fin de poder utilizarlos una vez ejecutada la FC, asigne el valor de salida a una posición de memoria global, p. ej. el área de marcas o un DB global.

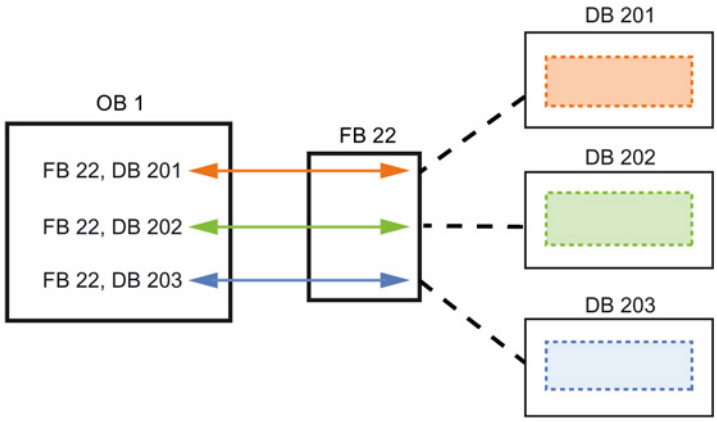
Un bloque de función (FB) es como una subrutina con memoria. Un FB es un bloque lógico cuyas llamadas pueden programarse mediante parámetros del bloque. El FB almacena los parámetros de entrada (IN), salida (OUT), y entrada/salida (IN_OUT) en una memoria variable integrada en un bloque de datos (DB), o en un DB "instancia". El DB de instancia ofrece un bloque de memoria asociado a esa instancia (o llamada) del FB y almacena datos una vez que haya finalizado el FB.

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 abre también un DB de instancia que almacena los valores de los parámetros del bloque y los datos locales estáticos de esa llamada o "instancia" del FB. Estos valores se almacenan en el DB de instancia una vez que finaliza el FB.

Los valores de arranque se asignan a los parámetros en la interfaz del FB. Estos valores se transfieren al DB de instancia asociado. Si no se asignan parámetros, se utilizan los valores almacenados actualmente en el DB de instancia. En algunos casos es necesario asignar parámetros.

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 de instancia. La CPU ejecuta entonces la lógica del programa en ese FB y almacena los parámetros del bloque y los datos locales estáticos en el DB de instancia. Cuando finaliza la ejecución del FB, la CPU regresa al bloque lógico que ha llamado el FB. El DB de instancia conserva los valores de esa instancia del FB. Si el FB se diseña para realizar tareas de control genéricas, es posible reutilizarlo para varios dispositivos, seleccionando diferentes DB de instancia para las distintas llamadas del FB.

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 de 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.

6.1.3 Los bloques de datos permiten almacenar fácilmente los datos del programa

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 de instancia almacena los datos de un bloque de función (FB) específico.

El programa de usuario puede almacenar los datos en las distintas áreas de memoria de la CPU, p. ej. en las áreas de entradas (I), salidas (Q) y marcas (M). Además, es posible utilizar un bloque de datos (DB) para acceder rápidamente a los datos almacenados en el programa.

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

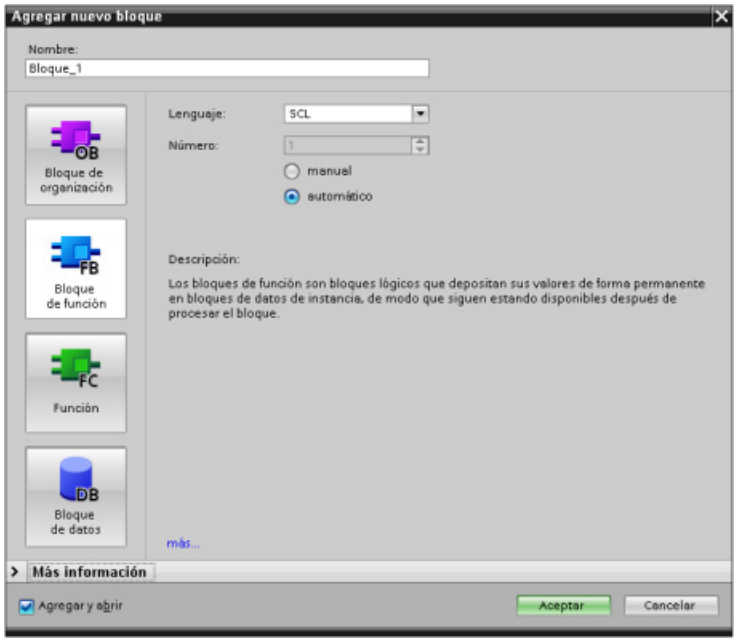
- Un DB global almacena los datos de los bloques lógicos en el programa. Cualquier OB, FB o FC puede acceder a los datos de un DB global.
- Un DB de instancia almacena los datos de un FB específico. La estructura de los datos en un DB de instancia refleja los parámetros (Input, Output e InOut) y los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

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

6.1.4 Crear un nuevo bloque lógico

Para agregar un nuevo bloque lógico al programa, proceda del siguiente modo:

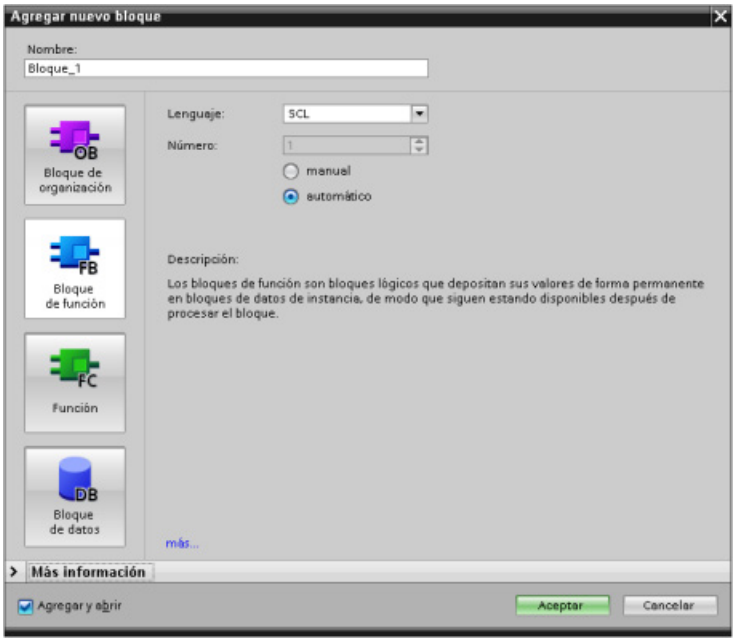
1. Abra la carpeta "Bloques de programa".
2. Haga doble clic en "Agregar nuevo bloque".
3. En el cuadro de diálogo "Agregar nuevo bloque", haga clic en el tipo de bloque que se desea agregar. Por ejemplo, haga clic en el icono "Función (FC)" para agregar una FC.
4. Seleccione el lenguaje de programación para el bloque lógico en el menú desplegable.



5. Haga clic en "Aceptar" para agregar el bloque al proyecto.

Al seleccionar la opción "Agregar nuevo y abrir" (predeterminada) STEP 7 abre el bloque lógico recién creado en el editor.

6.1.5 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 este solo almacena datos.

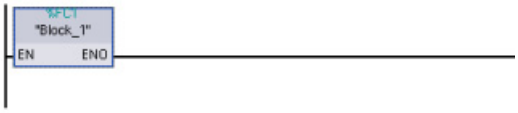
Activando la casilla de verificación "Agregar y abrir" (predeterminada) se abre el bloque lógico en la vista del proyecto.

Se pueden guardar en librerías objetos que se desea reutilizar. A cada proyecto hay conectada una librería de proyecto. Además de la librería del proyecto se puede crear un número indeterminado de librerías globales, que pueden utilizarse en varios proyectos. Dado que las librerías son compatibles entre sí, sus elementos pueden copiarse y moverse de una librería a otra.

Las librerías se utilizan, por ejemplo, para crear plantillas para bloques que primero se copian en la librería del proyecto y a continuación se desarrollan en él. Finalmente los bloques se copian de la librería del proyecto a una librería global. La librería global puede ser accesible para otros compañeros que trabajen en el mismo proyecto. Ellos utilizan los bloques y los adaptan a sus necesidades individuales, si es necesario.

Para más detalles sobre operaciones con librerías, consulte los temas relativos a librerías de la Ayuda en pantalla de STEP 7.

6.1.6 Llamar un bloque lógico desde otro bloque lógico



Se puede definir fácilmente la llamada de un FB o una FC de la CPU por cualquier bloque lógico (OB, FB o FC) en el programa de usuario.

1. Abra el bloque lógico que llamará otro bloque.
2. En el árbol del proyecto, seleccione el bloque lógico que se va a llamar.
3. Arrastre el bloque al segmento seleccionado para crear una llamada del bloque lógico.

Nota

El programa de usuario no puede llamar un OB porque los OBs están controlados por eventos (Página 62). La CPU inicia la ejecución del OB en respuesta a la recepción de un evento.

6.2 Lenguajes de programación fáciles de usar

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 106) de circuitos.
- FUP (diagrama de funciones) es un lenguaje de programación que se basa en los símbolos lógicos gráficos empleados en el álgebra (Página 107) booleana.
- SCL (structured control language) es un lenguaje de programación de alto nivel basado en texto (Página 108).

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.

6.2.1 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.

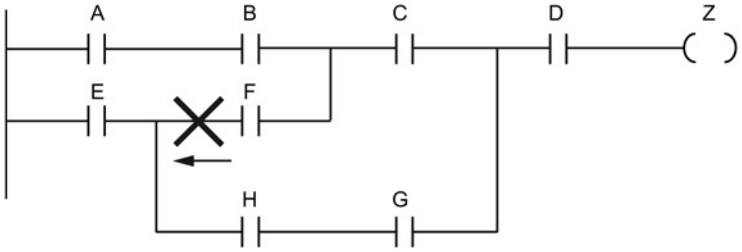
STEP 7 no limita el número de instrucciones (filas y columnas) de un segmento KOP.

Nota

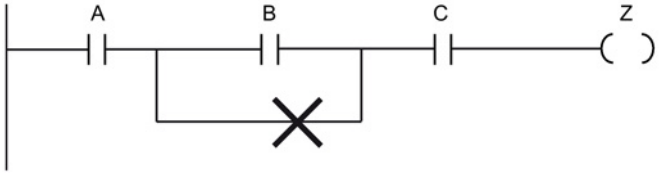
Todo segmento KOP debe terminar con una bobina o cuadro.

Tenga en cuenta las reglas siguientes al crear segmentos KOP:

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

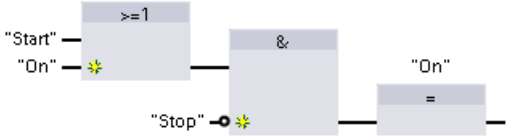


- No se permite programar ramas que causen cortocircuitos.



6.2.2 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.

6.2.3 Vista general SCL

El lenguaje de control estructurado (SCL, Structured Control Language) es un lenguaje de programación de alto nivel basado en PASCAL para las CPUs de SIMATIC S7. SCL soporta la estructura de los bloques de STEP 7. También se pueden 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 utiliza operaciones de control de programación PASCAL estándar, por ejemplo IF-THEN-ELSE, CASE, REPEAT-UNTIL, GOTO y RETURN. Existe la posibilidad de utilizar cualquier referencia de PASCAL para los 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.

Dado que SCL, al igual que PASCAL, ofrece estructuras de control condicionales de procesamiento, bucle y anidamiento, es más fácil implementar algoritmos complejos en SCL que en KOP y FUP.

Los ejemplos siguientes muestran diferentes expresiones para diversos usos:

`"C" := #A+#B;`	Asigna dos variables locales a una variable
`"Data_block_1".Tag := #A;`	Asignación a una variable de bloque de datos
`IF #A > #B THEN "C" := #A;`	Condición de la instrucción IF-THEN
`"C" := SQRT (SQR (#A) + SQR (#B));`	Parámetros de la instrucción SQRT

Como lenguaje de programación de alto nivel, SCL utiliza instrucciones estándar para tareas básicas:

- Instrucción de asignación: :=
- Funciones matemáticas: +, -, * y /
- Direccionamiento de variables globales: "<nombre de la variable>" (nombre de la variable o del bloque de datos entre comillas)
- Direccionamiento de variables locales: #<nombre de la variable> (nombre de la variable precedido por el carácter "#")
- Direccionamiento absoluto: %<dirección absoluta>, por ejemplo %I0.0 o %MW10

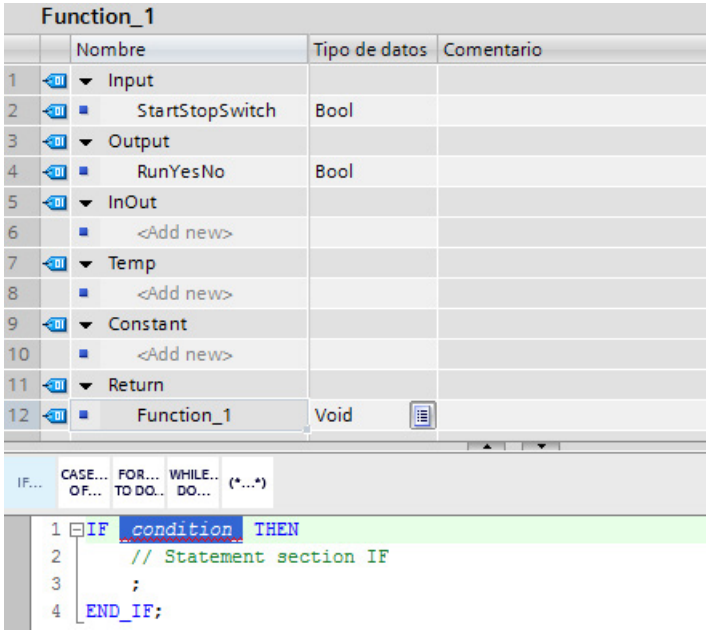
Los operadores aritméticos pueden procesar varios tipos de datos numéricos. El tipo de datos resultante lo determina el tipo de datos de los operandos más significativos. Por ejemplo, una operación de multiplicación que emplea un operando INT y un operando REAL da como resultado un valor REAL.

6.2.4 Editor de programas SCL

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. El editor incluye botones para las construcciones y comentarios de código habituales. 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.



En la sección de interfaz 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.
- Static (solo FB; la figura de arriba corresponde a una FC): el bloque lógico utiliza variables estáticas para almacenar resultados intermedios estáticos en el bloque de datos de instancia. El bloque retiene datos estáticos hasta que se sobrescriben, lo cual puede ocurrir después de varios ciclos. Los nombres de los bloques, que este bloque llama 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.
- Constant: son valores constantes con nombre para el 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.

6.3 Instrucciones eficaces que facilitan la programación

6.3.1 Disponer de las instrucciones básicas esperadas

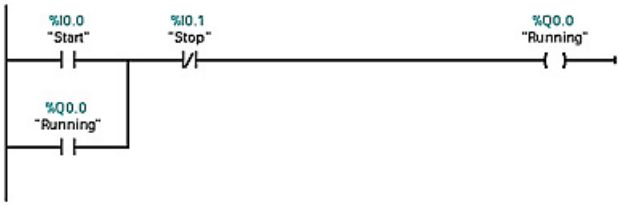
La CPU S7-1200 soporta numerosas instrucciones. Están disponibles en el árbol de instrucciones de STEP 7 en los siguientes grupos:

- Instrucciones básicas
- Instrucciones avanzadas
- Tecnología
- Instrucciones de comunicación

Encontrará una relación completa de todas las instrucciones en el *Manual de sistema del controlador programable S7-1200.* El manual describe muchas de las instrucciones comunes.

Operaciones lógicas con bits

La base de las instrucciones lógicas con bits son los contactos y las bobinas. Los contactos leen el estado de un bit y las bobinas escriben el estado de la operación en un bit.



Los contactos prueban el estado binario del bit: el resultado es "flujo de corriente" para on (1) o "sin flujo de corriente" para off (0).

El estado de la bobina refleja el estado de la lógica precedente.

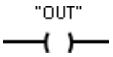
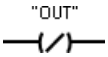
Si utiliza una bobina con la misma dirección en más de una ubicación del programa, el resultado del último cálculo en el programa de usuario determina el estado del valor que se escribe en la salida física durante la actualización de las salidas.

Contacto normalmente abierto	Contacto normalmente cerrado	
"IN"	"IN"	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.

La estructura básica de una operación lógica con bits puede ser una Y lógica o una O lógica. Los contactos conectados en serie crean segmentos lógicos Y. Los contactos conectados en paralelo crean segmentos lógicos O.

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 de control se cablean con los bornes de entrada del PLC. La CPU consulta las señales de entrada cableadas y actualiza continuamente los estados lógicos en la memoria imagen de proceso de las entradas.

La lectura inmediata de una entrada física se especifica introduciendo ":P" después de la variable de una entrada (p. ej. "Arranque_motor:P" o "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.

Bobina de relé, salida	Bobina de relé negada
	

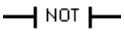
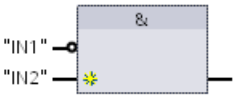
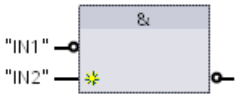
Anote los siguientes resultados de salida para flujo de corriente a través de bobinas de salida y de relé negadas:

- Si fluye corriente a través de una bobina de salida, el bit de salida se pone a 1.
- Si no fluye corriente a través de una bobina de salida, el bit de la bobina de salida se pone a 0.
- Si fluye corriente a través de una bobina de relé negada, el bit de salida se pone a 0.
- Si no fluye corriente a través de una bobina de relé negada, el bit de salida se pone a 1.

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 de salida del PLC. En el modo RUN, el sistema CPU explora las señales de entrada, 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.

La escritura inmediata de una salida física se especifica introduciendo ":P" después de la variable de una salida (p. ej. "Motor_on:P" o "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.

Las bobinas no tienen que estar necesariamente al final de un segmento. Se pueden insertar en el centro de un circuito del segmento KOP, entre contactos u otras instrucciones.

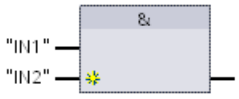
Contacto negado NOT (KOP)	Cuadro Y con una entrada lógica negada (FUP)	Cuadro Y con entrada y salida lógica negada (FUP)
		

El contacto NOT KOP invierte el estado lógico de la entrada de flujo de corriente.

- 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.

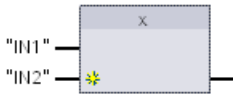
En la programación FUP es posible arrastrar la función "Invertir RLO" desde la barra de herramientas "Favoritos" o desde el árbol de instrucciones y soltarla en una entrada o salida para crear un inversor lógico en ese conector del cuadro.

Cuadro Y (FUP)



Cuadro O (FUP)

Cuadro O-exclusiva (FUP)



- Todas las entradas de un cuadro Y tienen que cumplirse para que la salida sea TRUE (verdadera).
- Una entrada cualquiera de un cuadro O tiene que cumplirse para que la salida sea TRUE (verdadera).
- Un número impar de entradas de un cuadro O-exclusiva tiene que cumplirse para que la salida sea TRUE (verdadera).

En la programación FUP, los segmentos de los contactos KOP se representan en segmentos con cuadros Y (&), O (>=1) y O-EXCLUSIVA (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 disponer 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".

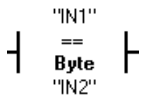
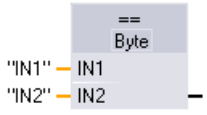
Es posible conectar las entradas y salidas de los cuadros con un cuadro lógico diferente, o bien introducir una dirección de bit o un nombre simbólico de bit para una entrada no conectada. Cuando se ejecuta el cuadro, los estados actuales de las entradas se aplican a la lógica del cuadro binario y, si se cumplen, la salida del cuadro será verdadera.

6.3.2 Instrucciones de comparación y desplazamiento

Las operaciones de comparación llevan a cabo una comparación de dos valores del mismo tipo de datos.

Tabla 6- 1 Operaciones de comparación

Instrucción	SCL	Descripción

KOP:
 
 	
FUP:
 
 	
	`out := in1 = in2;`
	`out := in1 <> in2;`
	`out := in1 >= in2;`
	`out := in1 <= in2;`
	`out := in1 > in2;`
	`out := in1 < in2;`
		• Igual a (==): la comparación es verdadera si IN1 es igual a IN2
		• Distinto de (<>): La comparación es verdadera si IN1 es distinto de IN2
		• Mayor o igual que (>=): la comparación es verdadera si IN1 es mayor o igual que IN2
		• Menor o igual que (>=): la comparación es verdadera si IN1 es mayor o igual que IN2
		• Mayor que (>): La comparación es verdadera si IN1 es mayor que IN2
		• Menor que (<): La comparación es verdadera si IN1 es menor que IN2

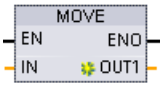
[1] En KOP y FUP: El contacto se activa (KOP) o la salida del cuadro es TRUE (FUP) si la comparación es TRUE.

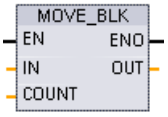
Para más operaciones de comparación, consulte el *Manual de sistema del controlador programable S7-1200*.

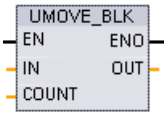
Las operaciones de desplazamiento permiten copiar elementos de datos a otra dirección de memoria y convertir un tipo de datos en otro. El proceso de desplazamiento no modifica los datos de origen.

- MOVE copia un elemento de datos almacenado en una dirección indicada a una dirección diferente. Para agregar otra salida, haga clic en el icono situado junto al parámetro OUT1.
- MOVE_BLK (desplazamiento con interrupciones) y UMOVE_BLK (desplazamiento sin interrupciones) copian un bloque de elementos de datos en otra dirección. 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 6- 2 Instrucciones MOVE, MOVE_BLK y UMOVE_BLK

KOP / FUP	SCL	Descripción
	`out1 := in;`	Copia un elemento de datos almacenado en una dirección indicada a una o varias direcciones diferentes. Para agregar otra salida en KOP o FUP, haga clic en el icono situado junto al parámetro de salida. Para SCL, utilice múltiples instrucciones de asignación. También se puede usar una de las construcciones de bucle.

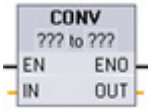
	`MOVE_BLK(in:=_variant_in,`
	`count:=_uint_in,`
	`out=>_variant_out);`
		Desplazamiento con interrupciones que copia un bloque de elementos de datos a otra dirección.

	`UMOVE_BLK(in:=_variant_in,`
	`count:=_uint_in`
	`out=>_variant_out);`
		Desplazamiento sin interrupciones que copia un bloque de elementos de datos a otra dirección.

Para más operaciones de desplazamiento, consulte el *Manual de sistema S7-1200*.

6.3.3 Operaciones de conversión

Tabla 6- 3 Operaciones de conversión

KOP / FUP	SCL	Descripción
	`out := <data type in>_TO_<data type out>(in);`	Convierte un elemento de datos de un tipo de datos a otro tipo de datos.

[1] En KOP y FUP: Haga clic debajo del nombre del cuadro y seleccione los tipos de datos en el menú desplegable. Tras haber seleccionado el tipo de datos que se desea convertir, las conversiones posibles aparecen en la lista desplegable (convertir a).

[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 6- 4 Instrucciones Round y Truncate

KOP / FUP	SCL	Descripción
	`out := ROUND (in);`	Convierte un número real (Real o LReal) en un entero. La instrucción redondea el número real al entero más cercano (IEEE - redondear al entero más cercano). Si el número se encuentra exactamente entre dos enteros (p. ej. 10,5), la instrucción redondeará el número al entero par. Ejemplo: • ROUND (10.5) = 10 • ROUND (11.5) = 12 Para KOP/FUP, haga clic en "???" del cuadro de la instrucción para seleccionar el tipo de datos para la salida, por ejemplo, "DInt". Para SCL, el tipo de datos predeterminado es DINT. Para redondear a otro tipo de datos de salida, introduzca el nombre de la instrucción con el nombre explícito del tipo de datos, por ejemplo, ROUND_REAL o ROUND_LREAL.
	`out := TRUNC(in);`	Convierte un número real (Real o LReal) en un entero. La parte fraccionaria del número real se trunca a cero (IEEE - redondear hacia cero).

Tabla 6- 5 Instrucciones límite (CEIL) y Floor

KOP / FUP	SCL	Descripción
CEIL Real to DInt EN ENO IN OUT	`out := CEIL(in);`	Convierte un número real (Real o LReal) en el siguiente entero mayor o igual a ese número real (IEEE - redondear hacia el infinito positivo).
FLOOR Real to DInt EN ENO IN OUT	`out := FLOOR(in);`	Convierte un número real (Real o LReal) en el siguiente entero menor o igual a ese número real (IEEE - redondear hacia el infinito negativo).

Tabla 6- 6 Instrucciones SCALE_X y NORM_X

KOP / FUP	SCL	Descripción

SCALE_X
Real to ???
EN ENO
MIN OUT
VALUE
MAX
	`out := SCALE_X(`
	`min:=_in_,`
	`value:=_in_,`
	`max:=_in_);`
		Escala el parámetro VALUE real normalizado (donde 0,0 <= VALUE <= 1,0) al tipo de datos y rango de valores especificados por los parámetros MIN y MAX:
		OUT = VALUE (MAX - MIN) + MIN

NORM_X
??? to Real
EN ENO
MIN OUT
VALUE
MAX
	`out := NORM_X(`
	`min:=_in_,`
	`value:=_in_,`
	`max:=_in_);`
		Normaliza el parámetro VALUE dentro del rango de valores especificado por los parámetros MIN y MAX:
		OUT = (VALUE - MIN) / (MAX - MIN),
		donde (0,0 <= OUT <= 1,0)

[1] Equivalente en SCL: `out := value (max-min) + min;` [2] Equivalente en SCL: `out := (value-min)/(max-min);`

6.3.4 Cálculos matemáticos sencillos con la instrucción Calcular

Tabla 6- 7 Instrucción CALCULATE

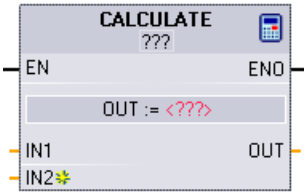
KOP / FUP	SCL	Descripción
	Utilice las expresiones matemáticas SCL estándar para crear la ecuación.	La instrucción CALCULATE permite crear una función matemática que funciona con entradas (IN1, IN2, ... INn) y genera el resultado en OUT, según la ecuación definida. • En primer lugar, seleccione un tipo 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.

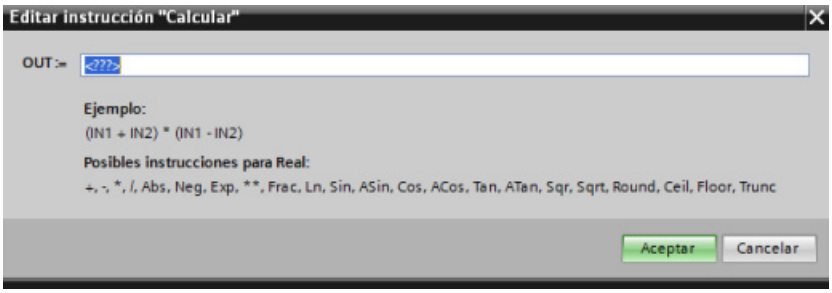
Tabla 6- 8 Tipos de datos de los parámetros

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.

El cuadro de diálogo muestra un ejemplo y una lista de posibles instrucciones que pueden incluirse en función del tipo de datos del parámetro OUT:



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.

Para un ejemplo de la instrucción CALCULATE, véase "Uso de la instrucción CALCULAR para una ecuación matemática compleja (Página 50)".

6.3.5 Operaciones de temporizadores

El S7-1200 soporta los temporizadores siguientes

- El temporizador TP genera un impulso con una duración predeterminada.
- El temporizador TON activa la salida (Q) en estado ON al cabo de un tiempo de retardo predeterminado.
- El temporizador TOF activa la salida (Q) en estado ON y, a continuación, la pone al estado OFF al cabo de un tiempo de retardo predeterminado.
- El temporizador TONR activa la salida (Q) en estado ON al cabo de un tiempo de retardo predeterminado. El tiempo transcurrido se acumula a lo largo de varios periodos de temporización hasta que la entrada de reset (R) se emplea para poner a cero el tiempo transcurrido.
- La bobina PT (Cargar tiempo) carga un nuevo valor de tiempo predeterminado en el temporizador especificado.
- La bobina RT (Inicializar temporizador) inicializa el temporizador especificado.

En KOP y FUP, estas instrucciones están disponibles como instrucción de cuadro o bobina de salida.

El número de temporizadores que pueden utilizarse en el programa de usuario está limitado solo por la cantidad de memoria disponible en la CPU. Cada temporizador utiliza 16 bytes de memoria.

Todos los temporizadores utilizan una estructura almacenada en un bloque de datos para mantener los datos. En SCL, hay que crear primero el DB de la instrucción de temporizador individual antes de poder emplearla. En KOP y FUP, STEP 7 crea automáticamente el DB al introducir la instrucción.

Al crear el DB, también puede usarse un DB multiinstancia. Como los datos del temporizador se incluyen en un DB sencillo y no hace falta crear un DB para cada temporizador, se reduce el tiempo de procesamiento en la gestión de los temporizadores. No hay interacción entre las estructuras de datos de los temporizadores en el DB multiinstancia compartido.

Tabla 6- 9 TP (impulso)

KOP / FUP	SCL	Cronograma

IEC_Timer_0
TP
Time
IN Q
PT ET
TP_DB
TP
"PRESET_Tag"
	`"timer_db".TP(`
	`IN:=_bool_in_,`
	`PT:=_time_in_,`
	`Q=>_bool_out_,`
	`ET=>_time_out_);`
		IN
		ET
		PT
		Q
		PT PT PT

Tabla 6- 10 TON (retardo al conectar)

KOP / FUP	SCL	Cronograma

IEC_Timer_1
TON
Time
IN Q
PT ET
TON_DB
TON
"PRESET_Tag"
	`"timer_db".TON(`
	`IN:=_bool_in_,`
	`PT:=_time_in_,`
	`Q=>_bool_out_,`
	`ET=>_time_out_);`
		IN
		ET
		PT
		Q
		PT PT

Tabla 6- 11 TOF (retardo al desconectar)

KOP / FUP	SCL	Cronograma

IEC_Timer_2
TOF
Time
IN Q
PT ET
TOF_DB
TOF
"PRESET_Tag"
	`"timer_db".TOF(`
	`IN:=_bool_in_,`
	`PT:=_time_in_,`
	`Q=>_bool_out_,`
	`ET=>_time_out_);`
		IN
		ET
		PT
		Q
		PT
		PT

Tabla 6- 12 TONR (acumulador de tiempo)

KOP / FUP	SCL	Cronograma

IEC_Timer_3
TONR
Time
IN Q
R ET
PT
TONR_DB
TONR
"PRESET_Tag"
	`"timer_db".TONR(`
	`IN:=_bool_in_,`
	`R:=_bool_in_,`
	`PT:=_time_in_,`
	`Q=>_bool_out_,`
	`ET=>_time_out_);`
		IN
		ET
		PT
		Q
		R

Tabla 6- 13 Instrucciones de bobina Cargar tiempo -(PT)- e Inicializar temporizador -(RT)-

KOP / FUP	SCL	Descripción

PT
PT
TON_DB
PT
"PRESET_Tag"
	`PRESET_TIMER(`
	`PT:=_time_in_,`
	`TIMER:=_iec_timer_in_)`
	`;`
		Las instrucciones de bobina Cargar temporizador -(PT)- E Inicializar temporizador -(RT)- pueden usarse con cualquier temporizador de bobina o cuadro. Estas instrucciones de bobina pueden colocarse en una posición media. El estado del flujo de salida de la bobina siempre es el mismo que el estado de entrada de la bobina.
		• Cuando se activa la bobina -(PT)-, el elemento de tiempo PRESET de los datos del DB IEC_Timer especificados se ajusta a la duración "PRESET_Tag".
		• Cuando se activa la bobina -(RT)-, el elemento de tiempo transcurrido de los datos de DB IEC_Timer especificados se pone a 0.

RT TON_DB RT	`RESET_TIMER(` `_iec_timer_in_);`	

Tabla 6- 14 Tipos de datos de 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	Solo cuadro TONR: 0=sin reset 1= resetear el tiempo transcurrido y poner 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 del cuadro Q o bit Q en los datos del DB de temporizador Bobina de temporizador: solo es posible direccionar el bit Q en los datos del DB de temporizador
Cuadro: ET Bobina: DBdata.ET	Time	Cuadro de temporizador: salida del cuadro ET (tiempo transcurrido) o valor de tiempo ET en los datos del DB de temporizador Bobina de temporizador: solo es posible direccionar el valor de tiempo ET en los datos del DB de temporizador.

Tabla 6- 15 Efecto de los cambios de valores en los parámetros PT e IN

Temporizador	Cambios en los parámetros de cuadro PT eIN y en los parámetros de bobina correspondientes
TP	• El cambio de PT no tiene efecto alguno durante el funcionamiento del temporizador. • El cambio de IN no tiene efecto alguno durante el funcionamiento del temporizador.
TON	• El cambio de PT no tiene efecto alguno durante el funcionamiento del temporizador. • Si IN cambia a FALSE durante el funcionamiento del temporizador, este se inicializará y se detendrá.
TOF	• El cambio de PT no tiene efecto alguno durante el funcionamiento del temporizador. • Si IN cambia a TRUE durante el funcionamiento del temporizador, este se inicializará y se detendrá.
TONR	• El 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, este 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 el dato IEC_TIMER DB especificado como enteros dobles con signo que representan el tiempo en milisegundos. Los datos TIME emplean el identificador T# y se pueden introducir como unidad de tiempo simple (T#200ms o 200) y como unidades de tiempo compuestas como T#2s_200ms.

Tabla 6- 16 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 del temporizador. Los valores PT (tiempo predeterminado) negativos se ponen a cero cuando se ejecuta la instrucción del temporización. ET (tiempo transcurrido) es siempre un valor positivo.

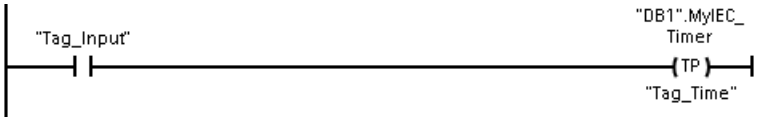
Programación de temporizadores

A la hora de planificar y crear el programa de usuario deben considerarse las siguientes consecuencias del manejo de temporizadores:

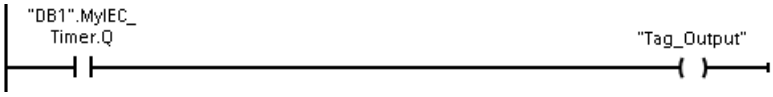
- Pueden producirse múltiples actualizaciones de un temporizador en el mismo ciclo. El temporizador se actualiza cada vez que la instrucción de temporizador (TP, TON, TOF, TONR) se ejecuta y cada vez que el miembro ELAPSED o Q de la estructura de temporizador se utiliza como parámetro para otra instrucción 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 solo 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 temporizador 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 este se actualiza debido al contacto normalmente cerrado, este inicializará la instrucción de temporizador inmediatamente. La salida Q de la instrucción de temporizador no estará conectada para el ciclo en cuestión, y los segmentos dependientes no se ejecutarán.

Las bobinas de temporizador -(TP)-, -(TON)-, -(TOF)- y -(TONR)- deben ser la última instrucción de una red. 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=1 y el valor de Tag_Output=1. Cuando haya transcurrido el valor de Tag_Time, DB1.MyIEC_Timer.Q=0 y el valor de Tag_Output=0.

6.3.6 Operaciones de contadores

Las instrucciones con contadores se utilizan para contar eventos del programa internos y eventos del proceso externos.

- El contador "ascendente" (CTU) se incrementa en 1 cuando el valor del parámetro de entrada CU cambia de 0 a 1.
- El contador "descendente" (CTD) se reduce en 1 cuando el valor del parámetro de entrada CD cambia de 0 a 1.
- El contador "ascendente y descendente" (CTUD) se incrementa o se reduce en 1 durante la transición de 0 a 1 de las entradas de conteo ascendente (CU) o descendente (CD).

S7-1200 también ofrece contadores rápidos (Página 136) (HSC) para el contaje de eventos que se producen con mayor rapidez que la frecuencia de ejecución del OB.

Las instrucciones CU, CD y CTUD utilizan contadores de software cuya frecuencia de contaje máxima está limitada por la frecuencia de ejecución del OB en el que se encuentran.

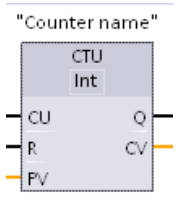
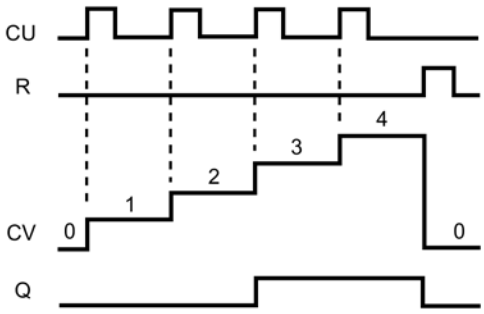
Nota

Si los eventos que se deben contar se producen con la frecuencia de ejecución del OB, utilice las instrucciones de contaje CTU, CTD o CTUD. Si el evento se produce con mayor rapidez que la frecuencia de ejecución del OB, utilice el HSC.

Todo contador utiliza una estructura almacenada en un bloque de datos para conservar sus datos. En SCL, hay que crear primero el DB de la instrucción de contador individual antes de poder emplearla. En KOP y FUP, STEP 7 crea automáticamente el DB al introducir la instrucción.

El número de contadores que pueden utilizarse en el programa de usuario está limitado solo por la cantidad de memoria disponible en la CPU. Los contadores individuales emplean 3 bytes (para SInt o USInt), 6 bytes (para Int o UInt) o 12 bytes (para DInt o UDInt).

Tabla 6- 17 Contador (ascendente) CTU

KOP / FUP	SCL	Funcionamiento
	"ctu_db".CTU(CU:=_bool_in, R:=_bool_in, PV:=_in_, Q=>_bool_out, CV=>_out_);	

El cronograma muestra el funcionamiento de un contador CTU con un valor de contaje de entero sin signo (donde PV = 3).

- 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 inicialización R cambia de 0 a 1, CV se pone a 0.

Tabla 6- 18 Contador (descendente) CTD

KOP / FUP	SCL	Funcionamiento

"Counter name"
CTD
Int
CD Q
LD CV
PV
	`"ctd_db".CTD(`
	`CD:=_bool_in,`
	`LD:=_bool_in,`
	`PV:=_in_,`
	`Q=>_bool_out,`
	`CV=>_out_);`
		CD
		LD
		CV 0 3 2 1 0 3 2
		Q

El cronograma muestra el funcionamiento de un contador CTD con un valor de contaje de entero sin signo (donde PV = 3).

- 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 LD cambia de 0 a 1, el valor del parámetro PV (valor predeterminado) se carga en el contador como nuevo CV.

Tabla 6- 19 Contador (ascendente y descendente) CTUD

KOP / FUP	SCL	Funcionamiento

"Counter name"
CTUD
Int
CU QU
CD QD
R CV
LD
PV
	`"ctud_db".CTUD(`
	`CU:=_bool_in,`
	`CD:=_bool_in,`
	`R:=_bool_in,`
	`LD:=_bool_in,`
	`PV:=_in_,`
	`QU=>_bool_out,`
	`QD=>_bool_out,`
	`CV=>_out_);`
		CU
		CD
		R
		LD
		CV 0 1 2 3 4 5 4 3 4 5 0
		QU
		QD

El cronograma muestra el funcionamiento de un contador CTUD con un valor de contaje de entero sin signo (donde PV = 4).

- Si el valor del parámetro CV (valor de contaje actual) es superior o igual que el del parámetro PV (valor predeterminado), 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 LD cambia de 0 a 1, el valor del parámetro PV se carga en el contador como nuevo CV.
- Si el valor del parámetro de inicialización R cambia de 0 a 1, CV se pone a 0.

6.3.7 Modulación del ancho de pulso (PWM)

La instrucción CTRL_PWM está disponible en el grupo Impulsos de las instrucciones avanzadas.

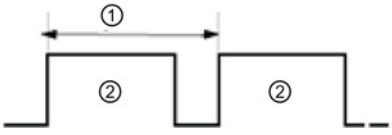
Tabla 6- 20 Instrucción CTRL_PWM

KOP / FUP	SCL	Descripción

	`"ctrl_pwm_db"(`
	`PWM:=W#16#0,`
	`ENABLE:=False,`
	`BUSY=>_bool_out_,`
	`STATUS=>_word_out_);`
		La instrucción CTRL_PWM ofrece una salida de tiempo de ciclo fijo 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.

Insertando la instrucción CTRL_PWM en al bloque lógico se crea el DB para la instrucción desde el cuadro de diálogo "Opciones de llamada". La instrucción CTRL_PWM almacena la información de parámetros en el DB y controla los parámetros del bloque de datos.

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 salida de palabra (Q) indicada en la configuración de dispositivos ("Direcciones de salida" / "Dirección inicial:") según sea necesario para modificar la duración de impulso. Utilice una instrucción (p. ej. Move, Convert, Math o PID) para escribir el ancho de impulso especificado en la salida (Q) adecuada. El valor de salida debe estar comprendido en el rango válido (porcentaje, milésimos, diezmilésimos o formato analógico S7).



El ciclo de carga puede expresarse, por ejemplo, porcentualmente respecto al tiempo de ciclo o como cantidad relativa (p. ej. 0:1000 o 0:10000). La duración de impulso puede variar entre 0 (sin impulso, siempre off) y escala completa (sin impulso, siempre on).

① Tiempo de ciclo

② Duración de impulso

La salida PWM puede variar entre 0 y escala completa, proporcionando así una salida digital que, en numerosos aspectos, es igual 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").

6.4 Registros de datos fáciles de crear

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.

Los archivos de registro se pueden abrir, editar, renombrar y borrar desde la página del explorador de archivos del servidor web. Se deben tener derechos de lectura para ver el explorador de archivos y derechos de modificación para editar, borrar o cambiar el nombre de los archivos de registro.

Use las instrucciones DataLog para que el programa almacene los datos del proceso de tiempo de ejecución en la memoria flash de la CPU. Los registros de datos se organizan como un archivo de registro circular con un tamaño predeterminado. Los registros nuevos se añaden al archivo de registro. Una vez que el archivo de registro ha almacenado el número máximo de registros, el siguiente registro sobrescribe el registro más antiguo. Para evitar la sobrescritura de los registros de datos, use la instrucción DataLogNewFile. Los registros de datos nuevos se almacenan en el archivo de registro nuevo, mientras que el archivo de datos antiguo permanece en la CPU.

Tabla 6- 21 Instrucción DataLogWrite

KOP / FUP	SCL	Descripción

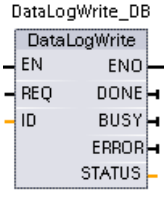
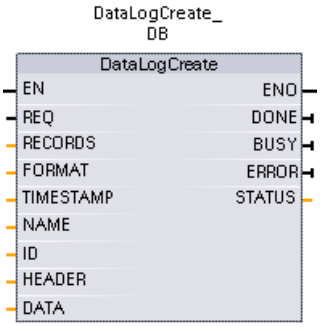
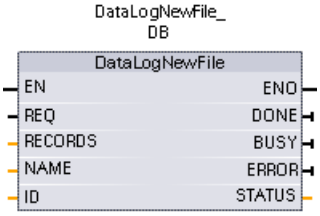
	`"DataLogWrite_DB"(`
	`req:=FALSE,`
	`done=>_bool_out_,`
	`busy=>_bool_out_,`
	`error=>_bool_out_,`
	`status=>_word_out_,`
	`ID:=_dword_inout_);`
		DataLogWrite escribe un registro de datos en el registro especificado. El registro de destino previo debe estar abierto.
		Hay que emplear el programa para cargar el búfer de registro con valores de datos de tiempo de ejecución actuales y, a continuación, ejecutar la instrucción DataLogWrite para desplazar los datos de registro nuevos del búfer al registro.
		Si se produce un fallo de alimentación durante una operación DataLogWrite incompleta, se podría perder el registro de datos que se estuviera transfiriendo al registro.

Tabla 6- 22 Instrucciones DataLogCreate y DataLogNewFile

KOP / FUP	SCL	Descripción

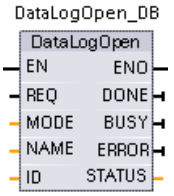
	`"DataLogCreate_DB"(`
	`req:=FALSE,`
	`records:=1,`
	`format:=1,`
	`timestamp:=1,`
	`done=>_bool_out_,`
	`busy=>_bool_out_,`
	`error=>_bool_out_,`
	`status=>_word_out_,`
	`name:=_variant_in_,`
	`ID:=_dword_inout_,`
	`header:=_variant_inout_,`
	`data:=_variant_inout_);`
		DataLogCreate[1] crea e inicializa un archivo de registro que se guarda en el directorio \DataLogs de la CPU. El archivo de registro se crea con un tamaño fijo predeterminado.

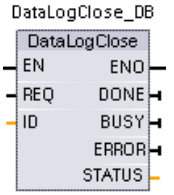
	`"DataLogNewFile_DB"(`
	`req:=FALSE,`
	`records:=1,`
	`done=>_bool_out_,`
	`busy=>_bool_out_,`
	`error=>_bool_out_,`
	`status=>_word_out_,`
	`name=:_variant_in_,`
	`ID:=_dword_inout_);`
		DataLogNewFile[1] permite al programa crear un archivo de registro nuevo basado en otro ya existente. Se creará y abrirá un nuevo registro con el valor NAME especificado. El registro del encabezado se duplicará a partir del registro original, junto con las propiedades del registro original. El archivo de registro original se cerrará.

[1] Las operaciones DataLogCreate y DataLogNewFile se extiende a lo largo de varios ciclos. El tiempo actual requerido para crear un archivo de registro depende de la estructura y del número de registros. Antes de que el nuevo archivo de registro pueda utilizarse para otras operaciones de registro, la lógica del programa debe monitorizar la transición del bit DONE a TRUE.

Tabla 6- 23 Instrucciones DataLogOpen y DataLogClose

KOP / FUP	SCL	Descripción

	`"DataLogOpen_DB"(`
	`req:=FALSE,`
	`mode:=0,`
	`name:=_variant_in_,`
	`done=>_bool_out_,`
	`busy=>_bool_out_,`
	`error=>_bool_out_,`
	`status=>_word_out_,`
	`ID:=_dword_inout_);`
		La instrucción DataLogOpen 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. Se pueden abrir ocho registros simultáneamente.

	`"DataLogClose_DB"(`
	`req:=FALSE,`
	`done=>_bool_out_,`
	`busy=>_bool_out_,`
	`error=>_bool_out_,`
	`status=>_word_out_,`
	`ID:=_dword_inout_);`
		La instrucción DataLogClose cierra un archivo de registro abierto. El empleo de DataLogWrite sobre un registro cerrado produce un error. No se permite realizar operaciones de escritura en este registro hasta que no se lleve a cabo otra operación DataLogOpen.
		La transición al estado operativo STOP cierra todos los archivos de registro abiertos.

6.5 Programa de usuario fácil de controlar y comprobar

6.5.1 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 350) 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").

STEP 7 también ofrece la capacidad de trazar y registrar las variables de programa en función de las condiciones de disparo (Página 364).

6.5.2 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 solo 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- 24 Elementos de la referencia cruzada

Columna	Descripción
Objeto	Nombre del objeto que utiliza los objetos del nivel inferior o que es utilizado por éstos.
Número	Número de utilizaciones
Ubicación (lugar de uso)	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

Dependiendo de los productos instalados, la tabla de referencias cruzadas mostrará columnas adicionales o diferentes.

6.5.3 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 solo muestra aquellos bloques que son llamados por un bloque lógico.

Dentro de la estructura de llamadas se pueden visualizar selectivamente solo 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.

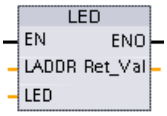
6.5.4 Instrucciones de diagnóstico para controlar el hardware

6.5.4.1 Lectura de los estados de los LEDs de la CPU

Con la instrucción LED el programa de usuario puede determinar el estado de los LEDs de la CPU. Esta información puede utilizarse para programar una variable para el dispositivo HMI.

Tabla 6- 25 Instrucción LED

KOP / FUP	SCL	Descripción

	`ret_val := LED(`
	`laddr:=_word_in_,`
	`LED:=_uint_in_);`
		RET_VAL devuelve los estados de LED siguientes para la CPU
		• RUN/STOP: verde o amarillo
		• Error: rojo
		• MAINT (mantenimiento): amarillo
		• Conexión: verde
		• Tx/Rx (transmisión/recepción): amarillo

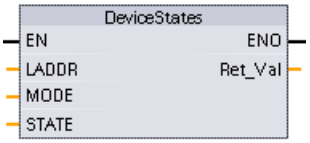
6.5.4.2 Instrucciones para leer el estado de diagnóstico de los dispositivos

STEP 7 también incluye instrucciones para leer la información de estado que ofrecen los dispositivos de hardware de la red.

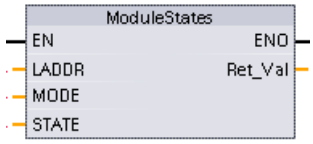
Tabla 6- 26 Instrucciones de diagnóstico

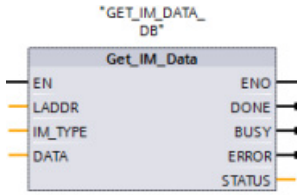
KOP / FUP	SCL	Descripción

	`ret_val := GET_DIAG(`
	`mode:=_uint_in_,`
	`laddr:=_word_in_,`
	`cnt_diag=>_uint_out_,`
	`diag:=_variant_inout_,`
	`de-`
	`tail:=_variant_inout_);`
		La instrucción GET_DIAG lee la información de diagnóstico de un dispositivo de hardware especificado.

	`ret_val := DeviceStates(`
	`laddr:=_word_in_,`
	`mode:=_uint_in_,`
	`state:=_variant_inout_);`
		La instrucción DeviceStates lee el estado de los dispositivos PROFINET o PROFIBUS.

KOP / FUP	SCL	Descripción

	`ret_val := ModuleStates(`
	`laddr:=_word_in_,`
	`mode:=_uint_in,`
	`state:=_variant_inout);`
		La instrucción ModuleStates lee el estado de los módulos PROFINET o PROFIBUS.

	`"GET_IM_DATA_DB"(LADDR:=16#0,`
	`IM_TYPE:=0,`
	`DONE=>_bool_out_,`
	`BUSY=>_bool_out_,`
	`ERROR=>_bool_out_,`
	`STATUS=>_word_out_,`
	`DATA:=_variant_inout_);`
		Utilice la instrucción Get_IM_Data para comprobar los datos de identificación y mantenimiento (I&M) del módulo o submódulo especificado.

6.6 Contador rápido (HSC)

Utilice los contadores rápidos (HSC) para el contaje de eventos que se producen con mayor rapidez que la frecuencia de ejecución del OB. Las instrucciones de contadores se encuentran en la sección tecnológica del árbol de instrucciones. La instrucción CTRL_HSC controla el funcionamiento del HSC.

Nota

Si los eventos que se deben contar se producen con la frecuencia de ejecución del OB, utilice las instrucciones de contaje CTU, CTD o CTUD. Si el evento se produce con mayor rapidez que la frecuencia de ejecución del OB, utilice el HSC.

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 o el período de impulsos.

Tabla 6- 27 Instrucción CTRL_HSC

KOP / FUP	SCL	Descripción

"Counter name"
CTRL_HSC
EN, HSC, DIR, CV, RV, PERIOD, NEW_DIR, NEW_CV, NEW_RV, NEW_PERIOD
ENO, BUSY, STATUS
	`"counter_name"(`
	`HSC:=W#16#0,`
	`DIR:=FALSE,`
	`CV:=FALSE,`
	`RV:=FALSE,`
	`Period:=FALSE,`
	`New_DIR:=0,`
	`New_CV:=L#0,`
	`New_RV:=L#0,`
	`New_Period:=0,`
	`Busy=>_bool_out_,`
	`Status=>_word_out_);`
		Cada instrucción CTRL_HSC utiliza una estructura almacenada en un bloque de datos para conservar sus datos.
		En SCL, hay que crear primero el DB de la instrucción de contador individual antes de poder emplearla. En KOP y FUP, STEP 7 crea automáticamente el DB al introducir la instrucción.

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 bloque lógico del OB de alarma de proceso ejecuta la instrucción CTRL_HSC y puede cambiar el valor de referencia cargando un valor NEW_RV.

Nota

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.

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 periodo (para el modo de medición de frecuencia) al valor NEW_PERIOD

Si los siguientes valores booleanos están definidos como 1 cuando se ejecuta la instrucción CTRL_HSC, el valor NEW_xxx correspondiente se carga en el contador. Las peticiones múltiples (varias marcas definidas simultáneamente) se procesan en una sola ejecución de la instrucción CTRL_HSC. Si los siguientes valores de marca booleanos se ajustan a 0 no habrá cambios.

- Al definir DIR = 1, se carga un valor NEW_DIR.
- Al definir CV = 1, se carga un valor NEW_CV.
- Al definir RV = 1, se carga un valor NEW_RV.
- Al definir PERIOD = 1, se carga un valor NEW_PERIOD.

Instrucción CTRL_HSC_EXT (controlar contador rápido (avanzada))

STEP 7 y la CPU S7-1200 también soportan una instrucción avanzada de contador rápido, CTRL_HSC_EXT. Esta instrucción permite al programa medir con precisión el período de los impulsos de entrada de un HSC especificado. Consulte el *Manual de sistema del controlador programable S7-1200* para más detalles.

6.6.1 Funcionamiento del contador rápido

Los contadores rápidos (HSC) pueden contar eventos que se producen con mayor rapidez que la frecuencia de ejecución del OB de ciclo. Si los eventos que se deben contar se producen más lentamente que la frecuencia de ejecución del OB, se pueden utilizar las instrucciones de contaje CTU, CTD o CTUD estándar. Si los eventos se producen con mayor rapidez que la frecuencia de ejecución del OB, utilice el dispositivo HSC más rápido. 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 contaje 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 contaje actual es menor que el valor predeterminado. El HSC pone a disposición una alarma cuando el contaje actual es igual al valor predeterminado, al ocurrir un reset y también al producirse un cambio de sentido.

Cuando el contaje 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 contaje 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.

Selección de un canal de entrada HSC

Utilice la tabla siguiente para garantizar que los canales de entrada de CPU y SB conectados soportan las frecuencias de impulso máximas de las señales del proceso.

Nota

Los canales de entrada de CPU y SB (firmware V4 o posterior) tienen tiempos de filtrado de entrada configurables

Las versiones de firmware anteriores tenían canales de entrada HSC y tiempos de filtrado fijos que no podían cambiarse.

Las versiones V4 o posteriores permiten asignar canales de entrada y tiempos de filtrado. El filtro de entrada predeterminado de 6,4 ms puede ser demasiado lento para las señales de proceso en cuestión. Hay que optimizar los tiempos de filtro de entrada digital de las entradas HSC para la aplicación HSC específica.

Tabla 6- 28 Entrada de CPU: frecuencia máxima

CPU	Canal de entrada de CPU	Modo de 1 o 2 fases	Modo de fase de cuadratura A/B
1211C	De Ia.0 a Ia.5	100 kHz	80 kHz
1212C	De Ia.0 a Ia.5	100 kHz	80 kHz
	Ia.6, Ia.7	30 kHz	20 kHz
1214C y 1215C	De Ia.0 a Ia.5	100 kHz	80 kHz
	De Ia.6 a Ib.5	30 kHz	20 kHz
1217C	De Ia.0 a Ia.5	100 kHz	80 kHz
	De Ia.6 a Ib.1	30 kHz	20 kHz
	De Ib.2 a Ib.5 De 0,2+, 0,2- a 0,5+, 0,5-)	1 MHz	1 MHz

Tabla 6- 29 Entrada de Signal Board SB: frecuencia máxima (tarjeta opcional)

Signal Board (SB)	Canal de entrada de SB	Modo de 1 o 2 fases	Modo de fase de cuadratura A/B
SB 1221, 200 kHz	De Ie.0 a Ie.3	200 kHz	160 kHz
SB 1223, 200 kHz	Ie.0, Ie.1	200 kHz	160 kHz
SB 1223	Ie.0, Ie.1	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. Modo de contador, control de sentido y sentido inicial se asignan en la configuración de dispositivos de la CPU para propiedades funcionales HSC.

Hay cuatro tipos básicos de HSC, 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 o 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.

- Función de medición de períodos: La medición de período se realiza durante el intervalo de medición configurado (10 ms, 100 ms o 1000 ms). HSC_Period SDT devuelve las mediciones de período y las proporciona en forma de dos valores: ElapsedTime y EdgeCount. Las entradas HSC ID1000 a ID1020 no se ven afectadas por las mediciones de periodo:

  – ElapsedTime es un valor de doble entero sin signo en nanosegundos, que representa el tiempo desde el primer evento de contaje hasta el último dentro del intervalo de medición. Si EdgeCount = 0, ElapsedTime será el tiempo desde el último evento de contaje en un intervalo anterior. ElapsedTime tiene un rango entre 0 y 4.294.967.280 ns (0x0000 0000 a 0xFFFF FFF0). El desbordamiento se indica con el valor 4.294967.295 (0xFFFF FFFF). Los valores de 0xFFFF FFF1 a 0xFFFF FFFE están reservados.

  – EdgeCount es un valor de doble entero sin signo, que representa el número de eventos de contaje dentro del intervalo de medición.

  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 HSC 1 está en un modo que utiliza dos entradas integradas, pero que no usa la tercera entrada de reset externa (asignada por defecto a I0.3), I0.3 puede utilizarse para alarmas de flanco o para HSC 2.

Tabla 6- 30 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	-	-	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

Al configurar la CPU, se pueden habilitar y configurar las "entradas de hardware" para cada HSC.

Todas las entradas HSC deben estar conectadas a terminales del módulo CPU o a una Signal Board opcional insertada en el frontal del módulo CPU.

Nota

Como se indica en las tablas siguientes, las asignaciones predeterminadas de las señales opcionales para los diferentes HSCs se solapan. Por ejemplo, el reset externo opcional del HSC 1 usa la misma entrada que una de las entradas del HSC 2.

Para CPUs V4 o posteriores se pueden reasignar las entradas HSC durante la configuración de la CPU. No es necesario utilizar las asignaciones de entradas predeterminadas.

Asegúrese siempre de haber configurado los HSCs de forma que **no** haya dos HSCs usando una entrada.

Las tablas siguientes muestran las asignaciones de entradas HSC predeterminadas tanto para las E/S integradas de las CPUs como para una SB opcional. (Si el modelo de SB seleccionado solo tiene 2 entradas, solo estarán disponibles las entradas 4.0 y 4.1).

Definiciones de la tabla de entradas HSC

- **Una fase: C es entrada de reloj**, **[d] es entrada de sentido** (opcional), y **[R] es entrada de reset externo** (opcional)
  (el reset solo está disponible para el modo "Contaje".)
- **Dos fases: CU es entrada de reloj hacia delante**, **CD es entrada de reloj hacia atrás**, y **[R] es entrada de reset externo** (opcional).
  (el reset solo está disponible para el modo "Contaje".)
- **Cuadratura de fases AB: A es la entrada de reloj A**, **B es la entrada de reloj B** y **[R] es la entrada de reset externo** (opcional) (el reset solo está disponible para el modo "Contaje").

Tabla 6- 31 CPU 1211C: asignaciones de direcciones predeterminadas del HSC

Modo de contador HSC		Entrada integrada de CPU (predeterminada 0.x)						Entrada de SB opcional (predeterminada 4.x) [1]			
		0	1	2	3	4	5	0	1	2	3
HSC 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 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]	C	[d]		[R]
	2 fases										
	Fase AB										
HSC4	1 fase					C	[d]	C	[d]		[R]
	2 fases					CU	CD				
	Fase AB					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] Una SB con solo dos entradas digitales únicamente ofrece las entradas 4.0 y 4.1.

Tabla 6- 32 CPU 1212C: asignaciones de direcciones predeterminadas del HSC

Modo de contador HSC		Entrada integrada de CPU (predeterminada 0.x)								Entrada de SB opcional (predeterminada 4.x) [1]			
		0	1	2	3	4	5	6	7	0	1	2	3
HSC 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 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]	C	[d]		[R]
	2 fases					CU	CD		[R]				
	Fase AB					A	B		[R]				
HSC 4	1 fase						[R]	C	[d]	C	[d]		[R]
	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] Una SB con solo dos entradas digitales únicamente ofrece las entradas 4.0 y 4.1.

Tabla 6- 33 CPU 1214C, CPU 1215C y CPU1217C:
asignaciones de dirección HSC predeterminadas
(solo entradas integradas, véase la tabla siguiente para direcciones SB opcionales)

Modo de contador HSC		Entrada digital byte 0 (predeterminada: 0.x)								Entrada digital byte 1 (predeterminada: 1.x)					
		0	1	2	3	4	5	6	7	0	1	2	3	4	5
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 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 fase									C	[d]	[R]			
	2 fases									CU	CD	[R]			
	Fase AB									A	B	[R]			
HSC 6	1 fase												C	[d]	[R]
	2 fases												CU	CD	[R]
	Fase AB												A	B	[R]

Tabla 6- 34 SB opcional en CPU en la tabla anterior: asignaciones de direcciones predeterminadas del HSC

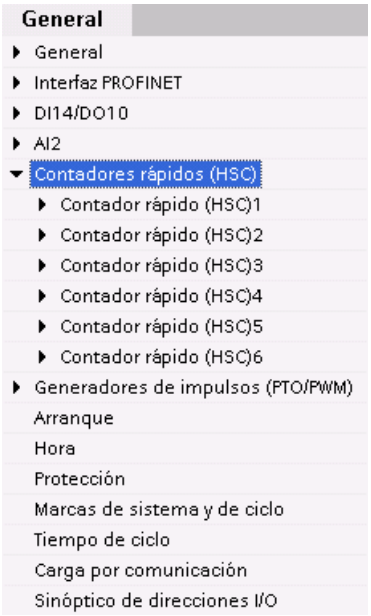
HSC		Entradas de SB opcionales (predeterminada: 4.x) [1]			
		0	1	2	3
HSC 1	1 fase	C	[d]		[R]
	2 fases	CU	CD		[R]
	Fase AB	A	B		[R]
HSC 2	1 fase		[R]	C	[d]
	2 fases		[R]	CU	CD
	Fase AB		[R]	A	B
HSC 5	1 fase	C	[d]		[R]
	2 fases	CU	CD		[R]
	Fase AB	A	B		[R]
HSC 6	1 fase		[R]	C	[d]
	2 fases		[R]	CU	CD
	Fase AB		[R]	A	B

[1] Una SB con solo dos entradas digitales únicamente ofrece las entradas 4.0 y 4.1.

Nota

Las E/S digitales utilizadas por los contadores rápidos se asignan durante la configuración de la CPU. Si se asignan direcciones E/S digitales a dispositivos HSC, los valores de las direcciones E/S asignadas no podrán modificarse utilizando la función de forzado permanente de una tabla de observación.

6.6.2 Configuración del HSC



Puede configurar hasta 6 contadores rápidos. Edite la configuración de dispositivos de la CPU y asigne las propiedades del HSC de cada HSC específico.

Active un HSC seleccionando la opción "Habilitar" de dicho HSC.

Utilice las instrucciones CTRL_HSC y/o CTRL_HSC_EXT en el programa de usuario para controlar el funcionamiento del HSC.

⚠ ADVERTENCIA

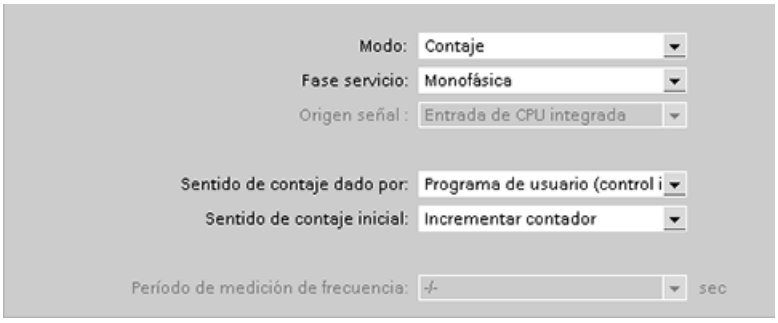
Riesgos asociados con los cambios en el ajuste de tiempos de filtro de los canales de entradas digitales

Si se reajusta el tiempo de filtro de un canal de entrada digital, 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 impulso 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 o daños a los equipos.

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

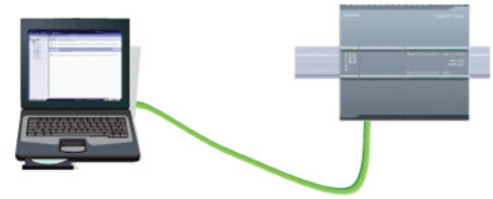
Tras habilitar el HSC se deben configurar los demás parámetros, tales como la función del contador, los valores iniciales, las opciones de reset y los eventos de alarma.



Para obtener información adicional sobre la configuración del HSC, consulte el apartado de configuración de la CPU (Página 84).

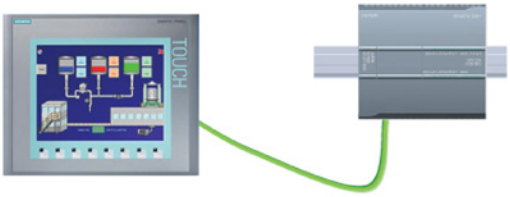
Fácil comunicación entre dispositivos

7



Para la conexión directa entre una programadora y una CPU:

- El proyecto debe incluir la CPU.
- La programadora no forma parte del proyecto, pero debe ejecutar STEP 7.



Para una conexión directa entre un panel HMI y una CPU, el proyecto debe incluir tanto la CPU como el HMI.

Para la conexión directa entre dos CPUs:

- El proyecto debe incluir ambas CPUs.
- Es necesario configurar una conexión de red entre ambas CPUs.

La CPU S7-1200 es un controlador de PROFINET IO que se comunica con STEP 7 desde una programadora, con dispositivos HMI y con otras CPU o dispositivos de terceros. Para la conexión directa entre una programadora o un HMI y una CPU no se requiere un switch Ethernet. Un switch Ethernet se requiere para una red que incorpore más de dos CPUs o dispositivos HMI.

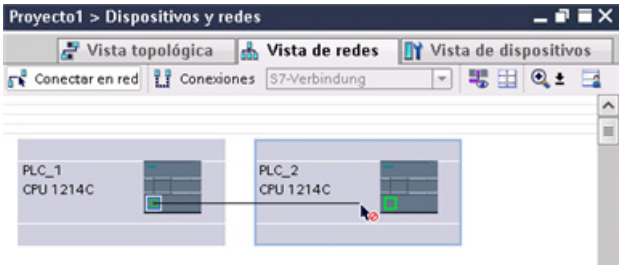
Al agregar un PROFIBUS CM, la CPU también puede funcionar como maestro o esclavo en una red PROFIBUS.

Otras interfaces de comunicación (CM, CP o CB) soportan diversos protocolos, como punto a punto (PTP, Point-to-Point), Modbus, USS y GPRS (módem), CP de seguridad y CP de control remoto.

7.1 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 7- 1 Crear una conexión de red

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

7.2 Opciones de comunicación

S7-1200 ofrece varios tipos de comunicación entre CPU, y programadoras, HMI y otras CPU.

ADVERTENCIA

Si un atacante puede acceder físicamente a las redes, posiblemente podrá leer y escribir datos.

El TIA Portal, la CPU y los HMIs (excepto los que utilizan GET/PUT) utilizan comunicación segura con protección contra reproducción y ataques "man-in-the-middle". Una vez que la comunicación está habilitada, el intercambio de los mensajes firmados se desarrolla en texto claro, lo que permite a cualquier atacante leer los datos, pero protege contra la escritura no autorizada de datos. El TIA Portal, no el proceso de comunicación, encripta los datos de los bloques con protección de know-how.

Todas las formas de comunicación restantes (intercambio E/S a través de PROFIBUS, PROFINET, AS-i u otro bus E/S, GET/PUT, bloque T y módulos de comunicación (CM)) carecen de funciones de seguridad. Este tipo de comunicaciones deben protegerse limitando el acceso físico. Si un atacante puede acceder físicamente a las redes con estos tipos de comunicación, posiblemente pueda leer y escribir datos.

Para más información sobre seguridad y recomendaciones, consulte nuestras "Guías operacionales sobre seguridad industrial" (http://www.industry.siemens.com/topics/global/en/industrial-security/Documents/operational_guidelines_industrial_security_en.pdf) en la página de atención al cliente de Siemens.

PROFINET

PROFINET se usa para intercambiar datos a través del programa de usuario con otros interlocutores vía Ethernet:

- En el S7-1200, PROFINET soporta 16 dispositivos E/S con un máximo de 256 submódulos, y PROFIBUS admite 3 maestros PROFIBUS DP independientes, con 32 esclavos por maestro DP y un máximo de 512 módulos por maestro DP.
- Comunicación S7
- Protocolo User Datagram Protocol (UDP)
- ISO on TCP (RFC 1006)
- Transport Control Protocol (TCP)

Controlador PROFINET IO

En calidad de controlador IO que utiliza PROFINET IO, la CPU se comunica con un máximo de 16 dispositivos PN en la red PN local o a través de un acoplador PN/PN (link). Véase PROFIBUS and PROFINET International, PI (www.profinet.com) para más información.

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.
- Los esclavos PROFIBUS DP, los maestros PROFIBUS DP, así como AS-i (los tres módulos de comunicación del lado izquierdo) y PROFINET son redes de comunicación separadas que no se limitan unas a otras.

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

Se puede crear una conexión con un interlocutor y utilizar las instrucciones GET y PUT para la comunicación con CPUs S7.

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.

IO-Link

El maestro S7-1200 SM 1278 4xIO-Link permite a los dispositivos IO-Link conectarse con un S7-1200 CPU.

7.3 Conexiones de comunicación asíncronas de V4.1

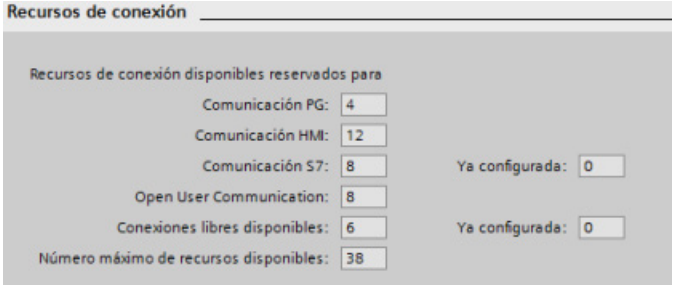
Sinopsis de los servicios de comunicación

La CPU soporta los siguientes servicios de comunicación:

Servicio de comunicación	Funcionalidad	Con PROFIBUS DP		Con Ethernet
		Módulo maestro CM 1243-5 DP	Módulo esclavo CM 1242-5 DP	
Comunicación PG	Puesta en marcha, test y diagnóstico	Sí	No	Sí
Comunicación HMI	Manejo y visualización	Sí	No	Sí
Comunicación S7	Intercambio de datos utilizando conexiones configuradas	Sí	No	Sí
Routing de funciones PG	Por ejemplo, test y diagnóstico más allá de los límites de la red	No	No	No
PROFIBUS DP	Intercambio de datos entre maestro y esclavo	Sí	Sí	No
PROFINET IO	Intercambio de datos entre controladores IO y dispositivos IO	No	No	Sí
Servidor web	Diagnóstico	No	No	Sí
SNMP (Simple Network Management Protocol)	Protocolo estándar para diagnóstico y parametrización de la red	No	No	Sí
Comunicación abierta vía TCP/IP	Intercambio de datos vía Industrial Ethernet con protocolo TCP/IP (con FB cargables)	No	No	Sí
Comunicación abierta vía ISO on TCP	Intercambio de datos vía Industrial Ethernet con protocolo ISO on TCP (con FB cargables)	No	No	Sí
Comunicación abierta vía UDP	Intercambio de datos vía Industrial Ethernet con protocolo UDP (con FB cargables)	No	No	Sí

Conexiones disponibles

La CPU soporta el siguiente número máximo de conexiones de comunicación asíncronas simultáneas para PROFINET y PROFIBUS. El número máximo de recursos de conexión asignados a cada categoría es fijo y no puede modificarse. Sin embargo, es posible configurar las 6 "conexiones libres disponibles" para aumentar el número de cualquier categoría tal y como requiere su aplicación.



En función de los recursos de conexión asignados está disponible el siguiente número de conexiones por dispositivo:

	Programadora (PG)	Interfaz hombre-máquina (HMI)	Servidor/cliente GET/PUT	Open User Communications	Navegador web
Nº máx. de recursos de conexión	3 (se garantiza el soporte de 1 PG)	12 (se garantiza el soporte de 4 dispositivos HMI)	8	8	30 (se garantiza el soporte de 3 navegadores web)

Así, por ejemplo, una programadora dispone de 3 recursos de conexión. Según sean las funciones de PG que se utilicen, esta puede utilizar 1, 2 o 3 de los recursos de conexión disponibles. En el S7-1200 se garantiza siempre una 1 PG como mínimo; sin embargo, no se permite más de 1 PG.

Otro ejemplo es el número de HMI, tal como se muestra en la figura inferior. Los dispositivos HMI disponen de 12 recursos de conexión. Según sea el tipo o modelo de HMI disponible y las funciones HMI que se utilicen, cada HMI podría utilizar 1, 2 o 3 de sus recursos de conexión disponibles. En función del número de recursos de conexión disponibles que se utilicen, existe la posibilidad de utilizar más de 4 dispositivos HMI a la vez. Sin embargo, siempre se garantizan 4 dispositivos HMI como mínimo. Un HMI puede emplear sus recursos de conexión disponibles (1 en cada caso de un total de 3) para las funciones siguientes:

- leer
- escribir
- alarmas más diagnóstico

Ejemplo	HMI 1	HMI 2	HMI 3	HMI 4	HMI 5	Total de recursos de conexión disponibles
Recursos de conexión utilizados	2	2	2	3	3	12

Nota

Conexiones de servidor web (HTTP): la CPU proporciona conexiones para varios navegadores web. El número de navegadores que soporta la CPU simultáneamente depende de las conexiones que solicite o utilice un navegador web concreto.

Nota

Las conexiones de comunicación de Open User Communication, S7, HMI, programadora y servidor web (HTTP) pueden aprovechar varios de los recursos de conexión basados en las funciones que se están usando actualmente.

7.4 Instrucciones PROFINET y PROFIBUS

Instrucciones PROFINET

Las instrucciones TSEND_C y TRCV_C simplifican las comunicaciones PROFINET al combinar las funciones de las instrucciones TCON y TDISCON con la instrucción TSEND o TRCV.

- 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. TSEND_C combina las funciones de las instrucciones TCON, TDISCON y TSEND en una instrucción.
- 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. La instrucción TRCV_C combina las funciones de las instrucciones TCON, TDISCON y TRCV en una instrucción.

También se soportan las instrucciones TCON, TDISCON, TSEND y TRCV.

Las instrucciones TUSEND y TURCV se pueden utilizar para transmitir o recibir datos a través de UDP. TUSEND y TURCV (así como TSEND, TRCV, TCON y TDISCON) funcionan de forma asíncrona, es decir, que el procesamiento de la tarea abarca varias llamadas de instrucciones.

La instrucción IP_CONF se puede usar para modificar los parámetros de configuración de IP desde el programa de usuario. IP_CONF funciona de forma asíncrona. La ejecución abarca varias llamadas.

Instrucciones PROFIBUS

La instrucción DPNRM_DG (leer diagnóstico) lee los datos de diagnóstico actuales de un esclavo DP en el formato especificado por EN 50 170 volumen 2, PROFIBUS.

Instrucciones de E/S distribuidas para PROFINET, PROFIBUS y AS-i

Con PROFINET, PROFIBUS y GPRS se pueden usar las instrucciones siguientes.

- Use las instrucciones RDREC (leer registro) y WRREC (escribir registro) para transferir un registro de datos especificado entre un componente, como puede ser el módulo de un rack central, o un componente distribuido (PROFIBUS DP o PROFINET IO).
- Use la instrucción RALRM (leer alarma) para leer una alarma y la información correspondiente desde un esclavo DP o un componente de dispositivo PROFINET IO. 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.
- Las instrucciones DPRD_DAT (leer datos coherentes) y DPWR_DAT (escribir datos coherentes) se utilizan para transferir áreas de datos coherentes que sobrepasen los 64 bytes desde o hasta un esclavo DP estándar.
- Exclusivamente para PROFIBUS, la instrucción DPNRM_DG se utiliza para leer los datos de diagnóstico actuales de un esclavo DP en el formato especificado por EN 50 170 volumen 2, PROFIBUS.

7.5 PROFINET

7.5.1 Open User Communication

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 7- 2 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	Solo TRCV_C y TRCV (V4.1 e instrucciones Legacy)	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(V4.1 e instrucciones Legacy)	
ISO on TCP	Comunicación de CPU a CPU Fragmentación y reensamblado de mensajes	Modo ad hoc	Solo TRCV_C y TRCV (V4.1 e instrucciones Legacy)	Asigna TSAPs a los dispositivos local (activo) e interlocutor (pasivo)
		Controlado por protocolo	TSEND_C, TRCV_C, TCON, TDISCON, TSEND y TRCV (V4.1 e instrucciones Legacy)	
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 IO	Comunicación de CPU a dispositivo PROFINET IO	Transmisión y recepción de datos con la longitud especificada	Incorporado	Incorporado

7.5.1.1 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" (solo simbólico), únicamente es posible recibir datos en matrices de tipos de datos Byte, Char, USInt y SInt.

Para configurar la instrucción TRCV_C o TRCV para el modo ad hoc, ajuste el parámetro de entrada de la instrucción ADHOC.

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.

7.5.1.2 ID de conexión para las instrucciones de Open User Communication

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" (Página 161) de la instrucción para configurar los parámetros de la conexión. Entre los parámetros se encuentra la ID de conexión para dicha conexión.

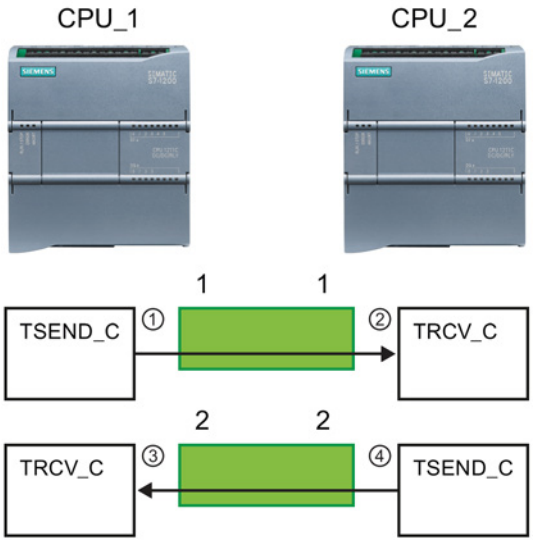
- La ID de conexión debe ser unívoca para la CPU. Cada conexión que se cree debe tener un DB y una ID de conexión distintos.
- Tanto la CPU local como la CPU interlocutora pueden utilizar el mismo número de ID de conexión para la misma conexión, aunque los números de ID de conexión no tienen que coincidir necesariamente. El número de ID de conexión solo 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 dos conexiones separadas para transmitir y recibir datos.

- La instrucción TSEND_C de la CPU_1 enlaza con TRCV_C 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).

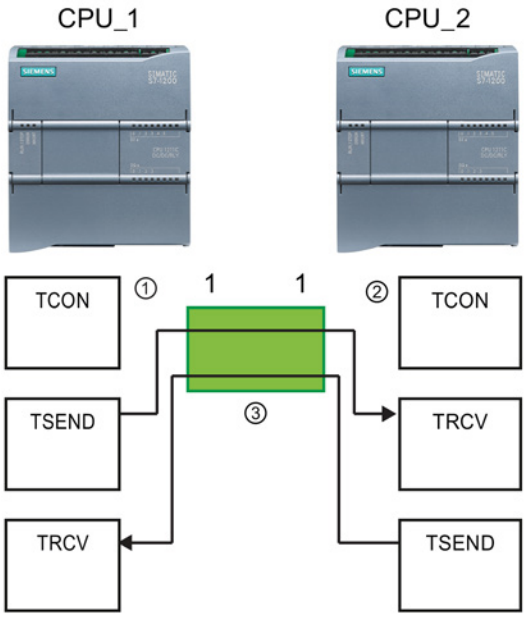
② TRCV_C en la CPU_2 crea la conexión para la CPU_2 y asigna la ID de conexión (ID de conexión 1 para la CPU_2).

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.



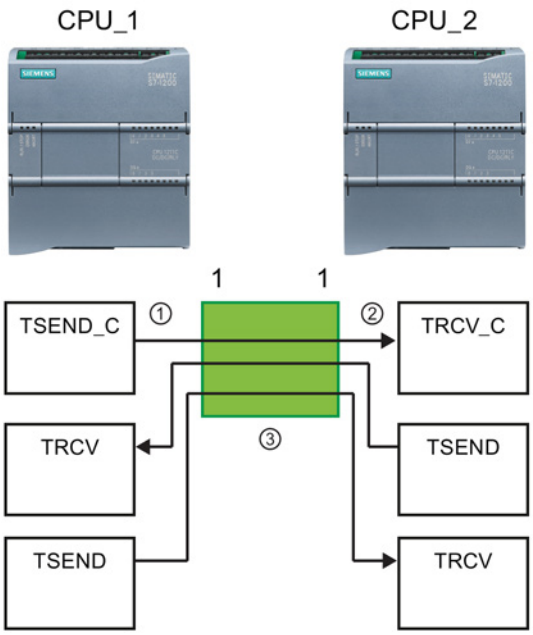
① TCON en la CPU_1 crea una conexión y asigna una ID de conexión a dicha conexión en la CPU_1 (ID=1).

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).

7.5.1.3 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 asignados 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.

TCON_Param

Tabla 7- 3 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: • 17: TCP (predeterminado) • 18: ISO on TCP • 19: UDP
5	active_est	Bool	ID del tipo de conexión: • TCP e ISO on TCP: – 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: • 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: • 0: sin especificar (el parámetro rem_staddr es irrelevante) • 4 (predeterminado): dirección IP válida en el parámetro rem_staddr (solo para TCP e ISO on TCP)
10	rem_tsap_id_len	USInt	Longitud del parámetro rem_tsap_id utilizado, en bytes; valores posibles: • 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.

Byte	Parámetro y tipo de datos		Descripción
12 … 27	local_tsap_id	Array [1..16] of Byte	Componente de conexión para la dirección local: • TCP e ISO on TCP: n.º de puerto local (valores posibles: de 1 a 49151; valores recomendados: 2000...5000): – 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: – 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. Nota: asegúrese de que todos los valores de local_tsap_id son unívocos dentro de la CPU.
28 … 33	rem_subnet_id	Array [1..6] of USInt	Este parámetro no se utiliza.
34 … 39	rem_staddr	Array [1..6] of USInt	Solo 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: • TCP: número de puerto del interlocutor. Rango: de 1 a 49151; valores recomendados: de 2000 a 5000): – 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: – 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

7.5.2 Configurar la vía de conexión local/de 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

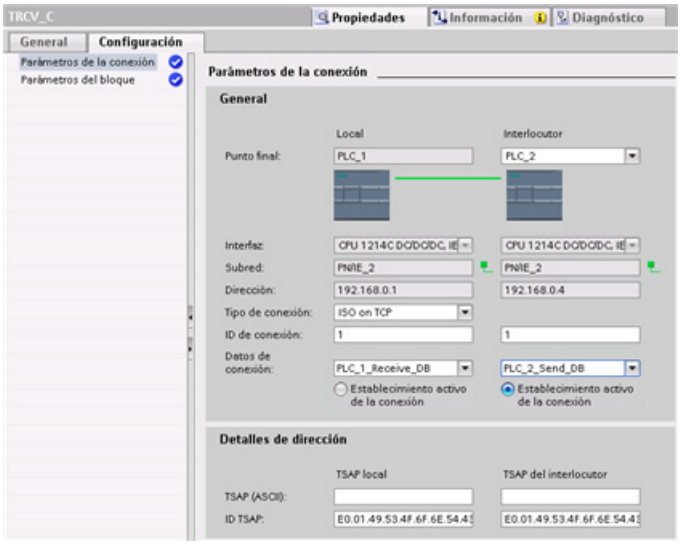
Los interlocutores ejecutan las instrucciones necesarias para configurar y establecer la conexión. El punto final activo y el punto final pasivo de la comunicación se especifican mediante parámetros. Una vez configurada y establecida la conexión, la CPU la mantiene y la vigila automáticamente.

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.

Rutas de conexión

Después de insertar una instrucción TSEND_C, TRCV_C o TCON en el programa de usuario, la ventana del inspector mostrará las propiedades de la conexión una vez que haya seleccionado cualquier parte de la instrucción. Los parámetros de comunicación se especifican en la ficha "Configuración" de "Propiedades" de la instrucción de comunicación.

Tabla 7- 4 Configurar la ruta de conexión (mediante las propiedades de la instrucción)

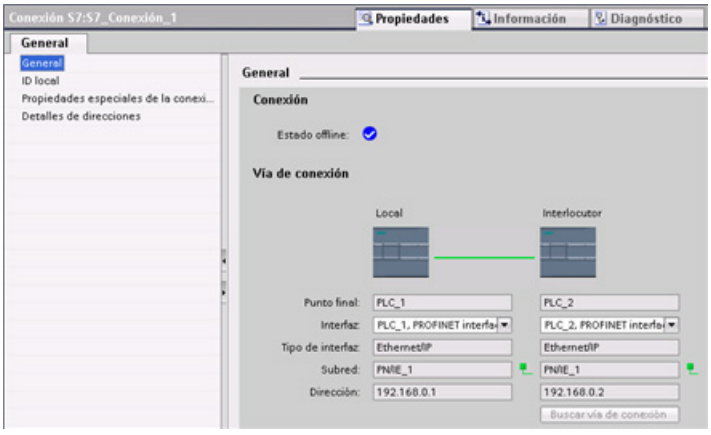
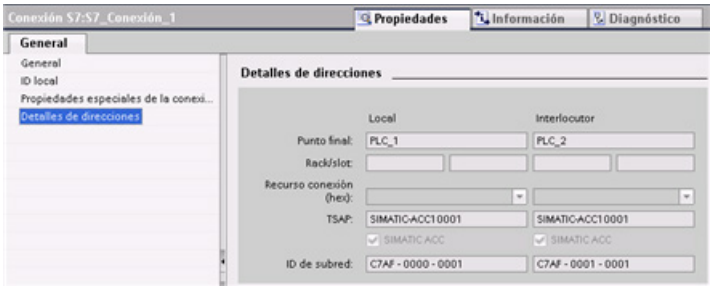
TCP, ISO on TCP y UDP	Propiedades de la conexión
Para los protocolos de Ethernet TCP, ISO on TCP y UDP, utilice las "Propiedades" de la instrucción (TSEND_C, TRCV_C o TCON) para configurar las conexiones "local/interlocutor". La figura muestra las "Propiedades de conexión" de la ficha "Configuración"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

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 7- 5 Configurar la ruta de conexión para la comunicación S7 (configuración del dispositivo)

Comunicación S7 (GET y PUT)	Propiedades de la conexión
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". La ficha "General" ofrece varias propiedades: • "General" (la que se muestra) • "ID local" • "Propiedades de conexión especiales" • "Detalles de direcciones" (la que se muestra)	 

Consulte los "Protocolos" (Página 156) de la sección "PROFINET" o "Crear una conexión S7" (Página 177) en la sección "Comunicación S7 para obtener más información y una lista de las instrucciones de comunicación disponibles.

Tabla 7- 6 Parámetros para la conexión múltiple de CPU

Parámetro		Definición
Dirección		Direcciones IP asignadas
General	Punto final	Nombre asignado a la CPU interlocutora (receptora)
	Interfaz	Nombre asignado a las interfaces
	Subred	Nombre asignado a las subredes
	Tipo de interfaz	*Solo comunicación S7*: 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	Punto final	*Solo comunicación S7*: Nombre asignado a la CPU interlocutora (receptora)
	Rack/slot	*Solo comunicación S7*: Ubicación de rack y slot
	Recurso de conexión	*Solo comunicación S7*: 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 UPD: Puerto de la CPU interlocutora en formato decimal
	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 solo 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".

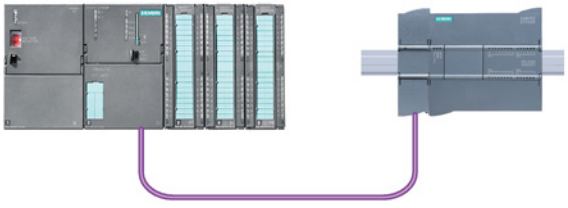
7.6 PROFIBUS

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 solo 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.

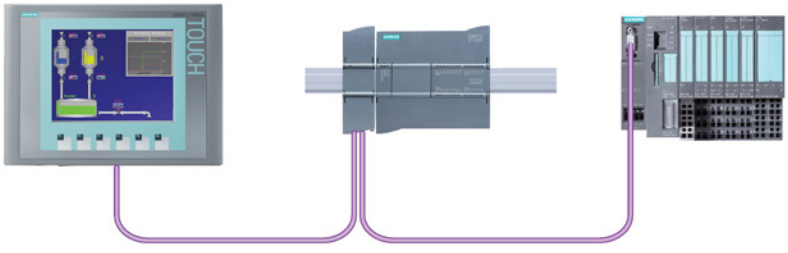
Un maestro PROFIBUS conforma una "estación activa" en la red. PROFIBUS DP define dos clases de maestro. Un maestro clase 1 (por lo general un controlador central programable (PLC) o un equipo dotado de un software especial) procesa la comunicación normal o intercambia datos con los esclavos que tiene 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. Si desea configurar el módulo en un sistema de otro fabricante, hay un archivo GSD disponible para el CM 1242-5 (esclavo DP) en el CD suministrado con el módulo y en las páginas del Siemens Automation Customer Support (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=en&objid=6GK72425DX300XE0&caller=view) de Internet.

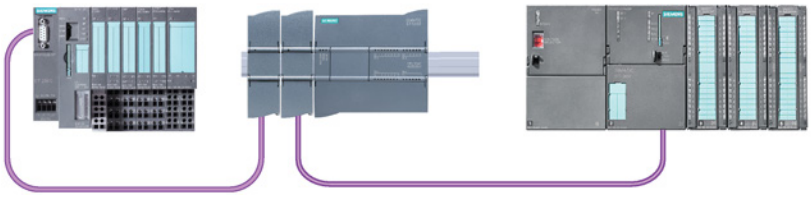
En la figura inferior el S7-1200 es un esclavo DP de un controlador S7-300:



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 inferior el S7-1200 es un maestro que controla a un esclavo DP ET200S:



Si un CM 1242-5 y un CM 1243-5 están instalados conjuntamente, un S7-1200 puede actuar simultáneamente tanto de un esclavo de un sistema maestro DP de nivel superior como de maestro de un sistema esclavo DP subordinado:



En V4.0 se pueden configurar tres CMs PROFIBUS como máximo por estación, pudiendo haber cualquier combinación de CMs maestros DP o esclavos DP. Los maestros DP en una implementación de firmware de CPU V3.0 o superior pueden controlar un máximo de 32 esclavos.

Los datos de configuración de los CMs PROFIBUS se guardan en la CPU local. Ello permite una sustitución rápida de estos módulos de comunicación en caso de necesidad.

7.6.1 Servicios de comunicaciones de CM PROFIBUS

Los CM PROFIBUS utilizan el protocolo PROFIBUS DP-V1.

Tipos de comunicación con DP-V1

Los siguientes tipos de comunicación están disponibles con DP-V1:

- Comunicación cíclica (CM 1242-5 y CM 1243-5)

  Los dos módulos PROFIBUS admiten la comunicación cíclica para la transferencia de datos de proceso entre el esclavo DP y el maestro DP.

  El sistema operativo de la CPU procesa la comunicación cíclica. Para ello, no se necesitan bloques de software. Los datos de E/S se leen o escriben directamente de/en la imagen de proceso de la CPU.

- Comunicación acíclica (solo en CM 1243-5)

  El módulo maestro DP también admite la comunicación acíclica mediante los bloques de software:

  – La instrucción "RALRM" está disponible para el procesamiento de alarmas.
  – Las instrucciones "RDREC" y "WRREC" están disponibles para transferir datos de configuración y diagnóstico.

Funciones que no admite el CM 1243-5: SYNC/FREEZE y Get_Master_Diag

Otros servicios de comunicaciones del CM 1243-5

El módulo maestro del CM 1243-5 DP admite los siguientes servicios de comunicaciones adicionales:

- Comunicación S7
  - Servicios PUT/GET

    El maestro DP funciona como un cliente y servidor para consultas de otros PC o controladores S7 mediante PROFIBUS.
  - Comunicación PG/OP

    Las funciones de PG permiten cargar los datos de configuración y los programas de usuario de un PG y transferir los datos de diagnóstico a un PG.

    Entre los posibles interlocutores para la comunicación OP se encuentran los paneles HMI, los PC de panel SIMATIC con WinCC flexible o los sistemas SCADA que admiten la comunicación S7.

7.6.2 Referencia a los manuales de usuario de CM PROFIBUS

Más información

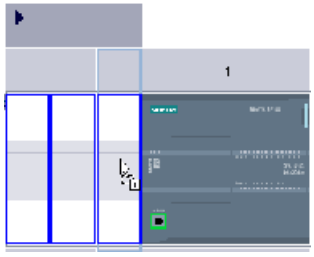
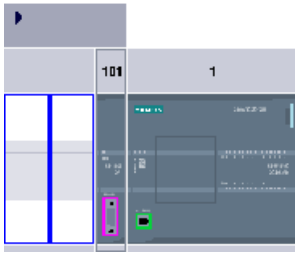
Puede encontrar información detallada sobre los CM PROFIBUS en los manuales de los dispositivos. Puede encontrarlos en Internet en las páginas de Customer Support de Siemens Industrial Automation con las ID de entrada siguientes:

- CM 1242-5 (http://support.automation.siemens.com/WW/view/en/49852105)
- CM 1243-5 (http://support.automation.siemens.com/WW/view/en/49851842)

7.6.3 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 7- 7 Agregar un módulo PROFIBUS CM 1243-5 (maestro DP) a la configuración de dispositivos

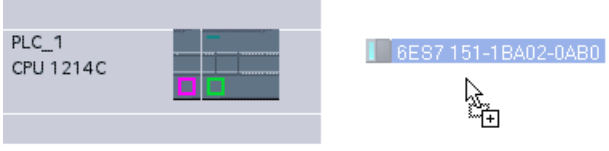
Módulo	Seleccionar el módulo	Insertar el módulo	Resultado
CM 1243-5 (maestro DP)			

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 7- 8 Agregar un esclavo DP ET200 S a la configuración de dispositivos

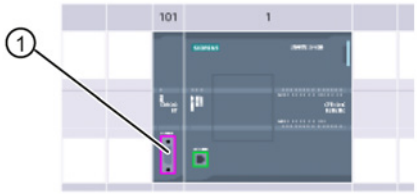
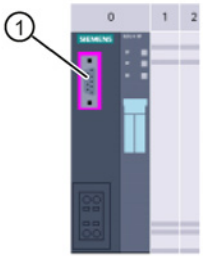
Inserte el esclavo DP	Resultado
	

7.6.4 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 7- 9 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.

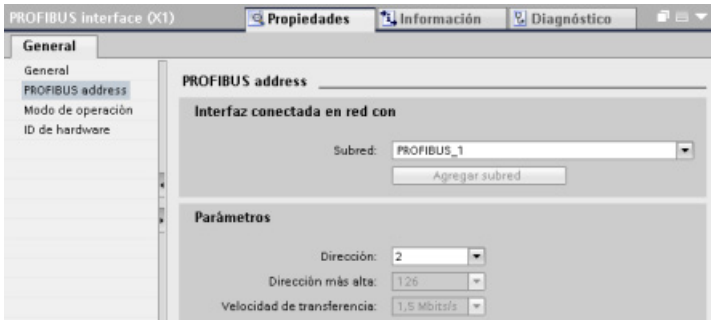


Tabla 7- 10 Parámetros de la dirección PROFIBUS

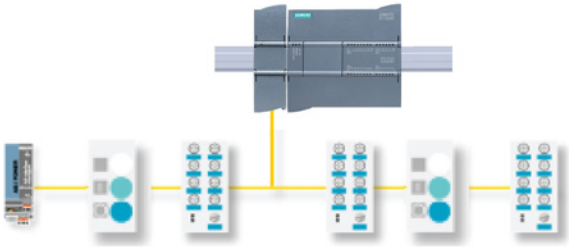
Parámetro		Descripción
Subred	Nombre de la subred a la que está conectada el dispositivo. Haga clic en el botón "Agregar nueva subred" para crear una subred nueva. La opción predeterminada es "No conectado". Son posibles dos tipos de conexión: • El ajuste predeterminado "no conectado" ofrece una conexión 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 solo se envía a estaciones activas. Al especificar la dirección PROFIBUS más alta se optimiza el bus.
	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).

7.7 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 solo están activos cuando el CM 1243-2 los llama.

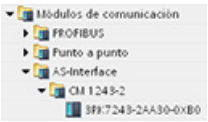
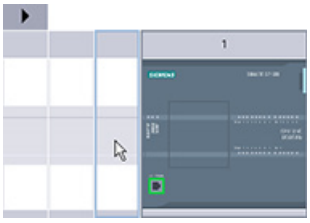
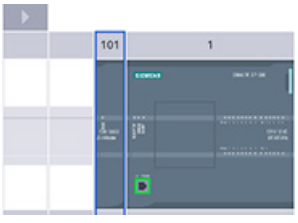
En la siguiente figura, el S7-1200 es un maestro AS-i que controla los dispositivos esclavos digitales/analógicos del módulo E/S AS-i.



7.7.1 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 7- 11 Agregar un módulo maestro AS-i CM1243-2 a la configuración de dispositivos

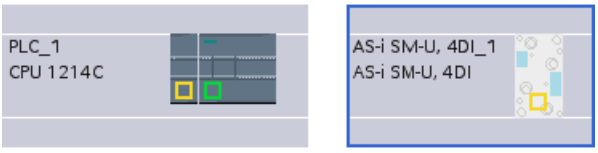
Módulo	Seleccionar el módulo	Insertar el módulo	Resultado
CM 1243-2 maestro AS-i			

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.

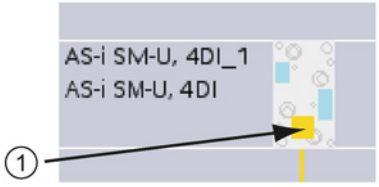
Tabla 7- 12 Agregar un esclavo AS-i a la configuración de dispositivos

Insertar el esclavo AS-i	Resultado
	

7.7.2 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.



① Puerto AS-i

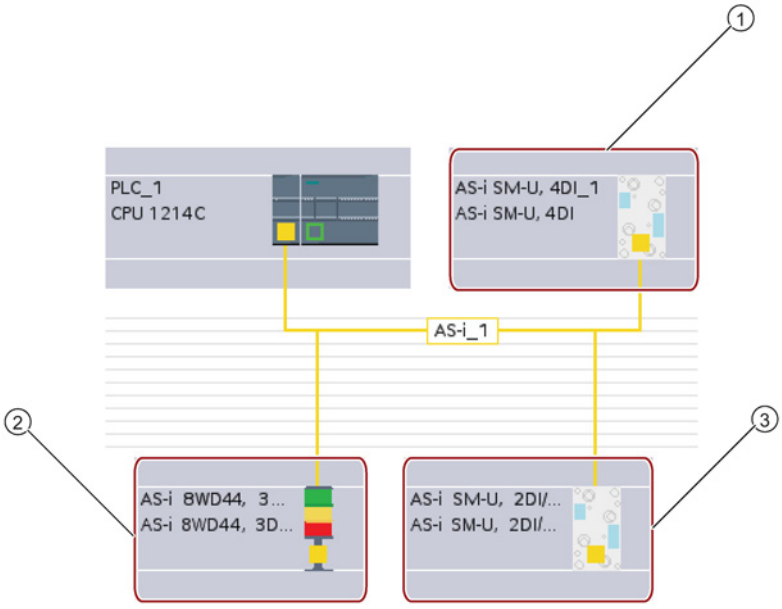
Asignar la dirección de esclavo AS-i

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.

Los dispositivos AS-i "estándar" utilizan la dirección completa, que es numérica y no tiene la designación A o B. Los dispositivos AS-i con "nodo A/B" utilizan la parte A o B de cada dirección, lo que permite utilizar dos veces cada una de las 31 direcciones. El rango de direcciones va de 1A a 31A y de 1B a 31B.

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.

En el ejemplo inferior, tres dispositivos AS-i se han direccionado como "1" (un dispositivo estándar), "2A" (un dispositivo con nodo A/B) y "3" (un dispositivo estándar):

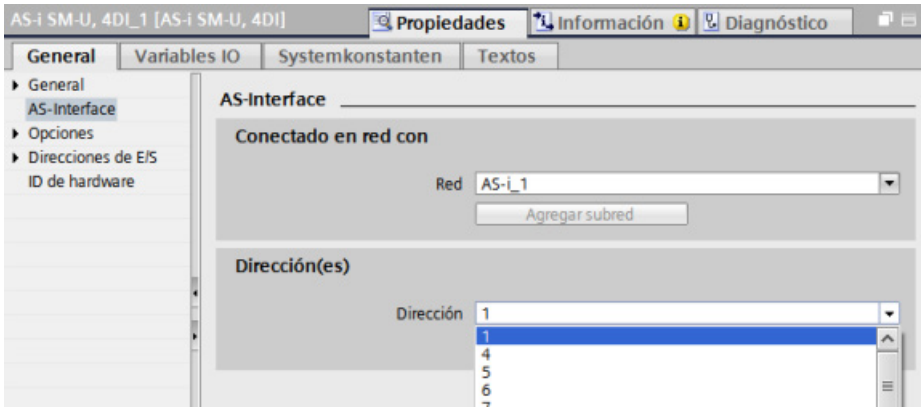


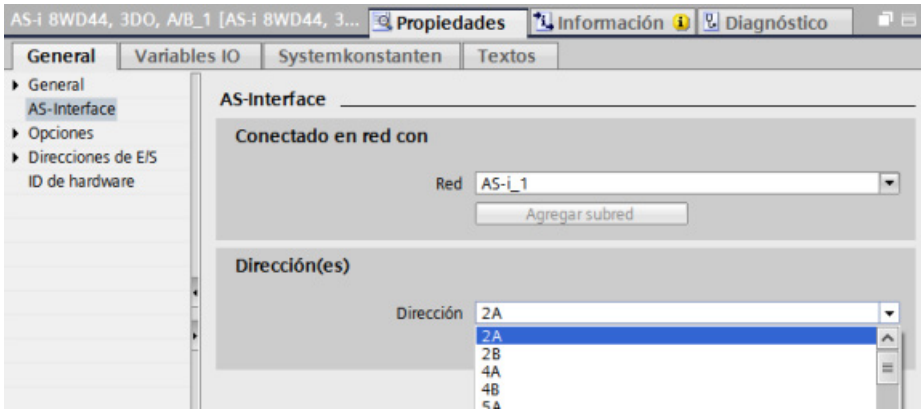
① Dirección de esclavo AS-i 1; dispositivo: AS-i SM-U, 4DI; referencia: 3RG9 001-0AA00

② Dirección de esclavo AS-i 2A; dispositivo: AS-i 8WD44, 3DO, A/B; referencia: 8WD4 428-0BD

③ Dirección de esclavo AS-i 3; dispositivo: AS-i SM-U, 2DI/2DO; referencia: 3RG9 001-0AC00

Asigne aquí la dirección de esclavo AS-i:





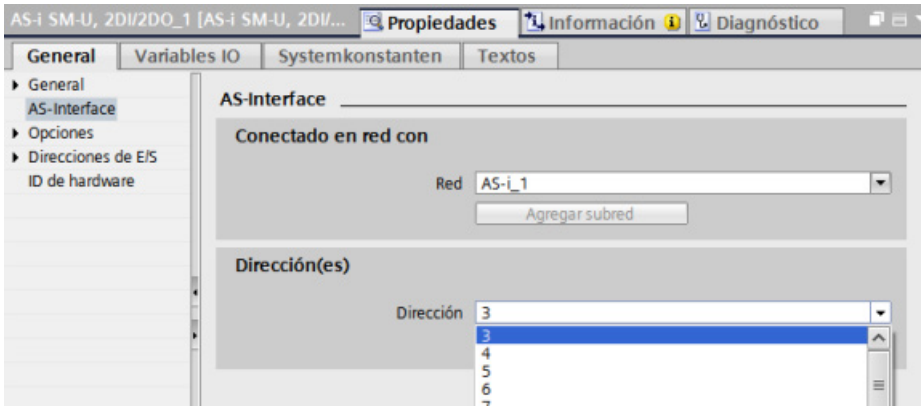


Tabla 7- 13 Parámetros de la interfaz AS-i

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

7.8 Comunicación S7

7.8.1 Instrucciones GET y PUT

Las instrucciones GET y PUT se pueden utilizar para comunicarse con CPUs S7 a través de conexiones PROFINET y PROFIBUS. Esto solo es posible si la función "Permitir acceso vía comunicación PUT/GET" está activada para la CPU del interlocutor en la propiedad "Protección" de las propiedades de la CPU local:

- 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 de un DB estándar: una CPU S7-1200 sólo puede utilizar direccionar 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.

STEP 7 crea automáticamente el DB al introducir la instrucción.

Nota

Para garantizar la coherencia de los datos, antes de acceder a los datos o de iniciar otra operación de lectura o escritura, compruebe siempre que la operación haya finalizado (NDR = 1 para GET o DONE = 1 para PUT).

Nota

La operación GET/PUT del programa de la CPU V4.0 no está activada automáticamente

Una operación GET/PUT del programa de una CPU V3.0 está activada automáticamente en una CPU V4.0.

Sin embargo, una operación GET/PUT del programa de una CPU V4.0 no está activada automáticamente en una CPU V4.0. Hay que ir a la "Configuración de dispositivos" de la CPU, ficha "Propiedades" de la ventana de inspección, propiedad "Protección" para activar Acceso GET/PUT (Página 91).

7.8.2 Crear una conexión S7

Mecanismos de conexión

Para acceder a interlocutores remotos con las instrucciones PUT/GET, el usuario también debe tener permiso.

De forma predeterminada, la opción "Permitir acceso vía comunicación PUT/GET" está desactivada. En ese caso, el acceso de lectura y escritura a los datos de la CPU solo es posible para conexiones que necesiten configuración o programación tanto para la CPU local como para el interlocutor. El acceso mediante instrucciones BSEND/BRCV es posible, por ejemplo.

En consecuencia, no se pueden realizar conexiones para las cuales la CPU local es solo un servidor cuando se usa la CPU, lo que significa que no hay configuración/programación de la comunicación con el interlocutor en la CPU local, por ejemplo:

- Acceso PUT/GET, FETCH/WRITE o FTP mediante módulos de comunicaciones
- Acceso con PUT/GET desde otras CPU S7
- Acceso a HMI mediante la comunicación PUT/GET

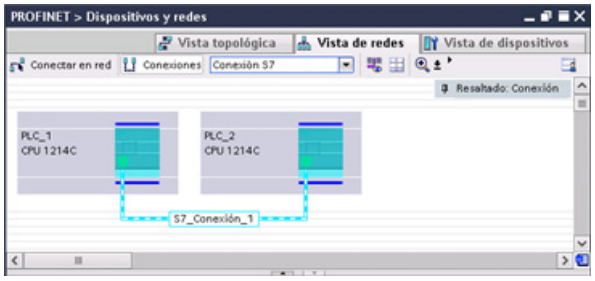
Si desea permitir el acceso a los datos de la CPU desde el lado del cliente, es decir, si no desea limitar los servicios de comunicación de la CPU, puede configurar la protección de acceso para la CPU S7-1200 (Página 91) para ese nivel de seguridad.

Tipos de conexión

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 148).

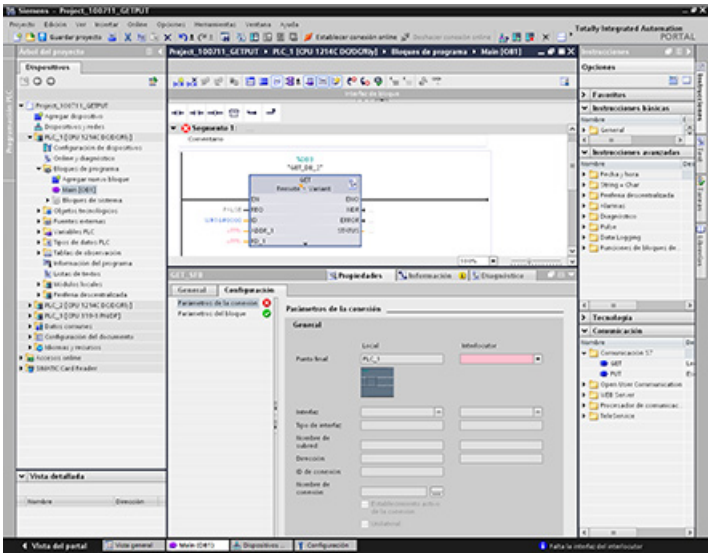


Haga clic en el botón "Resaltado: conexión" para acceder al cuadro de diálogo de configuración "Propiedades" de la instrucción de comunicación.

7.8.3 Asignación de parámetros de conexión GET/PUT

La asignación de parámetros de conexión de las instrucciones GET/PUT es una ayuda que permite al usuario configurar conexiones de comunicación S7 de CPU a CPU.

Después de 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.

Después de insertar un bloque GET o PUT, la ficha "Configuración" aparece automáticamente y se visualiza de inmediato la página "Parámetros de conexión". Esta página permite al usuario configurar la conexión S7 requerida, así como el parámetro "ID de la conexión" que está referenciado por el parámetro de bloque "ID". La página "Parámetros del bloque" le permite al usuario configurar parámetros adicionales para el bloque.

Nota

La operación GET/PUT del programa de la CPU V4.0 no está activada automáticamente

Una operación GET/PUT del programa de una CPU V3.0 está activada automáticamente en una CPU V4.0.

Sin embargo, una operación GET/PUT del programa de una CPU V4.0 no está activada automáticamente en una CPU V4.0. Hay que ir a la "Configuración de dispositivos" de la CPU, ficha "Propiedades" de la ventana de inspección, propiedad "Protección" para activar Acceso GET/PUT (Página 91).

7.9 GPRS

7.9.1 Conexión a una red GSM

Comunicación WAN basada en IP mediante GPRS

Empleando el procesador de comunicaciones CP 1242-7, el S7-1200 puede conectarse a las redes GSM. El CP 1242-7 permite la comunicación WAN de estaciones remotas con un centro de control y una comunicación entre estaciones.

La comunicación entre estaciones solo es posible mediante una red GSM. Para la comunicación entre una estación remota y una sala de control, el centro de control debe tener un PC con acceso a Internet.

El CP 1242-7 soporta los servicios siguientes para la comunicación mediante la red GSM:

- GPRS (General Packet Radio Service o servicio general de radio por paquetes)

  El servicio por paquetes para la transmisión de datos "GPRS" se maneja mediante la red GSM.

- SMS (Short Message Service o servicio de mensajería corta)

  El CP 1242-7 puede recibir y enviar mensajes SMS. El interlocutor de la comunicación puede ser un teléfono móvil o un S7-1200.

El CP 1242-7 es adecuado para un uso industrial en todo el mundo y soporta las bandas de frecuencia siguientes:

- 850 MHz
- 900 MHz
- 1800 MHz
- 1900 MHz

Requisitos

El equipo usado en las estaciones o el centro de control depende de la aplicación en concreto.

- Para la comunicación con o por medio de una sala de control central, el centro de control requiere un PC con acceso a Internet.
- Aparte del equipo de la estación, una estación S7-1200 remota con un CP 1242-7 debe cumplir los requisitos siguientes para ser capaz de comunicarse mediante la red GSM:
  - Un contrato con un proveedor de red GSM adecuado

    Si se usa GPRS, el contrato debe permitir el uso del servicio GPRS.

    Si solo se dispone de una comunicación directa entre las estaciones mediante una red GSM, el proveedor de la red GSM debe asignar una dirección IP fija a los CPs. En este caso, la comunicación entre estaciones no es mediante el centro de control.
  - La tarjeta SIM perteneciente al contrato

    La tarjeta SIM se inserta en el CP 1242-7.
  - Disponibilidad local de una red GSM en el rango de la estación

7.9.2 Aplicaciones del CP 1242-7

El CP 1242-7 puede usarse para las aplicaciones siguientes:

Aplicaciones de telecontrol

- Enviar mensajes por SMS

  Mediante el CP 1242-7, la CPU de una estación S7-1200 remota puede recibir mensajes SMS de la red GSM o enviar mensajes por SMS a un teléfono móvil configurado o un S7-1200.

- Comunicación con un centro de control

  Las estaciones S7-1200 remotas se comunican mediante la red GSM e Internet con un servidor de telecontrol en la estación maestra. Para la transferencia de datos mediante GPRS, la aplicación "TELECONTROL SERVER BASIC" se instala en el servidor de telecontrol en la estación maestra. El servidor de telecontrol se comunica con un sistema de control central de nivel superior mediante la función integrada de servidor OPC.

- Comunicación entre las estaciones S7-1200 mediante una red GSM

  La comunicación entre estaciones remotas con un CP 1242-7 puede manejarse de dos formas diferentes:

  – Comunicación entre estaciones mediante una estación maestra

    En esta configuración, una conexión segura permanente entre estaciones S7-1200 que se comunican entre sí y el servidor de telecontrol se establece en la estación maestra. La comunicación entre estaciones se realiza mediante el servidor de telecontrol. El CP 1242-7 funciona en modo "Telecontrol".

  – Comunicación directa entre estaciones

    Para una comunicación directa entre estaciones sin el desvío por medio de la estación maestra se usan tarjetas SIM con una dirección IP fija que permiten a las estaciones comunicarse entre sí directamente. Los servicios de comunicación y funciones de seguridad posibles (por ejemplo VPN) dependen de lo que ofrece el proveedor de red. El CP 1242-7 funciona en modo "GPRS directo".

TeleService vía GPRS

Una conexión TeleService puede establecerse entre una estación de ingeniería con STEP 7 y una estación S7-1200 remota con un CP 1242-7 a través de la red GSM e Internet. La conexión parte de la estación de ingeniería y pasa por un servidor de telecontrol o una pasarela de TeleService que actúa de intermediario redireccionando tramas y estableciendo la autorización. Estos PCs usan las funciones de la aplicación "TELECONTROL SERVER BASIC".

Es posible utilizar la conexión de TeleService para los fines siguientes:

- Descargar la configuración o los datos de programa del proyecto de STEP 7 a la estación
- Consultar datos de diagnóstico en la estación

7.9.3 Otras propiedades del CP 1242-7

Otros servicios y funciones del CP 1242-7

- Sincronización en tiempo real del CP mediante Internet

  Es posible ajustar el tiempo en el CP del siguiente modo:

  – En el modo "Telecontrol" la hora se transfiere por el servidor de telecontrol. El CP la usa para ajustar su hora.

  – En el modo "GPRS directo" el CP puede solicitar la hora usando SNTP.

  Para sincronizar la hora de la CPU es posible leer la hora actual del CP empleando un bloque.

- Respaldo intermedio de mensajes que deben enviarse si hay problemas de conexión
- Disponibilidad aumentada gracias a la opción de conectar a un servidor de telecontrol sustituto
- Volumen de datos optimizado (conexión temporal)

  Como alternativa a la conexión permanente del servidor de telecontrol, el CP puede configurarse en STEP 7 con una conexión temporal al servidor de telecontrol. En este caso, una conexión al servidor de telecontrol solo se establece cuando se requiere.

- Registro del volumen de datos

  El volumen de datos transferidos se registra y puede evaluarse para fines específicos.

7.9.4 Configuración y conexiones eléctricas

Configuración y sustitución de módulos

Para configurar el módulo se requiere la herramienta de configuración siguiente:

STEP 7 versión V11.0 SP1 o superior

Para STEP 7 V11.0 SP1 también se requiere el paquete de soporte "CP 1242-7" (HSP0003001).

Para transferir datos del proceso mediante GPRS se emplean las instrucciones de comunicación de telecontrol en el programa de usuario de la estación.

Los datos de configuración del CP 1242-7 se guardan en la CPU local. Ello permite una sustitución rápida del CP en caso de necesidad.

Es posible insertar hasta 3 módulos del tipo CP 1242-7 por S7-1200. Esto, por ejemplo, permite establecer vías de comunicación redundantes.

Conexiones eléctricas

- Alimentación del CP 1242-7

  El CP tiene una conexión separada para la alimentación externa de 24 V DC.

- Interfaz inalámbrica para la red GSM

  Se requiere una antena externa para la comunicación GSM. Esta se conecta al conector SMA del CP.

7.9.5 Más información

Más información

El manual del CP 1242-7 contiene información detallada. Lo encontrará en Internet, en las páginas del Customer Support de Siemens Industrial Automation con la ID de artículo siguiente:

45605894 (http://support.automation.siemens.com/WW/view/es/45605894)

7.9.6 Accesorios

Antena ANT794-4MR GSM/GPRS

Las antenas siguientes están disponibles para ser usadas en redes GSM/GPRS y pueden instalarse tanto en el interior como en el exterior:

- Antena cuatribanda ANT794-4MR

Nombre abreviado	Referencia	Explicación
ANT794-4MR	6NH9 860-1AA00	Antena cuatribanda (900, 1800/1900 MHz, UMTS); resistente a la intemperie para su uso en interiores y exteriores; cable de conexión de 5 m conectado permanentemente a la antena; conector SMA que incluye un soporte de instalación, tornillos y conectores de pared

- Antena plana ANT794-3M

Nombre abreviado	Referencia	Explicación
ANT794-3M	6NH9 870-1AA00	Antena plana (900, 1800/1900 MHz); resistente a la intemperie para su uso en interiores y exteriores; cable de conexión de 1,2 m conectado permanentemente a la antena; conector SMA que incluye un soporte adhesivo, posibilidad de montaje con tornillos

Las antenas deben pedirse por separado.

7.9.7 Consulte el manual de la antena GSM

Más información

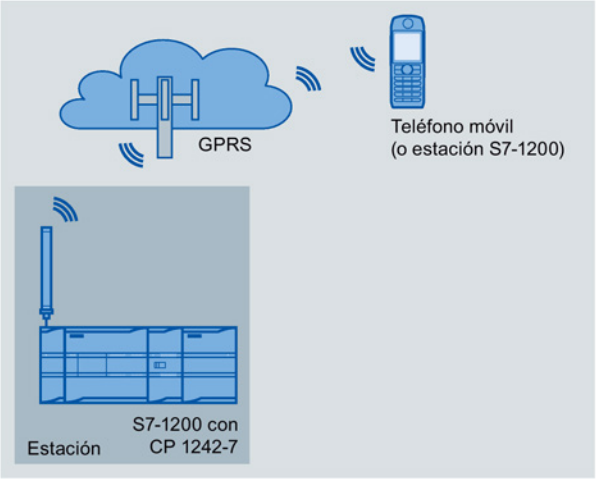
Encontrará información detallada en el manual del dispositivo. Lo encontrará en Internet, en las páginas del Customer Support de Siemens Industrial Automation con la ID de artículo siguiente:

23119005 (http://support.automation.siemens.com/WW/view/es/23119005)

7.9.8 Ejemplos de configuración de telecontrol

En la parte inferior encontrará varios ejemplos de configuración de estaciones con un CP 1242-7.

Enviar mensajes por SMS



Un SIMATIC S7-1200 con un CP 1242-7 puede enviar mensajes por SMS a un teléfono móvil o a una estación S7-1200 configurada.

Telecontrol mediante un centro de control

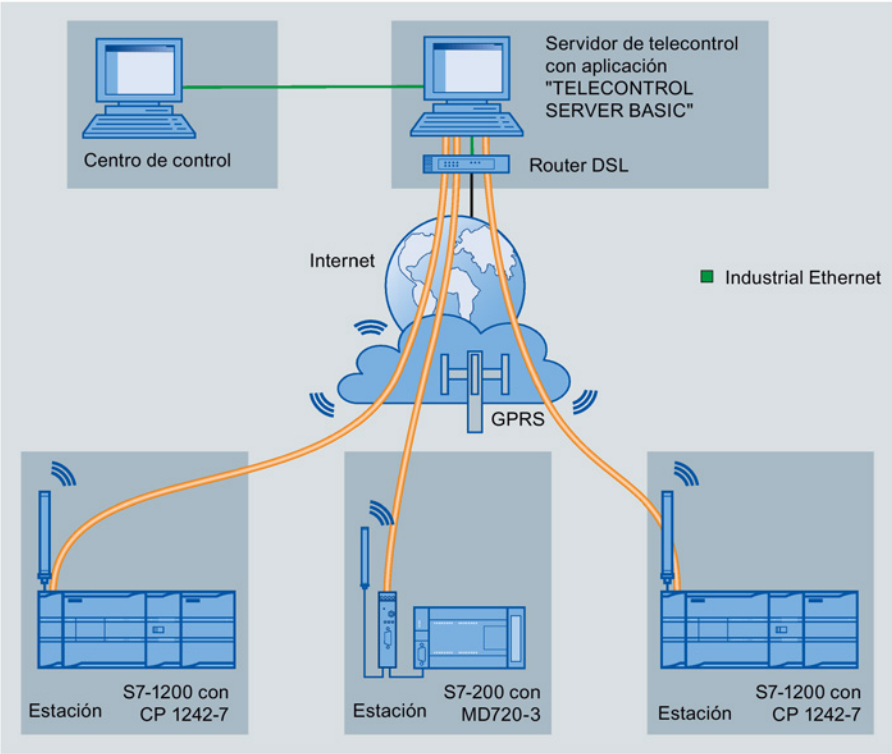


Figura 7-1 Comunicación entre las estaciones S7-1200 y un centro de control

En aplicaciones de telecontrol, las estaciones SIMATIC S7-1200 con un CP 1242-7 se comunican con un centro de control a través de la red GSM e Internet. La aplicación "TELECONTROL SERVER BASIC" (TCSB) se instala en el servidor de telecontrol en la estación maestra. Como resultado se producen las aplicaciones siguientes:

- Comunicación de telecontrol entre la estación y el centro de control

  En este caso, los datos del campo son enviados por las estaciones al servidor de telecontrol en la estación maestra a través de la red GSM e Internet. El servidor de telecontrol se usa para supervisar estaciones remotas.

- Comunicación entre una estación y una sala de control con un cliente OPC

  Igual que en el primer caso, la estación se comunica con el servidor de telecontrol. Mediante su servidor OPC integrado, el servidor de telecontrol intercambia datos con el cliente OPC de la sala de control.

  El cliente OPC y el servidor de telecontrol pueden encontrase en un mismo equipo, por ejemplo cuando TCSB se instala en un PC del centro de control con WinCC.

- Comunicación entre estaciones mediante un centro de control

  La comunicación entre estaciones es posible con estaciones S7 equipadas con un CP 1242-7.

  Para permitir la comunicación entre estaciones, el servidor de telecontrol transmite los mensajes de la estación emisora a la estación receptora.

Comunicación directa entre estaciones

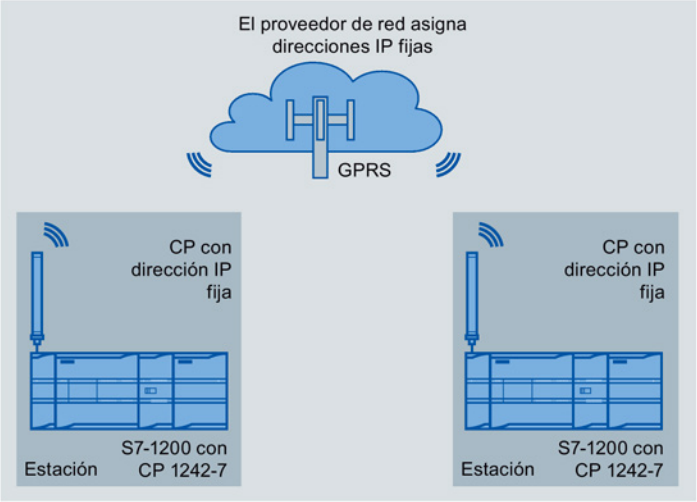


Figura 7-2 Comunicación directa entre dos estaciones S7-1200

En esta configuración, dos estaciones SIMATIC S7-1200 se comunican directamente entre sí usando el CP 1242-7 a través de la red GSM. Cada CP 1242-7 tiene asignada una dirección IP fija. El servicio relevante del proveedor de red GSM debe permitirlo.

TeleService vía GPRS

En TeleService vía GPRS, una estación de ingeniería en la que se ha instalado STEP 7 se comunica a través de la red GSM e Internet con el CP 1242-7 en el S7-1200.

Puesto que un cortafuegos suele estar cerrado para peticiones de conexión desde el exterior, se requiere una estación de conmutación entre la estación remota y la estación de ingeniería. Esta estación de conmutación puede ser un servidor de telecontrol o, si no se dispone de un servidor de telecontrol en la configuración, una pasarela de TeleService.

TeleService con servidor de telecontrol

La conexión se lleva a cabo mediante el servidor de telecontrol.

- La estación de ingeniería y el servidor de telecontrol están conectados vía Intranet (LAN) o Internet.
- El servidor de telecontrol y la estación remota están conectados vía Internet y vía red GSM.

La estación de ingeniería y el servidor de telecontrol también pueden ser el mismo equipo; en otras palabras, STEP 7 y TCSB están instalados en el mismo PC.

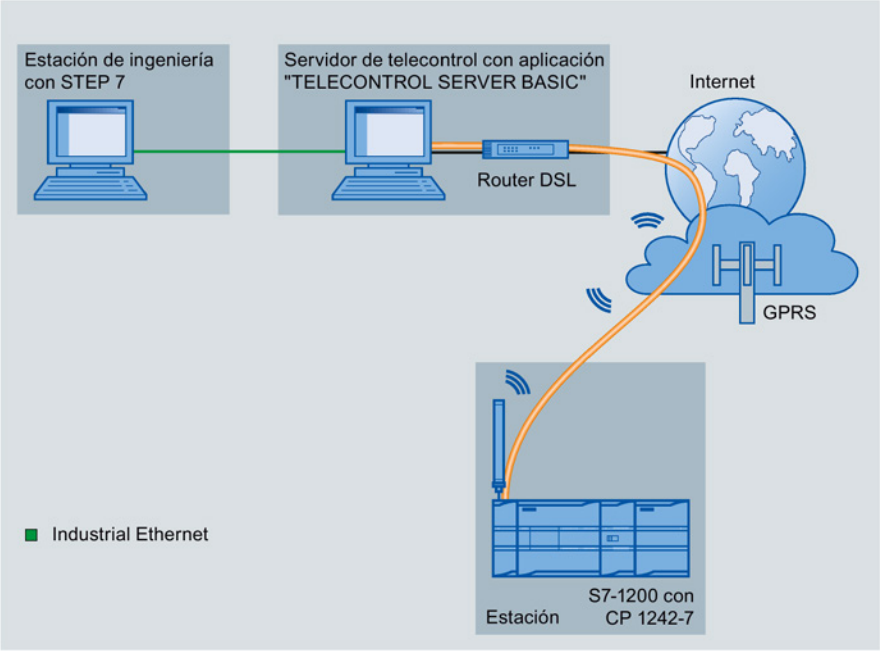


Figura 7-3 TeleService vía GPRS en una configuración con servidor de telecontrol

TeleService sin servidor de telecontrol

La conexión se lleva a cabo mediante la pasarela de TeleService.

La conexión entre la estación de ingeniería y la pasarela de TeleService puede ser local vía LAN o vía Internet.

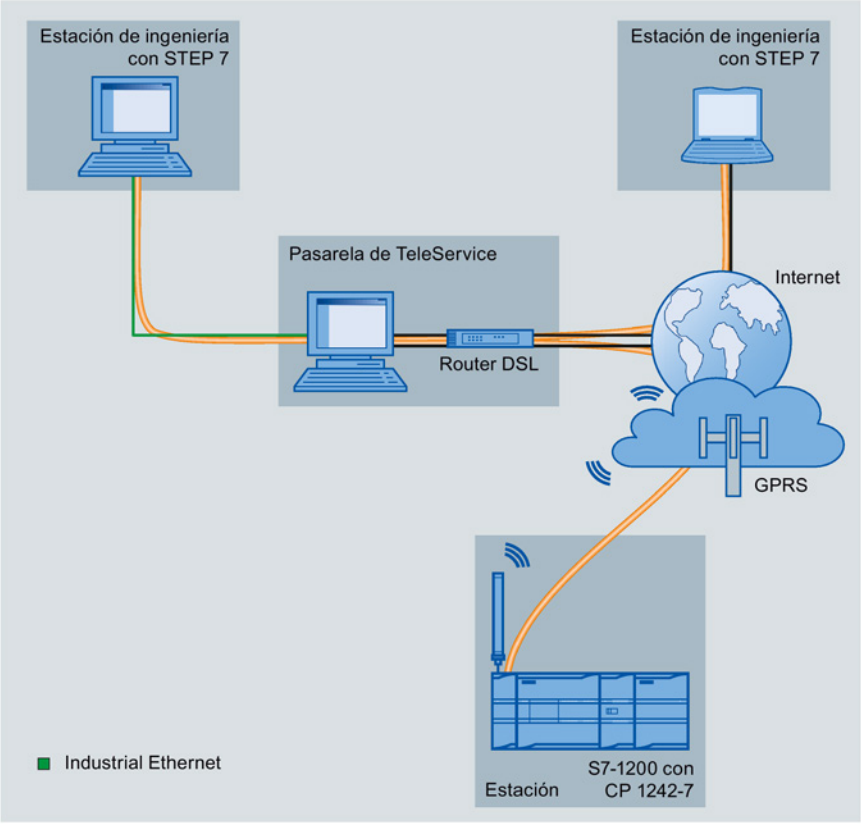


Figura 7-4 TeleService vía GPRS en una configuración con pasarela de TeleService

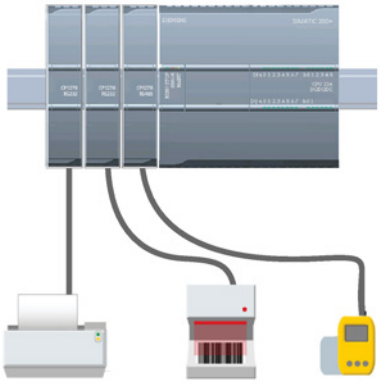
7.10 Protocolos de comunicación PtP, USS y Modbus

7.10.1 Comunicación punto a punto

La CPU admite la siguiente comunicación punto a punto (PtP) para protocolos serie basados en caracteres:

- PtP (Página 193)
- USS (Página 194)
- Modbus (Página 196)

PtP ofrece una libertad y flexibilidad máximas, pero requiere una implementación exhaustiva en el programa de usuario.



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

Este tipo de comunicación PtP es una comunicación serie que utiliza UART estándar para admitir distintas velocidades de transferencia y opciones de paridad. Los módulos de comunicaciones RS232 y RS422/485 (CM 1241) y las Communication Board RS485 (CB 1241) ofrecen las interfaces eléctricas para realizar la comunicación PtP.

PtP vía PROFIBUS o PROFINET

La versión V4.1 de la CPU S7-1200 junto con STEP 7 V13 SP1 amplía la capacidad de PtP para usar un rack de periferia descentralizada PROFINET o PROFIBUS para comunicarse con varios dispositivos (lectores RFID, dispositivos GPS y otros):

- PROFINET (Página 156): la interfaz Ethernet de la CPU S7-1200 se conecta a un módulo de interfaz PROFINET. Los módulos de comunicación PtP en el rack con el módulo de interfaz pueden ofrecer comunicaciones serie a los dispositivos PtP.
- PROFIBUS (Página 166): un módulo de comunicación PROFIBUS se inserta en el lado izquierdo del rack con la CPU S7-1200. El módulo de comunicación PROFIBUS se conecta al rack que contiene un módulo de interfaz PROFIBUS. Los módulos de comunicación PtP en el rack con el módulo de interfaz pueden ofrecer comunicaciones serie a los dispositivos PtP.

Por este motivo, S7-1200 soporta dos juegos de instrucciones PtP:

- Instrucciones Legacy punto a punto: estas instrucciones son anteriores a la versión V4.0 de S7-1200 y solo funcionan con la comunicaciones serie si se utiliza un módulo de comunicaciones CM 1241 o una Communication Board CB 1241.
- Instrucciones punto a punto (Página 193): estas instrucciones ofrecen toda la funcionalidad de las instrucciones Legacy más la capacidad de conectarse a la periferia descentralizada de PROFINET y PROFIBUS. Las instrucciones punto a punto permiten configurar la comunicación entre los módulos de comunicación PtP en el rack de periferia descentralizada y los dispositivos PtP.

Nota

Con la versión V4.1 de S7-1200 se pueden emplear las instrucciones punto a punto para todos los tipos de comunicación punto a punto: serie, serie mediante PROFINET y serie mediante PROFIBUS. STEP 7 ofrece las instrucciones Legacy punto a punto solo por su compatibilidad con programas existentes. Sin embargo, las instrucciones Legacy siguen funcionando con CPUs V4.1 así como con CPUs V4.0 o anteriores. No es necesario convertir programas anteriores de un juego de instrucciones a otro.

7.10.2 Utilizar las interfaces de comunicación serie

Dos módulos de comunicaciones (CM) y una Communication Board (CB) ofrecen la interfaz para la comunicación PtP:

- CM 1241 RS232 (Página 435)
- CM 1241 RS422/485 (Página 434)
- CB 1241 RS485 (Página 431)

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. Consulte las instrucciones de montaje (Página 20) para más información sobre el montaje y la retirada del módulo.

Las interfaces de comunicación serie tienen las siguientes características:

- Cuentan con un puerto aislado
- Soportan protocolos punto a punto
- Se configuran y programan mediante las instrucciones de procesador de comunicaciones punto a punto
- 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 423).

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 Communication Board cuenta con un LED de transmisión (TxD) y uno de recepción (RxD). No tiene ningún LED de diagnóstico.

7.10.3 Instrucciones PtP

Las instrucciones Port_Config, Send_Config y Receive_Config permiten cambiar la configuración desde el programa de usuario.

- Port_Config cambia los parámetros de puerto, como la velocidad de transferencia.
- Send_Config cambia la configuración de los parámetros de transmisión serie.
- Receive_Config cambia la configuración de los parámetros de un receptor serie en un puerto de comunicación. Esta instrucción configura las condiciones que indican el inicio y fin de un mensaje recibido. Los mensajes que cumplen esas condiciones son recibidos por la instrucción Receive_P2P.

Los cambios de la configuración dinámica no se almacenan de forma permanente en la CPU. Tras una desconexión y nueva conexión de la alimentación se utilizará la configuración estática inicial del dispositivo.

Las instrucciones Send_P2P, Receive_P2P y Receive_Reset controlan la comunicación PtP:

- Send_P2P transfiere el búfer indicado al CM o la CB. La CPU sigue ejecutando el programa de usuario mientras el módulo envía los datos a la velocidad de transferencia indicada.
- Receive_P2P comprueba si se han recibido mensajes en el CM o la CB. Si hay un mensaje disponible, se transfiere a la CPU.
- Receive_Reset inicializa el búfer de recepción.

Cada CM o CB puede almacenar como máximo 1 KB en un búfer. Este búfer puede asignarse a varios mensajes recibidos.

Las instrucciones Signal_Set y Signal_Get sólo son válidas para CM RS232. Use estas instrucciones para leer o establecer las señales de comunicación de RS232.

Las instrucciones Get_Features y Set_Features permiten al programa leer y ajustar propiedades de módulos.

7.10.4 Instrucciones USS

S7-1200 soporta el protocolo USS y proporciona instrucciones diseñadas específicamente para la comunicación con unidades a través del puerto RS485 de un CM o una CB. La unidad física y los parámetros de lectura/escritura se pueden controlar con las instrucciones USS. Cada CB o CM RS485 soporta como máximo 16 unidades.

- La instrucción USS_Port_Scan gestiona la comunicación real entre la CPU y todas las unidades conectadas a un CM o una CB. Inserte una instrucción USS_Port_Scan distinta para cada CM o CB de la aplicación. Asegúrese de que el programa de usuario ejecuta la instrucción USS_Port_Scan con la rapidez suficiente para impedir que se produzca un timeout en la unidad. Utilice la instrucción USS_Port_Scan en un ciclo o en cualquier OB de alarma.
- La instrucción USS_Drive_Control accede a una unidad específica de la red USS. Los parámetros de entrada y salida de la instrucción USS_Drive_Control corresponden a los estados y controles de la unidad. Si la red comprende 16 unidades, el programa debe tener como mínimo 16 instrucciones USS_Drive_Control, es decir, una para cada unidad.

  Asegúrese de que la CPU ejecuta la instrucción USS_Drive_Control a la velocidad necesaria para controlar las funciones de la unidad. Utilice la instrucción USS_Drive_Control únicamente en un OB de ciclo.
- Las instrucciones USS_Read_Param y USS_Write_Param leen y escriben los parámetros operativos de la unidad remota. Estos parámetros controlan el funcionamiento interno de la unidad. Estos parámetros se definen en el manual de la unidad.

  El programa de usuario puede contener tantas instrucciones de este tipo como sea necesario. No obstante, cada unidad sólo puede activar una petición de lectura o escritura en un momento determinado. Utilice las instrucciones USS_Read_Param y USS_Write_Param únicamente en un OB de ciclo.

Un DB instancia contiene búferes y memoria temporal para todas las unidades de la red USS conectadas a cada CM o CB. Todas las instrucciones USS para una unidad utilizan el DB instancia para compartir la información.

Calcular el tiempo necesario para la comunicación con la unidad

La comunicación con la unidad es asíncrona al ciclo de la CPU. Por lo general, la CPU completa varios ciclos antes de que finalice una transacción de comunicación con una unidad.

El intervalo USS_Port_Scan es el tiempo necesario para una transacción con una unidad. La tabla siguiente muestra el intervalo de USS_Port_Scan mínimo para cada velocidad de transferencia de comunicación. Si la función USS_Port_Scan se llama más frecuentemente que el intervalo de USS_Port_Scan , no se incrementará el número de transacciones. El intervalo de timeout de la unidad 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 7- 14 Calcular los requisitos de tiempo

Velocidad de transferencia	Intervalo mínimo calculado (milisegundos) para la llamada USS_Port_Scan	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

7.10.5 Instrucciones Modbus

La CPU admite comunicaciones Modbus vía varias redes:

- 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 solo 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.

- 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 comunicaciones adicionales.

  Modbus TCP utiliza conexiones cliente-servidor como camino 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 la CPU. Cada conexión MB_SERVER debe usar un DB de instancia y número de puerto IP únicos. Solo 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.

⚠ ADVERTENCIA

Si un atacante puede acceder físicamente a las redes, posiblemente podrá leer y escribir datos.

El TIA Portal, la CPU y los HMIs (excepto los que utilizan GET/PUT) utilizan comunicación segura con protección contra reproducción y ataques "man-in-the-middle". Una vez que la comunicación está habilitada, el intercambio de los mensajes firmados se desarrolla en texto claro, lo que permite a cualquier atacante leer los datos, pero protege contra la escritura no autorizada de datos. El TIA Portal, no el proceso de comunicación, encripta los datos de los bloques con protección de know-how.

Todas las formas de comunicación restantes (intercambio E/S a través de PROFIBUS, PROFINET, AS-i u otro bus E/S, GET/PUT, bloque T y módulos de comunicación (CM)) carecen de funciones de seguridad. Este tipo de comunicaciones deben protegerse limitando el acceso físico. Si un atacante puede acceder físicamente a las redes con estos tipos de comunicación, posiblemente pueda leer y escribir datos.

Para más información sobre seguridad y recomendaciones, consulte nuestras "Guías operacionales sobre seguridad industrial" en la página de atención al cliente: www.industry.siemens.com/topics/global/en/industrial-security/Documents/operational_guidelines_industrial_security_en.pdf (http://www.industry.siemens.com/topics/global/en/industrial-security/Documents/operational_guidelines_industrial_security_en.pdf)

Nota

Modbus TCP solo 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.

Tabla 7- 15 Instrucciones Modbus

Tipo de comunicaciones	Instrucción
Modbus RTU (RS232 o RS485)	Modbus_Comm_Load: una ejecución de Modbus_Comm_Load se utiliza para configurar parámetros de puerto PtP como velocidad de transferencia, paridad y control de flujo. Una vez que el puerto de CPU se configura para el protocolo Modbus RTU, solo puede ser utilizado por las instrucciones Modbus_Master o Modbus_Slave.
	Modbus_Master: la instrucción de maestro Modbus permite que la CPU actúe como maestro Modbus RTU y se comunique con uno o varios esclavos Modbus.
	Modbus_Slave: la instrucción de esclavo Modbus permite que la CPU actúe como esclavo Modbus RTU y se comunique con un dispositivo maestro Modbus.
Modbus TCP (PROFINET)	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.

Las instrucciones Modbus no utilizan eventos de alarma de comunicación para controlar el proceso de comunicación. El programa debe consultar las instrucciones Modbus_Master/Modbus_Slave o MB_CLIENT/MB_SERVER para transmitir y recibir condiciones completas.

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 necesite, iniciar la desconexión de cliente y servidor para permitir la conexión con otro servidor.

Instrucción PID sencilla 8

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.
- La instrucción PID_Temp ofrece un regulador PID universal que permite manejar los requisitos específicos de control de temperatura.

Nota

Los cambios realizados en la configuración PID y las descargas en estado RUN no son efectivos hasta que la CPU no pasa de STOP a RUN. Los cambios realizados en el cuadro de diálogo "Parámetros PID" con el "Control de valor de arranque" serán efectivos de inmediato.

Las tres instrucciones PID (PID_Compact, PID_3Step y PID_Temp) 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	Coeficiente de retardo derivativo (acción D)
T_1	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	Coeficiente de retardo derivativo (acción D)
T_1	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)

8.1 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 231). También deben ajustarse los parámetros de optimización ("optimización inicial" durante el arranque u "optimización fina" manual) para poner el regulador PID en servicio (Página 248).

Tabla 8- 1 Insertar la instrucción PID y el objeto tecnológico

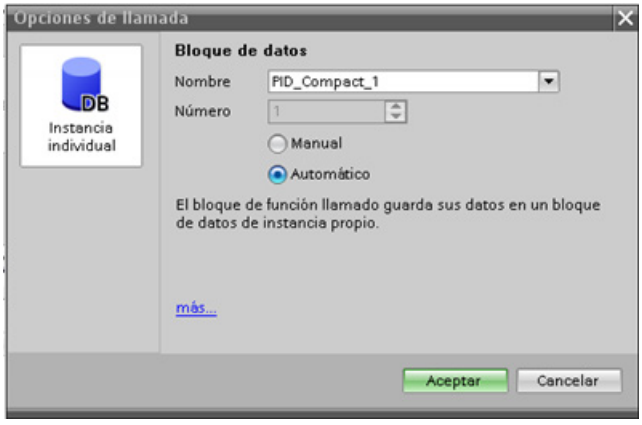
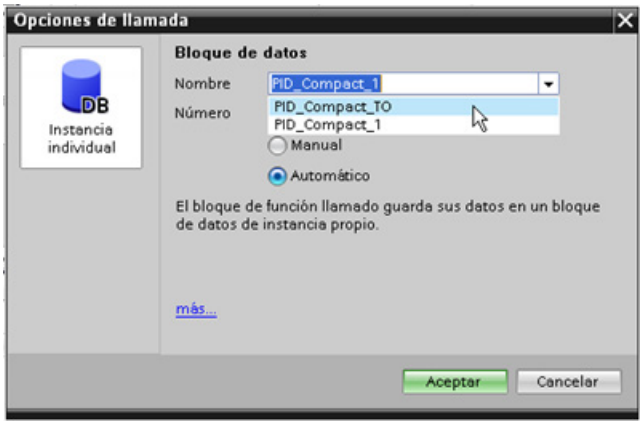
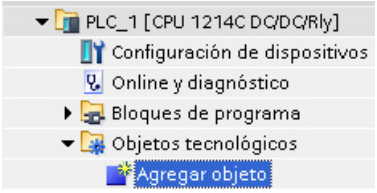
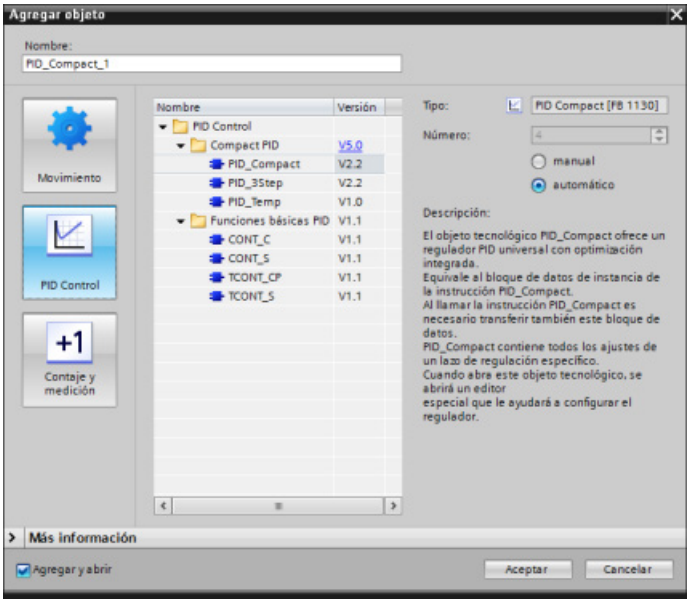
Cuando se inserta una instrucción PID en el programa de usuario, STEP 7 crea automáticamente un objeto tecnológico y un DB de instancia para dicha instrucción. El DB de instancia contiene todos los parámetros que se utilizan para la instrucción PID. Cada instrucción PID debe tener su propio DB de instancia 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 231).	

Tabla 8- 2 (Opcional) Crear un objeto tecnológico desde el árbol del proyecto

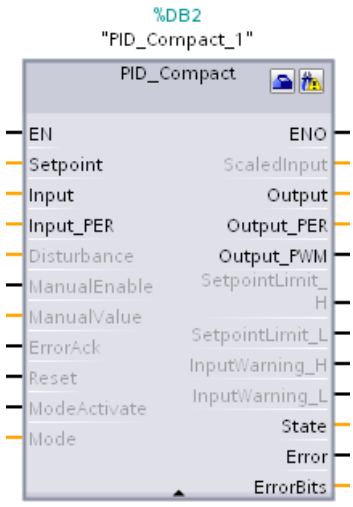
También es posible crear objetos tecnológicos para el proyecto **antes** de insertar la instrucción PID. Si se crea el objeto tecnológico antes de insertar una instrucción PID en el programa de usuario, puede seleccionarse dicho objeto tecnológico al insertar la instrucción PID.	
Para crear un objeto tecnológico, haga doble clic en el icono "Agregar objeto" del árbol del proyecto.	
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.	

8.2 Instrucción PID_Compact

La instrucción PID_Compact ofrece un regulador PID universal con autoajuste integrado para modo automático y manual.

Tabla 8- 3 Instrucción PID_Compact

KOP / FUP	SCL	Descripción

	`"PID_Compact_1"(`
	`Setpoint:=_real_in_,`
	`Input:=_real_in_,`
	`Input_PER:=_word_in_,`
	`Disturbance:=_real_in_,`
	`ManualEnable:=_bool_in_,`
	`ManualValue:=_real_in_,`
	`ErrorAck:=_bool_in_,`
	`Reset:=_bool_in_,`
	`ModeActivate:=_bool_in_,`
	`Mode:=_int_in_,`
	`ScaledInput=>_real_out_,`
	`Output=>_real_out_,`
	`Output_PER=>_word_out_,`
	`Output_PWM=>_bool_out_,`
	`SetpointLimit_H=>_bool_out_,`
	`SetpointLimit_L=>_bool_out_,`
	`InputWarn-`
	`ing_H=>_bool_out_,`
	`InputWarn-`
	`ing_L=>_bool_out_,`
	`State=>_int_out_,`
	`Error=>_bool_out_,`
	`ErrorBits=>_dword_out_);`
		PID_Compact ofrece un regulador PID con autoajuste para modo automático y modo manual. PID_Compact es un regulador PID T1 con anti-windup y ponderación de las acciones P y D.

[1] 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.

[2] En el ejemplo SCL, "PID_Compact_1" es el nombre del DB de instancia.

Tabla 8- 4 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	Una variable del programa de usuario se utiliza como fuente para el valor de proceso. (Valor predeterminado: 0,0) Si se utiliza el parámetro Input, hay que poner Config.InputPerOn = FALSE.
Input_PER	IN	Word	Una entrada analógica se utiliza como fuente del valor de proceso. (Valor predeterminado: W#16#0) Si se utiliza el parámetro Input_PER, hay que poner Config.InputPerOn = TRUE.
Disturbance	IN	Real	Variable de perturbación o valor de precontrol
ManualEnable	IN	Bool	Activa o desactiva el modo de operación manual. (Valor predeterminado: FALSE): • Un flanco de FALSE a TRUE activa el "modo manual", mientras que con State = 4 el modo permanece invariable. Mientras ManualEnable = TRUE, no se puede cambiar el estado operativo con un flanco ascendente en ModeActivate ni utilizar el cuadro de diálogo de puesta en marcha. • Un flanco de TRUE a FALSE activa el estado operativo asignado por Mode. Nota: recomendamos cambiar el estado operativo utilizando solo ModeActivate.
ManualValue	IN	Real	Valor de salida para operación manual. (Valor predeterminado: 0,0) Se pueden utilizar valores de Config.OutputLowerLimit a Config.OutputUpperLimit.
ErrorAck	IN	Bool	Resetea los ErrorBits y las salidas de advertencia. Flanco de FALSE a TRUE
Reset	IN	Bool	Reinicia el regulador. (Valor predeterminado: FALSE): • Flanco de FALSE a TRUE: – Cambia al modo "inactivo". – Resetea los ErrorBits y las salidas de advertencia – Borra la acción integral – Mantiene los parámetros PID • Mientras Reset = TRUE, PID_Compact permanece en modo "Inactivo" (estado = 0). • Flanco de TRUE a FALSE: – PID_Compact cambia al estado operativo guardado en el parámetro Mode.
ModeActivate	IN	Bool	PID_Compact cambia al estado operativo guardado en el parámetro Mode. Flanco de FALSE a TRUE:
Mode	IN	Int	El modo PID deseado; activado en el primer flanco de la entrada Mode Activate .
ScaledInput	OUT	Real	Valor de proceso escalado. (Valor predeterminado: 0,0)
Output[1]	OUT	Real	Valor de salida en formato REAL. (Valor predeterminado: 0,0)

Parámetro y tipo		Tipo de datos	Descripción
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) Los tiempos de activación y desactivación forman el valor de salida.
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 (Setpoint ≥ Config.SetpointUpperLimit). La consigna está limitada a Config.SetpointUpperLimit.
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 (Setpoint ≤ Config.SetpointLowerLimit). La consigna está limitada a Config.SetpointLowerLimit.
InputWarning_H	OUT	Bool	Si InputWarning_H = TRUE, el valor de proceso ha alcanzado o rebasado por exceso el límite superior de advertencia. (Valor predeterminado: FALSE)
InputWarning_L	OUT	Bool	Si InputWarning_L = TRUE, el valor de proceso ha alcanzado o rebasado por defecto el límite inferior de advertencia. (Valor predeterminado: FALSE)
State	OUT	Int	Estado operativo actual del regulador PID. (Valor predeterminado: 0) Se puede cambiar el estado operativo utilizando el parámetro de entrada Mode y un flanco ascendente en ModeActivate: • State = 0: Inactivo • State = 1: optimización inicial • State = 2: Optimización fina manual • State = 3: Modo automático • State = 4: Modo manual • State = 5: valor de salida sustitutivo con vigilancia de errores
Error	OUT	Bool	Si Error = TRUE, hay como mínimo un mensaje de error pendiente en este ciclo. (Valor predeterminado: FALSE) Nota: el parámetro Error en V1.x PID era el campo ErrorBits que contenía los códigos de error. Ahora es una marca booleana que indica que se ha producido un error.
ErrorBits	OUT	DWord	La tabla de parámetros (Página 207) ErrorBits de la instrucción PID_Compact define los mensajes de error pendientes. (Valor predeterminado: DW#16#0000 (ningún error)). ErrorBits es remanente y se resetea con un flanco ascendente en Reset o ErrorAck. Nota: en V1.x, el parámetro ErrorBits estaba definido como el parámetro Error y no existía.

[1] Las salidas de los parámetros Output, Output_PER y Output_PWM pueden utilizarse en paralelo.

Operación del regulador PID_Compact

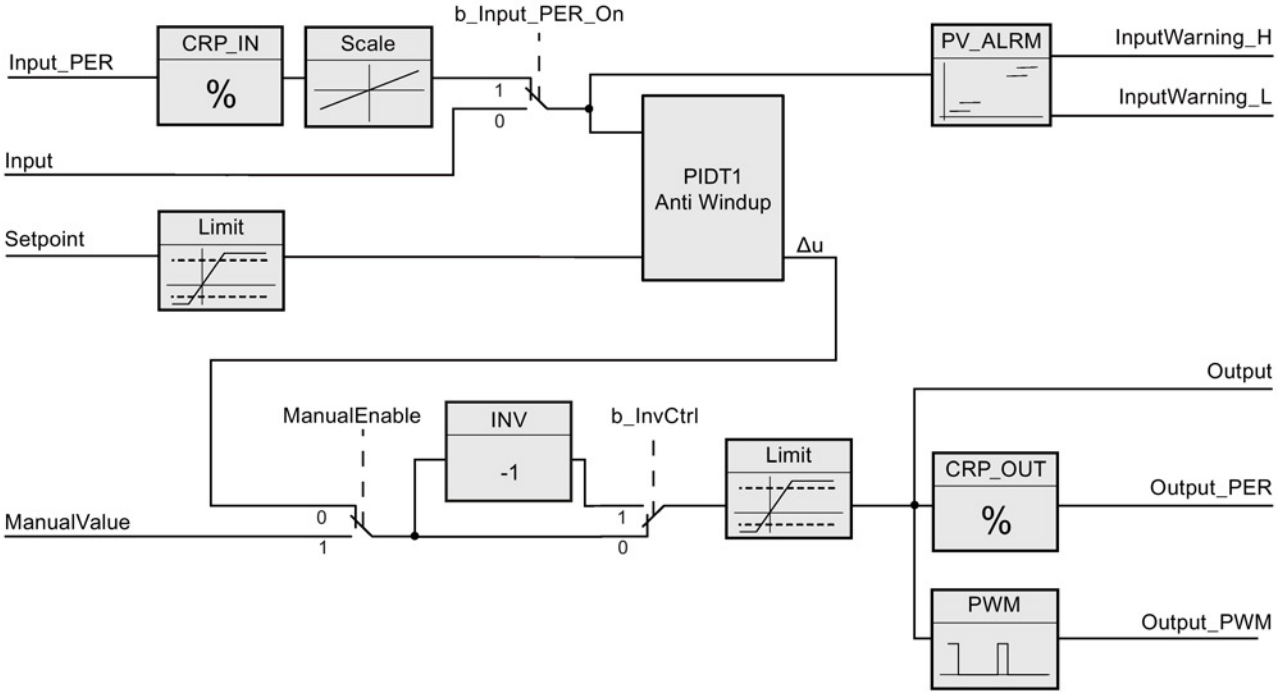


Figura 8-1 Operación del regulador PID_Compact

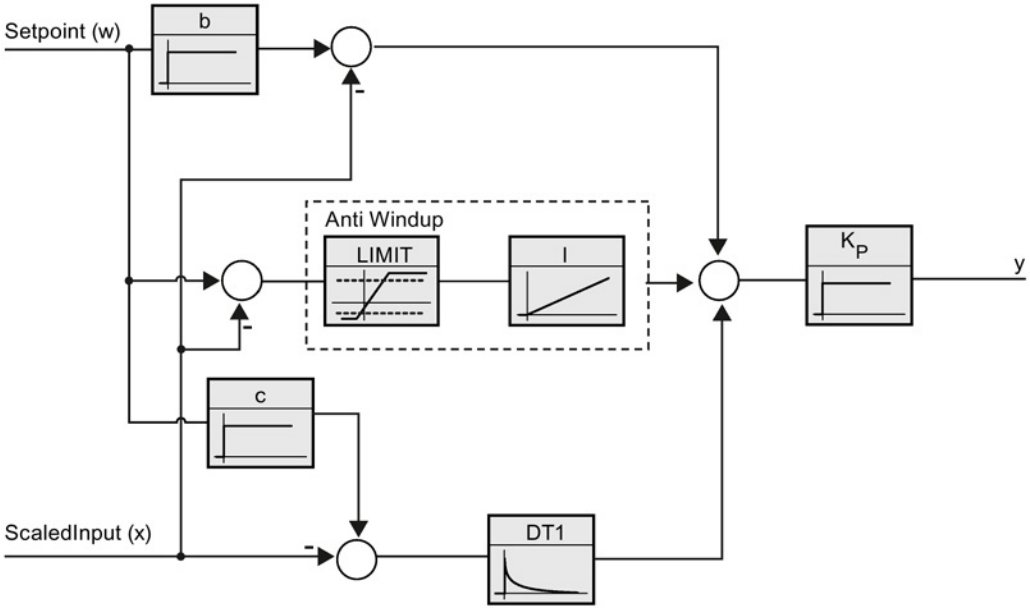


Figura 8-2 Operación del regulador PID_Compact como regulador PIDT1 con anti-windup

8.3 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 8- 5 Parámetros de la instrucción ErrorBit de PID_Compact

ErrorBit (DW#16#...)	Descripción
0000	No hay error
0001 [1, 2]	El parámetro Input se encuentra fuera de los límites del valor de proceso. Input > Config.InputUpperLimit Input < Config.InputLowerLimit
0002 [2, 3]	Valor no válido en el parámetro Input_PER. Compruebe si hay un error pendiente en la entrada analógica.
0004 [4]	Error durante la optimización fina. No ha sido posible mantener la oscilación del valor de proceso.
0008 [4]	Error al iniciar la optimización inicial. El valor de proceso está demasiado próximo a la consigna. Inicie la optimización fina.
0010 [4]	La consigna se ha cambiado durante la optimización. Nota: se puede ajustar la fluctuación permitida en la consigna en la variable CancelTuningLevel.
0020	La optimización inicial no está permitida durante la optimización fina. Nota: si ActivateRecoverMode = TRUE antes de que se produjera el error, PID_Compact permanece en el modo de optimización fina.
0080 [4]	Error durante la optimización inicial. Configuración incorrecta de los límites del valor de salida. Compruebe si los límites del valor de salida están configurados correctamente y coinciden con la lógica de control.
0100 [4]	Un error durante la optimización fina ha provocado parámetros no válidos.
0200 [2, 3]	Valor no válido en el parámetro Input: el valor tiene un formato de número no válido.
0400 [2, 3]	Fallo en el cálculo del valor de salida. Compruebe los parámetros PID.
0800 [1, 2]	Error de tiempo de muestreo: PID_Compact no se llama durante el tiempo de muestreo del OB de alarma cíclica.
1000 [2, 3]	Valor no válido en el parámetro Setpoint: el valor tiene un formato de número no válido.
10000	Valor no válido en el parámetro ManualValue: el valor tiene un formato de número no válido. Nota: si ActivateRecoverMode = TRUE antes de que se produjera el error, PID_Compact utiliza SubstituteOutput como valor de salida. En cuanto se asigna un valor válido en el parámetro ManualValue, PID_Compact lo utiliza como valor de salida.

ErrorBit (DW#16#...)	Descripción
20000	Valor no válido en la variable SubstituteValue: el valor tiene un formato de número no válido. PID_Compact utiliza el límite inferior de valor de salida como valor de salida. Nota: si el modo automático estaba activo antes de que ocurriera el error, ActivateRecoverMode = TRUE y el error ya no está pendiente, PID_Compact vuelve al modo automático.
40000	Valor no válido en el parámetro Disturbance: el valor tiene un formato de número no válido. Nota: si el modo automático estaba activo y ActivateRecoverMode = FALSE antes de que se produjera el error, Disturbance se pone a cero. PID_Compact permanece en modo automático. Nota: si el modo de optimización inicial o fina estaba activo y ActivateRecoverMode = TRUE antes de que se produjera el error, PID_Compact cambia al estado operativo guardado en el parámetro Mode. Si la perturbación en la fase actual no afecta al valor de salida, la optimización no se cancela.

[1] Nota: si el modo automático estaba activo antes de que se produjera el error y ActivateRecoverMode = TRUE, PID_Compact permanece en modo automático.

[2] Nota: si el modo de optimización inicial o fina estaba activo antes de que se produjera el error y ActivateRecoverMode = TRUE, PID_Compact cambia al estado operativo guardado en el parámetro Mode.

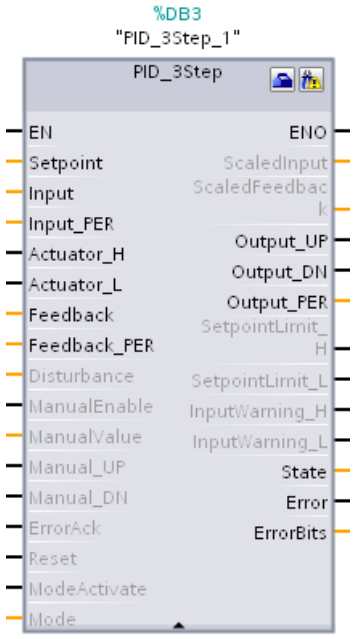
[3] Nota: si el modo automático estaba activo antes de que se produjera el error y ActivateRecoverMode = TRUE, PID_Compact emite el valor de salida sustitutivo configurado. En cuanto el error deja de estar pendiente, PID_Compact regresa al modo automático.

[4] Nota: si ActivateRecoverMode = TRUE antes de que se produjera el error, PID_Compact cancela la optimización y cambia al estado operativo guardado en el parámetro Mode.

8.4 Instrucción PID_3Step

La instrucción PID_3Step configura un regulador PID con capacidades de autoajuste que se ha optimizado para válvulas accionadas por motor y actuadores.

Tabla 8- 6 Instrucción PID_3Step

KOP / FUP	SCL	Descripción
	"PID_3Step_1"(SetpoInt:=_real_in_, Input:=_real_in_, ManualValue:=_real_in_, Feedback:=_real_in_, InputPer:=_word_in_, FeedbackPer:=_word_in_, Disturbance:=_real_in_, ManualEnable:=_bool_in_, ManualUP:=_bool_in_, ManualDN:=_bool_in_, ActuatorH:=_bool_in_, ActuatorL:=_bool_in_, ErrorAck:=_bool_in_, Reset:=_bool_in_, ModeActivate:=_bool_in_, Mode:=_int_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_, ErrorBits=>_dword_out_);	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. PID_3Step es un regulador PID T1 con anti-windup y ponderación de las acciones P y D.

[1] 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.

[2] En el ejemplo SCL, "PID_3Step_1" es el nombre del DB de instancia.

Tabla 8- 7 Tipos de datos de 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	Una variable del programa de usuario se utiliza como fuente para el valor de proceso. (Valor predeterminado: 0.0) Si se utiliza el parámetro Input, hay que poner Config.InputPerOn = FALSE.
Input_PER	IN	Word	Una entrada analógica se utiliza como fuente del valor de proceso. (Valor predeterminado: W#16#0) Si se utiliza el parámetro Input_PER, hay que poner Config.InputPerOn = TRUE.
Actuator_H	IN	Bool	Realimentación de posición digital de la válvula para la posición límite superior Si Actuator_H = TRUE, la válvula está en la posición límite superior y ya no se mueve en ese sentido. (Valor predeterminado: FALSE)
Actuator_L	IN	Bool	Realimentación de posición digital de la válvula para la posición límite inferior Si Actuator_L = TRUE, la válvula está en la posición límite inferior y ya no se mueve en ese sentido. (Valor predeterminado: FALSE)
Feedback	IN	Real	Realimentación de posición de la válvula. (Valor predeterminado: 0.0) Si se utiliza el parámetro Feedback, hay que poner Config.FeedbackPerOn = FALSE.
Feedback_PER	IN	Int	Realimentación analógica de la posición de la válvula. (Valor predeterminado: W#16#0) Si se utiliza el parámetro Feedback_PER, hay que poner Config.FeedbackPerOn = TRUE. Feedback_PER se escala en función de las siguientes variables: • Config.FeedbackScaling.LowerPointIn • Config.FeedbackScaling.UpperPointIn • Config.FeedbackScaling.LowerPointOut • Config.FeedbackScaling.UpperPointOut
Disturbance	IN	Real	Variable de perturbación o valor de precontrol
ManualEnable	IN	Bool	Activa o desactiva el modo de operación manual. (Valor predeterminado: FALSE): • Un flanco de FALSE a TRUE activa el "modo manual", mientras que con State = 4 el modo permanece invariable. Mientras ManualEnable = TRUE, no se puede cambiar el modo de operación con un flanco ascendente en ModeActivate ni utilizar el cuadro de diálogo de puesta en servicio. • Un flanco de TRUE a FALSE activa el modo de operación asignado por Mode. Nota: recomendamos cambiar el modo de operación utilizando solo ModeActivate.

Parámetro y tipo		Tipo de datos	Descripción
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 solo se evalúa si se utiliza OutputPer **o** si hay realimentación de posición.
ManualUP	IN	Bool	• Manual_UP = TRUE: – La válvula se abre incluso si se utiliza Output_PER o una realimentación de posición. La válvula deja de moverse cuando se alcanza la posición límite superior. – Consulte también Config.VirtualActuatorLimit • Manual_UP = FALSE: – Si se utiliza Output_PER o una realimentación de posición, la válvula se mueve a ManualValue. De lo contrario, la válvula ya no se mueve. Nota: si Manual_UP y Manual_DN se ponen a TRUE simultáneamente, la válvula no se mueve.
ManualDN	IN	Bool	• Manual_DN = TRUE: – La válvula se abre incluso si se utiliza Output_PER o una realimentación de posición. La válvula deja de moverse cuando se alcanza la posición límite superior. – Consulte también Config.VirtualActuatorLimit • Manual_DN = FALSE: – Si se utiliza Output_PER o una realimentación de posición, la válvula se mueve a ManualValue. De lo contrario, la válvula ya no se mueve.
ErrorAck	IN	Bool	Resetea los ErrorBits y las salidas de advertencia. Flanco de FALSE a TRUE
Reset	IN	Bool	Reinicia el regulador. (Valor predeterminado: FALSE): • Flanco de FALSE a TRUE: – Cambia al modo "inactivo". – Resetea los ErrorBits y las salidas de advertencia – Borra la acción integral – Mantiene los parámetros PID • Mientras Reset = TRUE, PID_3Step permanece en modo "Inactivo" (estado = 0). • Flanco de TRUE a FALSE: – PID_3Step cambia al modo de operación guardado en el parámetro Mode.
ModeActivate	IN	Bool	PID_3Step cambia al estado guardado en el parámetro Mode. Flanco de FALSE a TRUE:
Mode	IN	Int	El modo PID deseado; activado en el primer flanco de la entrada Mode Activate .
ScaledInput	OUT	Real	Valor de proceso escalado

Parámetro y tipo		Tipo de datos	Descripción
ScaledFeedback	OUT	Real	Realimentación de posición escalada de válvula Nota: para un actuador sin realimentación de posición, la posición del actuador indicada en ScaledFeedback es muy imprecisa. ScaledFeedback solo puede utilizarse para una estimación aproximada de la posición actual en este caso.
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.
Output_PER	OUT	Word	Valor de salida analógico. Si Config.OutputPerOn = TRUE, se usa el parámetro Output_PER.
SetpointLimitH	OUT	Bool	Límite superior de consigna. (Valor predeterminado: FALSE) Si SetpointLimitH = TRUE, se ha alcanzado el límite superior absoluto de la consigna (Setpoint ≥ Config.SetpointUpperLimit). Nota: la consigna está limitada a (Setpoint ≥ Config.SetpointUpperLimit).
SetpointLimitL	OUT	Bool	Límite inferior de consigna. (Valor predeterminado: FALSE) Si SetpointLimitL = TRUE, se ha alcanzado el límite inferior absoluto de la consigna (Setpoint ≥ Config.SetpointLowerLimit). Nota: la consigna está limitada a (Setpoint ≥ Config.SetpointLowerLimit).
InputWarningH	OUT	Bool	Si InputWarningH = TRUE, el valor de entrada ha alcanzado o rebasado por exceso el límite superior de advertencia. (Valor predeterminado: FALSE)
InputWarningL	OUT	Bool	Si InputWarningL = TRUE, el valor de entrada ha alcanzado o rebasado por defecto el límite inferior de advertencia. (Valor predeterminado: FALSE)
State	OUT	Int	Modo de operación actual del regulador PID. (Valor predeterminado: 0) Se puede cambiar el modo de operación utilizando el parámetro de entrada Mode y un flanco ascendente en ModeActivate: • 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: vigilancia de errores • State = 8: aproximación al valor de salida sustitutivo con vigilancia de errores\ • State = 10: modo manual sin señales de posición límite

Parámetro y tipo		Tipo de datos	Descripción
Error	OUT	Bool	Si Error = TRUE, hay como mínimo un mensaje de error pendiente. (Valor predeterminado: FALSE) Nota: el parámetro Error en V1.x PID era el campo ErrorBits que contenía los códigos de error. Ahora es una marca booleana que indica que se ha producido un error.
ErrorBits	OUT	DWord	La tabla de parámetros (Página 216) ErrorBits de la instrucción PID_3Step define los mensajes de error pendientes. (Valor predeterminado: DW#16#0000 (ningún error)). ErrorBits es remanente y se resetea con un flanco ascendente en Reset o ErrorAck. Nota: en V1.x, el parámetro ErrorBits estaba definido como el parámetro Error y no existía.

Operación del regulador PID_3Step

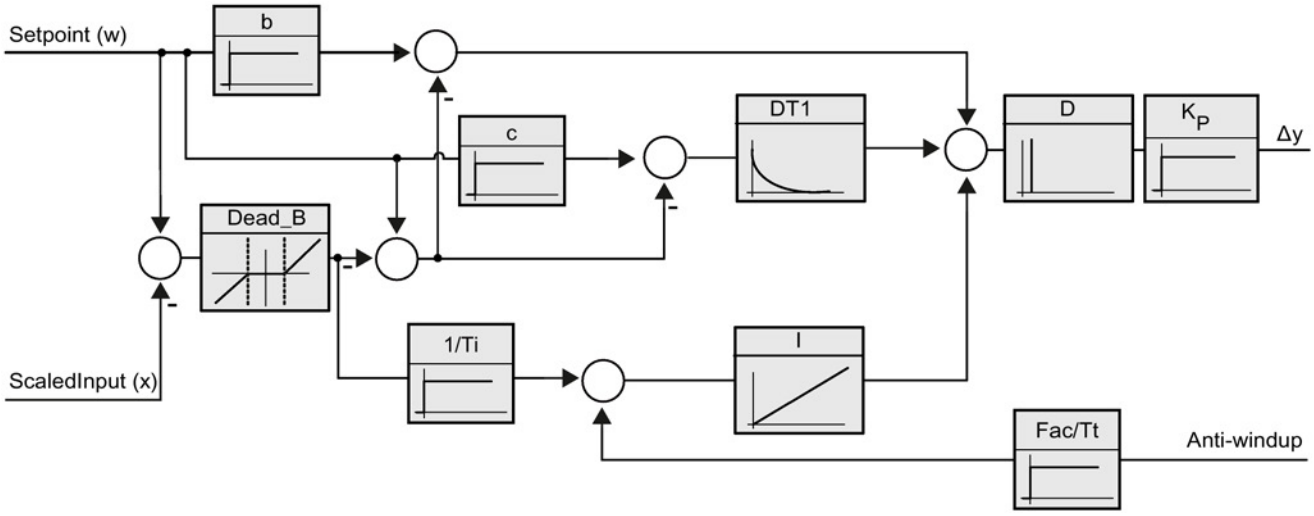


Figura 8-3 Operación del regulador PID_3Step como regulador PID T1 con anti-windup

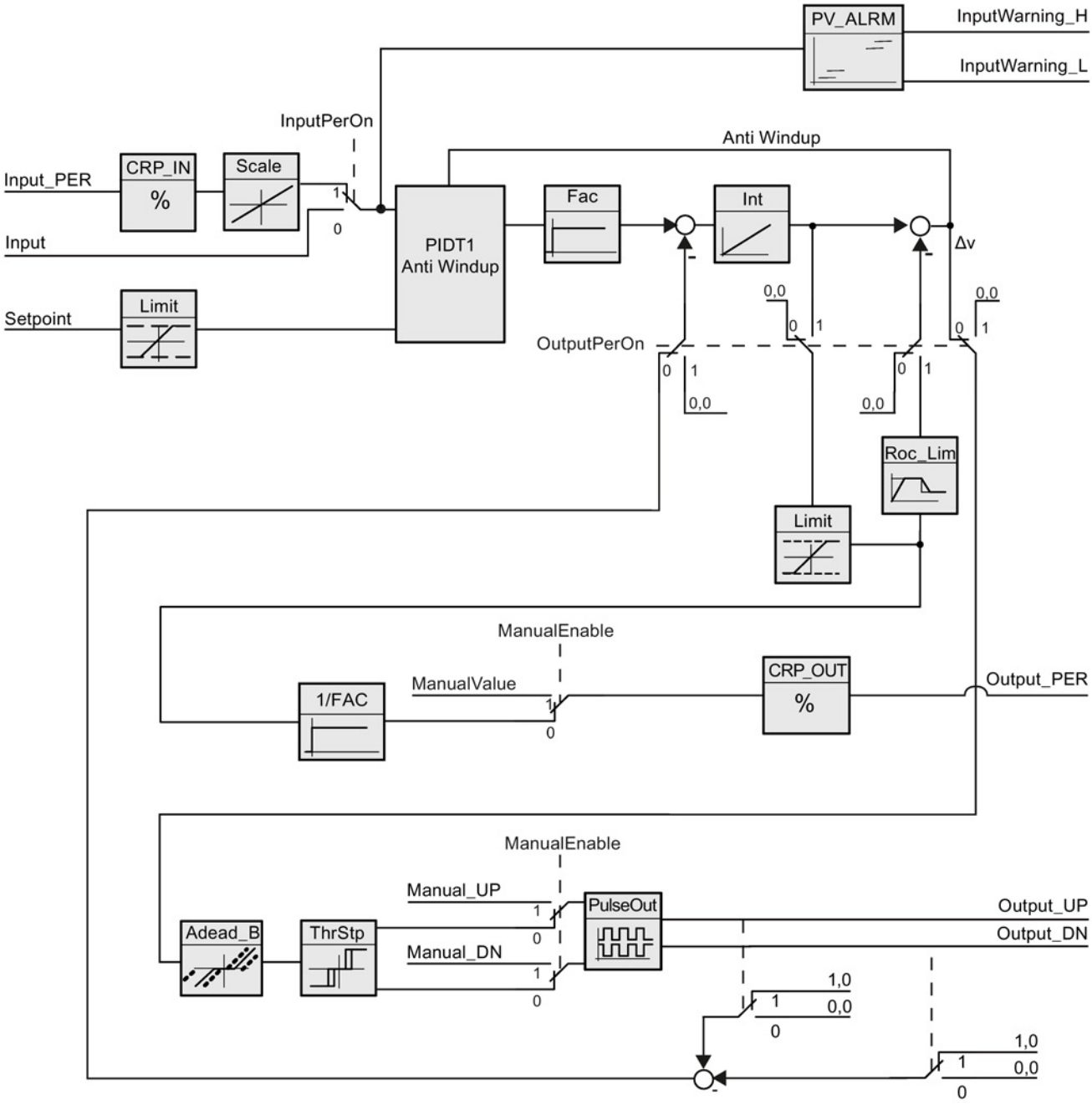


Figura 8-4 Operación del regulador PID_3Step sin realimentación de posición

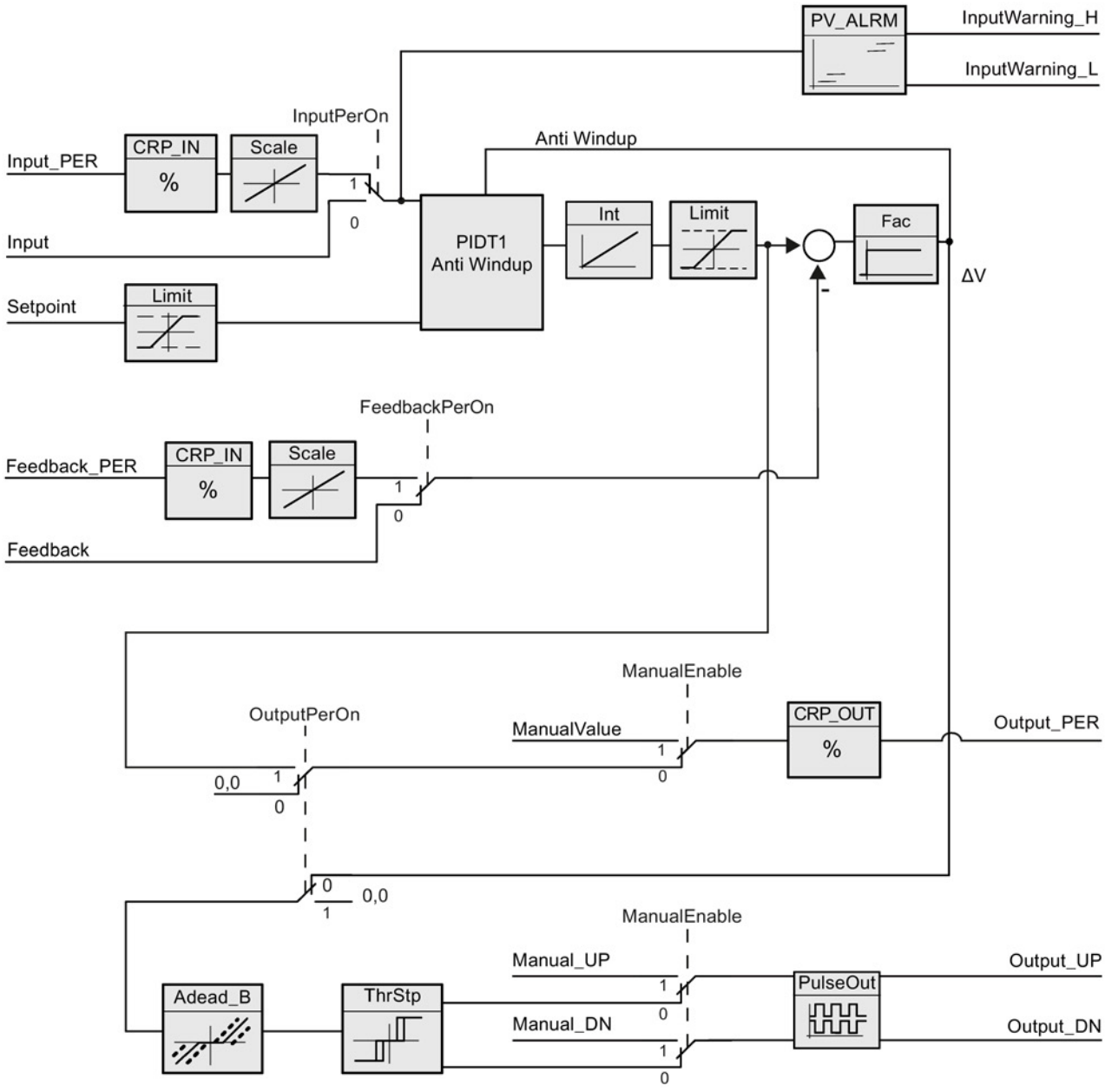


Figura 8-5 Operación del regulador PID_3Step con realimentación de posición habilitada

8.5 Parámetros ErrorBit de la instrucción 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 8- 8 Parámetros ErrorBit de la instrucción PID_3STEP

ErrorBit (DW#16#...)	Descripción
0000	No hay error
0001 [1, 2]	El parámetro Input se encuentra fuera de los límites del valor de proceso. Input > Config.InputUpperLimit Input < Config.InputLowerLimit
0002 [2, 3]	Valor no válido en el parámetro Input_PER. Compruebe si hay un error pendiente en la entrada analógica.
0004 [4]	Error durante la optimización fina. No ha sido posible mantener la oscilación del valor de proceso.
0010 [4]	La consigna se ha cambiado durante la optimización. Nota: se puede ajustar la fluctuación permitida en la consigna en la variable CancelTuningLevel.
0020	La optimización inicial no está permitida durante la optimización fina. Nota: si ActivateRecoverMode = TRUE antes de que se produjera el error, PID_3Step permanece en el modo de optimización fina.
0080 [4]	Error durante la optimización inicial. Configuración incorrecta de los límites del valor de salida. Compruebe si los límites del valor de salida están configurados correctamente y coinciden con la lógica de control.
0100 [4]	Un error durante la optimización fina ha provocado parámetros no válidos.
0200 [2, 3]	Valor no válido en el parámetro Input: el valor tiene un formato de número no válido.
0400 [2, 3]	Fallo en el cálculo del valor de salida. Compruebe los parámetros PID.
0800 [1, 2]	Error de tiempo de muestreo: PID_3Step no se llama durante el tiempo de muestreo del OB de alarma cíclica.
1000 [2, 3]	Valor no válido en el parámetro Setpoint: el valor tiene un formato de número no válido.
2000 [1, 2, 5]	Valor no válido en el parámetro Feedback_PER. Compruebe si hay un error pendiente en la entrada analógica.
4000 [1, 2, 5]	Valor no válido en el parámetro Feedback: el valor tiene un formato de número no válido.
8000 [1, 2]	Error en la realimentación de posición digital. Actuator_H = TRUE y Actuator_L = TRUE. El actuador no se puede mover al valor de salida sustitutivo y permanece en la posición actual. En este estado no es posible el modo manual. Para cambiar el estado del actuador hay que desactivar la "posición límite del actuador" (Config.ActuatorEndStopOn = FALSE) o cambiar al modo manual sin señales de posición límite (Mode = 10).

ErrorBit (DW#16#...)	Descripción
10000	Valor no válido en el parámetro ManualValue: el valor tiene un formato de número no válido. El actuador no se puede mover al valor manual y permanece en la posición actual. Asigne un valor válido en ManualValue o mueva el actuador en modo manual con Manual_UP y Manual_DN.
20000	Valor no válido en la variable SavePosition: el valor tiene un formato de número no válido. El actuador no se puede mover al valor de salida sustitutivo y permanece en la posición actual.
40000	Valor no válido en el parámetro Disturbance: el valor tiene un formato de número no válido. Nota: si el modo automático estaba activo y ActivateRecoverMode = FALSE antes de que se produjera el error, Disturbance se pone a cero. PID_3Step permanece en modo automático. Nota: si el modo de optimización inicial o fina estaba activo y ActivateRecoverMode = TRUE antes de que se produjera el error, PID_3Step cambia al estado operativo guardado en el parámetro Mode. Si la perturbación en la fase actual no afecta al valor de salida, la optimización no se cancela. El error no tiene ningún efecto durante la medición del tiempo de transición.

[1] Nota: si el modo automático estaba activo antes de que se produjera el error y ActivateRecoverMode = TRUE, PID_3Step permanece en modo automático.

[2] Nota: si el modo de optimización inicial o fina o bien la medición de tiempo de transición estaban activos y ActivateRecoverMode = TRUE antes de que se produjera el error, PID_3Step cambia al estado operativo guardado en el parámetro Mode.

[3] Nota: si el modo automático estaba activo antes de que se produjera el error y ActivateRecoverMode = TRUE, PID_3Step cambia a "Aproximación al valor de salida sustitutivo con vigilancia de errores" o "Vigilancia de errores". En cuanto el error deja de estar pendiente, PID_3Step regresa al modo automático.

[4] Nota: si ActivateRecoverMode = TRUE antes de que se produjera el error, PID_3Step cancela la optimización y cambia al estado operativo guardado en el parámetro Mode.

[5] El actuador no se puede mover al valor de salida sustitutivo y permanece en la posición actual. En el modo manual la posición del actuador solo se puede cambiar con Manual_UP y Manual_DN, pero no con ManualValue.

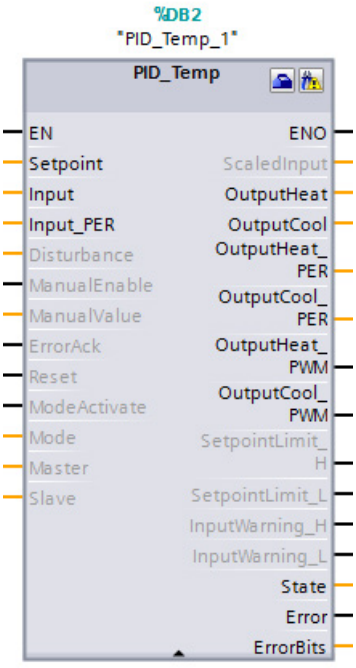
8.6 Instrucción PID_Temp

8.6.1 Vista general

La instrucción PID_Temp ofrece un regulador PID universal que permite manejar los requisitos específicos de control de temperatura.

Tabla 8- 9 Instrucción PID_Temp

KOP / FUP	SCL	Descripción

	`"PID_Temp_1"(`
	`Setpoint:=_real_in_,`
	`Input:=_real_in_,`
	`Input_PER:=_int_in_,`
	`Disturbance:=_real_in_,`
	`ManualEnable:=_bool_in_,`
	`ManualValue:=_real_in_,`
	`ErrorAck:=_bool_in_,`
	`Reset:=_bool_in_,`
	`ModeActivate:=_bool_in_,`
	`Mode:=_int_in_,`
	`Master:=_dword_in`
	`Save:=_dword_in`
	`ScaledInput=>_real_out_,`
	`OutputHeat=>_real_out_,`
	`OutputCool=>_real_out_,`
	`OutputHeat_PER=>_int_out_,`
	`OutputCool_PER=>_int_out_,`
	`Out-`
	`putHeat_PWM=>_bool_out_,`
	`Out-`
	`putCool_PWM=>_bool_out_,`
	`SetpointLimit_H=>_bool_out_,`
	`SetpointLimit_L=>_bool_out_,`
	`InputWarn-`
	`ing_H=>_bool_out_,`
	`InputWarn-`
	`ing_L=>_bool_out_,`
	`State=>_int_out_,`
	`Error=>_bool_out_,`
	`ErrorBits=>_dword_out_);`
		PID_Temp ofrece las funcionalidades siguientes:
		• Calentar y refrigerar el proceso con diferentes actuadores
		• Autotuning integrado para manejar procesos de temperatura
		• Función de cascada para procesar más de una temperatura que depende del mismo actuador.

[1] 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.

[2] En el ejemplo SCL, "PID_Temp_1" es el nombre del DB de instancia.

Tabla 8- 10 Tipos de datos de 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	Una variable del programa de usuario se utiliza como fuente para el valor de proceso. (Valor predeterminado: 0.0) Si se utiliza el parámetro Input, hay que poner Config.InputPerOn = FALSE.
Input_PER	IN	Int	Una entrada analógica se utiliza como fuente del valor de proceso. (Valor predeterminado: 0) Si se utiliza el parámetro Input_PER, hay que poner Config.InputPerOn = TRUE.
Disturbance	IN	Real	Variable de perturbación o valor de precontrol
ManualEnable	IN	Bool	Activa o desactiva el modo de operación manual. (Valor predeterminado: FALSE): • Un flanco de FALSE a TRUE activa el modo manual, mientras que si State = 4 el modo permanece invariable. Mientras ManualEnable = TRUE, no se puede cambiar el modo de operación con un flanco ascendente en ModeActivate ni utilizar el cuadro de diálogo de puesta en servicio. • Un flanco de TRUE a FALSE activa el modo de operación asignado por Mode. Nota: recomendamos cambiar el modo de operación utilizando solo ModeActivate.
ManualValue	IN	Real	Valor de salida para operación manual. (Valor predeterminado: 0.0) Se pueden utilizar valores de Config.OutputLowerLimit a Config.OutputUpperLimit.
ErrorAck	IN	Bool	Desactiva los ErrorBits y las salidas de advertencia con un flanco de FALSE a TRUE. (Valor predeterminado: FALSE)
Reset	IN	Bool	Reinicia el regulador. (Valor predeterminado: FALSE): • Flanco de FALSE a TRUE: – Cambia al modo "inactivo". – Resetea los ErrorBits y las salidas de advertencia – Borra la acción integral – Mantiene los parámetros PID • Mientras Reset = TRUE, PID_Temp permanece en modo inactivo (estado = 0). • Flanco de TRUE a FALSE: – PID_Temp cambia al modo de operación guardado en el parámetro Mode.
ModeActivate	IN	Bool	PID_Temp cambia al modo de operación guardado en el parámetro Mode con un flanco de FALSE a TRUE. (Valor predeterminado: FALSE)

Parámetro y tipo		Tipo de datos	Descripción
Mode	IN/OUT	Int	Activado en el primer flanco de la entrada Mode Activate. Selección del modo de operación (valor predeterminado: 0.0): • Mode = 0: inactivo • Mode = 1: optimización inicial • Mode = 2: optimización fina • Mode = 3: modo automático • Mode = 4: modo manual "Valor de salida sustitutivo con vigilancia de errores" (State = 5). El usuario no puede activarlo; solo es una reacción de error automática.
Master	IN/OUT	DWord	Conexión en cascada al maestro (condiciones AntiWindUp y de optimización). (Valor predeterminado: DW#16#0000) • Bits 0 - 15: no se usan en la instrucción PID_Temp • Bits 16 - 23: limitar contador: un esclavo incrementa este valor si alcanza su limitación. El número de esclavos en la limitación se procesa para el funcionamiento Anti-Windup (consulte el parámetro Config.Cascade.AntiWindUpMode). • Bit 24: IsAutomatic: este bit se pone a "1" si todos los esclavos de este controlador están en modo automático y son procesados para comprobar las condiciones de optimización en una cascada. Este bit es idéntico al parámetro AllSlaveAutomaticState. • Bit 25: "IsReplacement-Setpoint": este bit se pone a "1" si un esclavo de este controlador tiene activada la "consigna de sustitución" y es procesado para comprobar las condiciones de optimización en una cascada. El valor invertido se guarda en el parámetro NoSlaveReplacementSetpoint.
Slave	IN/OUT	DWord	
ScaledInput	OUT	Real	Valor de proceso escalado. (Valor predeterminado: 0.0)
OutputHeat[1]	OUT	Real	Valor de salida para calentamiento en formato REAL. (Valor predeterminado: 0.0) Este valor de salida se calcula con el parámetro Config.Output.Heat.Select, independientemente de la selección de salida.
OutputCool[1]	OUT	Real	Valor de salida para refrigeración en formato REAL. (Valor predeterminado: 0.0) Este valor de salida se calcula con el parámetro Config.Output.Cool.Select, independientemente de la selección de salida.
OutputHeat_PER[1]	OUT	Int	Valor de salida para calentamiento en formato periférico (valor predeterminado: 0) Este valor de salida solo se calcula si se selecciona usando el parámetro Config.Output.Heat.Select =2. Si no se selecciona, esta salida siempre es "0".
OutputCool_PER[1]	OUT	Int	Valor de salida para calentamiento en formato periférico (valor predeterminado: 0) Este valor de salida solo se calcula si se selecciona usando el parámetro Config.Output.Cool.Select =2. Si no se selecciona, esta salida siempre es "0".

Parámetro y tipo		Tipo de datos	Descripción
OutputHeat_PWM[1]	OUT	Bool	Valor de salida con modulación del ancho de impulso para calefacción. (Valor predeterminado: FALSE) Este valor de salida solo se calcula si se selecciona usando el parámetro Config.Output.Heat.Select =1 (valor predeterminado). Si no se selecciona, esta salida siempre es FALSE.
OutputCool_PWM[1]	OUT	Bool	Valor de salida con modulación del ancho de impulso para refrigeración. (Valor predeterminado: FALSE) Este valor de salida solo se calcula si se selecciona usando el parámetro Config.Output.Cool.Select =1 (valor predeterminado). Si no se selecciona, esta salida siempre es 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 (Setpoint ≥ Config.SetpointUpperLimit). La consigna está limitada a Config.SetpointUpperLimit.
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 (Setpoint ≤ Config.SetpointLowerLimit). La consigna está limitada a Config.SetpointLowerLimit.
InputWarning_H	OUT	Bool	Si InputWarning_H = TRUE, el valor de proceso ha alcanzado o rebasado por exceso el límite superior de advertencia. (Valor predeterminado: FALSE)
InputWarning_L	OUT	Bool	Si InputWarning_L = TRUE, el valor de proceso ha alcanzado o rebasado por defecto el límite inferior de advertencia. (Valor predeterminado: FALSE)
State	OUT	Int	Modo de operación actual del regulador PID. (Valor predeterminado: 0) Se puede cambiar el modo de operación utilizando el parámetro de entrada Mode y un flanco ascendente en ModeActivate: • State = 0: Inactivo • State = 1: optimización inicial • State = 2: optimización fina • State = 3: Modo automático • State = 4: Modo manual • State = 5: valor de salida sustitutivo con vigilancia de errores
Error	OUT	Bool	Si Error = TRUE, hay como mínimo un mensaje de error pendiente en este ciclo. (Valor predeterminado: FALSE) Nota: el parámetro Error en V1.x PID era el campo ErrorBits que contenía los códigos de error. Ahora es una marca booleana que indica que se ha producido un error.
ErrorBits	OUT	DWord	La instrucción PID_Temp, tabla de parámetros ErrorBits (Página 229), define los mensajes de error pendientes. (Valor predeterminado: DW#16#0000 (ningún error)). ErrorBits es remanente y se resetea con un flanco ascendente en Reset o ErrorAck. Nota: en V1.x, el parámetro ErrorBits estaba definido como el parámetro Error y no existía.

Parámetro y tipo		Tipo de datos	Descripción
Warning	OUT	DWord	La instrucción PID_Temp, tabla de parámetros de advertencia define los mensajes de advertencia pendientes que son relevantes para el usuario. (Valor predeterminado: DW#16#0000 (ninguna advertencia)).
WarningInternal	OUT	DWord	La instrucción PID_Temp, tabla de parámetros WarningInternal, define los mensajes de advertencia internos pendientes (incluye todas las advertencias). (Valor predeterminado: DW#16#0000 (ninguna advertencia interna)).

[1] Las salidas de los parámetros Output, Output_PER y Output_PWM pueden utilizarse en paralelo.

8.6.2 Operación del regulador PID_Temp

Selección de regulación de calentamiento/enfriamiento

Primero hay que seleccionar si se necesita un dispositivo de enfriamiento además de la salida de calentamiento en el parámetro "ActivateCooling". A continuación hay que definir si se desea utilizar dos juegos de parámetros PID (modo avanzado) o solo uno con un factor adicional de calentamiento/enfriamiento en el parámetro "AdvancedCooling".

Uso del parámetro CoolFactor

Si se desea aplicar un factor de calentamiento/enfriamiento, el valor debe definirse manualmente. Hay que determinar el valor a partir de los datos técnicos de la aplicación (relación de ganancia proporcional de los actuadores (por ejemplo, la relación de potencia máxima de calentamiento y enfriamiento de los actuadores) y asignarlo al parámetro "CoolFactor". Un factor de calentamiento/enfriamiento de 2.0 significa que el dispositivo de calentamiento es dos veces más efectivo que el dispositivo de enfriamiento. Si se utiliza un factor de enfriamiento, PID_Temp calculará la señal de salida y, dependiendo de su signo, multiplicará la señal de salida por el factor de calentamiento/enfriamiento (con signo negativo) o no (con signo positivo).

Uso de dos juegos de parámetros PID

Los juegos de parámetros PID diferentes para calentamiento y enfriamiento pueden detectarse automáticamente durante la puesta en servicio. Se puede esperar un mejor rendimiento de regulación en comparación con el factor de calentamiento/enfriamiento porque, además de ganancias proporcionales diferentes, con dos juegos de parámetros se pueden considerar tiempos de retardo diferentes. No obstante, la desventaja es que con este método puede requerirse más tiempo para el proceso de optimización. Si la conmutación de parámetros PID está activada (Config.AdvancedCooling = TRUE), el regulador PID_Temp detectará en "Modo automático" (regulación activada) si es necesario el calentamiento o enfriamiento en ese momento y utilizará juegos de parámetros PID para la regulación.

ControlZone

Con el regulador PID_Temp, se puede definir una zona de regulación para cada juego de parámetros en el parámetro "ControlZone". Si el error de regulación (consigna – entrada) se encuentra dentro de la zona de regulación, PID_Temp utiliza el algoritmo PID para calcular las señales de salida. Sin embargo, si el error de regulación abandona el rango definido, la salida adopta el valor máximo de calentamiento o enfriamiento (salida de enfriamiento activada) / valor de salida de calentamiento mínimo (salida de enfriamiento desactivada). Esta función se puede utilizar para alcanzar más rápidamente la consigna deseada, en especial para el calentamiento inicial de procesos de temperatura lentos.

DeadZone

En el parámetro "DeadZone" se puede definir un ancho del error de regulación para calentamiento y enfriamiento despreciado por el algoritmo PID. Así, un error de regulación dentro de ese rango se suprime y el regulador PID_Temp se comporta como si la consigna y los valores del proceso fueran idénticos. Con ello se pueden minimizar intervenciones innecesarias por parte del regulador alrededor de la consigna y proteger el actuador. Si se desea aplicar una zona muerta (DeadZone), el valor debe definirse manualmente. El autoajuste no activa el valor DeadZone automáticamente. DeadZone es simétrica (entre -Retain.CtrlParams.Heat.DeadZone y +Retain.CtrlParams.Heat.DeadZone) para reguladores de calentamiento sin enfriamiento o reguladores de calentamiento/enfriamiento con CoolFactor. DeadZone puede ser asimétrica (entre -Retain.CtrlParams.Cool.DeadZone y +Retain.CtrlParams.Heat.DeadZone) para reguladores de calentamiento/enfriamiento con dos juegos de parámetros PID.

Operaciones de regulador PID_Temp

Los siguientes diagramas de bloques muestran las operaciones estándar y en cascada de la instrucción PID_Temp:

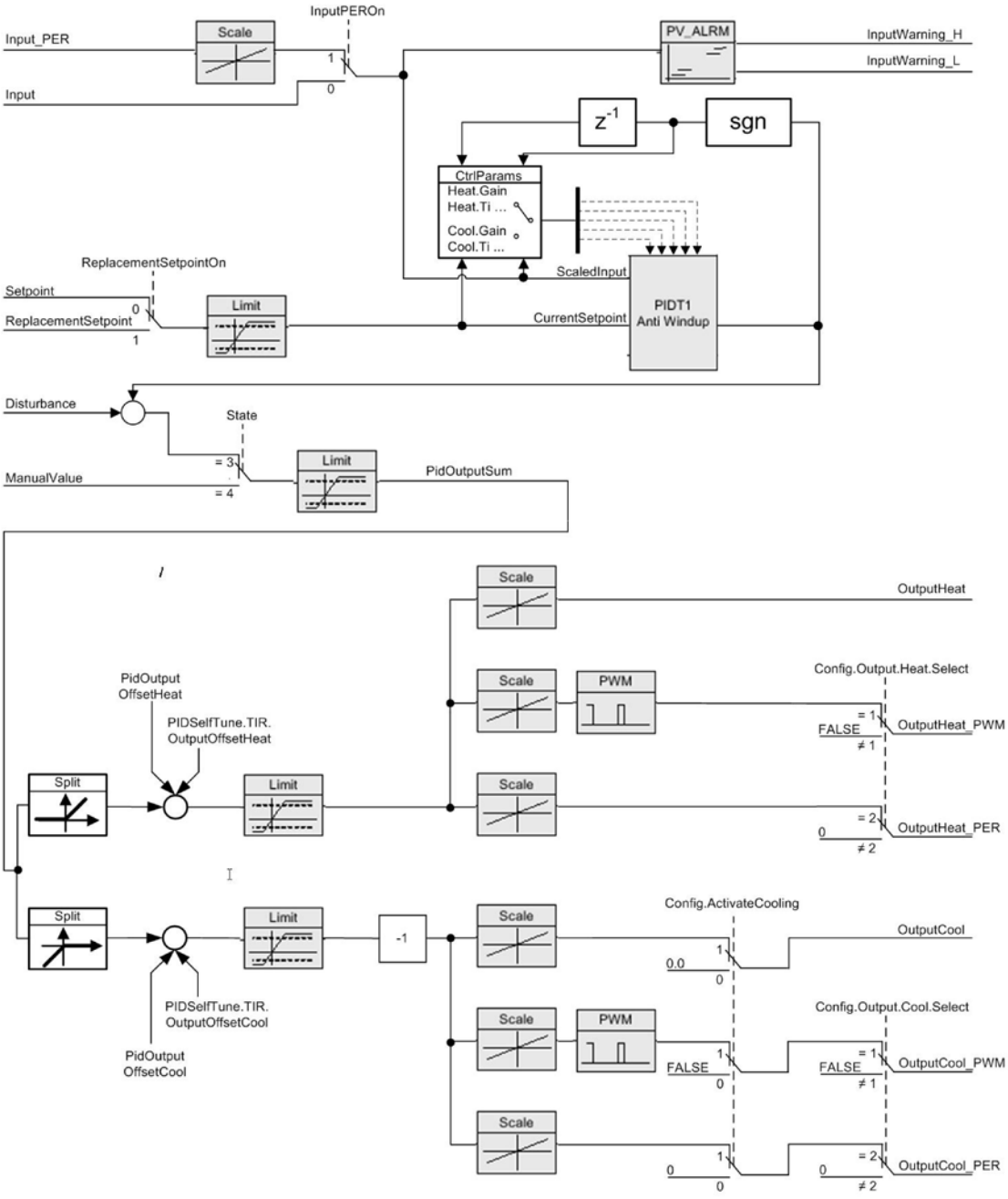


Figura 8-6 PID_Temp_Operation_Block_Diagram

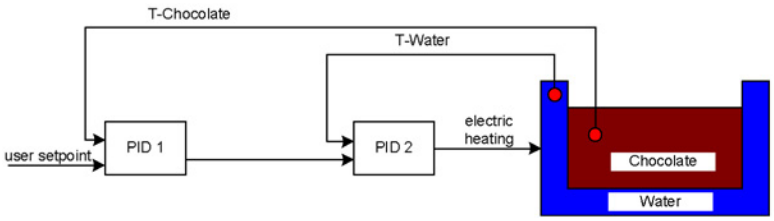


Figura 8-7 PID_Temp_Cascade_Operation_Block_Diagram

8.6.3 Reguladores en cascada

Los reguladores PID de temperatura se pueden ajustar en cascada para procesar más de una temperatura que dependa del mismo actuador.

Orden de llamada

Los reguladores PID en cascada se pueden llamar en el mismo ciclo de OB. Primero hay que llamar el maestro, a continuación el siguiente esclavo o los siguientes esclavos en el flujo de señal de regulación, y así hasta llamar finalmente el último esclavo de la cascada. La instrucción PID_Temp no comprueba automáticamente el orden de llamada.

Conexiones de comunicación

Si se ajustan reguladores en cascada, hay que conectar maestro y esclavo de forma que puedan compartir información entre sí. Hay que conectar el parámetro IN/OUT “Master” de un esclavo con el parámetro IN/OUT “Slave” de su maestro en el sentido de flujo de las señales.

Esto muestra una conexión de reguladores PID_Temp en cascada con dos subcascadas: "PID_Temp1" proporciona la consigna. La configuración conecta las salidas de "PID_Temp2", "PID_Temp3", "PID_Temp5", "PID_Temp6" y "PID_Temp8" al proceso:

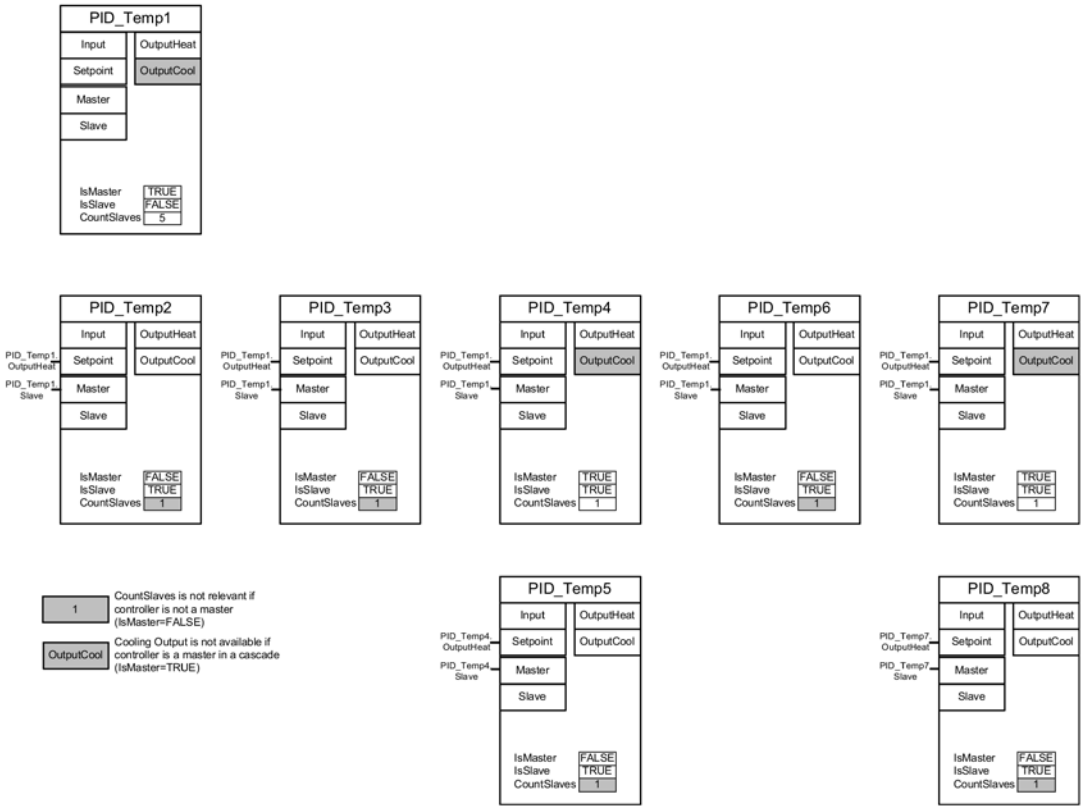


Figura 8-8 PID_Temp_Cascading_communication_connection

Consigna de sustitución

La instrucción PID_Temp proporciona una segunda entrada de consigna en el parámetro “ReplacementSetpoint”, que puede activarse ajustando el parámetro “ReplacementSetpointOn” = TRUE. Se puede utilizar “ReplacementSetpoint” como entrada de consigna durante la puesta en servicio o la optimización de un regulador esclavo sin necesidad de desconectar la conexión salida-a-consigna entre maestro y esclavo. La conexión es necesaria para el funcionamiento normal de la cascada.

De este modo, no es necesario cambiar el programa ni descargarlo si se desea desconectar temporalmente un esclavo de su maestro. Tan solo hay que activar "ReplacementSetpoint" y desactivarlo de nuevo al acabar. La consigna se hará efectiva para el algoritmo PID cuando pueda verse el valor en el parámetro “CurrentSetpoint”.

Autoajuste

El autoajuste de reguladores maestros en cascada debe cumplir los siguientes requisitos:

- Ser puesto en servicio desde su esclavo interior al primer maestro.
- Todos los esclavos del maestro deben estar en "Modo automático".
- La salida del maestro debe ser la consigna de los esclavos.

La instrucción PID_Temp proporciona la siguiente ayuda para el autoajuste en cascada:

- Si se inicia un autoajuste para un regulador maestro, el maestro comprueba si todos los esclavos se encuentran en “Modo automático” y si la función de consigna de sustitución está desactivada para todos los esclavos (“ReplacementSetpointOn” = FALSE). Si no se cumplen estas condiciones, el maestro no podrá autoajustarse. El maestro cancela el ajuste, pasa al modo “Inactivo” (si “ActivateRecoverMode” = FALSE) o regresa al modo guardado en el parámetro "Mode" (si “ActivateRecoverMode” = TRUE). El maestro muestra el mensaje de error 200000hex ("Error del maestro en la cascada. Los esclavos no están en modo automático o han activado la consigna sustitutiva e impiden la optimización del maestro".
- Cuando todos los esclavos están en “Modo automático", el sistema pone el parámetro "AllSlaveAutomaticState" = TRUE. Se puede aplicar este parámetro en el programa o localizar la causa del error 200000hex.
- Si el "ReplacementSetpoint" está desactivado para todos los esclavos, el sistema ajusta el parámetro "NoSlaveReplacementSetpoint" = TRUE. Se puede aplicar este parámetro en el programa o localizar la causa del error 200000hex.

Si se utiliza el cuadro de diálogo de puesta en servicio de la instrucción PID_Temp, se obtendrá más ayuda para el ajuste en cascada (Página 250).

Modos de operación y tratamiento de errores

El regulador PID_Temp no permite la conmutación del modo de operación en su maestro o sus esclavos. Así, el maestro de una cascada permanece en su modo actual cuando un esclavo genera un error. Esto supone una ventaja si dos más esclavos paralelos funcionan con el regulador maestro en cuestión; un error en una cadena no desconecta la cadena paralela.

De forma similar, un esclavo de la cascada permanece en su modo de operación actual aunque su maestro genere un error. Sin embargo, en ese caso la operación posterior del esclavo dependerá de la configuración del maestro, ya que la consigna del esclavo es la salida del maestro. Así, si se configura el maestro con "ActivateRecoverMode" = TRUE y se produce un error, el maestro depositará en la salida el último valor valido o un valor de salida sustitutivo como consigna para el esclavo. Si el maestro se configura con "ActivateRecoverMode" = FALSE, el maestro conmutará a “Modo inactivo" y pondrá todas las salidas a "0.0", de modo que el esclavo utilizará "0.0" como su consigna.

Dado que solo los reguladores esclavos tienen acceso directo a los actuadores y que permanecen en su estado de operación actual en caso de error del maestro, se evitan daños en el proceso. Por ejemplo, para dispositivos de transformación de plásticos resulta fatal para los esclavos dejar de funcionar, desconectar los actuadores y permitir que el plástico se endurezca lentamente dentro del dispositivo porque el maestro ha generado un error.

Anti-windup

Un esclavo en cascada recibe su consigna de la salida de su maestro. Si el esclavo alcanza sus propios límites de salida mientras el maestro todavía ve un error de regulación (consigna – entrada), el maestro congela o reduce su contribución a la integración para impedir el efecto llamado "WindUp". En caso de "WindUp", el maestro incrementa su contribución a la integración a un valor muy elevado y primero debe reducirlo antes de que el regulador pueda volver a tener una reacción normal. Un efecto "WindUp" de este tipo repercute negativamente en la dinámica de la regulación. PID_Temp ofrece maneras de prevenir este efecto en una cascada configurando el parámetro “Config.Cascade.AntiWindUpMode” del regulador maestro:

Valor	Descripción
0	Desactiva la función Anti-Windup.
1	Reduce la contribución a la integración del regulador maestro en la relación entre "esclavos en limitación" y "esclavos existentes" (parámetro “CountSlaves”).
2	Congela la contribución a la integración del maestro en cuanto un esclavo alcanza su limitación. Solo es relevante si "Config.Cascade.IsMaster" = TRUE.

8.7 Parámetros ErrorBit de la instrucción PID_Temp

Si el regulador PID tiene diferentes advertencias pendientes, mostrará los valores de los códigos de error mediante suma binaria. La indicación del código de error 0003, por ejemplo, indica que están pendientes los errores 0001 y 0002.

Tabla 8- 11 Parámetros ErrorBit de la instrucción PID_Temp

ErrorBit (DW#16#...)	Descripción
0000	No hay error
0001 [1, 2]	El parámetro Input se encuentra fuera de los límites del valor de proceso. Input > Config.InputUpperLimit Input < Config.InputLowerLimit
0002 [2, 3]	Valor no válido en el parámetro Input_PER. Compruebe si hay un error pendiente en la entrada analógica.
0004 [4]	Error durante la optimización fina. No ha sido posible mantener la oscilación del valor de proceso.
0008 [4]	Error al iniciar la optimización inicial. El valor de proceso está demasiado próximo a la consigna. Inicie la optimización fina.
0010 [4]	La consigna se ha cambiado durante la optimización. Nota: se puede ajustar la fluctuación permitida en la consigna en la variable CancelTuningLevel.
0020	La optimización inicial no está permitida durante la optimización fina. Nota: si ActivateRecoverMode era = TRUE antes de que se produjera el error, PID_Temp permanece en el modo de optimización fina.
0040 [4]	Error durante la optimización inicial. El enfriamiento no ha podido reducir el valor del proceso.
0080 [4]	Error durante la optimización inicial. Configuración incorrecta de los límites del valor de salida. Compruebe si los límites del valor de salida están configurados correctamente y coinciden con la lógica de control.
0100 [4]	Un error durante la optimización fina ha provocado parámetros no válidos.
0200 [2, 3]	Valor no válido en el parámetro Input: el valor tiene un formato de número no válido.
0400 [2, 3]	Fallo en el cálculo del valor de salida. Compruebe los parámetros PID.
0800 [1, 2]	Error de tiempo de muestreo: PID_Temp no se llama durante el tiempo de muestreo del OB de alarma cíclica.
1000 [2, 3]	Valor no válido en el parámetro Setpoint: el valor tiene un formato de número no válido.
10000	Valor no válido en el parámetro ManualValue: el valor tiene un formato de número no válido. Nota: si ActivateRecoverMode era = TRUE antes de que se produjera el error, PID_Temp utiliza SubstituteOutput como valor de salida. En cuanto se asigna un valor válido en el parámetro ManualValue, PID_Temp lo utiliza como valor de salida.

ErrorBit (DW#16#...)	Descripción
20000	Valor no válido en la variable SubstituteValue: el valor tiene un formato de número no válido. PID_Temp utiliza el límite inferior del valor de salida como valor de salida. Nota: si el modo automático estaba activo antes de que se produjera el error, ActivateRecoverMode = TRUE y el error ya no está pendiente, PID_Temp vuelve al modo automático.
40000	Valor no válido en el parámetro Disturbance: el valor tiene un formato de número no válido. Nota: si el modo automático estaba activo y ActivateRecoverMode = FALSE antes de que se produjera el error, Disturbance se pone a cero. PID_Temp permanece en modo automático. Nota: si el modo de optimización inicial o fina estaba activo y ActivateRecoverMode = TRUE antes de que se produjera el error, PID_Temp cambia al estado operativo guardado en el parámetro Mode. Si la perturbación en la fase actual no afecta al valor de salida, la optimización no se cancela.
200000	Error con maestro en la cascada. Los esclavos no están en modo automático o tienen habilitada una consigna de sustitución e impiden el ajuste del maestro.
400000	El regulador PID no permite la optimización inicial para calentamiento mientras está activo el enfriamiento.
800000	El valor del proceso debe estar cercano a la consigna para iniciar la optimización inicial para el enfriamiento.
1000000	Error al iniciar la optimización. “Heat.EnableTuning” y “Cool.EnableTuning” no están ajustados o no coinciden con la configuración.
2000000	La optimización inicial para el enfriamiento requiere una optimización inicial correcta para el calentamiento.
4000000	Error al iniciar la optimización fina. “Heat.EnableTuning” y “Cool.EnableTuning” no pueden ajustarse al mismo tiempo.
8000000	Un error durante el cálculo del parámetro PID ha resultado en parámetros no válidos (por ejemplo, ganancia negativa; los parámetros PID actuales permanecen invariables y la optimización no tiene efecto).

[1] Nota: si el modo automático estaba activo antes de que se produjera el error y ActivateRecoverMode = TRUE, PID_Temp permanece en modo automático.

[2] Nota: si el modo de optimización inicial o fina estaba activo antes de que se produjera el error y ActivateRecoverMode = TRUE, PID_Temp cambia al estado operativo guardado en el parámetro Mode.

[3] Nota: si el modo automático estaba activo antes de que se produjera el error y ActivateRecoverMode = TRUE, PID_Compact emite el valor de salida sustitutivo configurado. En cuanto el error deja de estar pendiente, PID_Temp regresa al modo automático.

[4] Nota: si ActivateRecoverMode era = TRUE antes de que se produjera el error, PID_Temp cancela la optimización y cambia al estado operativo guardado en el parámetro Mode.

8.8 Configuración de los reguladores PID_Compact y PID_3Step

Los parámetros del objeto tecnológico determinan el funcionamiento del regulador PID. Utilice el icono para abrir el editor de configuración.

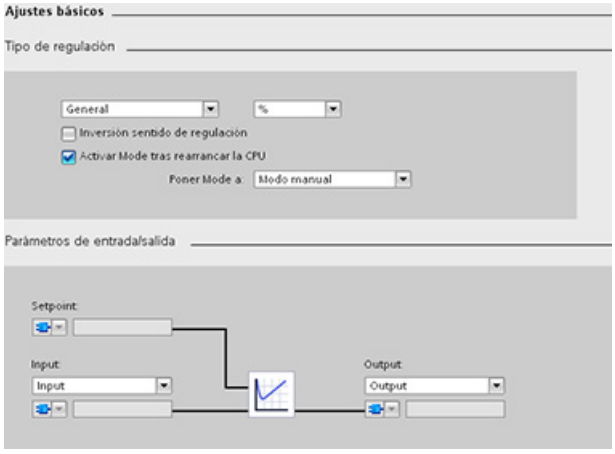


Tabla 8- 12 Ejemplo de ajustes de configuración de la instrucción PID_Compact

Ajustes		Descripción
Básicos	Tipo de regulador	Selecciona las unidades de ingeniería.
	Invertir la lógica de control	Permite seleccionar un lazo PID de acción inversa. • 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).	

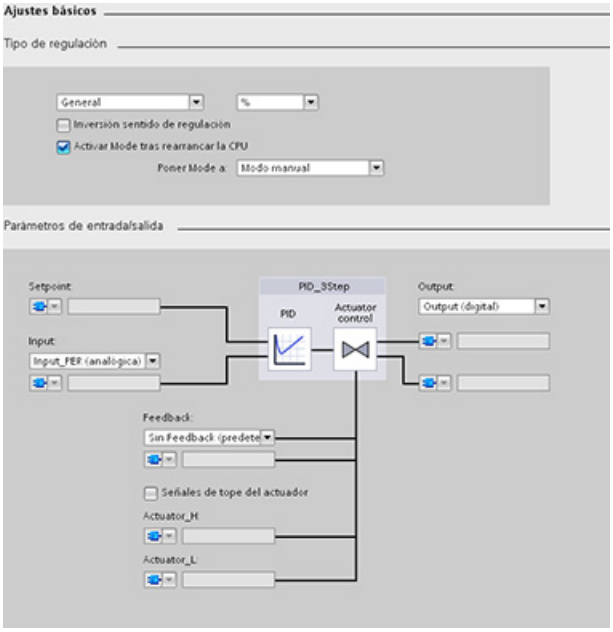


Tabla 8- 13 Ejemplo de ajustes de configuración de la instrucción PID_3Step

Configuración		Descripción
Básicos	Tipo de regulador	Selecciona las unidades de ingeniería.
	Invertir la lógica de control	Permite seleccionar un lazo PID de acción inversa. • 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.
	Activar modo tras rearranque de 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. Poner modo a: define el modo al que el usuario desea que pase el PID tras un rearranque.
	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: • Sin realimentación (predeterminado) • Realimentación • Feedback_PER
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).	

Configuración		Descripción
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 el faceplate 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 el faceplate 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 el faceplate de la válvula.)
	Reacción a 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 sustitutiva, 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 solo entre 0% (off) y 100% (on).
	Escalar realimentación de posición[1]	• "Posición límite superior" y "Posición límite inferior" definen la posición positiva máxima (completamente abierta) y la posición negativa máxima (completamente cerrada). La "Posición límite superior" debe ser mayor que la "Posición límite inferior". • "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".
Avanzada	Vigilancia del valor de proceso	Ajusta los límites superior e inferior de advertencia para el valor de proceso.
	Parámetros PID	Si el usuario lo desea, puede introducir sus propios parámetros de optimización PID en esta ventana. Para que eso sea posible, la casilla de verificación "Permitir entrada manual" debe estar activada.

[1] "Escalar realimentación de posición" solo es editable si se ha activado "Realimentación" en los ajustes "básicos".

8.9 Configuración del regulador PID_Temp

Los parámetros del objeto tecnológico determinan el funcionamiento del regulador PID. Utilice el icono para abrir el editor de configuración.

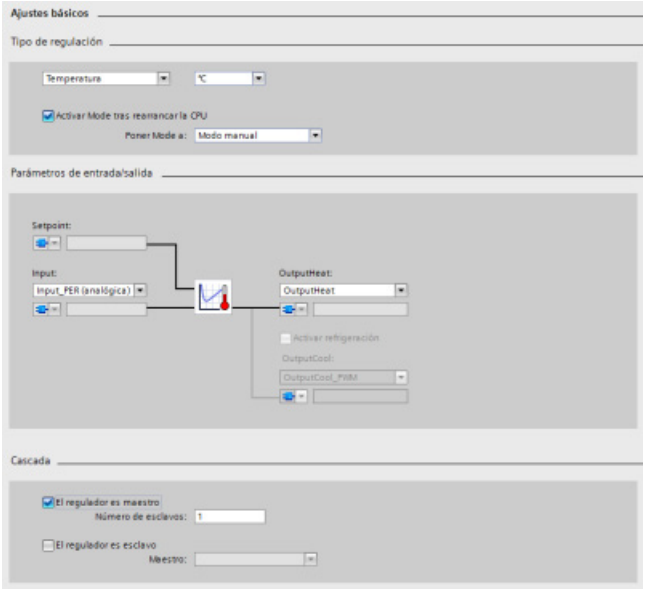


Tabla 8- 14 Ejemplo de ajustes de configuración de la instrucción PID_Temp

Configuración		Descripción
Básicos	Tipo de regulador	Selecciona las unidades de ingeniería.
	Activar modo tras rearranque de 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. Poner modo a: define el modo al que el usuario desea que pase el PID tras un rearranque.
	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 Calen-tamiento	Selecciona si deben utilizarse las salidas digitales (OutputHeat y OutputHeat_PWM) o la salida analógica (OutputHeat_PER (analog)) para el valor de salida.
	Salida Enfri-amiento	Selecciona si deben utilizarse las salidas digitales (OutputCool y OutputCool_PWM) o la salida analógica (OutputCool_PER (analog)) para el valor de salida.
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).	
Cascada	El regulador es maestro	Ajusta el regulador como maestro y selecciona el número de escla-vos.
	El regulador es esclavo	Ajusta el regulador como esclavo y selecciona el número de maes-tros.

Tipo de regulador

Ajuste	Parámetro TO-DB	Tipo de datos	Rango de valores	Descripción
Cantidad física	"PhysicalQuantity"	Int (Enum)	• General • Temperatura (=predeterminada)	Preselección de valor de unidad física Sin regulación multivalor y no editable en modo online de la vista funcional.
Unidad de medida	"PhysicalUnit	Int (Enum)	• General: Unidades = % • Temperatura: Unidades (selecciones posibles) = – °C (=predeterminada) – °F – K	La selección de unidad del usuario se vuelve a poner a “0” al cambiar la cantidad física.
Activar modo tras rearranque de la CPU	"RunModeByStartup"	Bool	Casilla de verificación	Si se ajusta TRUE (=predeterminada), el regulador cambia al estado guardado en la variable “Mode” tras Power on - off - on tras pasar el PLC de STOP a RUN. De lo contrario, PID-Temp permanece en modo "Inactivo".
Poner modo a	"Mode"	Int (Enum)	Modos (selecciones posibles): • 0: Inactivo • 1: Optimización inicial • 2: Optimización fina • 3: Modo automático • 4: Modo manual (=predeterminada)	La estación de ingeniería (ES) ajusta el valor de arranque de la variable “Mode” conforme a la selección del usuario. El valor predeterminado de Mode (guardado en TO-DB) es Modo manual.

Parámetros de entrada/salida

Ajuste	Parámetro TO-DB	Tipo de datos	Rango de valores	Descripción
Consigna	Consigna	Real)	Real	Solo accesible en la página de propiedades. Sin regulación multivalor en modo online de la vista funcional.
Entrada de selección	"Config.InputPerOn"	Bool (Enum)	Bool	Selecciona el tipo de entrada que se va a utilizar. Selecciones posibles: • FALSE: “Input” (Real) • TRUE: “Input_PER (analog)”
Entrada	Input o Input_PER	Real o Int	Real o Int	Solo accesible en la página de propiedades. Sin regulación multivalor en modo online de la vista funcional.
Salida de selección (calentamiento)	"Config.Output.Heat.Select"	Int (Enum)	2 >= Config.Output.Heat.Select >= 0	Selecciona el tipo de salida que se va a utilizar para el calentamiento. Selecciones posibles: • “OutputHeat” (Real) • “OutputHeat_PWM” (Bool) (=predeterminada) • “OutputHeat_PER (analog)” (Word) Se pone una vez a “OutputHeat” si el usuario activa la casilla de verificación “Este regulador es maestro” en la sección “Cascada”.
Salida (calentamiento)	OutputHeat, OutputHeat_PER o OutputHeat_PWM	Real o Int o Bool	Real, Int o Bool	Solo accesible en la página de propiedades. Sin regulación multivalor en modo online de la vista funcional.

Ajuste	Parámetro TO-DB	Tipo de datos	Rango de valores	Descripción
Activar salida (enfri-amiento)	"Config.ActivateCooling"	Bool	Bool	Si se activa esta casilla de verificación: • Se pone "Con-fig.Output. Heat.PidLowerLimit = 0.0 una vez. • Se pone el parámetro "Con-fig.ActivateCooling" a TRUE en lugar de FALSE si no está se-leccionada (=predeterminada). • Se activan todos los controles “Salida (en-friamiento)” (en “Ajustes básicos” y otras vistas). • Se cambia la línea del símbolo PID a los controles de gris a negro. • La casilla de verifica-ción “Este regulador es maestro” en la sección “Cascada” está desactivada. Nota: Solo disponible si el regulador no se confi-gura como maestro de la cascada (la casilla de verificación “Este regu-lador es maestro” en la sección “Cascada” está desactivada; "Con-fig.Cascade.IsMaster" = FALSE).

Ajuste	Parámetro TO-DB	Tipo de datos	Rango de valores	Descripción
Salida de selección (enfriamiento)	"Config.Output.Cool.Select"	Int (Enum)	2 >= Config.Output.Heat.Select >= 0	Selecciona el tipo de salida que se va a utilizar para el enfriamiento. Selecciones posibles: • “OutputCool” (Real) • “OutputCool_PWM” (Bool) (=predeterminada) • “OutputCool_PER (analog)” (Word) Solo disponible si se activa “Activar salida (enfriamiento)”; (Config.ActivateCooling = TRUE).
Salida (enfriamiento)	OutputCool, OutputCool_PER o OutputCool_PWM	Real o Int o Bool	Real, Int o Bool	Solo accesible en la página de propiedades. Sin regulación multivalor en modo online de la vista funcional.

Parámetros de cascada

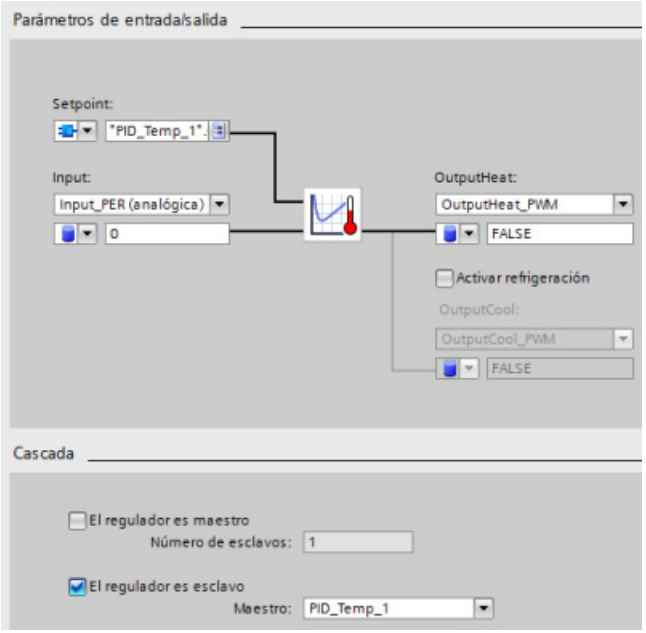
Los siguientes parámetros permiten seleccionar reguladores como maestros o esclavos y determinar el número de reguladores esclavos que reciben la consigna directamente del regulador maestro:

Ajuste	Parámetro TO-DB	Tipo de datos	Rango de valores	Descripción
Este regulador es maestro	"Config.Cascade.IsMaster"	Bool	Bool	Indica si este regulador es maestro en la cascada. Si se marca esta casilla de verificación, ocurre lo siguiente: • El parámetro "Config.Cascade.IsMaster" se pone a TRUE, en lugar de FALSE si no está seleccionada (=predeterminada). • “Salida de selección (calentamiento)” en la sección “Parámetros de entrada/salida” se pone una vez a “Output-Heat” (Config.Output.Heat.Select = 0). • Se habilita el campo de entrada “Número de esclavos”. • Se deshabilita la casilla de verificación “Activar salida (enfriamiento)” en la sección “Parámetros de entrada/salida”. Nota: Solo disponible si la salida de enfriamiento de este regulador está desactivada (casilla de verificación “Activar salida (enfriamiento)” en la sección “Parámetros de entrada/salida” desactivada (Config.ActivateCooling = FALSE).

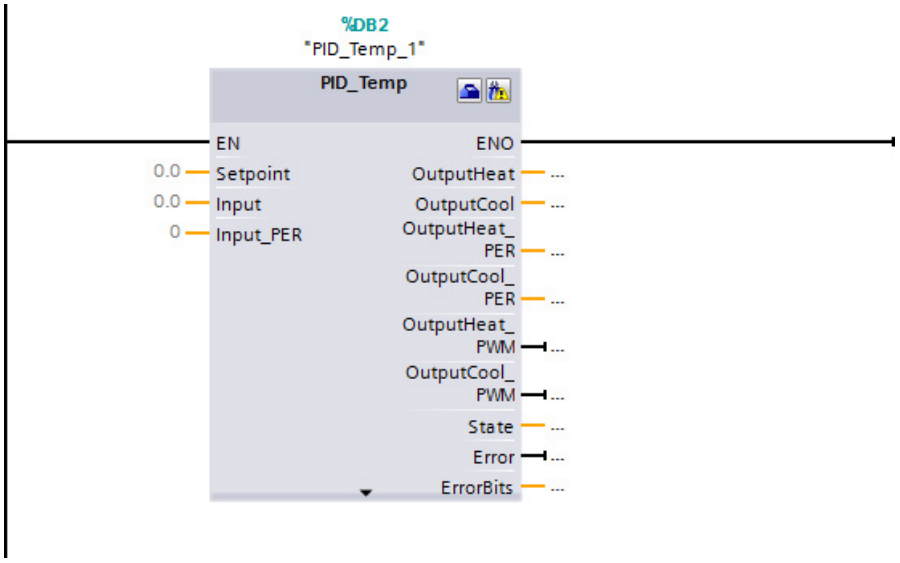
Ajuste	Parámetro TO-DB	Tipo de datos	Rango de valores	Descripción
Número de esclavos	"Config.Cascade.CountSlaves"	Int	255 >= Config.Cascade. CountSlaves >= 1	Número de reguladores esclavos que obtienen la consigna directamente de este regulador maestro. La instrucción PID_Temp procesa este valor, junto con otros, para el tratamiento anti-windup. "Número de esclavos" solo está disponible si la casilla de verificación “Este regulador es maestro” está activada (Config.Cascade.IsMaster = TRUE).
Este regulador es esclavo	"Config.Cascade.IsSlave"	Bool	Bool	Indica si este regulador es esclavo en la cascada. Si se activa esta casilla de verificación, el parámetro "Config.Cascade.IsSlave" se pone a TRUE, en lugar de FALSE si no está seleccionada (=predeterminada). Active esta casilla de verificación en la página de propiedades para habilitar la lista desplegable “SelectionMaster”.

Ejemplo: Reguladores en cascada

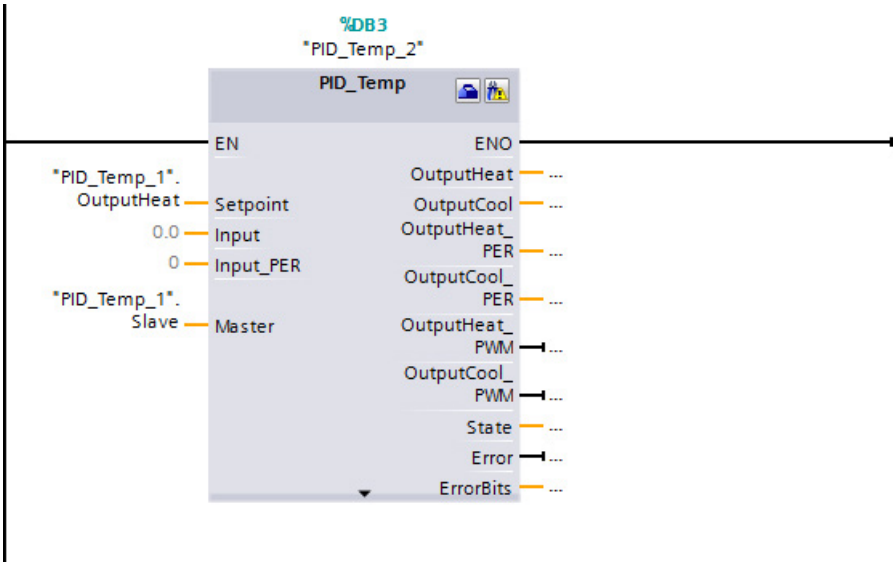
En el cuadro de diálogo “Ajustes básicos” que aparece a continuación se pueden ver las secciones “Parámetros de entrada/salida” y “Cascada” para el regulador esclavo "PID_Temp_2" tras seleccionar "PID_Temp_1" como maestro. Se realiza la conexión entre los reguladores maestro y esclavo:



Segmento 1: En estos segmentos se establece la conexión entre el maestro "PID_Temp_1" y el esclavo "PID_Temp_2" en el editor de programación:



Segmento 2: Se establece la conexión entre los parámetros “OutputHeat” y "slave" del maestro "PID_Temp_1" y los parámetros “Setpoint” y "Master" del esclavo "PID_Temp_2" respectivamente:



Autoajuste de procesos de temperatura

La instrucción PID_Temp proporciona dos modos de autoajuste:

- “Optimización inicial” (parámetro “Mode” = 1)
- “Optimización fina” (parámetro “Mode” = 2)

Dependiendo de la configuración del regulador, habrá disponibles diferentes variantes de estos métodos de optimización:

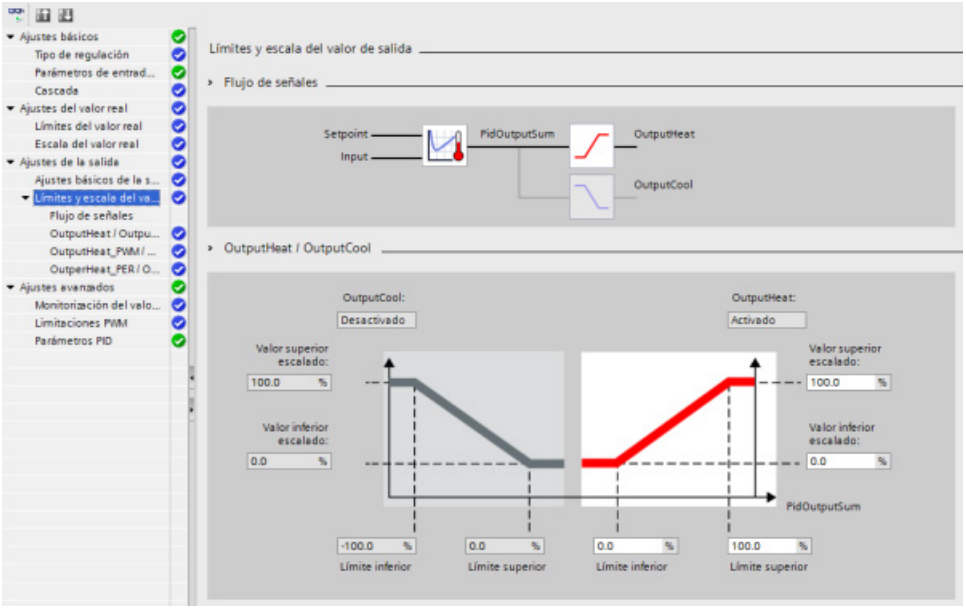
Configuración	Regulador con salida de calentamiento	Regulador con salida de calentamiento y enfriamiento usando factor de enfriamiento	Regulador con salida de calentamiento y enfriamiento usando dos juegos de parámetros PID
Valores TO-DB asociados	• Config.ActivateCooling = FALSE • Con-fig.AdvancedCooling = irrelevante	• Config.ActivateCooling = TRUE • Con-fig.AdvancedCooling = FALSE	• Config.ActivateCooling = TRUE • Con-fig.AdvancedCooling = TRUE
Métodos de optimización disponibles	• “Optimización inicial calefacción” • “Optimización fina calefacción" (no puede usarse off-set de enfriamiento)	• “Optimización inicial calefacción” • “Optimización fina calefacción" (puede usarse offset de enfriamiento)	• “Optimización inicial calefacción y refrigera-ción” • “Optimización inicial calefacción” • “Optimización inicial refrigeración” • “Optimización fina calefacción" (puede usarse offset de enfriamiento) • “Optimización fina refrigeración" (puede usarse offset de calentamiento)

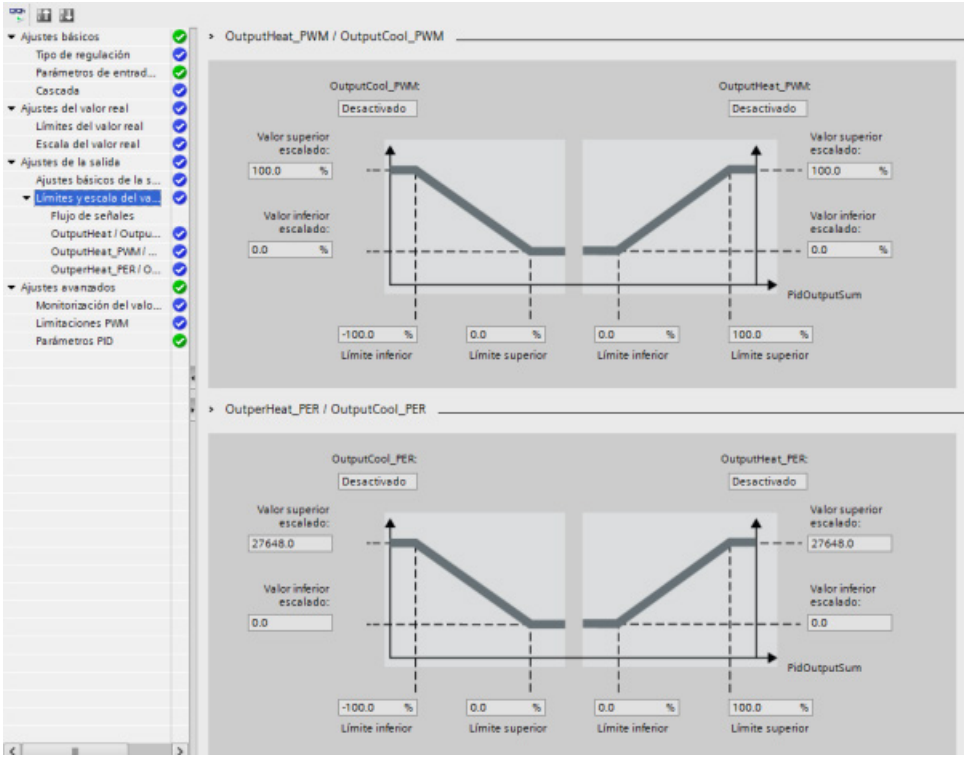
Límites y escala del valor de salida

Activación de enfriamiento deshabilitada

Si configura la instrucción PID_Temp como maestro para una cascada, la casilla de verificación “Activar salida (refrigeración)” de la vista “Ajustes básicos” estará desactivada y deshabilitada, todos los ajustes de la vista “Ajustes de la salida” que dependan de la activación de refrigeración también estarán deshabilitados.

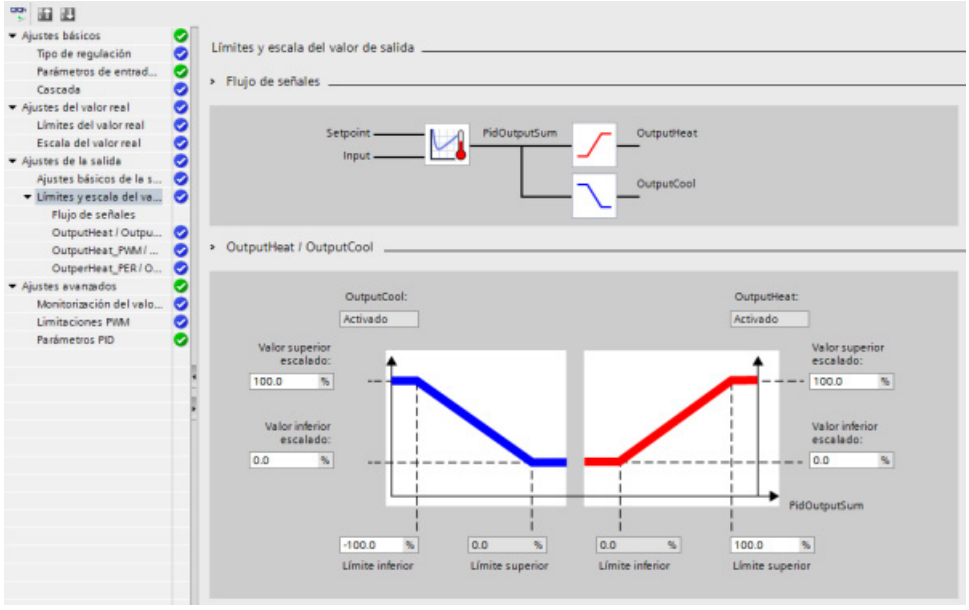
La siguiente figura muestra la sección "Límites y escala del valor de salida” en la vista “Ajustes de la salida” con enfriamiento desactivado (OutputHeat_PWM seleccionado en la vista “Parámetros de entrada/salida” y OutputHeat siempre habilitado):

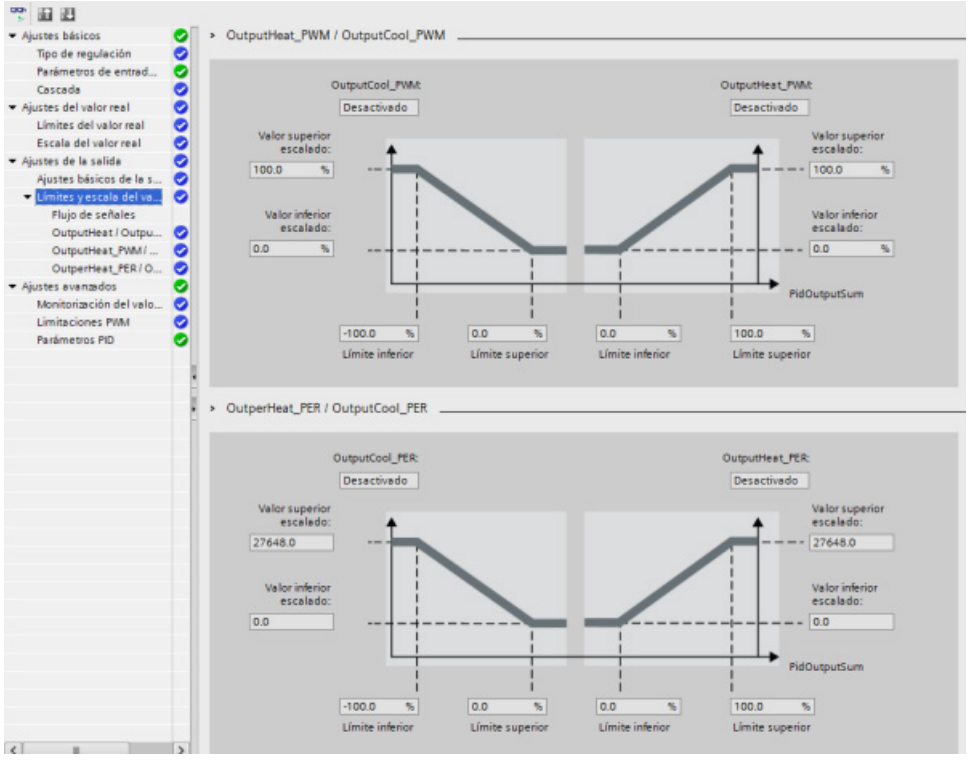




Activación de enfriamiento habilitada

La siguiente figura muestra la sección "Límites de valor de salida y escalado” en la vista “Ajustes de la salida” con enfriamiento activado (OutputCool_PER y OutputHeat_PWM seleccionados en la vista “Parámetros de entrada/salida”; OutputCool y OutputHeat siempre habilitados):





Modos de operación

Para cambiar el modo de operación manualmente, el usuario tiene que ajustar el parámetro de entrada/salida “Mode” del regulador y activarlo cambiando “ModeActivate” de FALSE a TRUE (disparo por flanco ascendente). Es necesario resetear “ModeActivate” antes de volver a cambiar el modo; no se resetea automáticamente.

El parámetro de salida “State” muestra el modo de operación actual y, si es posible, se pone en el modo (“Mode”) solicitado. El parámetro “State” no puede cambiarse directamente; solo se cambia con el parámetro “Mode” o con cambios de modo de operación automáticos del regulador.

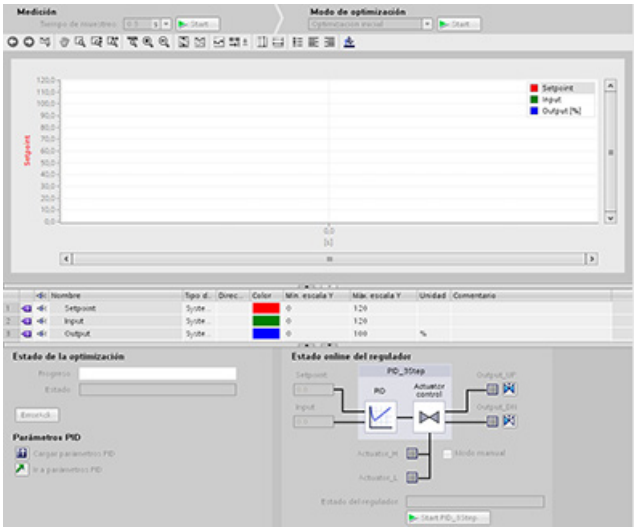
"Mode" / "State"	Nombre	Descripción
0	Inactivo	La instrucción PID_Temp: • Desactiva el algoritmo PID y la modulación por ancho de impulso • Pone a "0" (FALSE) todas las salidas de regulador (OutputHeat, OutputCool, OutputHeat_PWM,OutputCool_PWM, OutputHeat_PER, OutputCool_PER), independientemente de los límites de salida u offsets configurados. Este modo se puede alcanzar poniendo “Mode” = 0, “Reset” = TRUE o en caso de error.
1	Optimización inicial (ajuste de arranque / SUT)	Este modo determina los parámetros durante el primer arranque del regulador. Al contrario que en el caso de PID_Compact, para PID_Temp es necesario seleccionar si es necesaria una optimización de calentamiento, una optimización de enfriamiento o ambas con los parámetros “Heat.EnableTuning” y “Cool.EnableTuning”. "Optimización inicial" se puede activar desde los modos Inactivo, Automático o Manual. Si la optimización es correcta, PID_TEMP cambia al modo automático. Si la optimización no es correcta, la conmutación del modo de operación dependerá de “ActivateRecoverMode”.
2	Optimización fina (optimización en RUN / TIR)	Este modo determina la parametrización óptima del regulador PID en la consigna. Al contrario que en el caso de PID_Compact, para PID_Temp es necesario seleccionar si es necesaria una optimización de calentamiento o una optimización de enfriamiento con los parámetros “Heat.EnableTuning” y “Cool.EnableTuning”. "Optimización fina" se puede activar desde los modos Inactivo, Automático o Manual. Si la optimización es correcta, PID_TEMP cambia al modo automático. Si la optimización no es correcta, la conmutación del modo de operación dependerá de “ActivateRecoverMode”.
3	Modo automático	En el modo Automático (el modo estándar de la regulación PID), el resultado del algoritmo PID determina los valores de salida. PID_Temp cambia a inactivo si ocurre un error y “ActivateRecoverMode” = FALSE. Si se produce un error y “ActivateRecoverMode” = TRUE, la conmutación del modo de operación dependerá del error. Consulte la instrucción PID_Temp Parámetros ErrorBit (Página 229) para más información.

"Mode" / "State"	Nombre	Descripción
4	Modo manual	Es este modo, el regulador PID escala, limita y transfiere el valor del parámetro “ManualValue” a las salidas. El regulador PID asigna “ManualValue” en el escalado del algoritmo PID (igual que “PidOutputSum”), de ese modo su valor decide si es efectivo en las salidas de calentamiento o enfriamiento. Este modo se puede alcanzar poniendo “Mode” = 4 o “ManualEnable” = TRUE.
5	Valor de salida sustitutivo con vigilancia de error (modo de recuperación)	Este modo se activa ajustando "Mode" = 5. El modo es una reacción automática al error del regulador si el modo automático está activo en el momento en que se produce el error: • SetSubstituteOutput = FALSE (último valor de salida válido) • SetSubstituteOutput = TRUE (valor guardado en el parámetro "SubstituteOutput") Si PID_Temp está en “Modo automático” y el parámetro “ActivateRecoverMode” = TRUE, PID_Temp cambia a ese modo si se producen los siguientes errores: • "Valor no válido en el parámetro "Input_PER". Compruebe si hay un error en la entrada analógica (por ejemplo, rotura de hilo)." (ErrorBits = DW#16#0002) • "Valor no válido en el parámetro "Input". El valor no es un número.” (ErrorBits = DW#16#0200) • "Error en el cálculo del valor de salida. Compruebe los parámetros PID." (ErrorBits = DW#16#0400) • "Valor no válido en el parámetro "Setpoint". El valor no es un número.” (ErrorBits = DW#16#1000) Si el error ya no está pendiente, PID_Temp regresará automáticamente al modo automático.

8.10 Puesta en servicio de los reguladores PID_Compact y PID_3Step

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 8- 15 Ejemplo de pantalla de puesta en servicio (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.

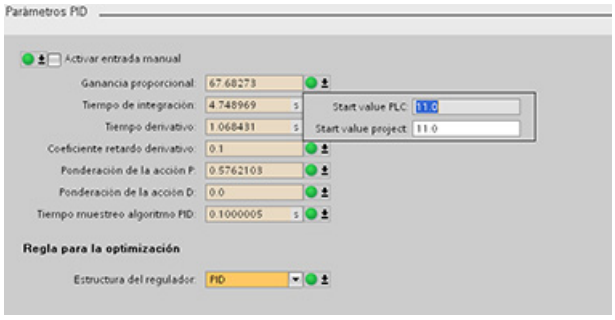
Control del valor de arranque PID

Los valores reales de los parámetros de configuración PID se pueden editar para que el comportamiento del regulador PID pueda optimizarse en modo online.

Abra los "Objetos tecnológicos" del regulador PID y su objeto "Configuración". Para acceder al control del valor de arranque, haga clic en el icono "monóculo" de la esquina superior izquierda del cuadro de diálogo:

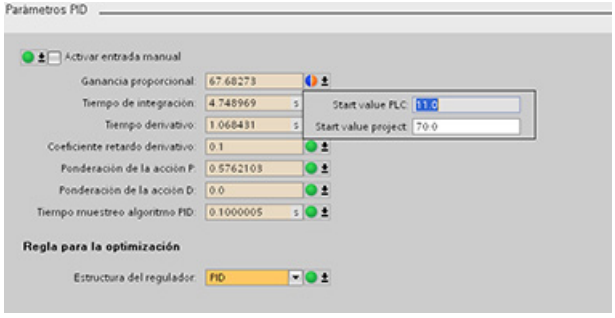
Ahora puede cambiar el valor de cualquier parámetro de configuración del regulador PID como muestra la figura inferior.

Se puede comparar el valor actual con el valor de arranque del proyecto (offline) y el valor de arranque del PLC (online) de cada parámetro. Esto es necesario para detectar las diferencias online/offline del bloque de datos del objeto tecnológico (TO-DB) y para estar informado sobre los valores que se utilizarán como valores actuales en el siguiente cambio de STOP a ARRANQUE del PLC. Además, un icono de comparación permite identificar fácilmente las diferencias online/offline:



La figura inferior muestra la pantalla de parámetros PID con iconos de comparación que muestran qué valores difieren en los proyectos online y offline. Un icono verde indica que los valores son los mismos; un icono azul/naranja indica que los valores difieren.

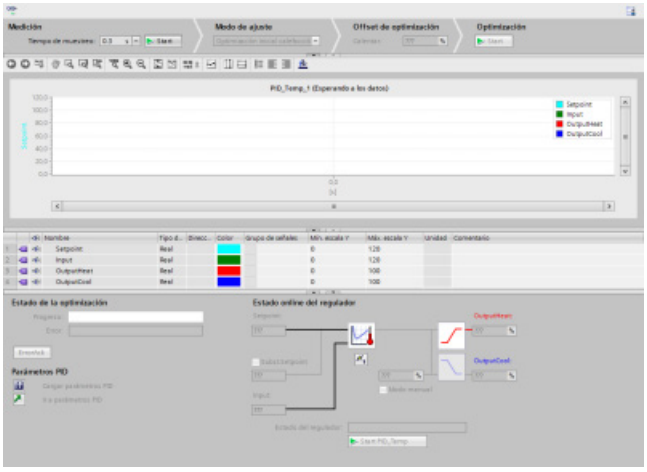
Además, haga clic en el botón de parámetros con la flecha hacia abajo y abra la pequeña ventana que muestra el valor de arranque del proyecto (offline) y el valor de arranque del PLC (online) para cada parámetro:



8.11 Puesta en servicio del regulador PID_Temp

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 8- 16 Ejemplo de pantalla de puesta en servicio (PID_Temp)

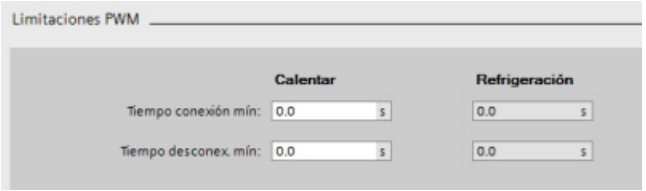
	Medición: para visualizar la consigna, el valor de proceso (valor de entrada) y el valor de salida en una curva de tiempo real, introduzca el tiempo de muestreo y haga clic en el botón "Iniciar". Modo de optimización: para optimizar el lazo PID_Temp, seleccione "Optimización inicial" u "Optimización 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". Entonces el modo PID se pone en modo "inactivo". El estado indica el error.

Límites PWM

Los actuadores controlados con la función de software PWM del PID_Temp pueden requerir protección frente a duraciones de impulso demasiado cortas (por ejemplo, un relé tiristor tiene que activarse durante más de 20 ms antes de que pueda reaccionar); para ello se asigna un mínimo de tiempo. El actuador también puede obviar impulsos breves y reducir así la calidad de la regulación. Puede ser necesario un tiempo mínimo de desconexión (por ejemplo, para prevenir sobrecalentamientos).

Para visualizar la vista de límites PWM hay que abrir la vista funcional en la configuración de objetos tecnológicos (TO) y seleccionar “Límites PWM” en el nodo “Ajustes avanzados” del árbol de navegación.

Si se abre la vista “Límites PWM” en la vista funcional y se activa la vigilancia (botón “gafas”), todos los reguladores mostrarán el valor de vigilancia online de TO-DB con fondo naranja y regulación multivalor, y los valores podrán editarse (si se cumplen las condiciones de configuración; consulte la tabla siguiente).



Ajuste	Parámetro TO-DB	Tipo de datos	Rango de valores	Descripción
Tiempo conexión mín (calefacción) [1,2]	"Config.Output.Heat. MinimumOnTime"	Real	100000.0 >= "Config.Output. Heat. MinimumOnTime >= 0.0	Un impulso en OutputHeat_PWM" nunca será inferior a este valor.
Tiempo desconexión mínimo (calefacción) [1,2]	"Config.Output.Heat. MinimumOffTime"	Real	100000.0 >= "Config.Output. Heat. MinimumOffTime >= 0.0	Una rotura en Output-Heat_PWM nunca será inferior a este valor.
Tiempo conexión mín (refrigeración) [1,3,4]	"Config.Output.Cool. MinimumOnTime"	Real	100000.0 >= Config.Output. Cool. MinimumOnTime >= 0.0	Un impulso en OutputCool_PWM" nunca será inferior a este valor.
Tiempo desconexión mínimo (refrigeración) [1,3,4]	"Config.Output.Cool. MinimumOffTime"	Real	100000.0 >= Config.Output. Cool. MinimumOffTime >= 0.0	Una rotura en Output-Cool_PWM nunca será inferior a este valor.

1 El campo muestra "s" (segundos) como unidad de tiempo.

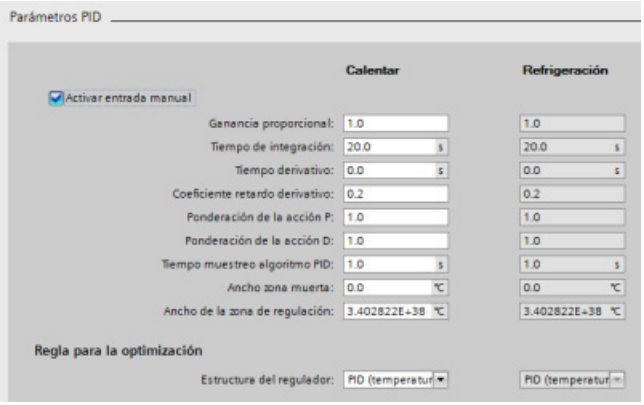
2 Si la salida de selección (calefacción) en la vista “Ajustes básicos” no es “OutputHeat_PWM” (Config.Output.Heat.Select = TRUE), conviene poner este valor a "0.0".

3 Si la salida de selección (refrigeración) en la vista “Ajustes básicos” no es “OutputCool_PWM” (Config.Output.Cool.Select = TRUE), conviene poner este valor a "0.0".

4 Solo disponible si se activa “Activar salida (refrigeración)” en la vista “Ajustes básicos” (Config.ActivateCooling = TRUE).

Parámetros PID

La vista “Ajustes avanzados”, sección "Parámetros PID” se muestra a continuación con la función de enfriamiento y/o “Conmutación de parámetros PID” desactivada.



Ajuste	Parámetro TO-DB	Tipo de datos	Rango de valores	Descripción
Activar Entrada manual	"Retain.CtrlParams. SetByUser"	Bool	Bool	Hay que activar esta casilla de verificación para introducir los parámetros PID manualmente.
Ganancia proporcional (calefacción) [2]	"Retain.CtrlParams. Heat.Gain"	Real	Ganancia >= 0.0	Ganancia proporcional PID para calentamiento
Tiempo de integración (calefacción) [1,2]	"Retain.CtrlParams. Heat.Ti"	Real	100000.0 >= Ti >= 0.0	Acción integral PID para calentamiento.
Tiempo derivativo (calefacción) [1,2]	"Retain.CtrlParams. Heat.Td"	Real	100000.0 >= Td >= 0.0	Tiempo de acción derivativa PID para calentamiento.
Coeficiente retardo derivativo (calefacción) [2]	"Retain.CtrlParams. Heat.TdFiltRatio"	Real	TdFiltRatio >= 0.0	Coeficiente de retardo derivativo PID para calentamiento que define el tiempo de retardo derivativo como coeficiente del tiempo derivativo PID.
Ponderación de la acción P (calefacción) [2]	"Retain.CtrlParams. Heat.PWeighting"	Real	1.0 >=PWeighting >= 0.0	Ponderación de la ganancia proporcional PID para calentamiento en el sistema regulado ya sea directamente o en loopback.

Ajuste	Parámetro TO-DB	Tipo de datos	Rango de valores	Descripción
Ponderación de la acción D (calefacción) [2]	"Retain.CtrlParams. Heat.DWeighting"	Real	1.0 >=DWeighting >= 0.0	Ponderación de la acción derivativa PID para calentamiento en el sistema regulado ya sea directamente o en loopback.
Tiempo de muestreo algoritmo PID (calefacción) [1,2]	"Retain.CtrlParams. Heat.Cycle"	Real	100000.0 >=Cycle > 0.0	Ciclo de llamada interno del regulador PID para calentamiento. Redondeado a un entero múltiplo del tiempo de ciclo de llamada de FB.
Ancho zona muerta (calefacción) [2,3]	"Retain.CtrlParams. Heat.DeadZone"	Real	DeadZone>= 0.0	Ancho de la zona muerta para error de regulación de calentamiento.
Ancho de la zona de regulación (calefacción)[2,3]	"Retain.CtrlParams. Heat.ControlZone"	Real	ControlZone> 0.0	Ancho de la zona del error de regulación para calentamiento en la que la regulación PID está activa. Si el error de regulación sale de este rango, la salida adopta valores de salida máximos. El valor predeterrminado es "MaxReal", de modo que la zona de regulación está desactivada mientras no se ejecute el autoajuste. El valor "0.0" está prohibido para la zona de regulación; con el valor "0.0", PID_Temp se comporta como un regulador de dos posiciones que siempre está calentando o enfriando a plena potencia.

Ajuste	Parámetro TO-DB	Tipo de datos	Rango de valores	Descripción
Estructura del regulador (calefacción)	"PIDSelfTune.SUT. TuneRuleHeat", "PIDSelfTune.TIR. TuneRuleHeat"	Int	"PIDSelf-Tune.SUT. TuneRuleHeat" = 0..2, "PIDSelf-Tune.TIR. TuneRuleHeat" = 0..5	Se puede seleccionar el algoritmo de optimización para calentamiento. Selecciones posibles: • PID (temperatura) (=predeterminada) ("PIDSelfTune.SUT. TuneRuleHeat" = 2) ("PIDSelfTune.TIR. TuneRuleHeat" = 0) • PID ("PIDSelfTune.SUT. TuneRuleHeat" = 0) ("PIDSelfTune.TIR. TuneRuleHeat" = 0) • PI ("PIDSelfTune.SUT. TuneRuleHeat" = 1) ("PIDSelfTune.TIR. TuneRuleHeat" = 4) Con cualquier otra combinación se mostrará “Definido por usuario”, pero “Definido por usuario” no está disponible por defecto. “PID (Temperatura)” es nuevo en PID_Temp, con un método de optimización inicial específico (SUT) para procesos de temperatura.
Ganancia proporcional (refrigeración) [4]	"Retain.CtrlParams. Cool.Gain"	Real	Ganancia >= 0.0	Ganancia proporcional PID para enfriamiento
Tiempo de integración (refrigeración) [1,4]	"Retain.CtrlParams. Cool.Ti"	Real	100000.0 >=Ti >= 0.0	Acción integral PID para enfriamiento.
Tiempo derivativo (refrigeración) [1,4]	"Retain.CtrlParams. Cool.Td"	Real	100000.0 >=Td >= 0.0	Tiempo de acción derivativa PID para enfriamiento.

Ajuste	Parámetro TO-DB	Tipo de datos	Rango de valores	Descripción
Coeficiente retardo derivativo (enfri-amiento) [4]	Retain.CtrlParams. Cool.TdFiltRatio"	Real	TdFiltRatio>= 0.0	Coeficiente de retardo derivativo PID para enfriamiento que define el tiempo de retardo derivativo como coefi-ciente del tiempo deri-vativo PID.
Pondera-ción de acción pro-porcional (refrigera-ción) [4]	"Retain.CtrlParams. Cool.PWeighting"	Real	1.0 >=PWeighting >= 0.0	Ponderación de la ganancia proporcional PID para enfriamiento en el sistema regulado ya sea directamente o en loopback.
Pondera-ción de la acción D (refrigera-ción) [4]	Retain.CtrlParams. Cool.DWeighting"	Real	1.0 >=DWeighting >= 0.0	Ponderación de la acción derivativa PID para enfriamiento en el sistema regulado ya sea directamente o en loopback.
Tiempo de muestreo algoritmo PID (refrige-ración) [1,4]	"Retain.CtrlParams. Cool.Cycle"	Real	100000.0 >=Cycle > 0.0	Ciclo de llamada in-terno del regulador PID para enfriamiento. Redondeado a un ente-ro múltiplo del tiempo de ciclo de llamada de FB.
Ancho zona muerta (refriger-ación) [3,4]	"Retain.CtrlParams. Cool.DeadZone"	Real	DeadZone>= 0.0	Ancho de la zona muerta para error de regulación de enfria-miento.

Ajuste	Parámetro TO-DB	Tipo de datos	Rango de valores	Descripción
Ancho de la zona de regulación (refrigeración)[3,4]	"Retain.CtrlParams. Cool.ControlZone"	Real	ControlZone> 0.0	Ancho de la zona del error de regulación para enfriamiento en la que la regulación PID está activa. Si el error de regulación sale de este rango, la salida adopta valores de salida máximos. El valor predeterrminado es "MaxReal", de modo que la zona de regulación está desactivada mientras no se ejecute el autoajuste. El valor "0.0" está prohibido para la zona de regulación; con el valor "0.0", PID_Temp se comporta como un regulador de dos posiciones que siempre está calentando o enfriando a plena potencia.

Ajuste	Parámetro TO-DB	Tipo de datos	Rango de valores	Descripción
Estructura del regulador (refrigeración)	"PIDSelfTune.SUT. TuneRuleCool", "PIDSelfTune.TIR. TuneRuleCool"	Int	"PIDSelf-Tune.SUT. TuneRuleHeat" = 0..2, "PIDSelf-Tune.TIR. TuneRuleHeat" = 0..5	Se puede seleccionar el algoritmo de optimización para enfriamiento. Selecciones posibles: • PID (temperatura) (=predeterminada) ("PIDSelfTune.SUT. TuneRuleCool" = 2) ("PIDSelfTune.TIR. TuneRuleCool = 0) • PID ("PIDSelfTune.SUT. TuneRuleCool" = 0) ("PIDSelfTune.TIR. TuneRuleCool" = 0) • PI ("PIDSelfTune.SUT. TuneRuleCool" = 1) ("PIDSelfTune.TIR. TuneRuleCool" = 4) Con cualquier otra combinación se mostrará “Definido por usuario”, pero “Definido por usuario” no está disponible por defecto. “PID (Temperatura)” es nuevo en PID_Temp, con un método de optimización inicial (SUT) específico para procesos de temperatura. Solo disponible si se han seleccionado los siguientes elementos: “Activar salida (refrigeración)” en la vista “Ajustes básicos” ("Con-fig.ActivateCooling" = TRUE), y “Conmutación de parámetros PID” en la vista “Ajustes de la salida” (Con-fig.AdvancedCooling = TRUE).

Ajuste	Parámetro TO-DB	Tipo de datos	Rango de valores	Descripción

[1] El campo muestra "s" (segundos) como unidad de tiempo.

[2] Solo disponible si se ha activado "Activar entrada manual" en los parámetros PID ("Retain.CtrlParams.SetByUser" = TRUE).

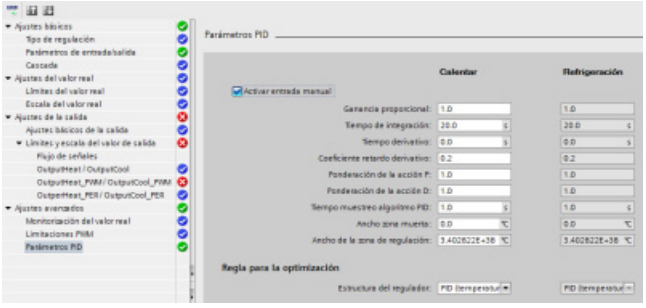
[3] La unidad de medida se indica al final del campo de la forma seleccionada en la vista “Ajustes básicos”.

[4] Solo disponible si se han seleccionado los siguientes elementos: "Activar entrada manual" en los parámetros PID ("Retain.CtrlParams.SetByUser" = TRUE), “Activar salida (refrigeración)” en la vista “Ajustes básicos” ("Config.ActivateCooling" = TRUE), y “Conmutación de parámetros PID” en la vista “Ajustes de la salida” (Config.AdvancedCooling = TRUE).

Control del valor de arranque PID

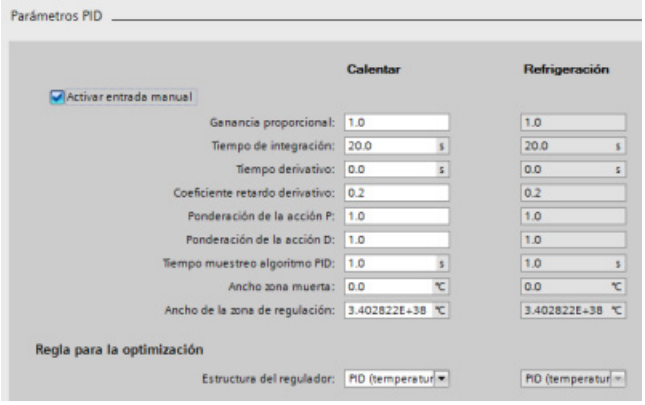
Los valores reales de los parámetros de configuración PID se pueden editar para que el comportamiento del regulador PID pueda optimizarse en modo online.

Abra los "Objetos tecnológicos" del regulador PID y el objeto "Configuración". Para acceder al control del valor de arranque, haga clic en el icono "monóculo" de la esquina superior izquierda del cuadro de diálogo:



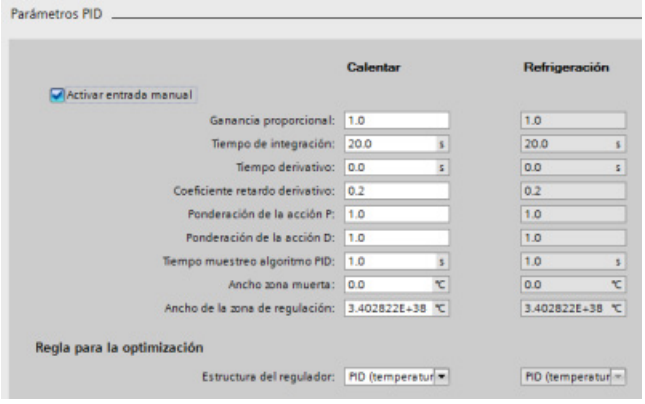
Ahora puede cambiar el valor de cualquier parámetro de configuración del regulador PID como muestra la figura inferior.

Se puede comparar el valor actual con el valor de arranque del proyecto (offline) y el valor de arranque del PLC (online) de cada parámetro. Esto es necesario para detectar las diferencias online/offline del bloque de datos del objeto tecnológico (TO-DB) y para estar informado sobre los valores que se utilizarán como valores actuales en el siguiente cambio de STOP a ARRANQUE del PLC. Además, un icono de comparación permite identificar fácilmente las diferencias online/offline:



La figura anterior muestra la pantalla de parámetros PID con iconos de comparación que muestran qué valores difieren en los proyectos online y offline. Un icono verde indica que los valores son idénticos; un icono azul/naranja indica que los valores difieren.

Además, haga clic en el botón de parámetros con la flecha hacia abajo y abra la pequeña ventana que muestra el valor de arranque del proyecto (offline) y el valor de arranque del PLC (online) para cada parámetro:



Servidor web para una conexión a Internet sencilla 9

El servidor web ofrece a la página web acceso a datos de la CPU y datos de proceso de la CPU. Con estas páginas web se puede acceder a la CPU (o CP habilitado para web) con el navegador web del PC o dispositivo móvil. Las páginas web estándar permiten a los usuarios autorizados llevar a cabo estas y otras funciones:

- Cambiar el estado operativo (STOP y RUN) de la CPU.
- Vigilar y modificar variables PLC, variables de bloques de datos y valores E/S
- Visualizar y descargar registros
- Visualizar el búfer de diagnóstico de la CPU.
- Actualizar el firmware de la CPU.

El servidor web también permite crear páginas web definidas por el usuario que pueden acceder a los datos de la CPU. Estas páginas se pueden desarrollar con el software para crear contenido HTML que se desee usar. Se pueden introducir comandos "AWP" (Automation Web Programming) predefinidos en el código HTML para acceder a los datos de la CPU.

Los usuarios y los niveles de autorización para el servidor web pueden configurarse en la configuración de la CPU en STEP 7.

Requisitos del navegador web

El servidor web soporta los siguientes navegadores web para PC:

- Internet Explorer 8.0
- Internet Explorer 9.0
- Mozilla Firefox 17.0.1
- Google Chrome 23.0
- Apple Safari 5.1.7 (Windows)
- Apple Safari 6.0.2 (Mac)

El servidor web soporta los siguientes navegadores web para dispositivos móviles:

- Internet Explorer 6.0 o inferior, para paneles HMI
- Mobile Safari 7534.48.3 (iOS 5.0.1)
- Mobile Android Browser 2.3.4
- Mobile Google Chrome 23.0

Para más información sobre las restricciones que rigen para navegadores web que pueden interferir en la visualización de páginas web definidas por el usuario, consulte los temas sobre restricciones (Página 265).

9.1 Páginas web estándar fáciles de usar

Resulta muy fácil utilizar páginas web estándar. Tan solo tiene que habilitar el servidor web al configurar la CPU y configurar usuarios del servidor web con derechos para realizar las tareas necesarias.

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 dispone de CP habilitados para servidor web, estos también se muestran en la página de arranque y es posible conectarse con páginas web desde dichos CP.

Si el usuario dispone de los derechos necesarios, podrá cambiar el estado operativo de la CPU (STOP y RUN) o encender los LED.

La página de estado de las variables permite controlar o modificar cualquier dato de memoria o E/S en la CPU. Para observar valores es necesario el permiso "Leer estado de valores", y para modificarlos se requiere el permiso "escribir estado de valores". Se puede introducir una dirección directa (como I0.0), un nombre de variable PLC o una variable de un bloque de programa determinado. Los valores de datos se actualizan automáticamente hasta que se deshabilita la opción de actualización automática.

La página del búfer de diagnóstico muestra el búfer de diagnóstico y es accesible para usuarios con permisos para consultar diagnósticos. Es posible seleccionar el intervalo de entradas de diagnóstico que deben mostrarse.

Las entradas de diagnóstico muestran los eventos que han tenido lugar, así como la fecha y hora de la CPU en que se ha producido cada evento. Seleccione un evento para visualizar información detallada de dicho evento.

La página del explorador de archivos permite visualizar, descargar o editar archivos en la memoria de carga de la CPU, tales como registros de datos (Página 129) y recetas. Si la CPU no tiene protección de nivel 4, todos los usuarios pueden ver los archivos desde la página del explorador de archivos. Los usuarios con permisos para modificar archivos pueden borrar, editar y cambiar el nombre de archivos.

Además de mostrar la información sobre los módulos de la estación, la página de información de módulos permite actualizar la versión del firmware de la CPU o de otros módulos que soporten actualizaciones de firmware. Los usuarios con permisos para consultar diagnósticos pueden acceder a la página de información de módulos. Los usuarios con permisos para actualizar el firmware pueden actualizarlo.

Otras páginas web estándar muestran información sobre la CPU (como el número de serie, la versión y la referencia), los parámetros de comunicación (como las direcciones de red, las propiedades físicas de las interfaces de comunicación).

⚠ ADVERTENCIA

Acceso no autorizado a la CPU desde el servidor web

El acceso no autorizado a la CPU o el cambio de variables PLC a valores no válidos puede afectar al funcionamiento del proceso y puede producir la muerte, lesiones corporales graves o daños materiales.

Puesto que la habilitación del servidor web permite a usuarios autorizados realizar cambios de estado operativo, escribir en datos de PLC y actualizar el firmware, Siemens recomienda observar estas prácticas de seguridad:

- Permitir el acceso al servidor web solo con el protocolo HTTPS.
- Proteger con contraseña las IDs de usuario del servidor web utilizando una contraseña segura. Las contraseñas seguras tienen diez 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. La contraseña debe mantenerse en secreto y cambiarse con frecuencia.
- No extender los privilegios mínimos predeterminados del usuario "Cualquiera".
- Realice comprobaciones de errores y de rango de las variables usadas en la lógica del programa, puesto que los usuarios de páginas web pueden cambiar variables PLC a valores no válidos.
- Utilice una red privada virtual (VPN = Virtual Private Network) para conectar al PLC S7-1200 un servidor web desde una ubicación externa a la red protegida.

9.2 Limitaciones que pueden afectar al uso del servidor web

Los factores de TI siguientes pueden afectar al uso del servidor web:

- Generalmente, debe usar la dirección IP de la CPU para acceder a las páginas web estándar o las páginas web definidas por el usuario, o la dirección IP de un router inalámbrico con un número de puerto. 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 JavaScript y cookies. Si la configuración del navegador web deshabilita el JavaScript o las cookies, habilítelos. Si no se pueden habilitar, algunas de las funciones están restringidas. El uso de JavaScript y cookies en las páginas web definidas por el usuario es opcional. En caso de usarlos, es necesario habilitarlos en el navegador.
- El servidor web admite 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, se puede cargar e importar el certificado en las opciones de Internet del navegador web. 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. Varias acciones consumen las 30 conexiones, según el navegador web utilizado y el número de objetos diferentes por página (archivos .css, imágenes, archivos .html adicionales). Algunas conexiones se mantienen mientras el servidor web está mostrando una página; otras conexiones no se mantienen después de 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.

Nota

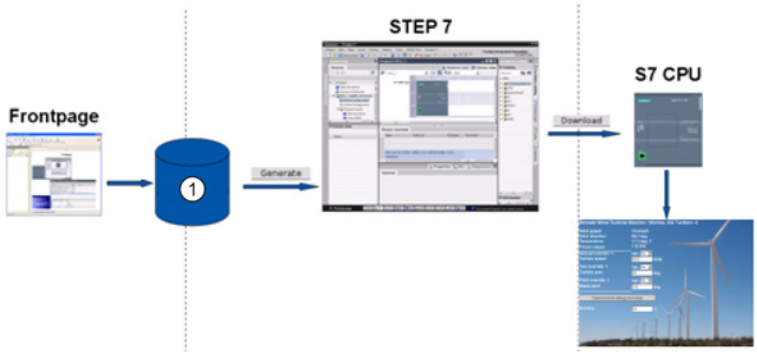
Cierre de sesión antes de cerrar el servidor web

Si ha iniciado sesión en el servidor web, asegúrese de cerrarla antes de cerrar el navegador web. El servidor web soporta un máximo de siete sesiones activas al mismo tiempo.

9.3 Páginas web definidas por el usuario fáciles de crear

9.3.1 Páginas web definidas por el usuario personalizadas fáciles de crear

El servidor web de S7-1200 también ofrece medios para crear páginas HTML específicas para la aplicación que incorporan datos del PLC. El editor de HTML deseado puede utilizarse para crear estas páginas y, a continuación, cargarlas en la CPU en la que estarán accesibles desde las páginas web estándar.



① Archivos HTML con comandos AWP incrustados

Este proceso incluye varias tareas:

- Creación de las páginas HTML con un editor HTML
- incluya comandos AWP en comentarios HTML con código HTML: Los comandos AWP son un conjunto fijo de comandos para acceder a la información de la CPU.
- Configure STEP 7 para leer y procesar las páginas HTML.
- Genere bloques de programa desde las páginas HTML.
- Programe STEP 7 para controlar el uso de las páginas HTML.
- Compile y cargue los bloques de programa en la CPU.
- Acceda a las páginas web definidas por el usuario desde el PC o dispositivo móvil.

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 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 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 como comentarios HTML. Al acabar, guarde las páginas HTML en el PC y anote la ruta de carpetas en la que se han guardado.

Nota

El límite de tamaño para los archivos HTML que contienen comandos AWP es de 64 KB. El tamaño del archivo debe mantenerse por debajo de ese límite.

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 JavaScript 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.

9.3.2 Limitaciones específicas de las páginas web definidas por el usuario

Las limitaciones para páginas web estándar 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.

Comillas en cadenas de texto

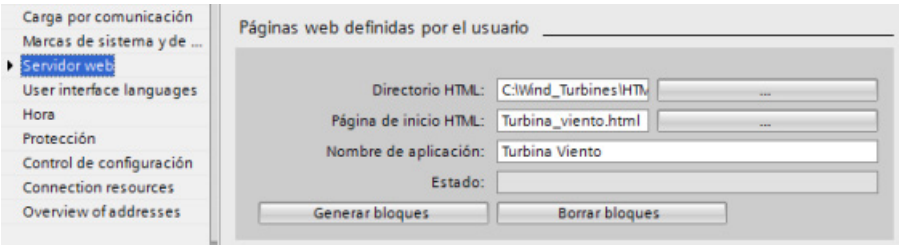
Evite el empleo de cadenas de texto que contengan comillas simples o dobles en las variables de los bloques de datos que se utilizan para cualquier fin en páginas web definidas por el usuario. Puesto que la sintaxis de HTML suele usar comillas simples o dobles como delimitadores, las comillas en las cadenas de texto pueden crear saltos de línea en las páginas web definidas por el usuario.

Para las variables del bloque de datos del tipo String que se usan en páginas web definidas por el usuario, observe las normas siguientes:

- No introduzca comillas simples o dobles en el valor de cadena de la variable del bloque de datos en STEP 7.
- No deje que el programa de usuario haga asignaciones de cadenas que contienen comillas a estas variables de bloque de datos.

9.3.3 Configuración de una página web definida por el usuario

Para configurar páginas web definidas por el usuario, hay que editar las propiedades del servidor web de la CPU.



Tras habilitar las funciones del servidor web, introduzca la información siguiente:

- Nombre y ubicación actual de la página HTML de inicio predeterminada para generar los DBs para las páginas web definidas por el usuario.
- Nombre de la aplicación (opcional). El nombre de la aplicación se emplea para agrupar o categorizar con más detalle las páginas web. Al indicar un nombre de aplicación, el servidor web crea una URL para la página definida por el usuario con el siguiente formato:

  http[s]://ww.xx.yy.zz/awp/<nombre de aplicación>/<nombre de página>.html
- Extensiones de nombres de archivos que contienen comandos AWP. De forma predeterminada, STEP 7 analiza archivos con extensiones .htm, .html o .js. Si dispone de otras extensiones de archivos, añádalas.
- Números de identificación para el número del DB de control y el DB de fragmentos inicial.

Tras configurar el servidor web, haga clic en el botón "Generar bloques" para generar los DB de las páginas HTML. Tras generar los DB, las páginas web forman parte del programa de usuario. El bloque de datos de control para el funcionamiento de las páginas web y los DB de fragmentos contienen todas las páginas HTML.

9.3.4 Uso de la instrucción WWW

La instrucción WWW permite acceder a las páginas web definidas por el usuario desde las páginas web estándar. El programa de usuario solo tiene que ejecutar una vez la instrucción WWW para habilitar el acceso a las páginas web definidas por el usuario. Sin embargo, existe la posibilidad de que las páginas web definidas por el usuario solo sean accesibles bajo determinadas circunstancias. En ese caso, el programa de usuario puede llamar la instrucción WWW según los requisitos de la aplicación.

Tabla 9- 1 Instrucción WWW

KOP / FUP	SCL	Descripción

	`ret_val := #WWW(`
	
	`ctrl_db:=_uint_in_);`
		Identifica el DB de control que se va a utilizar en las páginas web definidas por el usuario.
		El bloque de datos de control es el parámetro de salida 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.

El programa de usuario suele emplear el DB de control directamente, tal y como se creó en el proceso de generación de bloques, sin requerir manipulación adicional. No obstante, el programa de usuario 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, el programa de usuario debe controlar el comportamiento de dichas páginas mediante una tabla de peticiones en el DB de control, tal y como se describe en el *Manual de sistema del controlador programable S7-1200*.

El control de movimiento es muy fácil

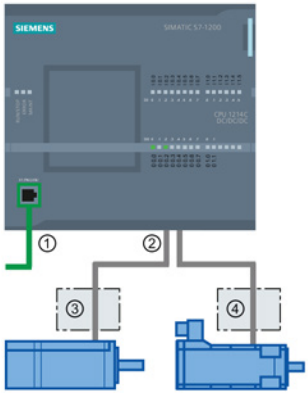
10

La CPU ofrece funciones de control de movimiento para el uso de motores paso a paso y servomotores con interfaz por impulsos. 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 salidas de impulso y sentido para controlar un motor. Puede emplearse una SB con cuatro salidas digitales como salidas de impulso y sentido para controlar dos motores. No se pueden usar salidas de relé integradas como generadores de impulsos para controlar motores. Tanto si usa E/S incorporadas o E/S de SB, o bien una combinación de ambas E/S, puede tener un número máximo de cuatro generadores de impulsos.

Los cuatro generadores de impulsos tienen asignaciones de E/S predeterminadas. Sin embargo, se pueden configurar para cualquier salida digital de la CPU o SB. Los generadores de impulsos de la CPU no se pueden asignar a SM o a las E/S descentralizadas.

Nota

Los trenes de impulsos no pueden ser utilizados por otras instrucciones del programa de usuario

Si las salidas de la CPU o la Signal Board se configuran como generadores de impulsos (para el uso de instrucciones PWM o Motion Control), las direcciones de salida correspondientes dejarán de controlar las salidas. 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 10- 1 Número máximo de unidades controlables

Tipo de CPU		E/S incorporadas/ Ninguna SB instalada		Con una SB (2 salidas DC)		Con una SB (4 salidas DC)	
		Con sentido	Sin sentido	Con sentido	Sin sentido	Con sentido	Sin sentido
CPU 1211C	DC/DC/DC	2	4	3	4	4	4
	AC/DC/relé	0	0	1	2	2	4
	DC/DC/relé	0	0	1	2	2	4
CPU 1212C	DC/DC/DC	3	4	3	4	4	4
	AC/DC/relé	0	0	1	2	2	4
	DC/DC/relé	0	0	1	2	2	4
CPU 1214C	DC/DC/DC	4	4	4	4	4	4
	AC/DC/relé	0	0	1	2	2	4
	DC/DC/relé	0	0	1	2	2	4
CPU 1215C	DC/DC/DC	4	4	4	4	4	4
	AC/DC/relé	0	0	1	2	2	4
	DC/DC/relé	0	0	1	2	2	4
CPU 1217C	DC/DC/DC	4	4	4	4	4	4

Nota

El número máximo de generadores de impulsos es cuatro.

Tanto si usa E/S incorporadas, E/S de SB, o bien una combinación de ambas E/S, puede tener un número máximo de cuatro generadores de impulsos.

Tabla 10- 2 Salida de la CPU: frecuencia máxima

CPU	Canal de salida de la CPU	Salida de impulsos y sentido	A/B, cuadratura, arriba/abajo y impulso/sentido
1211C	De Qa.0 a Qa.3	100 kHz	100 kHz
1212C	De Qa.0 a Qa.3	100 kHz	100 kHz
	Qa.4, Qa.5	20 kHz	20 kHz
1214C y 1215C	De Qa.0 a Qa.3	100 kHz	100 kHz
	De Qa.4 a Qb.1	20 kHz	20 kHz
1217C	De DQa.0 a DQa.3 (de .0+, .0- a .3+, .3-)	1 MHz	1 MHz
	De DQa.4 a DQb.1	100 kHz	100 kHz

Tabla 10- 3 Salida de la Signal Board (SB): frecuencia máxima (tarjeta opcional)

Signal Board (SB)	Canal de salida de la SB	Salida de impulsos y sentido	A/B, cuadratura, arriba/abajo y impulso/sentido
SB 1222, 200 kHz	De DQe.0 a DQe.3	200 kHz	200 kHz
SB 1223, 200 kHz	DQe.0, DQe.1	200 kHz	200 kHz
SB 1223	DQe.0, DQe.1	20 kHz	20 kHz

Tabla 10- 4 Frecuencias límite de salidas de impulsos

Salida de impulsos	Frecuencia
Integrada	4 PTO: 2 Hz ≤ f ≤ 1 MHz, 4 PTO: 2 Hz ≤ f ≤ 100 kHz o una combinación de estos valores para 4 PTO.[1] [2]
SB estándar	2 Hz ≤ f ≤ 20 kHz
SB rápidas	2 Hz ≤ f ≤ 200 kHz

1 Consulte la tabla que se muestra más abajo respecto a cuatro posibles combinaciones de velocidad de salida para la CPU 1217C.

2 Consulte la tabla que se muestra más abajo respecto a cuatro posibles combinaciones de velocidad de salida de las CPU 1211C, CPU 1212C, CPU 1214C o CPU 1215C.

Ejemplo: configuraciones de velocidad de salida de impulsos de la CPU 1217C

Nota

La CPU 1217C puede generar salidas de impulsos de hasta 1 MHz mediante las salidas diferenciales incorporadas.

En los ejemplos que se muestran a continuación se muestran cuatro posibles combinaciones de velocidad de salida:

- Ejemplo 1: PTO a 4 - 1 MHz, sin salida de sentido
- Ejemplo 2: PTO a 1 - 1 MHz, 2 - 100 kHz y 1 - 20 kHz, todos con salida de sentido
- Ejemplo 3: PTO a 4 - 200 kHz, sin salida de sentido
- Ejemplo 4: PTO a 2 - 100 kHz y PTO a 2 - 200 kHz, todos con salida de sentido

P = Impulso D = Sentido		Salidas integradas en CPU										Salidas de SB rápidas				Salidas de SB estándar	
		Salidas a 1 MHz (Q)				Salidas a 100 kHz (Q)						Salidas a 200 kHz (Q)				Salidas a 20 kHz (Q)	
		0.0+	0.1+	0.2+	0.3+	0.4	0.5	0.6	0.7	1.0	1.1	4.0	4.1	4.2	4.3	4.0	4.1
		0.0-	0.1-	0.2-	0.3-												
Ej. 1: 4 - 1 MHz (sin salida de sentido)	PTO1	P															
	PTO2		P														
	PTO3			P													
	PTO4				P												
Ej. 2: PTO a 1 - 1 MHz, 2 - 100 kHz y 1 - 20 kHz (todos con salida de sentido)	PTO1	P	D														
	PTO2					P	D										
	PTO3							P	D								
	PTO4															P	D
Ej. 3: 4 - 200 kHz (sin salida de sentido)	PTO1											P					
	PTO2												P				
	PTO3													P			
	PTO4														P		
Ej. 4: 2 - 100 kHz;	PTO1					P	D										
	PTO2							P	D								
	PTO3											P	D				

P = Impulso D = Sentido		Salidas integradas en CPU										Salidas de SB rápidas				Salidas de SB estándar	
2 - 200 kHz (todos con salida de sentido)	PTO4													P	D		

Ejemplo: configuraciones de velocidad de salida de impulsos de CPU 1211C, CPU 1212C, CPU 1214C y CPU 1215C

En los ejemplos que se muestran a continuación se muestran cuatro posibles combinaciones de velocidad de salida:

- Ejemplo 1: PTO a 4 - 100 kHz, sin salida de sentido
- Ejemplo 2: PTO a 2 - 100 kHz y PTO a 2 - 20 kHz, todos con salida de sentido
- Ejemplo 3: PTO a 4 - 200 kHz, sin salida de sentido
- Ejemplo 4: PTO a 2 - 100 kHz y PTO a 2 - 200 kHz, todos con salida de sentido

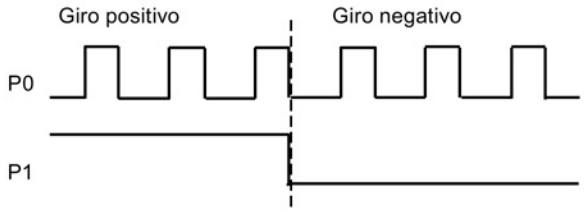
P = Impulso D = Sentido		Salidas integradas en CPU										Salidas de SB rápidas				Salidas de SB lentas	
		Salidas a 100 kHz (Q)				Salidas a 20 kHz (Q)						Salidas a 200 kHz (Q)				Salidas a 20 kHz (Q)	
		0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	1.0	1.1	4.0	4.1	4.2	4.3	4.0	4.1
		CPU 1211C															
		CPU 1212C				CPU 1212C											
		CPU 1214C				CPU 1214C		CPU 1214C									
		CPU 1215C				CPU 1215C		CPU 1215C									
Ej. 1: 4 - 100 kHz (sin salida de sentido)	PTO1	P															
	PTO2		P														
	PTO3			P													
	PTO4				P												
Ej. 2: 2 - 100 kHz; 2 - 20 kHz (todos con salida de senti-do)	PTO1	P	D														
	PTO2			P	D												
	PTO3					P	D										
	PTO4							P	D								
Ej. 3: 4 - 200 kHz (sin	PTO1											P					
	PTO2												P				
	PTO3													P			

P = Impulso D = Sentido		Salidas integradas en CPU										Salidas de SB rápidas				Salidas de SB lentas	
salida de sentido)	PTO4														P		
Ej. 4: 2 - 100 kHz; 2 - 200 kHz (todos con salida de senti-do)	PTO1	P	D														
	PTO2			P	D												
	PTO3											P	D				
	PTO4													P	D		

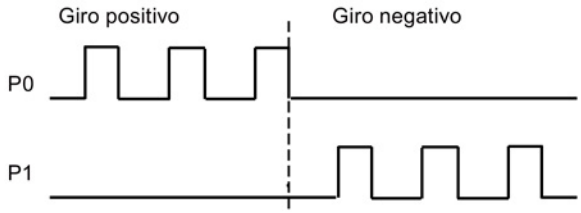
10.1 Escalonamiento

Hay cuatro opciones para la interfaz de "sincronización" con el motor paso a paso/servoaccionamiento. Las opciones son las siguientes:

- PTO (impulso A y sentido B): si selecciona una opción PTO (impulso A y sentido B), una salida (P0) controla los impulsos y otra salida (P1) el sentido. P1 es high (activo) si hay impulsos en sentido positivo. P1 es low (inactivo) si hay impulsos en sentido negativo:

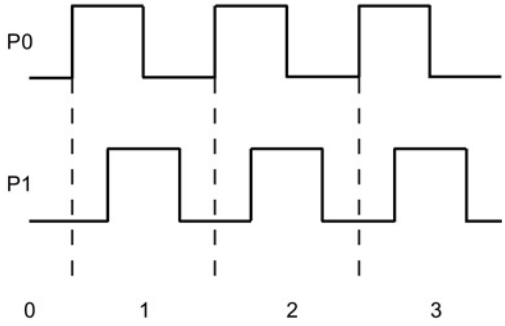
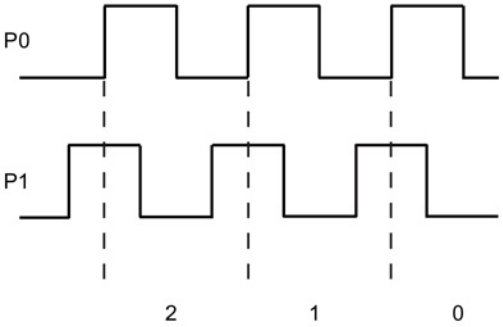


- PTO (incrementar A y decrementar B): si selecciona una opción PTO (incrementar A y decrementar B), una salida (P0) controla los impulsos para sentidos positivos y una salida diferente (P1) los impulsos para sentidos negativos.



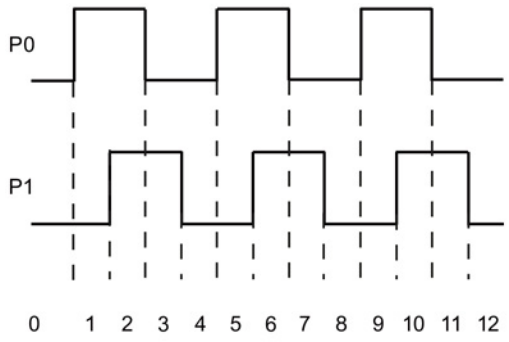
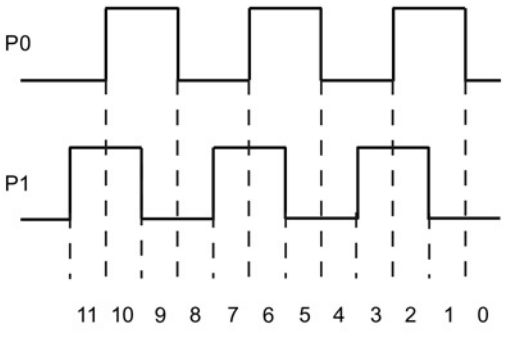
- PTO (con desplazamiento de fase A/B): si selecciona una opción PTO (A/B con desplazamiento de fase), ambas salidas emiten impulsos a la velocidad especificada, pero con un desfase de 90 grados. Se trata de una configuración 1X, lo que significa que un impulso es la cantidad de tiempo entre transiciones positivas de P0. En este caso, el sentido viene determinado por la salida que cambie primero a high (activo). P0 va delante de P1 en el sentido positivo. P1 va delante de P0 en el sentido negativo.

  El número de impulsos generados varía en función del número de transiciones de 0 a 1 de fase A. La relación entre fases determina el sentido del movimiento:

PTO (A/B con desplazamiento de fase)	
La fase A va delante de la fase B (movimiento positivo)	La fase A va detrás de la fase B (movimiento negativo)
	
Número de impulsos	Número de impulsos

- PTO (A/B con desplazamiento de fase - cuádruple): si selecciona una opción PTO (A/B con desplazamiento de fase - cuádruple), ambas salidas emiten impulsos a la velocidad especificada, pero con un desfase de 90 grados. La configuración cuádruple es una configuración 4X, lo que significa que un impulso es la transición de cada salida (tanto positiva como negativa). En este caso, el sentido viene determinado por la salida que cambie primero a high (activo). P0 va delante de P1 en el sentido positivo. P1 va delante de P0 en el sentido negativo.

  La configuración cuádruple se basa en transiciones positivas y negativas de ambas fases A y B. El número de transiciones se configura. La relación entre fases (A delante de B o B delante de A) determina el sentido del movimiento.

PTO (A/B con desplazamiento de fase - cuádruple)	
La fase A va delante de la fase B (movimiento positivo)	La fase A va detrás de la fase B (movimiento negativo)
	
Número de impulsos	Número de impulsos

- PTO (impulso y sentido (sentido deseleccionado)): si deselecciona la salida de sentido en un PTO (impulso y sentido (sentido deseleccionado)), la salida (P0) controlará la emisión de impulsos. La salida P1 no se utiliza y queda disponible para otros usos del programa. En este modo la CPU solo acepta comandos de movimiento positivos. El control de movimiento impide la realización de configuraciones negativas no válidas con este modo seleccionado. Se puede guardar una salida si la aplicación de movimiento solo es en un sentido. La fase simple (una salida) se muestra en la figura siguiente (partiendo de una polaridad positiva):

10.2 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.

     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.

   - 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 solo 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, V2.1 y V2.2; para las CPU S7-1200, V3.0 y V4.0 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

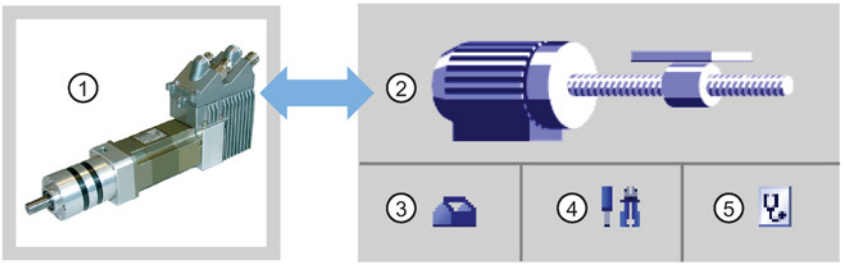
La CPU calcula las tareas de movimiento en "fragmentos" o segmentos de 10 ms. Una vez se ha ejecutado un fragmento, el próximo ya está esperando en la cola para ser ejecutado. Si se interrumpe la tarea de movimiento de un eje (ejecutando otra tarea de movimiento para dicho eje), la nueva tarea de movimiento no puede ejecutarse durante un máximo de 20 ms (el resto de la fracción actual más la fracción en cola).

10.3 Control de movimiento en lazo abierto

10.3.1 Configurar el eje

El eje de lazo abierto del PLC y el accionamiento se conectan a través de una PTO (Pulse Train Output).

STEP 7 ofrece las herramientas de configuración, puesta en servicio y diagnóstico del objeto tecnológico Eje.



① Accionamiento
② Objeto tecnológico
③ Configuración
④ Puesta en servicio
⑤ Diagnóstico

Nota

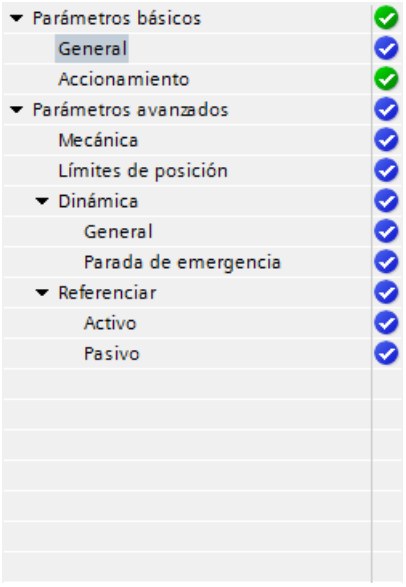
Para las versiones de firmware V2.2 y anteriores de CPU, el PTO necesita la funcionalidad interna de un contador rápido (HSC). Esto significa que el HSC rápido correspondiente no se puede utilizar en ningún otro lugar.

La asignación entre PTO y HSC es fija. Cuando se activa PTO1, se conecta a HSC1. Cuando se activa PTO2, se conecta a HSC2. No se puede controlar el valor actual (por ejemplo, en ID1000) cuando se producen impulsos.

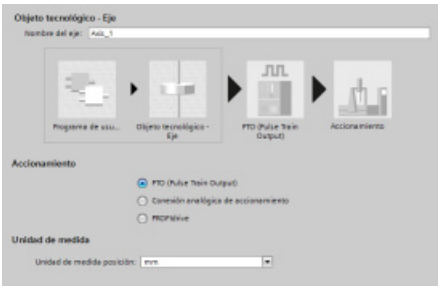
Tanto las CPU S7-1200 de la versión V3.0 como las posteriores no tienen esta restricción; todos los HSC estarán disponibles para usarlos con el programa cuando las salidas de impulsos estén configuradas en estas CPU.

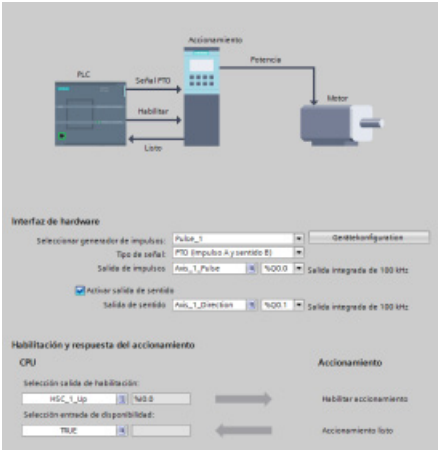
Tabla 10- 5 Herramientas de STEP 7 para control de movimiento

Herramienta	Descripción
Configuración	Configura las propiedades siguientes del objeto tecnológico "Eje": • 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 Guarde la configuración en el bloque de datos del objeto tecnológico.
Puesta en servicio	Prueba la función del eje sin tener que crear un programa de usuario. Cuando se inicia la herramienta, aparece el panel de mando. En el panel de mendo se dispone de los comandos siguientes: • Habilitar y deshabilitar el eje • Desplazar el eje en modo Jog • Posicionar el eje en términos absolutos y relativos • Referenciar el eje • Acusar errores 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.
Diagnóstico	Observa el estado actual y la información de error del eje y del accionamiento.



El selector de árbol del eje PTO no incluye los menús de configuración Encóder, Módulo, Vigilancia de posicionamiento y Lazo de regulación.



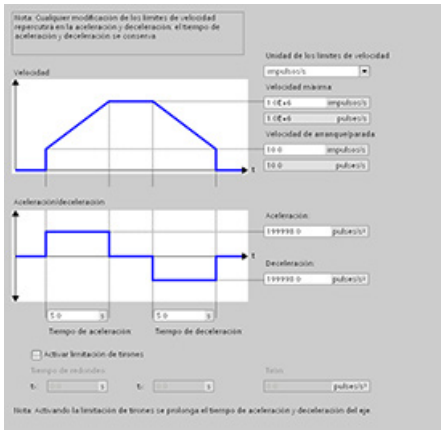


Después de crear el objeto tecnológico para el eje, se configura el eje definiendo los parámetros básicos, como el PTO y la configuración de la interfaz del accionamiento. También se configuran las demás propiedades del eje, como los límites de posición, las animaciones y la referenciación.

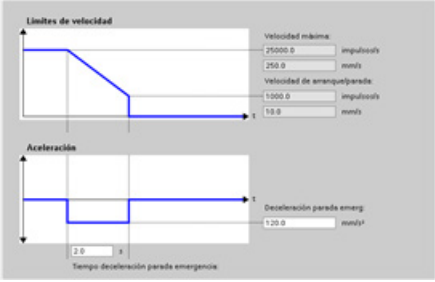
Nota

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.

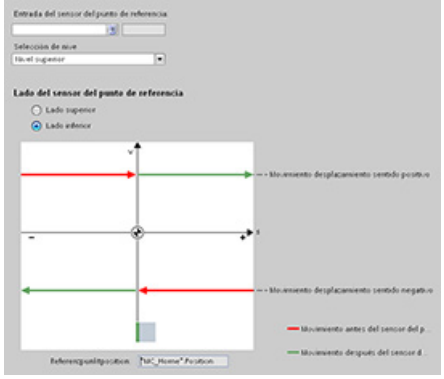
Configure las propiedades de las señales y los mecanismos del accionamiento, así como la vigilancia de posición (finales de carrera por hardware y software).

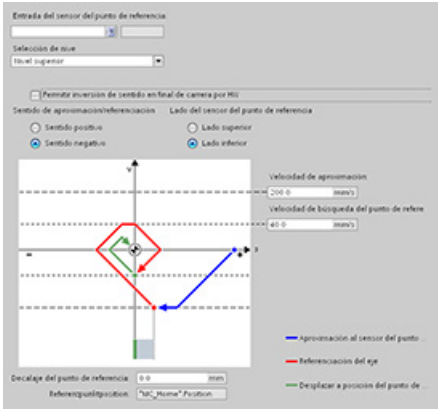


Se configuran las animaciones del movimiento y el comportamiento del comando de parada de emergencia.



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 mando muestra el estado actual del eje. No solo 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 referenciado y Jog. El panel de mando permite además acusar errores.

10.3.2 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 10- 6 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: • 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 10- 7 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 10- 8 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 10- 9 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 10- 10 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 ya utilizado	Un segundo eje utiliza el mismo PTO 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 10- 11 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" solo 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)

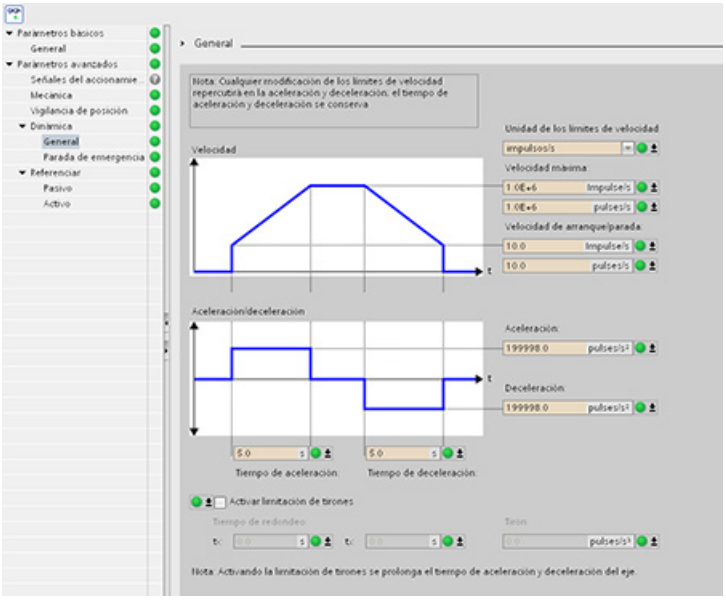
Tabla 10- 12 Límites dinámicos

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)

Control de valor inicial de movimiento

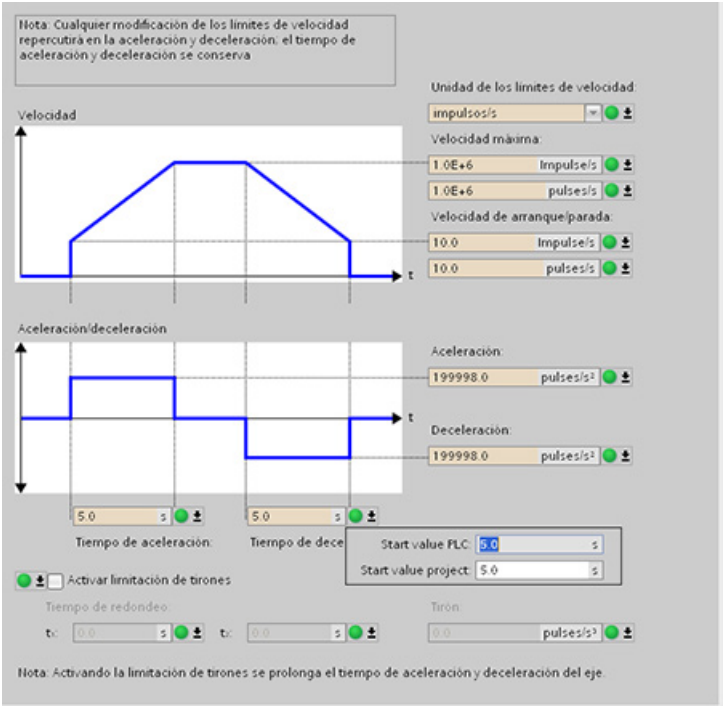
Los valores reales de los parámetros de configuración de movimiento se pueden editar para que el comportamiento del proceso pueda optimizarse en modo online.

Abra los "Objetos tecnológicos" para Motion Control y el objeto "Configuración". Para acceder al control de valor inicial, haga clic en el icono "monóculo" de la esquina superior izquierda del cuadro de diálogo:



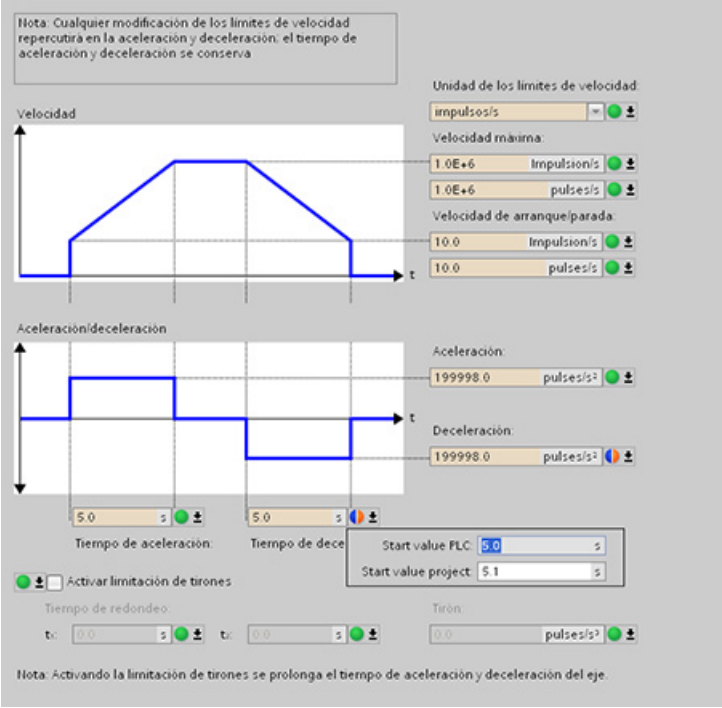
Ahora puede cambiar el valor de cualquiera de los parámetros de configuración de Motion Control de la forma representada en la figura inferior.

Se puede comparar el valor real con el valor inicial del proyecto (offline) y el valor inicial del PLC (online) de cada parámetro. Esto es necesario para detectar diferencias online/offline del bloque de datos del objeto tecnológico (TO-DB) y para estar informado sobre los valores que se utilizarán como actuales en la siguiente transición de STOP a ARRANQUE del PLC. Además, un icono de comparación proporciona una indicación visual que ayuda a identificar fácilmente las diferencias online/offline.



La figura inferior muestra la pantalla de parámetros Motion con iconos de comparación que muestran qué valores son diferentes entre los proyectos online y offline. Un icono verde indica que los valores son los mismos; un icono azul/naranja indica que los valores son diferentes.

Asimismo, se puede hacer clic en el botón de parámetros con la flecha de descarga para abrir una pequeña ventana que muestra el valor inicial del proyecto (offline) y el valor inicial del PLC (online) para cada parámetro.

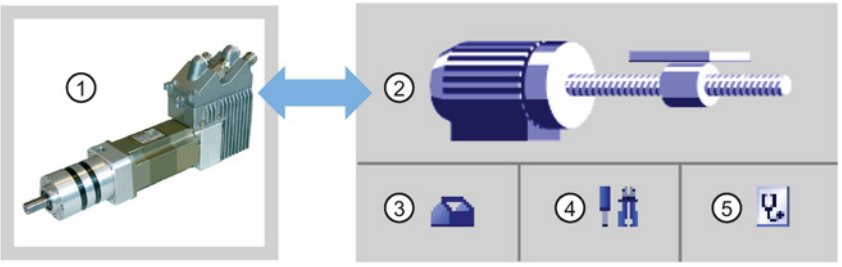


10.4 Control de movimiento en lazo cerrado

10.4.1 Configurar el eje

El eje de lazo cerrado del PLC y el accionamiento se conectan a través del accionamiento analógico o PROFIdrive. El eje de lazo cerrado también requiere un encóder.

STEP 7 ofrece las herramientas de configuración, puesta en servicio y diagnóstico del objeto tecnológico Eje.



① Accionamiento
② Objeto tecnológico
③ Configuración
④ Puesta en servicio
⑤ Diagnóstico

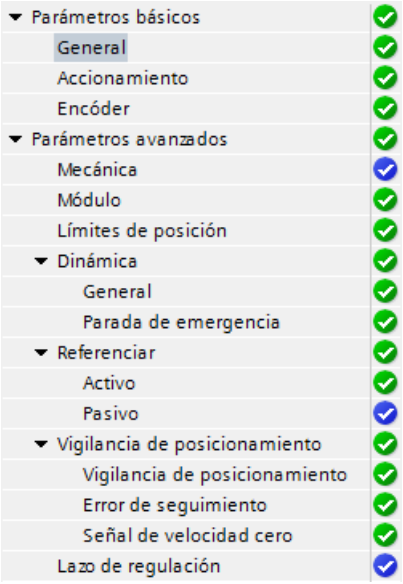
Tabla 10- 13 Herramientas de STEP 7 para control de movimiento en lazo cerrado

Herramienta	Descripción
Configuración	Configura las propiedades siguientes del objeto tecnológico "Eje": • Selección de la conexión del accionamiento analógico o PROFIdrive que se va a utilizar y configuración de la interfaz del accionamiento y del encóder • Propiedades de los mecanismos y de la velocidad de transmisión del accionamiento y el encóder (o máquina o sistema) • Propiedades de los límites de posición, dinámicas y referenciado Guarde la configuración en el bloque de datos del objeto tecnológico.
Puesta en servicio	Prueba la función del eje sin tener que crear un programa de usuario. Cuando se inicia la herramienta, aparece el panel de mando. Los comandos siguientes están disponibles en el panel de mando: • Habilitar y deshabilitar eje • Desplazar el eje en modo Jog • Posicionar el eje en términos absolutos y relativos • Referenciar el eje • Acusar errores Para los comandos de desplazamiento es posible especificar la velocidad y la aceleración o deceleración. El panel de mando también muestra el estado actual del eje.
Diagnóstico	Observa el estado actual y la información de error del eje y del accionamiento.

Nota

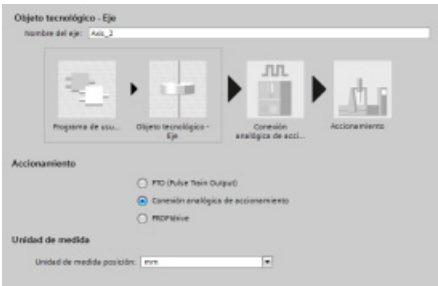
Puede que haya que adaptar los valores de los parámetros de entrada de las instrucciones de control de movimiento a la nueva unidad del programa de usuario.

Después de crear el objeto tecnológico para el eje, se configura el eje definiendo los parámetros básicos, ya sea el accionamiento analógico o la conexión PROFIdrive y la configuración del accionamiento y el encóder.



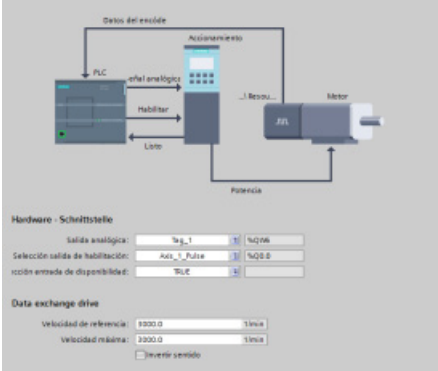
El árbol de selección del accionamiento analógico o la conexión PROFIdrive incluye los menús de configuración Encóder, Módulo, Vigilancia de posicionamiento y Lazo de regulación.

Configuración de la conexión del accionamiento analógico



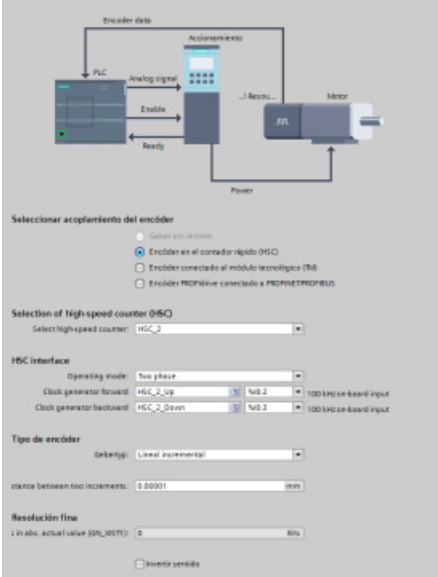
En el cuadro de diálogo de configuración General se seleccionan los siguientes parámetros:

- Botón de opción "Conexión de accionamiento analógico"
- Unidad de medida



En el cuadro de diálogo de configuración Accionamiento se seleccionan los siguientes parámetros:

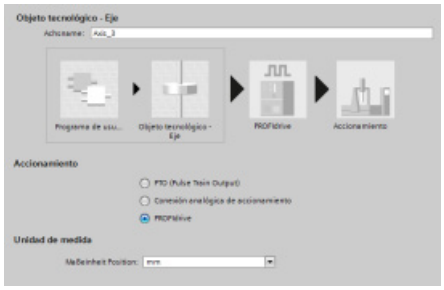
- Salidas hardware del accionamiento analógico
- Velocidades del accionamiento, intercambio de datos



En el cuadro de diálogo de configuración Encóder se seleccionan los siguientes parámetros:

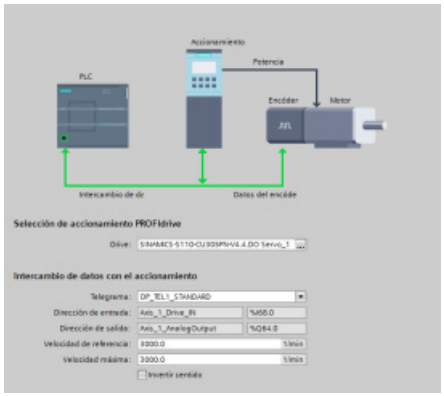
- Acoplamiento del encóder del accionamiento analógico (por ejemplo, un contador rápido (HSC))
- Interfaz HSC
- Tipo de encóder
- Resolución fina

Configuración de PROFIdrive



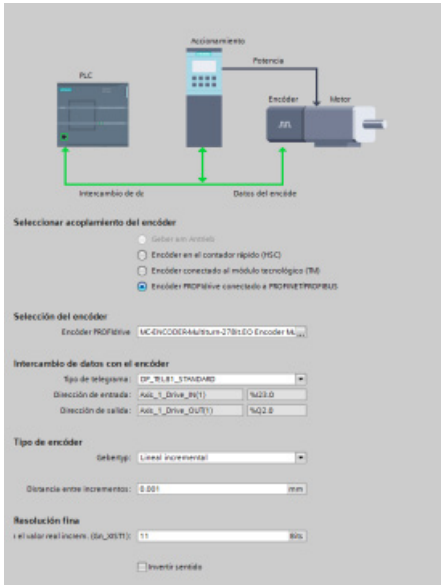
En el cuadro de diálogo de configuración General se seleccionan los siguientes parámetros:

- Botón de opción "PROFIdrive"
- Unidad de medida



En el cuadro de diálogo de configuración Accionamiento se seleccionan los siguientes parámetros:

- Accionamiento PROFIdrive
- Intercambio de datos con el accionamiento



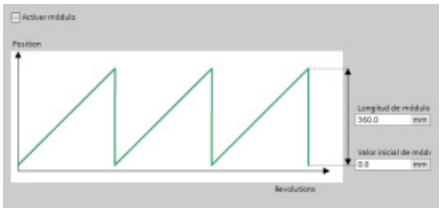
En el cuadro de diálogo de configuración Encóder se seleccionan los siguientes parámetros:

- Acoplamiento de encóder PROFIdrive (por ejemplo, un encóder PROFIdrive en PROFINET)
- Encóder PROFIdrive
- Intercambio de datos con el encóder
- Tipo de encóder
- Resolución fina

Parámetros avanzados

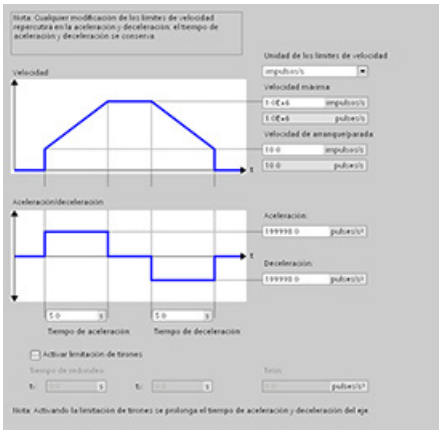
También es posible configurar las siguientes propiedades del eje de lazo cerrado:

- Modulo
- Límites de posición
- Dinámica
- Referenciado
- Vigilancia de posición
- Error de seguimiento
- Señal de parada
- Lazo de regulación

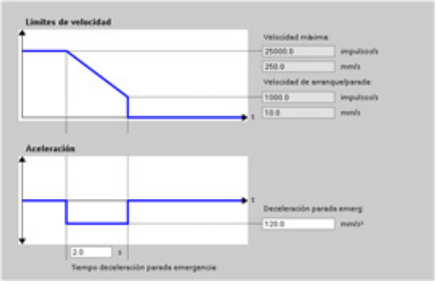


Modulo: Se puede configurar un eje "Modulo" para mover la carga en un área cíclica con un valor o posición de arranque y una longitud dada. Si la posición de la carga llega al final de esta área, se vuelve a poner automáticamente al valor de arranque. Los campos "Longitud" y "Valor inicial de módulo" se activan al hacer clic en la casilla de verificación "Activar Modulo".

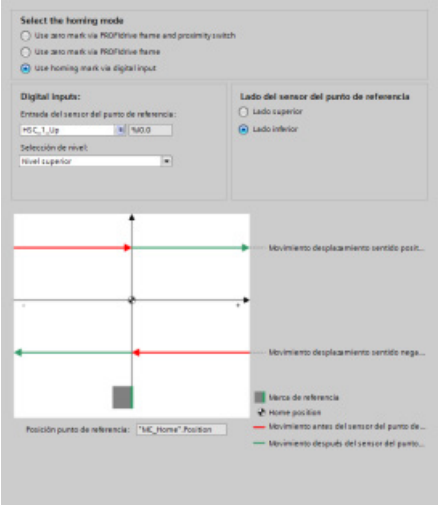
Límites de posición: Se pueden configurar las propiedades de las señales y los mecanismos del accionamiento, así como la vigilancia de posición (finales de carrera por hardware y software).

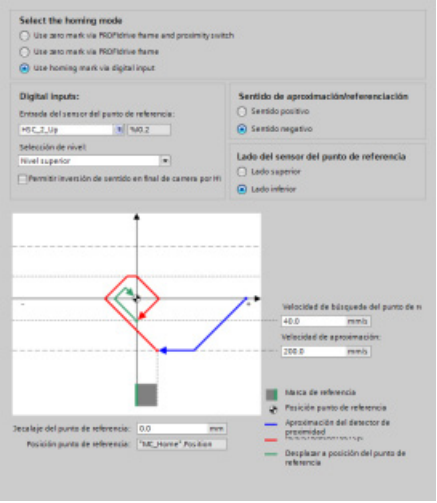


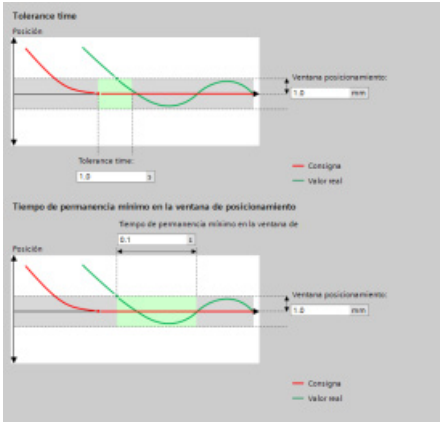
Dinámica: Se pueden configurar las animaciones del movimiento y el comportamiento del comando de parada de emergencia.



Referenciado: Se puede configurar el comportamiento de referenciado (pasivo y activo).



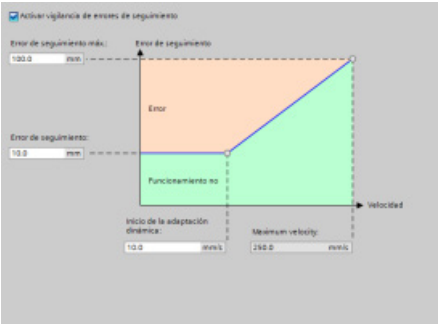




"Vigilancia de posición": Se puede configurar un tiempo de tolerancia y un tiempo de permanencia mínimo para la ventana de posicionamiento.

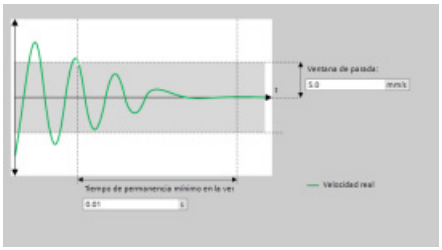
El sistema conecta los tres parámetros siguientes directamente con el eje TO-DB:

- Ventana de posicionamiento
- Tiempo de tolerancia
- Tiempo de permanencia mínimo en la ventana de posicionamiento



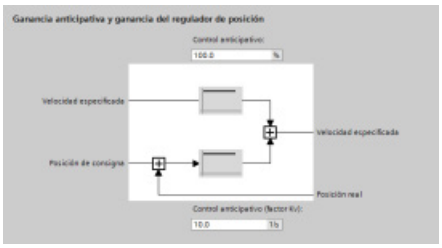
"Error de seguimiento": Se puede configurar la diferencia de la distancia de error permitida en un rango de velocidad. Para activar el error de seguimiento, seleccione la casilla de verificación "Activar vigilancia de errores de seguimiento". Se pueden configurar los siguientes parámetros:

- Error de seguimiento máximo
- Error de seguimiento
- Iniciar ajuste dinámico
- Velocidad máxima



"Señal de parada": Se pueden configurar los siguientes parámetros:

- Tiempo de permanencia mínimo en la ventana de parada
- Ventana de parada.



"Lazo de regulación": Se puede configurar la ganancia de velocidad conocida como "Control previo (factor Kv)".

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 mando muestra el estado actual del eje. No solo es posible habilitar y deshabilitar el eje, sino también probar su posicionamiento (tanto en términos absolutos como relativos) y especificar la velocidad, aceleración y deceleración. También pueden probarse las tareas de referenciado y Jog. El panel de mando permite además acusar errores.

10.4.2 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 10- 14 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: • 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 10- 15 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 10- 16 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 10- 17 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 10- 18 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 ya utilizado	Un segundo eje utiliza el mismo PTO 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 10- 19 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" solo 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)

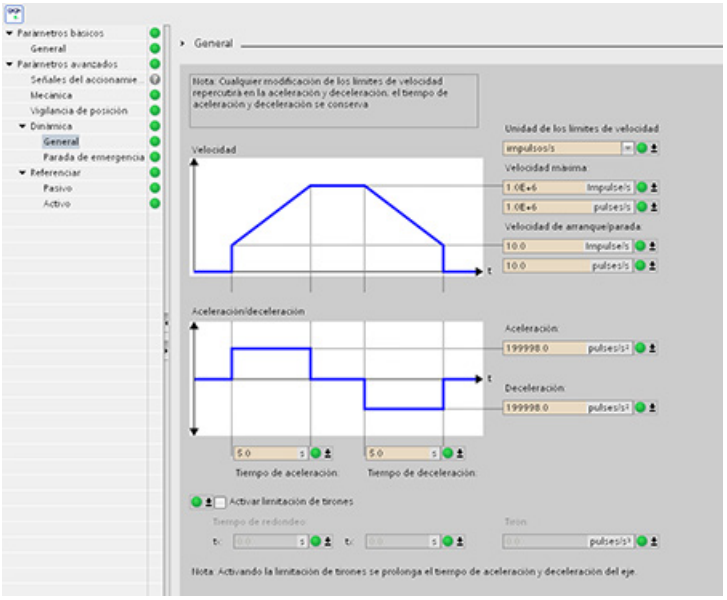
Tabla 10- 20 Límites dinámicos

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)

Control de valor inicial de movimiento

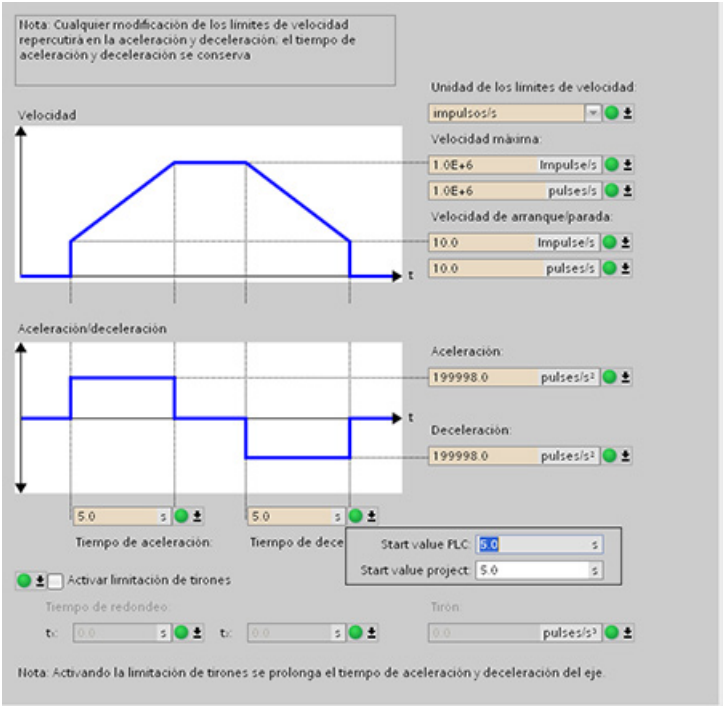
Los valores reales de los parámetros de configuración de movimiento se pueden editar para que el comportamiento del proceso pueda optimizarse en modo online.

Abra los "Objetos tecnológicos" para Motion Control y el objeto "Configuración". Para acceder al control de valor inicial, haga clic en el icono "monóculo" de la esquina superior izquierda del cuadro de diálogo:



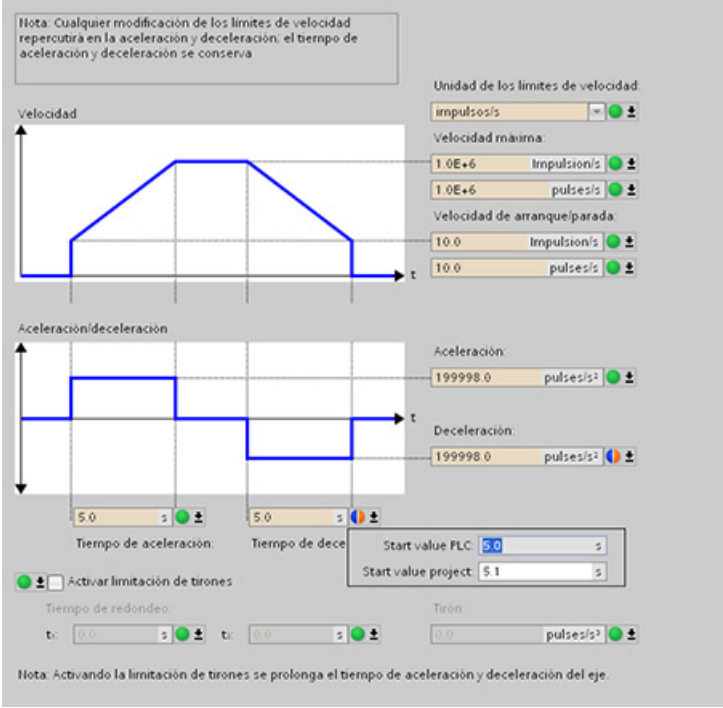
Ahora puede cambiar el valor de cualquiera de los parámetros de configuración de Motion Control de la forma representada en la figura inferior.

Se puede comparar el valor real con el valor inicial del proyecto (offline) y el valor inicial del PLC (online) de cada parámetro. Esto es necesario para detectar diferencias online/offline del bloque de datos del objeto tecnológico (TO-DB) y para estar informado sobre los valores que se utilizarán como actuales en la siguiente transición de STOP a ARRANQUE del PLC. Además, un icono de comparación proporciona una indicación visual que ayuda a identificar fácilmente las diferencias online/offline.



La figura inferior muestra la pantalla de parámetros Motion con iconos de comparación que muestran qué valores son diferentes entre los proyectos online y offline. Un icono verde indica que los valores son los mismos; un icono azul/naranja indica que los valores son diferentes.

Asimismo, se puede hacer clic en el botón de parámetros con la flecha de descarga para abrir una pequeña ventana que muestra el valor inicial del proyecto (offline) y el valor inicial del PLC (online) para cada parámetro.

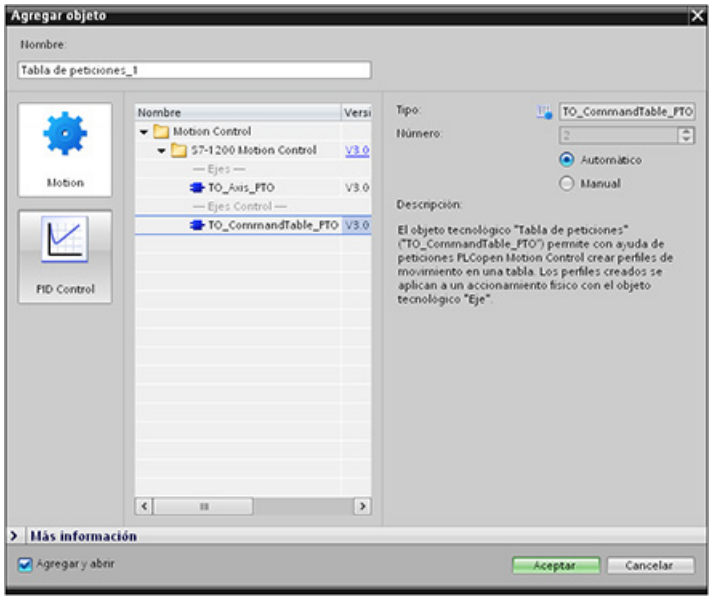


10.5 Configurar TO_CommandTable_PTO

Puede configurar una instrucción MC_CommandTable mediante los objetos tecnológicos. El ejemplo siguiente muestra cómo se hace.

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.

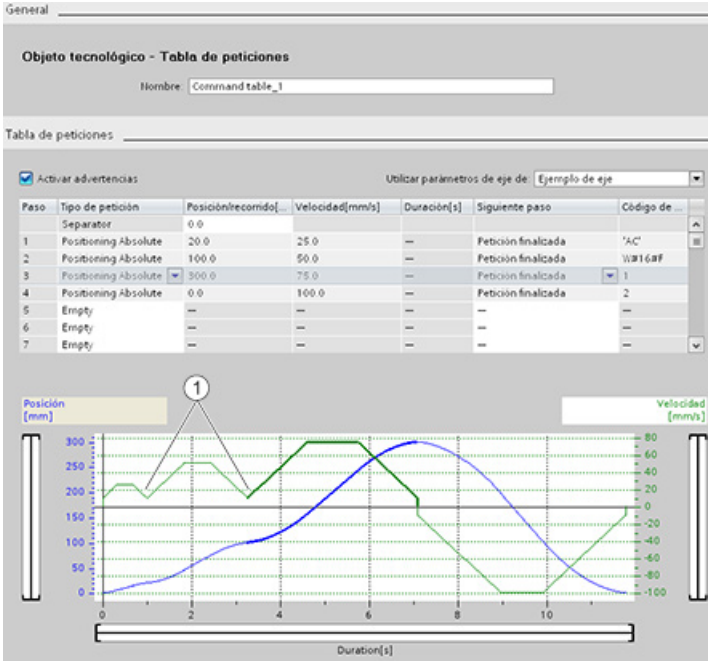
Se pueden seleccionar los tipos de comandos que se van a utilizar para procesar la tabla de comandos. Se pueden introducir hasta 32 pasos. Los comandos se procesan en secuencia, lo que permite generar con facilidad un complejo perfil de movimiento.

Tabla 10- 21 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	Detener eje. Nota: El comando solo 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.

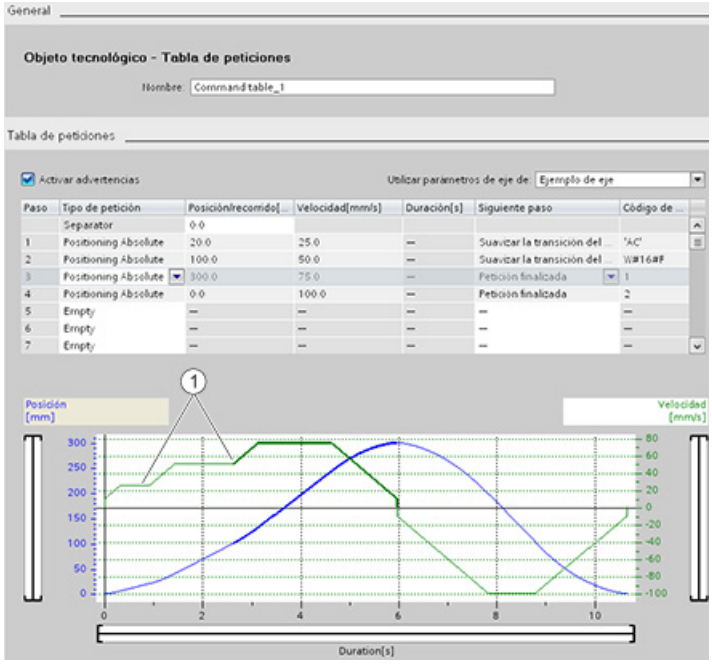
Tipo de comando	Descripción
Positioning Absolute	Coloca el eje basándose en la posición. El comando mueve el eje a la posición indicada, a la velocidad especificada.
Velocity setpoint	Mueve el eje a la velocidad dada.
Wait	Espera hasta que finaliza el período determinado. El comando "Wait" no detiene 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.

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.



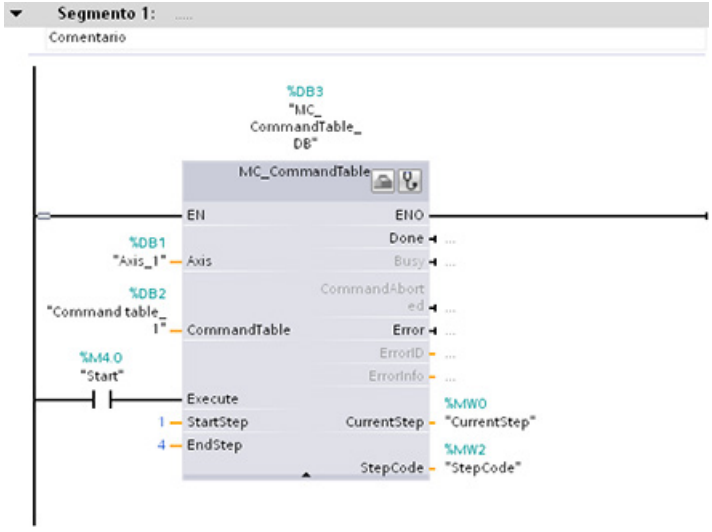
① El eje decelera a la velocidad de arranque/parada entre pasos.

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:



10.6 Operación de Motion Control para S7-1200

10.6.1 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.

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.

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). Solo PTO (Pulse Train Output) se aplica al control de movimiento.

En la tabla que se muestra más abajo se muestran las asignaciones de E/S predeterminadas; sin embargo, los cuatro generadores de impulsos se pueden configurar para cualquier salida digital.

Nota

Los trenes de impulsos no pueden ser utilizados por otras instrucciones dentro del programa de usuario.

Si las salidas de la CPU o la Signal Board se configuran como generadores de impulsos (para el uso de instrucciones PWM o Motion Control), las direcciones de salida correspondientes dejarán de controlar las salidas. 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.

Nota

Las salidas de sentido de PTO se pueden liberar para usarlas en cualquier otro lugar del programa.

Para cada PTO se necesita asignar dos salidas: una como salida de impulsos y otra como salida de sentido. Puede usar la salida de impulsos, pero no la salida de sentido. Puede liberar la salida de sentido para otros fines en el programa del usuario. La salida no se puede usar para la salida de sentido de PTO y en el programa del usuario de forma simultánea.

Tabla 10- 22 Asignaciones de direcciones predeterminadas de las salidas de impulsos y sentido

Utilización de salidas para el control de movimiento		
	Impulso	Sentido
PTO1		
E/S incorporadas	Q0.0	Q0.1
E/S de la SB	Q4.0	Q4.1
PTO2		
E/S incorporadas	Q0.2	Q0.3
E/S de la SB	Q4.2 [1]	Q4.3 [1]
PTO3		
E/S incorporadas	Q0.4 [2]	Q0.5 [2]
E/S de la SB	Q4.0	Q4.1
PTO4		
E/S incorporadas	Q0.6 [3]	Q0.7 [3]
E/S de la SB	Q4.2	Q4.3

1 Las salidas Q4.2 y Q4.3 solo están disponibles en la SB1222 DQ4.

2 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.

3 La CPU 1212C no tiene las salidas Q0.6 ni Q0.7. Por lo tanto, esas salidas no pueden usarse en la CPU 1212C.

4 Esta tabla es válida para las funciones de PTO de la CPU 1211C, CPU 1212C, CPU 1214C, CPU 1215C y CPU 1217C.

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

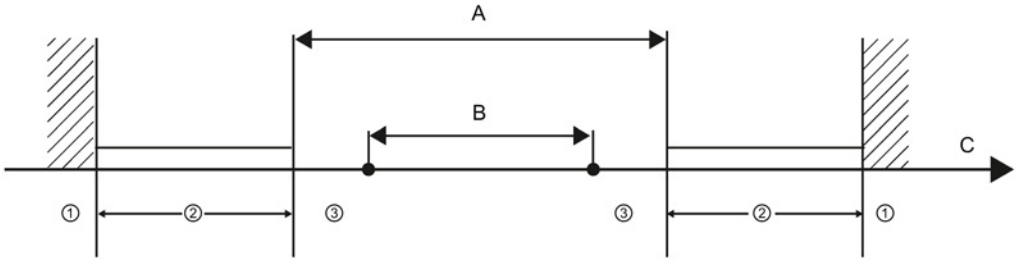
El firmware tomará el control mediante las salidas de impulsos y sentido correspondientes si se ha seleccionado el PTO (Pulse Train Output) y se ha asignado a un eje.

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, esta 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.

10.6.2 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.



①	Parada mecánica	A	Rango de desplazamiento permitido para el eje
②	Límites de hardware inferior y superior	B	Rango de trabajo del eje
③	Límites de software inferior y superior	C	Distancia

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 solo están activos tras referenciar el eje.

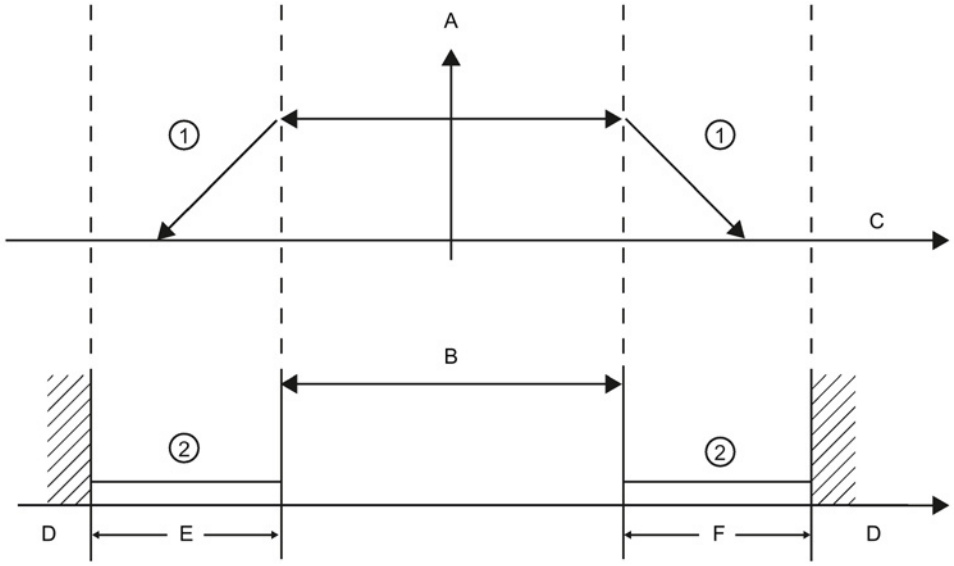
Finales de carrera por hardware

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 solo finales de carrera por hardware que permanezcan activados permanentemente tras la aproximación. Este estado de conmutación solo puede ser revocado una vez que se regrese al rango de desplazamiento válido.

Tabla 10- 23 Entradas disponibles para límites de hardware

Descripción	RPS	LIM-	LIM+
E/S incorporadas	I0.0 - I1.5		
E/S de la SB	I4.0 - I4.3		

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

Riesgos asociados con los cambios en el tiempo de filtro para el canal de entrada digital

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.

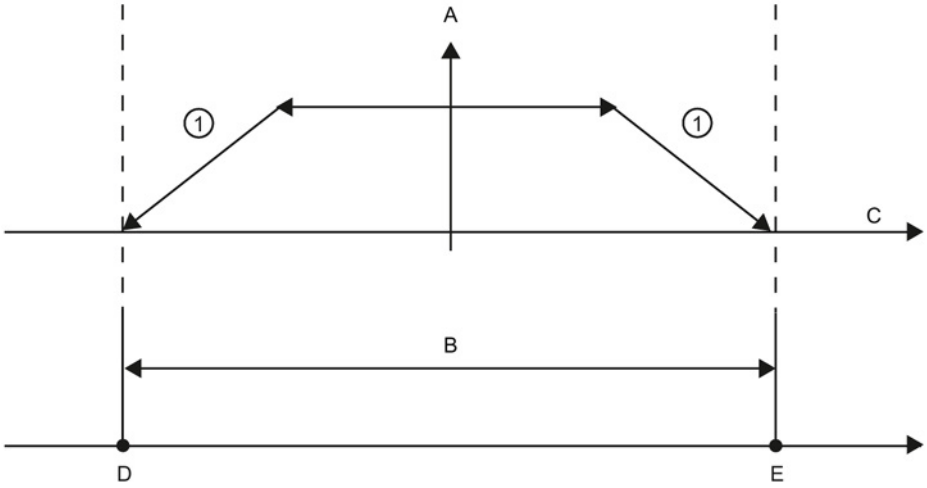
Este cambio de los tiempos de filtro puede originar un funcionamiento inesperado de los equipos o del proceso, lo que puede causar la muerte o lesiones graves al personal y/o daños a los equipos.

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/**PositonLimits_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/**Position Limits_SW**". El estado de la variable "Active" 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**".

10.6.3 Referenciación

10.6.3.1 Referenciar el eje

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. El sentido y la 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 puede situarse en el área de trabajo del eje o fuera del área normal de trabajo, pero dentro del rango de movimiento.

10.6.3.2 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.

Nota

Utilice una de las siguientes medidas para garantizar que la máquina no se desplaza 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 10- 24 Parámetros de configuración para referenciar el eje

Parámetro	Descripción
Entrada interruptor de punto de referencia (Referenciación activa y pasiva)	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. 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. El tiempo de filtrado especificado debe ser menor que la duración de la señal de entrada en el interruptor del punto de referencia.
Inversión automática tras alcanzar los finales de carrera por hardware (Sólo referenciación activa)	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. 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. 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.

Parámetro	Descripción
Dirección de aproximación (Referenciación activa y pasiva)	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.
Interruptor de punto de referencia (Referenciación activa y pasiva)	• 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: – 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)	Especifique la velocidad a la que debe localizarse el interruptor de punto de referencia durante la aproximación al punto de referencia. Valores límite (independientemente de la unidad seleccionada por el usuario): Velocidad inicio/parada ≤ velocidad de aproximación ≤ velocidad máxima
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 10- 25 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

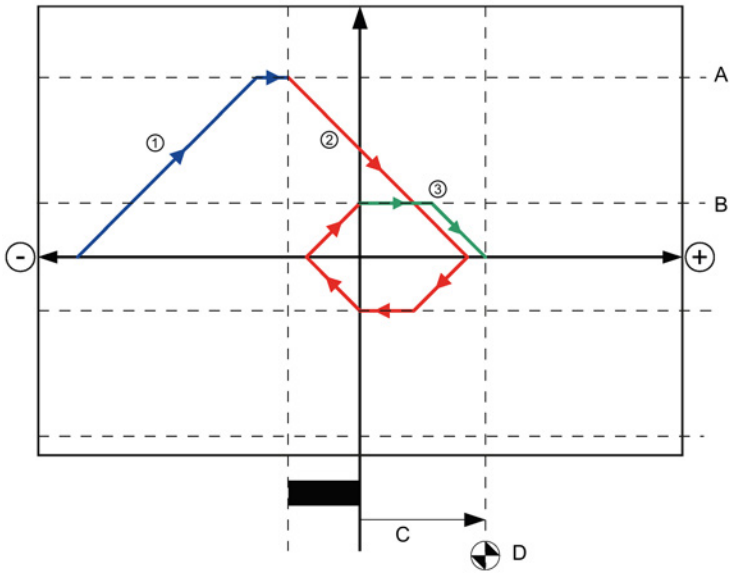
10.6.3.3 Secuencia de referenciación activa

La referenciación activa se inicia con la instrucción de control de movimiento "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 10- 26 Características de velocidad del referenciado 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 deshabilidadas 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.

10.7 Instrucciones de Motion Control

10.7.1 Vista general de instrucciones de MC

Las instrucciones de control de movimiento utilizan un bloque de datos tecnológico asociado y el PTO (salidas del tren de impulsos) específico de la CPU para controlar el movimiento de un eje.

- MC_Power (Página 318) activa y desactiva un eje de control de movimiento.
- MC_Reset (Página 321) resetea todos los errores de control de movimiento. Se acusan todos los errores de control de movimiento que pueden acusarse.
- MC_Home (Página 322) establece la relación entre el programa de control del eje y el sistema de posicionamiento mecánico del eje.
- MC_Halt (Página 325) cancela todos los procesos de movimiento y detiene el movimiento del eje. La posición de parada no está definida.
- MC_MoveAbsolute (Página 327) inicia el movimiento hacia una posición absoluta. La tarea finaliza cuando se alcanza la posición de destino.
- MC_MoveRelative (Página 329) inicia un movimiento de posicionamiento relativo a la posición inicial.
- MC_MoveVelocity (Página 331) hace que el eje se mueva a la velocidad indicada.
- MC_MoveJog (Página 334) ejecuta el modo jog para fines de test y arranque.
- MC_CommandTable (Página 336) ejecuta comandos de eje como una secuencia de movimientos.
- MC_ChangeDynamic (Página 339) cambia la configuración de la dinámica del eje.
- MC_WriteParam (Página 341) escribe un número de parámetros determinado para cambiar las funciones del eje desde el programa de usuario.
- MC_ReadParam (Página 343) lee un número determinado de parámetros que indican la posición y la velocidad (entre otros valores actuales) del eje definidos en la entrada del eje.

Niveles de firmware de CPU

Si tiene una CPU S7-1200 con firmware V4.1, seleccione la versión V5.0 de cada instrucción de movimiento.

Si tiene una CPU S7-1200 con firmware V4.0 o anterior, seleccione la versión correspondiente V4.0, V3.0, V2.0 o V1.0 de cada instrucción de movimiento.

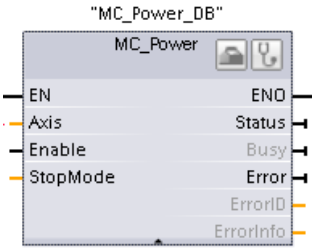
10.7.2 Instrucción MC_Power (Habilitar/bloquear eje)

Nota

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 10- 27 Instrucción MC_Power

KOP / FUP	SCL	Descripción

	`"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_,`
	`ErrorIn-`
	`fo=>_word_out_);`
		La instrucción MC_Power de Motion Control habilita o deshabilita un eje. Antes de poder habilitar o deshabilitar el eje, garantice las siguientes condiciones:
		• El objeto tecnológico está configurado correctamente.
		• No hay ningún error pendiente que impida la habilitación.
		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.

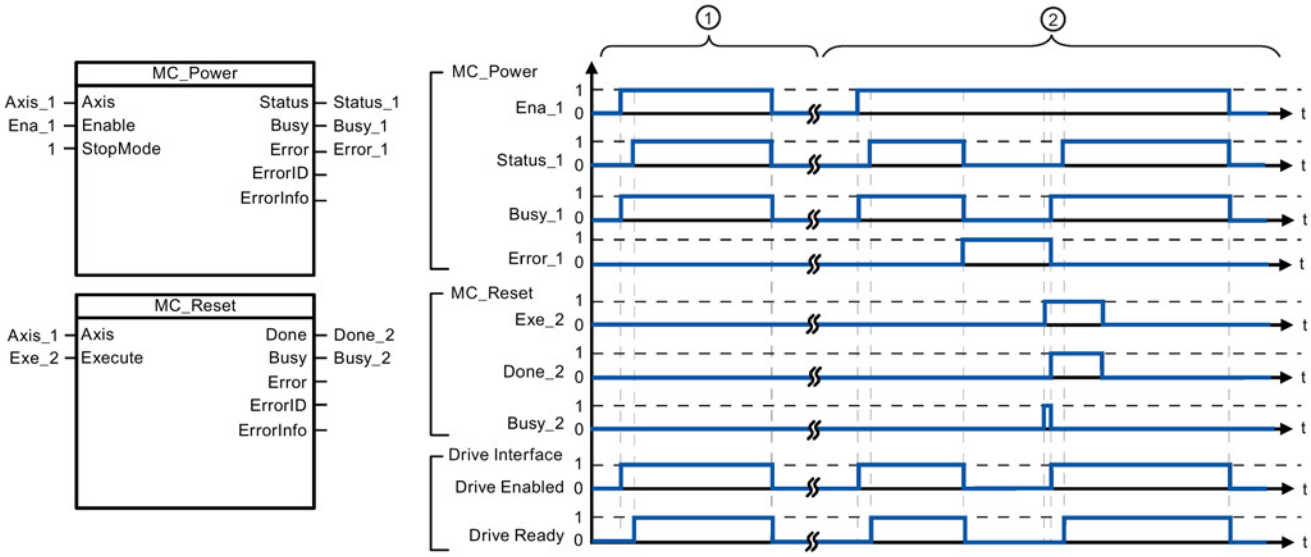
1 STEP 7 crea el DB automáticamente al insertar la instrucción.

2 En el ejemplo SCL, "MC_Power_DB" es el nombre del DB de instancia.

Tabla 10- 28 Parámetros de la instrucción MC_Power

Parámetro y tipo		Tipo de datos	Descripción
Axis	IN	TO_Axis_1	Objeto tecnológico Eje
Enable	IN	Bool	• 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	• 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.

Parámetro y tipo		Tipo de datos	Descripción
Status	OUT	Bool	Se habilita Status del eje: • FALSE: El eje está deshabilitado: – 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: – 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	FALSE: MC_Power no está activo. TRUE: MC_Power está activo.
Error	OUT	Bool	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".
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.

10.7.3 Instrucción MC_Reset (Confirmar error)

Tabla 10- 29 Instrucción MC_Reset

KOP / FUP	SCL	Descripción

	`"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_);`
		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".
		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).
		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.

[1] STEP 7 crea el DB automáticamente al insertar la instrucción.

[2] 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 10- 30 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 solo 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"

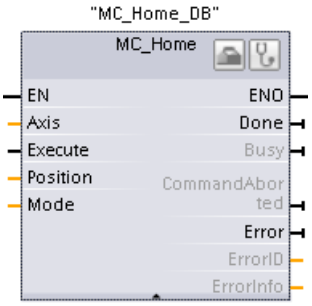
Para acusar un error con MC_Reset, proceda del siguiente modo:

1. Compruebe que se dan las condiciones arriba indicadas.
2. Inicie el acuse del error con un flanco ascendente en el parámetro de entrada Execute.
3. El error ha sido acusado cuando Done es TRUE y la variable <Nombre de eje>.StatusBits.Error del objeto tecnológico es FALSE.

10.7.4 Instrucción MC_Home (Referenciar eje)

Tabla 10- 31 Instrucción MC_Home

KOP / FUP	SCL	Descripción

	`"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_);`
		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:
		Para utilizar la instrucción MC_Home primero es necesario haber habilitado el eje.

[1] STEP 7 crea el DB automáticamente al insertar la instrucción.

[2] 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 10- 32 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	• 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: $-1.0e^{12} \leq$ Position $\leq 1.0e^{12}$
Mode	IN	Int	Modo de referenciación • 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.

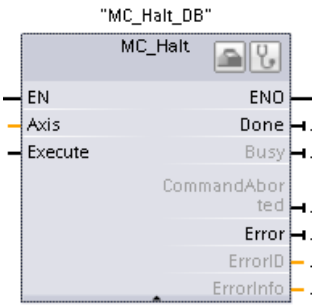
Tabla 10- 33 Respuesta de corrección

Modo	Descripción	
0 o 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: • 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: • Modo MC_Home = 2, 3 • MC_Halt • MC_MoveAbsolute • MC_MoveRelative • MC_MoveVelocity • MC_MoveJog

10.7.5 Instrucción MC_Halt (Detener eje)

Tabla 10- 34 Instrucción MC_Halt

KOP / FUP	SCL	Descripción

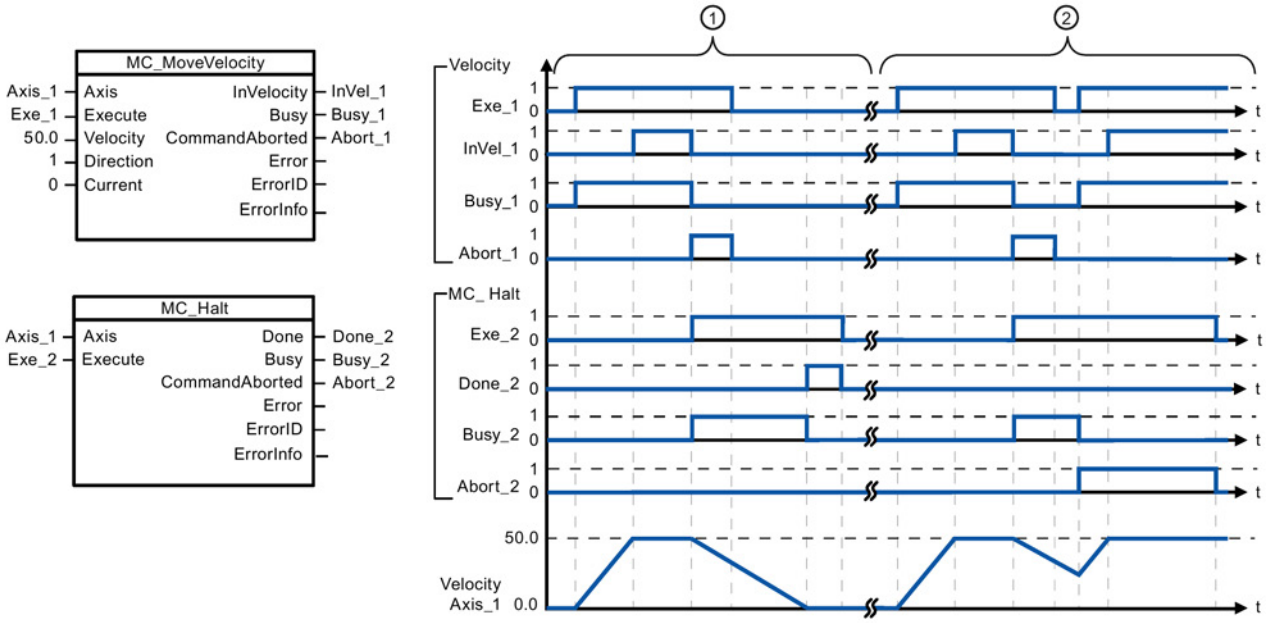
	`"MC_Halt_DB"(`
	`Axis:=_multi_fb_in_,`
	`Execute:=_bool_in_,`
	`Done=>_bool_out_,`
	`Busy=>_bool_out_,`
	`CommandAborted=>_bool_out_,`
	`Error=>_bool_out_,`
	`ErrorID=>_word_out_,`
	`ErrorInfo=>_word_out_);`
		Utilice la instrucción MC_Halt para parar todo el movimiento y detener el eje por completo. La posición de eje totalmente detenido no está definida.
		Para utilizar la instrucción MC_Halt primero es necesario haber habilitado el eje.

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 10- 35 Parámetros de la instrucción MC_Halt

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
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 este 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

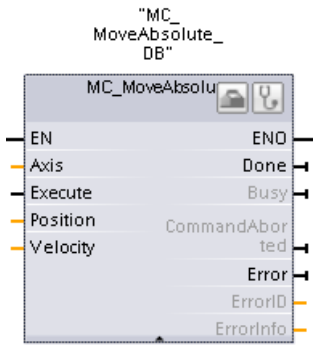
La tarea MC_Halt nueva interrumpe las siguientes tareas de Motion Control activas:

- MC_Home Mode = 3
- MC_Halt
- MC_MoveAbsolute
- MC_MoveRelative
- MC_MoveVelocity
- MC_MoveJog

10.7.6 Instrucción MC_MoveAbsolute (Posicionamiento absoluto del eje)

Tabla 10- 36 Instrucción MC_MoveAbsolute

KOP / FUP	SCL	Descripción

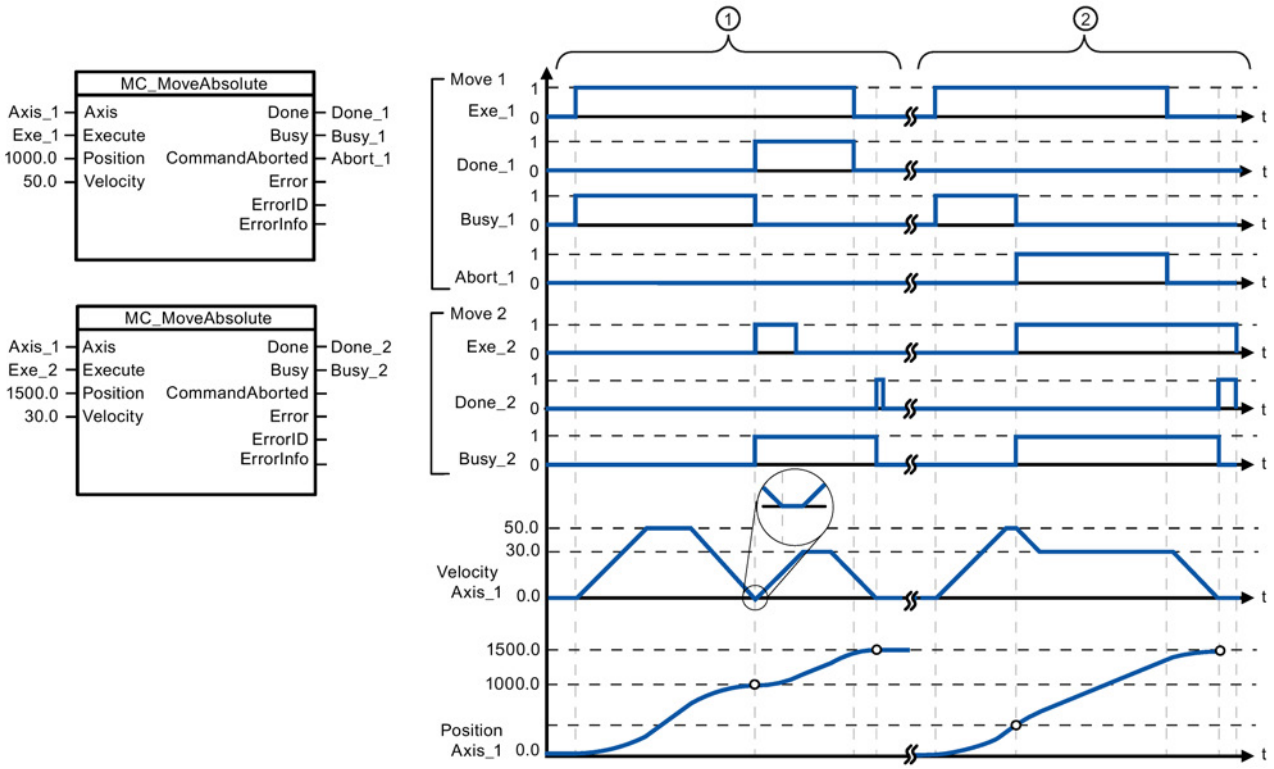
	`"MC_MoveAbsolute_DB"(`
	`Axis:=_multi_fb_in_,`
	`Execute:=_bool_in_,`
	`Position:=_real_in_,`
	`Velocity:=_real_in_,`
	`Done=>_bool_out_,`
	`Busy=>_bool_out_,`
	`CommandAborted=>_bool_out_,`
	`Error=>_bool_out_,`
	`ErrorID=>_word_out_,`
	`ErrorInfo=>_word_out_);`
		Utilice la instrucción MC_MoveAbsolute para iniciar un movimiento de posicionamiento del eje a una posición absoluta.
		Para utilizar la instrucción MC_MoveAbsolute primero es necesario haber habilitado y referenciado el eje.

[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 10- 37 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: $-1.0e^{12} \leq$ 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: velocidad inicio/parada ≤ Velocity ≤ 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

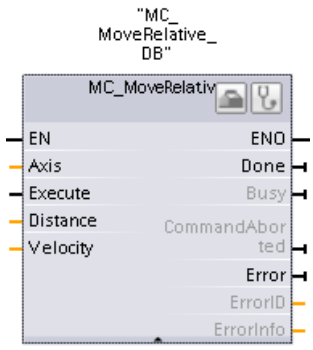
La tarea MC_MoveAbsolute nueva interrumpe las siguientes tareas de Motion Control activas:

- MC_Home Mode = 3
- MC_Halt
- MC_MoveAbsolute
- MC_MoveRelative
- MC_MoveVelocity
- MC_MoveJog

10.7.7 Instrucción MC_MoveRelative (Posicionamiento relativo del eje)

Tabla 10- 38 Instrucción MC_MoveRelative

KOP / FUP	SCL	Descripción

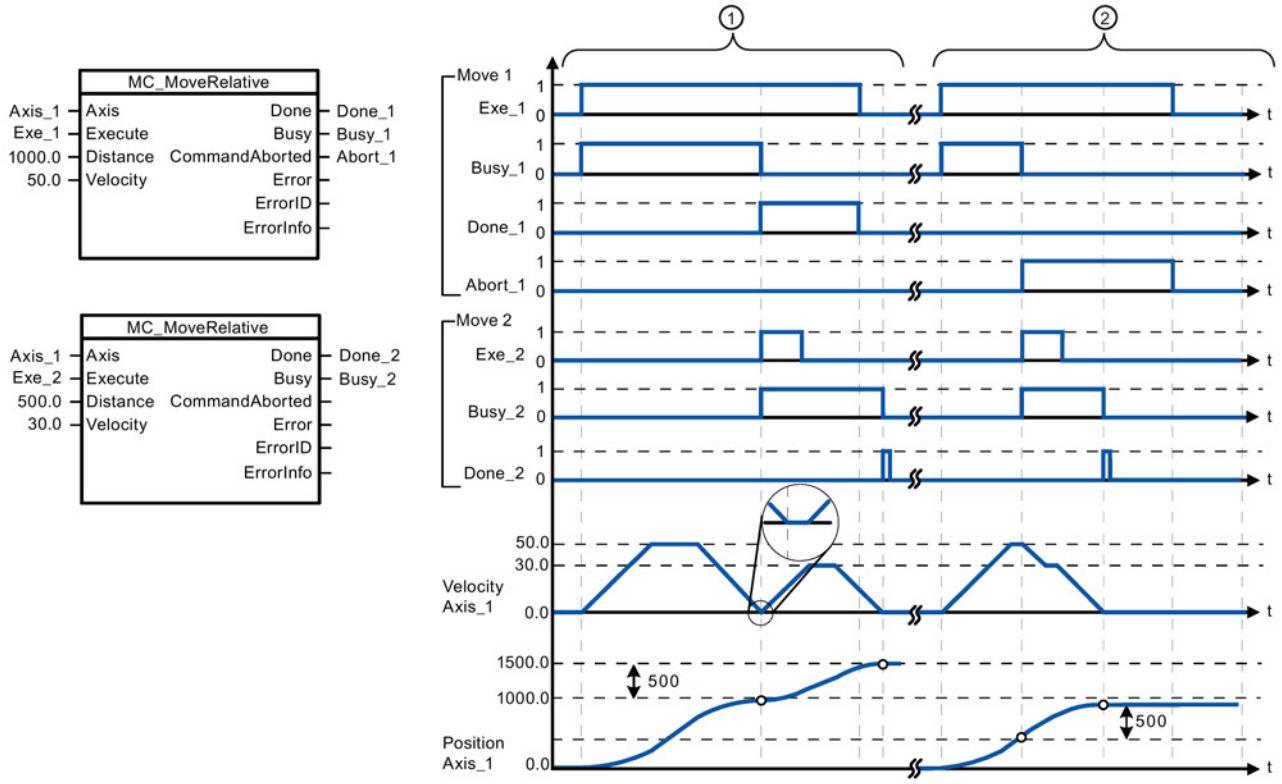
	`"MC_MoveRelative_DB"(`
	`Axis:=_multi_fb_in_,`
	`Execute:=_bool_in_,`
	`Distance:=_real_in_,`
	`Velocity:=_real_in_,`
	`Done=>_bool_out_,`
	`Busy=>_bool_out_,`
	`CommandAborted=>_bool_out_,`
	`Error=>_bool_out_,`
	`ErrorID=>_word_out_,`
	`ErrorInfo=>_word_out_);`
		Utilice la instrucción MC_MoveRelative para iniciar un movimiento de posicionamiento relativo a la posición inicial.
		Para utilizar la instrucción MC_MoveRelative primero es necesario haber habilitado el eje.

[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 10- 39 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: $-1.0e^{12} \leq$ 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 ≤ Velocity ≤ 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

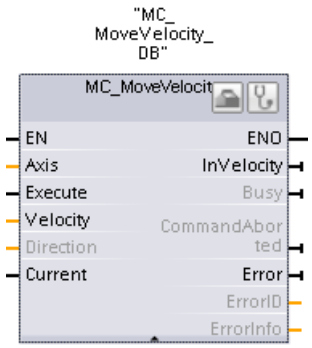
La tarea MC_MoveRelative nueva interrumpe las siguientes tareas de Motion Control activas:

- MC_Home Mode = 3
- MC_Halt
- MC_MoveAbsolute
- MC_MoveRelative
- MC_MoveVelocity
- MC_MoveJog

10.7.8 Instrucción MC_MoveVelocity (Mover el eje a la velocidad predefinida)

Tabla 10- 40 Instrucción MC_MoveVelocity

KOP / FUP	SCL	Descripción

	`"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_);`
		Utilice la instrucción MC_MoveVelocity para mover el eje constantemente a la velocidad especificada.
		Para utilizar la instrucción MC_MoveVelocity primero es necesario haber habilitado el eje.

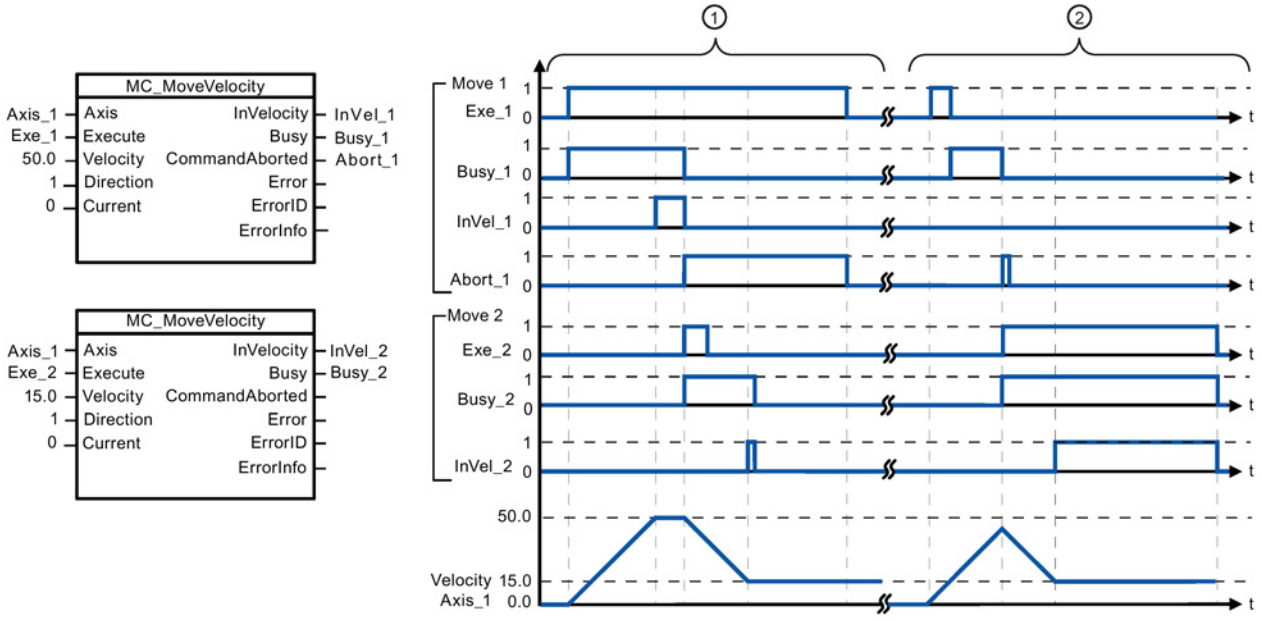
[1] STEP 7 crea el DB automáticamente al insertar la instrucción.

[2] En el ejemplo SCL, "MC_MoveVelocity_DB" es el nombre del DB de instancia.

Tabla 10- 41 Parámetros de la instrucción MC_MoveVelocity

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)
Velocity	IN	Real	Especificación de velocidad para movimiento del eje (valor predeterminado: 10,0) Valores límite: Velocidad de inicio/parada ≤ |Velocity| ≤ velocidad máxima (Se permite Velocity = 0,0)
Direction	IN	Int	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).
Current	IN	Bool	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: • 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 señaliza 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.

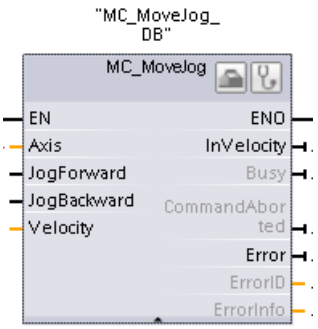
"Busy" indica el valor TRUE durante la operación de deceleración y cambia a FALSE junto con "InVelocity". Si se activa el parámetro "Execute" = TRUE, "InVelocity" y "Busy" están enclavados.

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.

10.7.9 Instrucción MC_MoveJog (Desplazamiento del eje en modo Jog)

Tabla 10- 42 Instrucción MC_MoveJog

KOP / FUP	SCL	Descripción

	`"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_);`
		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.
		Para utilizar la instrucción MC_MoveJog primero es necesario haber habilitado el eje.

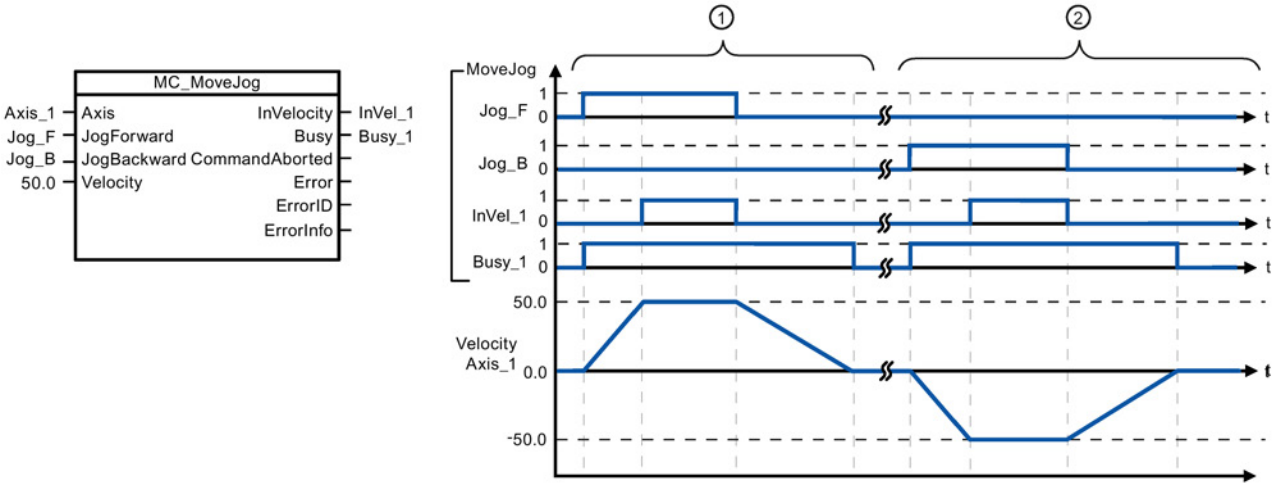
[1] STEP 7 crea el DB automáticamente al insertar la instrucción.

[2] En el ejemplo SCL, "MC_MoveJog_DB" es el nombre del DB de instancia.

Tabla 10- 43 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

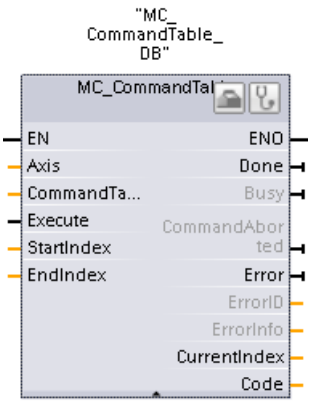
La tarea MC_MoveJog nueva interrumpe las siguientes tareas de Motion Control activas:

- MC_Home Mode = 3
- MC_Halt
- MC_MoveAbsolute
- MC_MoveRelative
- MC_MoveVelocity
- MC_MoveJog

10.7.10 Instrucción MC_CommandTable (Ejecutar comandos de eje como secuencia de movimientos)

Tabla 10- 44 Instrucción MC_CommandTable

KOP / FUP	SCL	Descripción

	`"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_);`
		Ejecuta una serie de movimientos individuales para un eje controlado por motor que se combinan en una secuencia de movimientos.
		Los movimientos individuales están configurados en la tabla de comandos de un objeto tecnológico para la salida del tren de impulsos (TO_CommandTable_PTO).

[1] STEP 7 crea el DB automáticamente al insertar la instrucción.

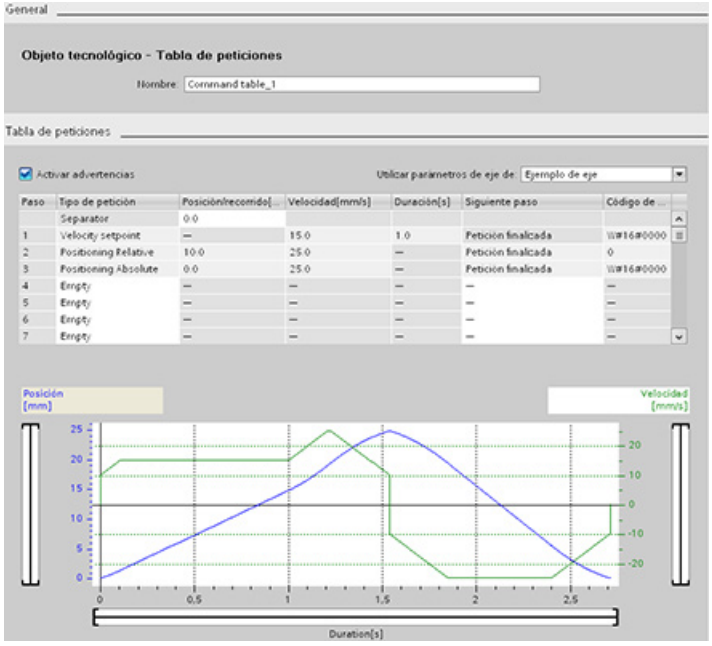
[2] En el ejemplo SCL, "MC_CommandTable_DB" es el nombre del DB de instancia.

Tabla 10- 45 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 ≤ **StartIndex** ≤ EndIndex
EndIndex	IN	Int	32	Finalizar el procesamiento de la tabla de comandos con este paso Límites: StartIndex ≤ **EndIndex** ≤ 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

Parámetro y tipo		Tipo de datos	Valor inicial	Descripción
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 10- 46 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	Detener eje. Nota: El comando solo 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

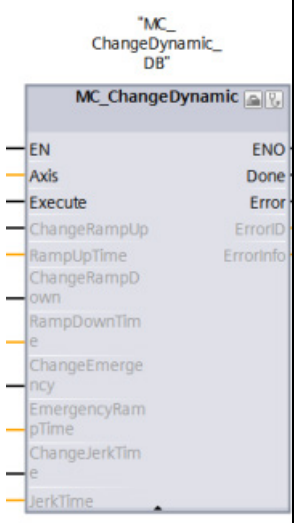
La tarea MC_CommandTable nueva interrumpe las siguientes tareas de Motion Control activas:

- 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".

10.7.11 Instrucción MC_ChangeDynamic (Cambiar la configuración de la dinámica del eje)

Tabla 10- 47 Instrucción MC_ChangeDynamic

KOP / FUP	SCL	Descripción

	`"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_);`
		Modifica los ajustes dinámicos de un eje de control de movimiento:
		• 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 10- 48 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.

Parámetro y tipo		Tipo de datos	Descripción
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".

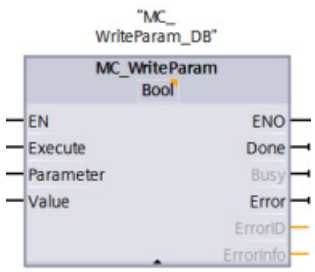
Asegúrese de mantener los parámetros MC_ChangeDynamic dentro de los límites de la configuración dinámica para el objeto tecnológico Eje.

10.7.12 Instrucción MC_WriteParam (Escribir parámetros de un objeto tecnológico)

La instrucción MC_WriteParam se usa para escribir un número determinado de parámetros para cambiar la funcionalidad del eje desde el programa del usuario.

Tabla 10- 49 Instrucción MC_WriteParam

KOP / FUP	SCL	Descripción

	`"MC_WriteParam_DB"(`
	`Parameter:=_variant_in_,`
	`Value:=_variant_in_,`
	`Execute:=_bool_in_,`
	`Done:=_bool_out_,`
	`Error:=_real_out_,`
	`ErrorID:=_word_out_,`
	`ErrorInfo:=_word_out_);`
		La instrucción MC_WriteParam se utiliza para escribir en parámetros públicos (por ejemplo, los valores de DB de aceleración y usuario).

1 STEP 7 crea el DB automáticamente al insertar la instrucción.

2 En el ejemplo de SCL, "MC_WriteParam_DB" es el nombre del DB de instancia.

Se puede escribir en los parámetros que son públicos. No puede escribir en "MotionStatus" ni en "StatusBits". Los parámetros válidos se listan en la tabla siguiente:

Nombre de parámetro escribible	Nombre de parámetro escribible
Actor.InverseDirection	DynamicDefaults.Acceleration
Actor.DirectionMode	DynamicDefaults.Deceleration
Actor.DriveParameter.PulsesPerDriveRevolution	DynamicDefaults.Jerk
Sensor[1].ActiveHoming.Mode	DynamicDefaults.EmergencyDeceleration
Sensor[1].ActiveHoming.SideInput	PositionLimitsHW.Active
Sensor[1].ActiveHoming.Offset	PositionLimitsHW.MaxSwitchedLevel
Sensor[1].ActiveHoming.SwitchedLevel	PositionLimitsHW.MinSwitchedLevel
Sensor[1].PassiveHoming.Mode	PositionLimitsSW.Active
Sensor[1].PassiveHoming.SideInput	PositionLimitsSW.MinPosition
Sensor[1].PassiveHoming.SwitchedLevel	PositionLimitsSW.MaxPosition
Units.LengthUnit	Homing.AutoReversal
Mechanics.LeadScrew	Homing.ApproachDirection
DynamicLimits.MinVelocity	Homing.ApproachVelocity
DynamicLimits.MaxVelocity	Homing.ReferencingVelocity

Tabla 10- 50 Parámetros de la instrucción MC_WriteParam

Parámetro y tipo		Tipo de datos	Descripción
PARAMNAME	IN	Variant	Nombre del parámetro en el que se escribe el valor.
VALUE	IN	Variant	Valor que se escribe en el parámetro asignado.
EXECUTE	IN	Bool	Inicia la instrucción. Valor predeterminado: FALSE
DONE	OUT	Bool	Se ha escrito el valor. Valor predeterminado: FALSE
BUSY	OUT	Bool	Si es TRUE, la instrucción está funcionando. Valor predeterminado: FALSE
ERROR	OUT	Real	Si es TRUE, se produce un error. Valor predeterminado: FALSE
ERRORID	OUT	Word	ID del error
ERRORINFO	OUT	Word	Información relacionada con ERRORID

Tabla 10- 51 Códigos de condición de ERRORID y ERRORINFO

ERRORID (W#16#...)	ERRORINFO (W#16#...)	Descripción
0	0	Se ha cambiado correctamente un parámetro TO-DB de eje.
8410[1]	0028[1]	Establece un parámetro no válido (parámetros de TO-DB de eje de longitud incorrecta).
8410[1]	0029[1]	Establece un parámetro no válido (sin parámetro TO-DB de eje).
8410[1]	002B[1]	Establece un parámetro no válido (sin parámetro TO-DB de eje)
8410[1]	002C[1]	Establece un parámetro válido, pero el eje no está desactivado.
Config Error[2]	Config Error[2]	Establece un parámetro válido (parámetro TO-DB de eje público de solo lectura) fuera de rango.
Config Error[3]	Config Error[3]	Establece un parámetro válido (parámetro TO-DB de eje público) fuera de rango.

[1] Error en MC_WriteParam

[2] Error en MC_Power

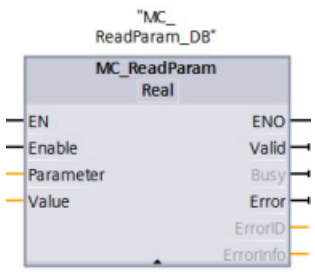
[3] Error en MC_Power y MC_MoveXXX o MC_CommandTable

10.7.13 Instrucción MC_ReadParam (leer parámetros de un objeto tecnológico)

La instrucción MC_ReadParam se usa para leer un número determinado de parámetros que indican la posición y la velocidad (entre otros valores actuales) del eje definidos en la entrada del eje.

Tabla 10- 52 Instrucción MC_ReadParam

KOP / FUP	SCL	Descripción

	`"MC_ReadParam_DB"(`
	`Enable:=_bool_in_,`
	`Parameter:=_variant_in_,`
	`Value:=_variant_in_out_,`
	`Valid:=_bool_out_,`
	`Busy:=_bool_out_,`
	`Error:=_real_out_,`
	`ErrorID:=_word_out_,`
	`ErrorInfo:=_word_out_);`
		La instrucción MC_ReadParam se utiliza para leer valores de estado únicos, al margen del punto de control del ciclo.

1 STEP 7 crea el DB automáticamente al insertar la instrucción.

2 En el ejemplo SCL, "MC_ReadParam_DB" es el nombre del DB de instancia.

La instrucción MC_ReadParam funciona con un comportamiento de activación. Siempre y cuando la entrada "Enable" sea TRUE la instrucción leerá el "parámetro" especificado en la ubicación de "valor".

El valor "Position" de "MotionStatus" actualiza cada punto de control del ciclo (CCP) en función del valor de HSC actual.

El valor "Velocity" de "MotionStatus" es comando de velocidad al final del segmento actual (actualizado ~10 ms). El parámetro MC_ReadParam también puede leer este valor.

Si se produce un error, la instrucción cambia a un estado de error que solo se puede inicializar con un nuevo flanco ascendente en la entrada "Enable".

Tabla 10- 53 Parámetros de la instrucción MC_ReadParam

Parámetro y tipo		Tipo de datos	Descripción
ENABLE	IN	Bool	Inicia la instrucción. Valor predeterminado: FALSE
PARAMETER	IN	Variant	Puntero al parámetro TO que se va a leer.
VALID	OUT	Bool	Si es TRUE, se habrá leído el valor. Valor predeterminado: FALSE
BUSY	OUT	Bool	Si es TRUE, la instrucción está funcionando. Valor predeterminado: FALSE
ERROR	OUT	Real	Si es TRUE, se produce un error. Valor predeterminado: FALSE
ERRORID	OUT	Word	ID del error. Valor predeterminado: 0
ERRORINFO	OUT	Word	Información relacionada con ERRORID. . Valor predeterminado: 0
VALUE	INOUT	Variant	Puntero a la ubicación donde se almacena el valor leído.

Tabla 10- 54 Códigos de condición de ERRORID y ERRORINFO

ERRORID (W#16#...)	ERRORINFO (W#16#...)	Descripción
0	0	Lectura correcta de un parámetro
8410	0028	Parámetro no válido (longitud incorrecta)
8410	0029	Parámetro no válido (no TO-DB)
8410	0030	Parámetro no válido (no legible)
8411	0032	Parámetro no válido (valor incorrecto)

Parámetros TO

El eje "MotionStatus" consta de cuatro valores. Puede supervisar los cambios en estos valores, que pueden leerse mientras se ejecuta el programa:

Nombre de variable	Tipo de datos	Legibles mediante MC_ReadParam
MotionStatus:	Estructura	No
• Posición	REAL	Sí
• Velocidad	REAL	Sí
• Distancia	REAL	Sí
• TargetPosition	REAL	Sí

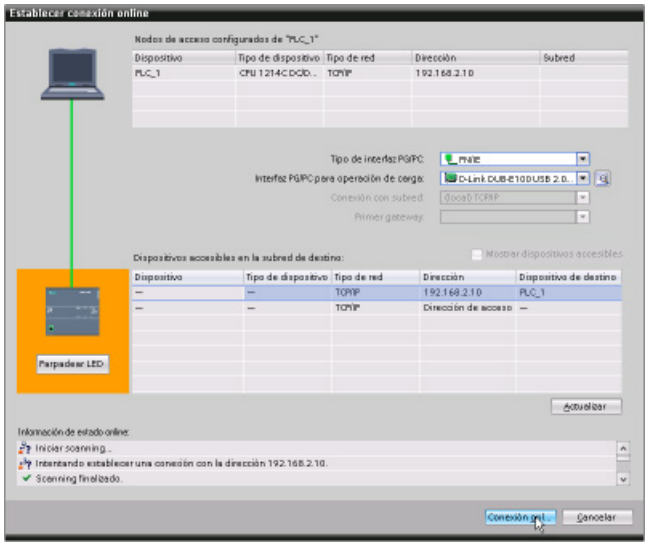
Facilidad de uso de las herramientas online 11

11.1 Establecer una conexión online con una CPU

Debe establecerse 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 346)
- Visualizar y ajustar la fecha y hora de la CPU (Página 356)
- Visualizar la información del módulo
- Comparar y sincronizar (Página 355) bloques de programa de offline a online
- Cargar y descargar bloques de programa
- Mostrar diagnóstico y el búfer de diagnóstico (Página 356)
- Usar una tabla de observación (Página 348) para probar el programa de usuario vigilando y modificando valores
- Utilizar una tabla de forzado permanente para forzar valores en la CPU (Página 350)

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.

Se ha conectado la programadora 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.

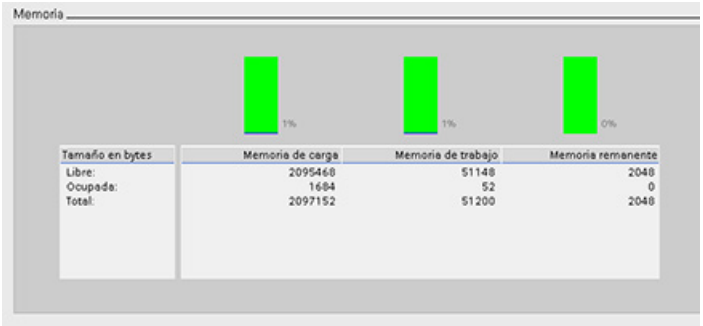
11.2 Interactuar con la CPU online

La Task Card "Herramientas online" de la vista del proyecto muestra un panel de operador en el que se indica el estado operativo de la CPU online. El panel también permite cambiar el estado operativo de la CPU. Utilice el botón del panel de operador para cambiar el estado operativo (STOP o RUN). El panel de operador también dispone de un botón MRES para resetear la memoria.

El color del indicador RUN/STOP muestra el estado operativo actual de la CPU: Amarillo indica STOP y verde RUN.

Para utilizar el panel de operador es necesario establecer una conexión online entre STEP 7 y la CPU. Una vez seleccionada la CPU en la configuración de dispositivos o visualizando un bloque lógico de la CPU online es posible abrir el panel de operador desde la Task Card "Herramientas online".

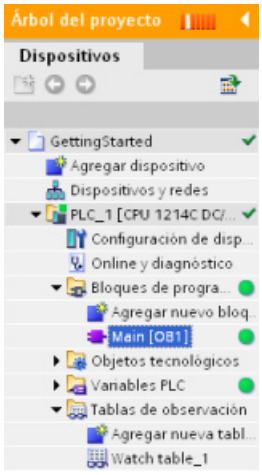
Es posible vigilar el tiempo de ciclo de una CPU online.



También es posible ver la carga de memoria de la CPU.

11.3 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.

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.

El campo "Valor de observación" muestra el valor de cada variable.

11.4 Visualización sencilla del estado del programa de usuario

También es posible observar el estado de un máximo de 50 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.

11.5 Utilizar una tabla de observación para vigilar la CPU

Una tabla de observación permite vigilar y forzar datos a medida que la CPU ejecuta el programa. Los datos pueden ser entradas (I), salidas (Q), marcas, DB o entradas de la periferia (como p. ej., "On:P", "I 3.4:P" o "Q3.4:P"). No es posible monitorizar con precisión las salidas físicas (como p. ej., Q0.0:P) debido a que la función de observación sólo puede visualizar el último valor escrito de 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. También es posible utilizar la función "Forzar" para probar la ejecución del programa de usuario.

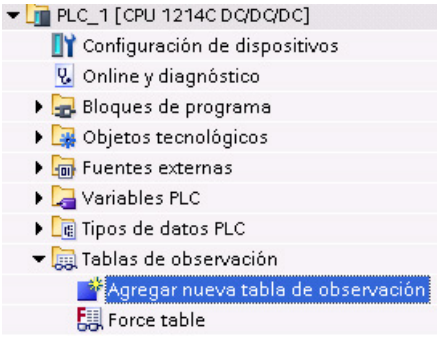
	Nombre	Dirección	Formato visualización	Valor de observación	Observar con disparo	Forzar con disparador	Valor de forzado
1	"Start"	%I0.0	Bool		Permanente	Permanente	
2	"Stop"	%I0.1	Bool		Permanente	Permanente	
3	"Running"	%M0.0	Bool		Permanente	Permanente	

Nota

Las entradas y salidas digitales utilizadas por el contador rápido (HSC) y los dispositivos con modulación del ancho de pulso (PWM) y tren de impulsos (PTO) se asignan en la configuración de dispositivos. Al asignar las direcciones de las entradas y salidas digitales a los dispositivos anteriores, los valores de las direcciones E/S asignadas no pueden modificarse utilizando la función "Forzar permanentemente" de la tabla de observación.

Una tabla de observación permite monitorizar y forzar individualmente los valores de las diferentes variables, seleccionando para ello una de las siguientes opciones:

- Al principio o al final del ciclo
- Cuando la CPU cambia al estado operativo STOP
- "Permanentemente" (el valor no es reseteado tras una transición de STOP a RUN)



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.

Para forzar las variables debe existir una conexión online con la CPU. Las siguientes opciones están disponibles para forzar variables:

- "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" permite activar las salidas de periferia cuando la CPU se encuentra en estado operativo STOP. Esta función es útil para probar el cableado de los módulos de salida.

Las distintas funciones pueden seleccionarse mediante los botones situados en la parte 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".

11.6 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).

---

	i	Nombre	Dirección	Formato visualización	Valor de observación	Valor de forzado permanente	F
1		"On":P	%I0.0:P	Bool		TRUE	☑
2		"Off":P	%I0.1:P	Bool			☐
3		"Run":P	%Q0.1:P	Bool			☐

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.

Nota

Cuando 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.

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.

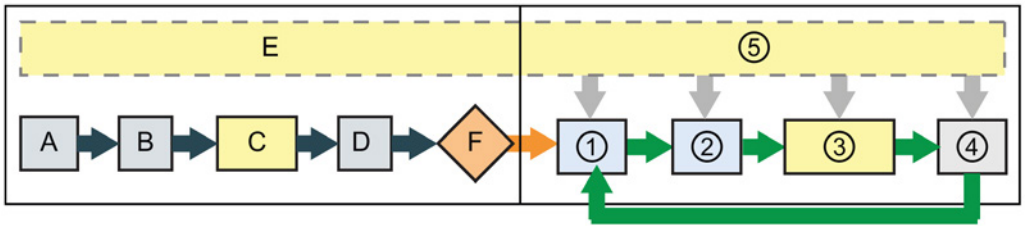
Cuando una entrada o salida se fuerza permanentemente en la tabla de forzado permanente, las acciones de forzado permanente se convierten en parte del programa de usuario. Aunque se haya cerrado el software de programación, las selecciones de forzado permanente permanecen activadas en el programa de 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 forzado.



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.

11.7 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.)
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

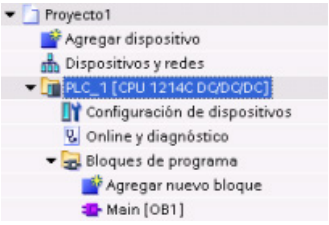
Los valores que se muestran en la columna "Valor de observación" son siempre copiados de la CPU. STEP 7 no comprueba si todos los valores provienen del mismo ciclo de la CPU.

11.8 Carga de elementos del proyecto

También se pueden copiar bloques de programa de una CPU online o una Memory Card conectada a la programadora.

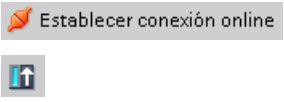
Prepare el proyecto offline para los bloques de programa copiados:

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".

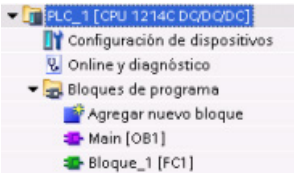


Para cargar los bloques de programa desde la CPU online al proyecto offline, siga estos pasos:

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 345).



Una vez se ha completado la carga, STEP 7 muestra todos los bloques de programa cargados en el proyecto.

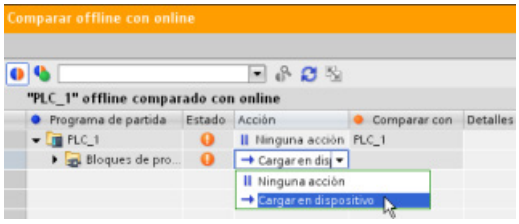


11.9 Comparar CPUs online y offline

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.

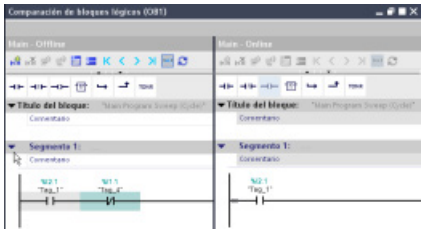
Seleccione la CPU del proyecto.

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.



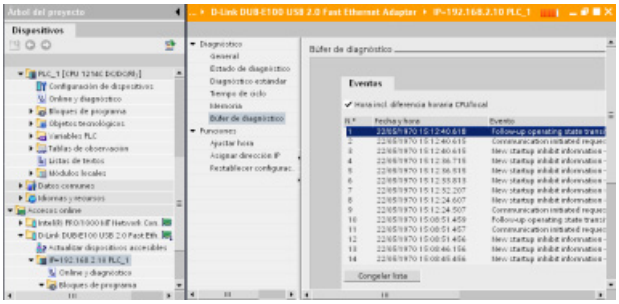
Haga clic con el botón derecho del ratón en un objeto en la columna "Comparar con" y seleccione el botón "Iniciar comparación detallada" para mostrar los bloques lógicos uno junto a otro.

La comparación detallada destaca las diferencias entre los bloques lógicos de la CPU online y los de la CPU del proyecto.

11.10 Visualizar los eventos de diagnóstico

La CPU dispone de un búfer de diagnóstico que contiene una entrada para cada evento de diagnóstico, como p. ej. un cambio en el estado operativo de la CPU o errores detectados por la CPU o los módulos. Para acceder al búfer de diagnóstico es preciso estar online.

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.



Estando conectada la alimentación de la CPU, los 50 eventos más recientes están disponibles en este búfer. Cuando se llena el búfer, un evento nuevo reemplaza al evento más antiguo.

Cuando se corta la alimentación, se almacenan los diez eventos más recientes.

11.11 Ajustar la dirección IP y la hora

Es posible ajustar la dirección IP 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.

11.12 Restablecimiento de los ajustes de fábrica

Se puede restablecer un S7-1200 a sus ajustes originales de fábrica bajo las siguientes condiciones:

- 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 "Borrar 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 restablece los ajustes de fábrica. La CPU realiza las siguientes acciones:

Con tarjeta de memoria instalada en la CPU	Sin tarjeta de memoria instalada en la CPU
• Borra el búfer de diagnóstico • Resetea la hora • Restaura la memoria de trabajo desde la tarjeta de memoria • Ajusta todas las áreas de operandos a los valores de arranque configurados • Ajusta todos los parámetros a sus valores configurados • Conserva o borra la dirección IP en función de la selección que haya hecho. (La dirección MAC es fija y nunca se cambia.)[1] • Borra el registro de control, si lo hay	• Borra el búfer de diagnóstico • Resetea la hora • Borra la memoria de trabajo y la memoria de carga interna • Ajusta todas las áreas de operandos a los valores de arranque configurados • Ajusta todos los parámetros a sus valores configurados • Conserva o borra la dirección IP en función de la selección que haya hecho. (La dirección MAC es fija y nunca se cambia.)[1] • Borra el registro de control, si lo hay

Con tarjeta de memoria instalada en la CPU	Sin tarjeta de memoria instalada en la CPU

[1] Si ha seleccionado "Conservar dirección IP", la CPU ajusta la dirección IP, máscara de subred y la dirección de router (si se utiliza) a los ajustes de la configuración de hardware, a no ser que haya modificado esos valores en el programa de usuario o con otra herramienta, en cuyo caso la CPU restaura los valores modificados.

11.13 Actualización del firmware

Puede actualizar el firmware de la CPU conectada desde las herramientas online y de diagnóstico de STEP 7.

Para realizar una actualización de firmware, proceda del siguiente modo:

1. Abra la vista Online y diagnóstico de la CPU conectada.
2. Seleccione "Actualización de firmware" en la carpeta "Funciones".
3. Haga clic en el botón Examinar y vaya hasta la ubicación que contiene el archivo de actualización del firmware. Esta podría ser una ubicación de su disco duro en la que ha descargado un archivo de actualización del firmware de S7-1200 (http://support.automation.siemens.com/WW/view/es/34612486/133100) del sitio web de Service & Support (http://www.siemens.com/automation/).
4. Seleccione un archivo que sea compatible con el módulo. En el caso de un archivo seleccionado, la tabla muestra los módulos compatibles.
5. Haga clic en el botón "Iniciar actualización". Siga los diálogos, si es necesario, para cambiar el estado operativo de la CPU.

STEP 7 muestra los diálogos de progreso mientras carga la actualización del firmware. Cuando haya finalizado, le indicará que arranque el módulo con el nuevo firmware.

Nota

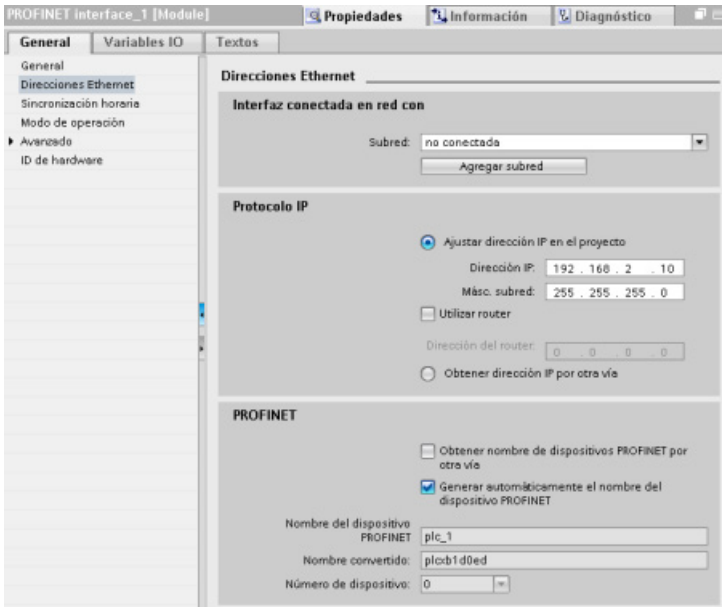
Si no elige arrancar el módulo con el nuevo firmware, el firmware anterior permanecerá activo hasta que inicialice el módulo, por ejemplo desconectando y conectando la alimentación. El nuevo firmware solo pasará a estar activo después de inicializar el módulo.

También puede realizar una actualización de firmware siguiendo uno de los métodos adicionales siguientes:

- Utilizar una Memory Card (Página 65)
- Utilizar la página web estándar "Información del módulo" del servidor web (Página 262)

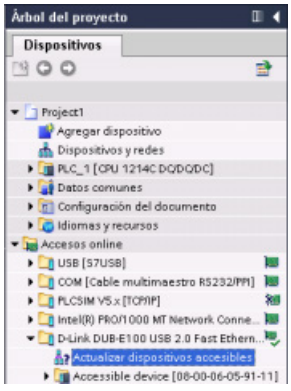
11.14 Descargar una dirección IP permanente en una CPU online

Para asignar una dirección IP, proceda del siguiente modo:



- Configure la dirección IP de la CPU (Página 89)
- Guarde y descargue la configuración en la CPU.

La dirección IP y la máscara de subred de la CPU deben ser compatibles con la dirección IP y la máscara de subred de la programadora. Consulte a su especialista en redes la dirección IP y la máscara de subred de su CPU.



Si la CPU no se ha configurado anteriormente, es posible utilizar también "Accesos online" para establecer la dirección IP.

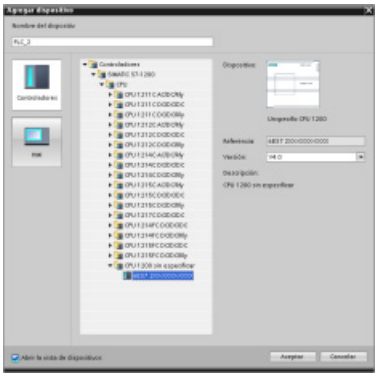
Si con la configuración del dispositivo se ha descargado una dirección IP, esta no se perderá al desconectar y volver a conectar la alimentación del PLC.

Una vez que se ha descargado la configuración del dispositivo, se puede ver la dirección IP en la carpeta "Accesos online".

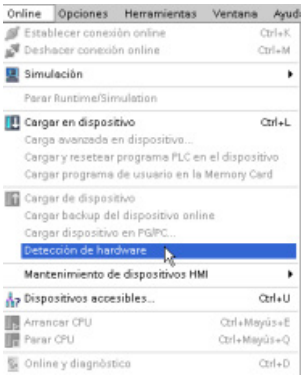
11.15 Utilizar la "CPU sin especificar" para cargar la configuración hardware

Si se dispone de una CPU física que puede conectarse a una programadora, la configuración hardware se carga fácilmente.

En primer lugar, conecte la CPU a la programadora y cree un proyecto nuevo.



En la configuración de dispositivos (vista del proyecto o vista del portal), agregue un dispositivo nuevo, pero seleccione la "CPU sin especificar" en lugar de una específica. STEP 7 crea una CPU sin especificar.



Una vez creada la CPU sin especificar se puede cargar la configuración hardware desde la CPU online.

- En el editor de programación, seleccione el comando "Detección de hardware" del menú "Online".
- En el editor de configuración de dispositivos, seleccione la opción para detectar la configuración del dispositivo conectado.

Tras seleccionar la CPU en el cuadro de diálogo online, STEP 7 carga la configuración de hardware de la CPU, incluidos todos los módulos (SM, SB o CM). La dirección IP **no** está cargada. Vaya a "Configuración de dispositivos" para configurar la dirección IP manualmente.

11.16 Cargar 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

Riesgos asociados con la carga en estado operativo RUN

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, borrar y sobrescribir bloques de datos (DB) y bloques de datos de instancia para bloques de función (FB). Puede añadirlos a las estructuras de los DB y cargarlos en estado operativo RUN. La CPU puede conservar los valores de las variables de bloques existentes e inicializar las nuevas variables de bloques de datos a sus valores iniciales o la CPU puede establecer las variables de bloques de datos en los valores iniciales en función de los ajustes de configuración. 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.

Puede cargar como máximo veinte bloques a la vez en estado operativo RUN. Si debe cargar más de veinte bloques, debe pasar la CPU al estado operativo 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

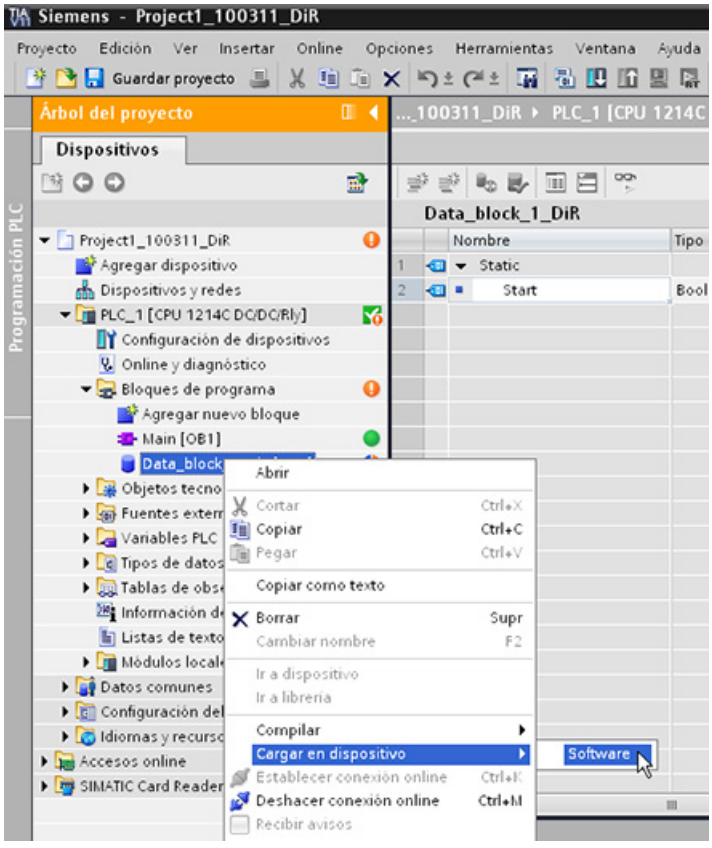
Si la CPU está en estado operativo RUN y se realizan cambios en el programa, STEP 7 intentará siempre cargar primero en RUN. Si no desea que esto suceda, deberá poner la CPU en estado operativo STOP.

Si los cambios realizados no se soportan en "Cargar en RUN", STEP 7 pedirá al usuario que ponga la CPU a STOP.

11.16.1 Modificar el programa en el estado operativo RUN

Para cambiar el programa en estado operativo RUN, en primer lugar debe asegurarse de que la CPU y el programa cumplan los requisitos previos y luego proceder del siguiente modo.

1. Para descargar el programa en RUN, seleccione uno de los métodos siguientes:
   - Seleccione el comando "Cargar en dispositivo" en el menú "Online".
   - Haga clic en el 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".



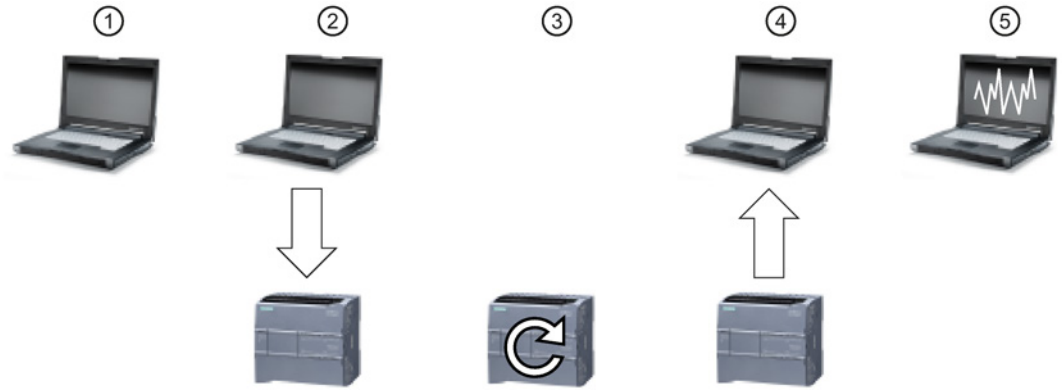
   Si el programa se compila correctamente, STEP 7 empieza a cargar el programa en la CPU.

2. Cuando STEP 7 le solicite que cargue el programa o cancele el funcionamiento, haga clic en "Cargar" para cargar el programa en la CPU.

11.17 Trazas y registros de datos de la CPU en condiciones de disparo

STEP 7 proporciona funciones de analizador lógico y de trazas que le permitirán configurar variables para que el PLC pueda trazarlas o registrarlas. A continuación, puede cargar los datos de seguimiento registrados en el dispositivo de programación y usar las herramientas de STEP 7 para analizar y administrar los datos y representarlos gráficamente. La carpeta Trazas del árbol del proyecto STEP 7 se usa para crear y administrar trazas.

En la figura siguiente se muestran los diferentes pasos de la función de trazas:



① Configure el seguimiento en el editor de trazas de STEP 7. Puede configurar los valores de datos que se registrarán, la duración del registro, la frecuencia de registro y la condición del disparo.

② Transfiera la configuración de trazas de STEP 7 al PLC.

③ El PLC ejecuta el programa y, cuando se produce la condición de disparo, empieza a registrar los datos de trazas.

④ Transfiera los valores registrados del PLC a STEP 7.

⑤ Use las herramientas de STEP 7 para analizar los datos, visualizarlos gráficamente y guardarlos.

El tamaño máximo de una traza son 512 KB por traza.

Acceso a ejemplos

Consulte el sistema de información de STEP 7 para obtener información detallada sobre cómo programar trazas, cargar la configuración, cargar los datos de traza y visualizarlos en el analizador lógico. Es posible consultar ejemplos detallados en el capítulo "Uso de funciones online y diagnóstico > Uso de funciones de traza y analizador lógico".

Asimismo, el manual online "Industry Automation SIMATIC/SINAMICS Uso de la función Trace y de analizador lógico" (http://support.automation.siemens.com/WW/view/es/64897128) también es una referencia excelente.

IO-Link es fácil 12

12.1 Sinopsis de la tecnología IO-Link

IO-Link es una innovadora tecnología de comunicación para sensores y actuadores definida por la Organización de usuarios de PROFIBUS (PNO). IO-Link es un estándar internacional conforme a IEC 61131-9. Se basa en una conexión punto a punto entre los sensores y actuadores (esclavos) y el controlador (maestro). Por ello, no representa un sistema de bus, sino que es una conexión punto a punto convencional más desarrollada.

Además de los datos de operación cíclicos, los sensores y actuadores conectados transmiten un gran número de parámetros y datos de diagnóstico. El mismo cable de conexión a 3 hilos que se utiliza para la tecnología de sensor estándar se utiliza para la transmisión de datos.

12.2 Componentes de un sistema IO-Link

Un sistema IO-Link se compone de dispositivos IO-Link (generalmente sensores, actuadores o combinaciones de ambos), un cable a 3 hilos estándar de sensor/actuador y un maestro IO-Link. El maestro puede ser un dispositivo con cualquier diseño y grado de protección.

Un maestro IO-Link puede tener uno o más puertos. El SM 1278 4xIO-Link Master tiene cuatro puertos. A cada puerto puede conectarse un dispositivo IO-Link o un sensor o actuador estándar. IO-Link es un sistema de comunicación punto a punto.

12.3 Después del arranque

Al arrancar, el dispositivo IO-Link siempre está en modo SIO (modo I/O estándar). Los puertos del maestro pueden tener configuraciones diferentes. Consulte el capítulo dedicado a IO-Link en el manual de sistema del controlador programable *S7-1200* para más detalles.

Si un puerto está ajustado al modo SIO, en ese puerto el maestro funcionará como una entrada digital normal. Si el puerto está ajustado al modo IO-Link (modo de comunicación), el maestro intenta encontrar el dispositivo IO-Link conectado. Este proceso se conoce como "despertar".

Durante el proceso de despertar, el maestro envía una señal definida y espera a que el dispositivo esclavo responda. Inicialmente, el maestro intenta hacerlo con la máxima velocidad posible. Si no puede, el maestro intenta la siguiente velocidad más baja. El maestro intenta direccionar el dispositivo tres veces con cada velocidad. El dispositivo siempre soporta una única velocidad determinada. Si el maestro recibe una respuesta (es decir, si el dispositivo se ha despertado), ambos inician una comunicación. Primero intercambian los parámetros de comunicación y a continuación inician el intercambio cíclico de datos del proceso.

Si el dispositivo esclavo se retira durante el funcionamiento, el maestro detecta la interrupción de la comunicación, lo notifica al controlador con especificidad de bus de campo e intenta volver a despertar el dispositivo cíclicamente. Tras un nuevo despertar correcto, los parámetros de comunicación se leen de nuevo, se validan si procede, y a continuación se inicia de nuevo el canal de comunicación cíclica.

12.4 Protocolo IO-Link

El sistema IO-Link puede intercambiar tres tipos de datos:

- Datos de proceso cíclicos (entradas y salidas de datos de proceso) → datos cíclicos
- Parámetros de dispositivo (objetos de datos según demanda) → datos acíclicos
- Eventos → datos acíclicos

El dispositivo IO-Link solo envía datos si el maestro IO-Link lo solicita. Los datos de proceso se envían tras el frame IDLE del maestro, y el maestro solicita explícitamente parámetros de datos y eventos del dispositivo.

12.5 Configuración en el bus de campo

El maestro IO-Link aparece en el bus de campo como nodo normal de bus de campo y se integra a través de una descripción de dispositivo adecuada en el configurador de red correspondiente. Estos archivos describen las propiedades de comunicación y otras propiedades del maestro IO-Link, como p. ej. el número de puertos. No indican qué dispositivos IO-Link están conectados.

Sin embargo, la descripción de dispositivo IO-Link (IODD = IO-Link Device Description) se ha definido para una representación completamente transparente de la arquitectura del sistema hasta el dispositivo IO-Link. Con ayuda de la IODD y la herramienta de configuración de IO-Link S7-PCT se puede configurar qué dispositivo IO-Link se conecta a qué puerto del maestro IO-Link.

Consulte el sistema de ayuda S7-PCT y el *S7-1200 Programmable Controller System Manual* para conocer información detallada sobre la configuración.

12.6 IO-Link y el programa STEP 7

El maestro IO-Link programa una comunicación acíclica con un dispositivo IO-Link mediante el bloque de función (FB) IOL_CALL en el programa del controlador STEP 7 S7-1200. El FB IOL_CALL indica qué maestro IO-Link utiliza el programa y qué puertos utiliza el maestro para el intercambio de datos.

Visite la página web Siemens Industry Online Support (http://support.automation.siemens.com) para conocer más detalles sobre cómo trabajar con el FB IOL_CALL. Introduzca "IO-Link" en el campo de búsqueda de la web para acceder a información sobre los productos IO-Link y su uso.

12.7 El SM 1278 4 maestros IO-Link

El SM 1278 4xIO-Link Master es un módulo de 4 puertos que funciona como un módulo de señales y como un módulo de comunicación. Cada puerto puede funcionar en el modo IO-Link, entrada digital de 24 V DC o salida digital de 24 V DC únicas. Se pueden conectar hasta cuatro dispositivos IO-Link (conexión a 3 hilos) o cuatro actuadores o encoders estándar.

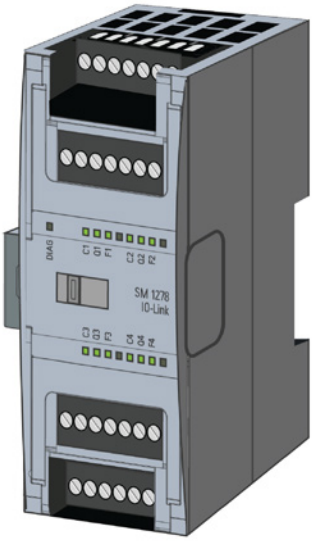
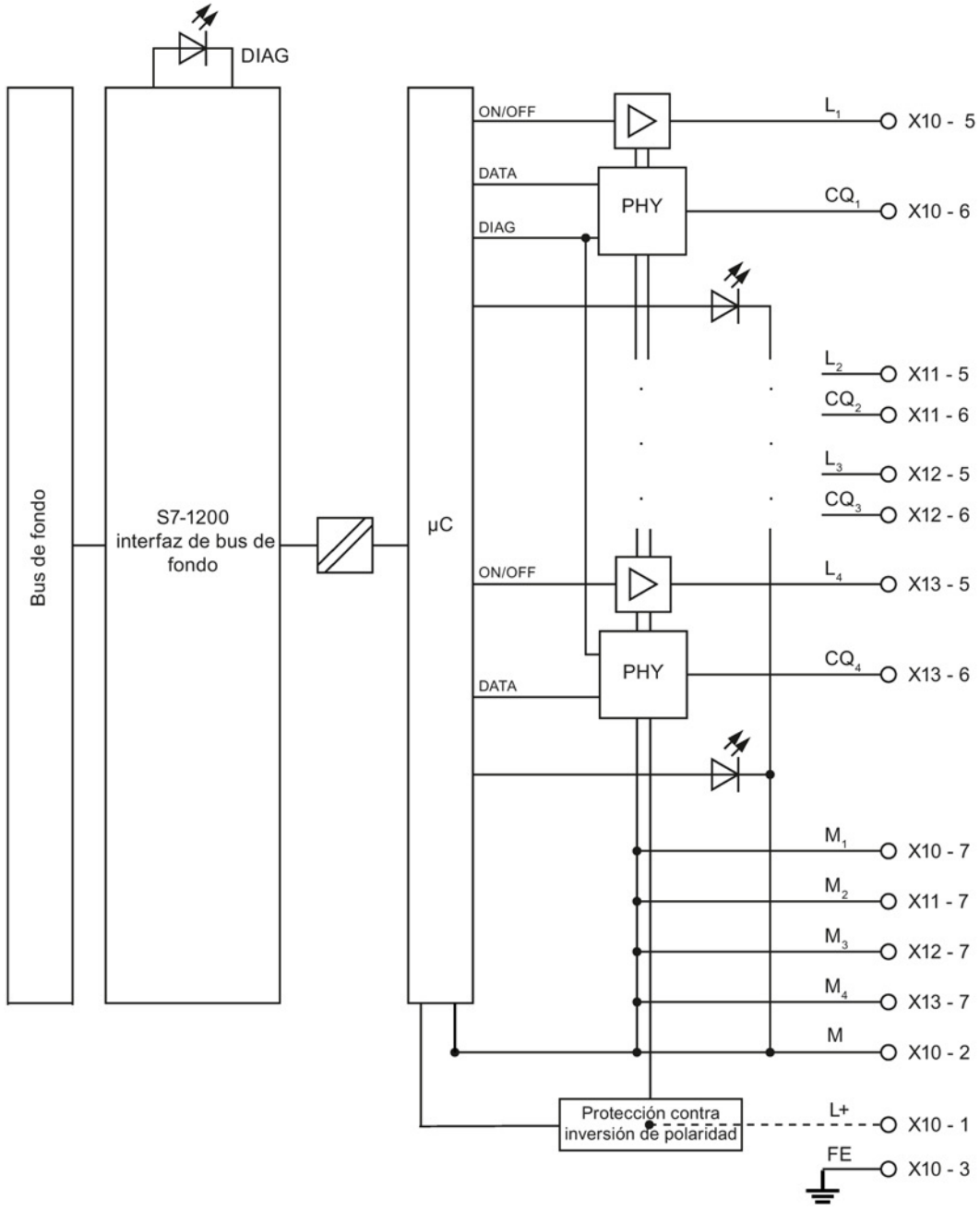
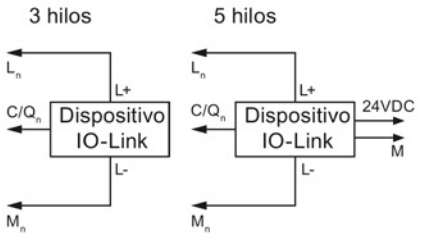


Diagrama de bloques del SM 1278 4 maestros IO-Link

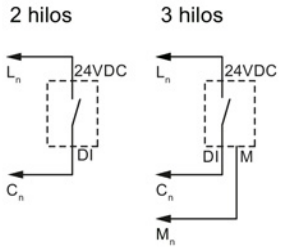


Ejemplos de conexión

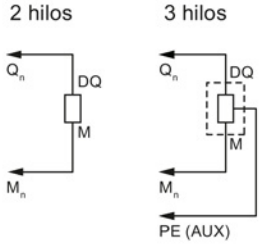
La siguiente ilustración muestra la configuración para el estado operativo del IO-Link (a 3 y a 5 hilos), siendo n = número de puerto:



La siguiente ilustración muestra la configuración para el estado operativo DI (a 2 y a 3 hilos), siendo n = número de puerto:



La siguiente ilustración muestra la configuración para el estado operativo DQ (a 2 y a 3 hilos), siendo n = número de puerto:



Información detallada sobre el uso y la configuración del SM 1278 4 maestros IO-Link

Para más información sobre el SM 1278 4xIO-Link Master, incluidos diagramas, conexión, parametrización, alarmas de diagnóstico, etc., consulte el manual de sistema del controlador programable *S7-1200*.

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. Es responsabilidad del usuario 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+A1:2011: 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 60079-15:2010: Tipo de protección 'n'

La Declaración de conformidad CE se encuentra a disposición de las autoridades competentes en:

Siemens AG
Sector Industry
I IA AS FA DH AMB
Postfach 1963
D-92209 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)

Nota

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:
Clase I, División 2, Grupo de gas A, B, C, D, Clase de temperatura T3C Ta = 60 °C
Clase I, Zona 2, IIC, Clase de temperatura T3 Ta = 60 °C
Clase Canadiense I, Instalación en zona 2 según CEC 18-150

EXCEPCIÓN IMPORTANTE: Consulte en las especificaciones técnicas el número de entradas y salidas que pueden estar activadas simultáneamente. Algunos modelos tienen características reducidas para Ta = 60 °C.

⚠ ADVERTENCIA

Si se sustituyen componentes, podría perderse la idoneidad para Class I, Division 2 y Zone 2.

Solo un centro de asistencia Siemens autorizado puede reparar estas unidades.

Homologación IECEx

EN 60079-0: Atmósferas explosivas - Requisitos generales

EN60079-15: Material eléctrico para atmósferas potencialmente explosivas;
Tipo de protección 'nA'
ECEX FMG14.0012X
Ex nA IIC Tx Gc

En el producto puede indicarse información de clasificación IECEx con la información sobre ubicación peligrosa FM.

Solo están homologados los productos con clasificación IECEx. Contacte con el representante de Siemens más próximo para obtener una lista de las homologaciones actuales con las referencias respectivas.

Los modelos de tipo relé no están incluidos en la homologación IECEx.

Consulte la temperatura en el marcado específico del producto.

Monte los módulos en una envolvente adecuada que proporcione un grado de protección mínimo IP54 conforme a IEC 60079-15.

Homologación ATEX

La homologación ATEX solo es válida para los modelos DC. La homologación ATEX no es válida para los modelos AC y de relé.

EN 60079-0:2009: Atmósferas explosivas - Requisitos generales

EN 60079-15:2010: Material eléctrico para atmósferas de gas explosivas;
Tipo de protección 'nA'
II 3 G Ex nA IIC T4 o T3 Gc

Instale los módulos en una caja adecuada con un grado de protección mínimo de IP54 conforme a EN 60529, o en una ubicación que aporte un grado de protección equivalente.

Los cables y conductores conectados deben ser aptos para la temperatura real medida en condiciones nominales.

La instalación debe garantizar que las crestas de tensión estén limitadas a valores inferiores a 119 V. Consulte Inmunidad a ondas de choque en esta sección.

EXCEPCIÓN IMPORTANTE: Consulte en las especificaciones técnicas el número de entradas y salidas que pueden estar activadas simultáneamente. Algunos modelos tienen características reducidas para Ta = 60 °C.

Aprobación C-Tick

El S7-1200 cumple los requisitos de loes estándares según AS/NZS CISPR16 (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.

Homologación para la Unión Aduanera Euroasiática (Bielorrusia, Kazajstán, Federación Rusa)

EAC (Eurasion Conformity): Declaración de conformidad según el Reglamento Técnico de la Comunidad Económica (TR CU)

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)
- Korean Register of Shipping

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 de emisión	Requisitos de inmunidad	Requisitos respecto a la inmunidad a interferencias
Industrial	EN 61000-6-4:2007+A1:2011	EN 61000-6-2:2005	EN 61000-6-2:2005

Nota

El sistema de automatización S7-1200 está diseñado para ser usado en áreas industriales; el uso en zonas residenciales puede causar interferencias en la recepción de TV y radio. Si se usa el S7-1200 en zonas residenciales debe asegurarse de que la emisión de interferencias de radio cumple con el valor límite de clase B conforme a EN 55011.

Los ejemplos de medidas adecuadas para alcanzar un nivel de interferencia en frecuencias de radio de clase B incluyen:

- Instalación del S7-1200 en un armario de distribución puesto a tierra

- Uso de filtros de ruido en la líneas de suministro

Asegúrese de que las emisiones de interferencias de radio cumplen con la clase B conforme a EN 55011.

Se requiere recepción individual (el montaje final debe cumplir todos los requisitos de seguridad y CEM de una instalación en entorno residencial).

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 radiofre-cuencia 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, 1 kV en modo diferencial Sistemas DC - 2 kV en modo común, 1 kV en modo diferencial Para los sistemas DC, consulte Inmunidad a ondas de choque a continuación.
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 interrupcio-nes breves	Sistemas AC 0% durante 1 ciclo, 40% durante 12 ciclos y 70% durante 30 ciclos a 60 Hz

Inmunidad a ondas de choque

Los sistemas de cableado sujetos a ondas de choque por rayos deben estar equipados con protección externa. Una especificación para la evaluación de la protección frente a ondas de choque de tipo rayo se recoge en EN 61000-4-5, con límites operacionales establecidos en EN 61000-6-2. Las CPUs S7-1200 DC y los módulos de señales requieren protección externa para garantizar el funcionamiento seguro en caso de sobretensiones definidas en este estándar.

A continuación aparece una lista de algunos dispositivos que soportan la protección necesaria de inmunidad a ondas de choque. Estos dispositivos solo proporcionan protección si están correctamente instalados conforme a las recomendaciones del fabricante. También pueden utilizarse dispositivos de otros fabricantes con especificaciones iguales o mejores:

Tabla A- 3 Dispositivos que soportan la protección de inmunidad a sobretensiones

Subsistema	Dispositivo de protección
Tensión +24 V DC	BLITZDUCTOR VT, BVT AVD 24, referencia 918 422
Industrial Ethernet	DEHNpatch DPA M CLE RJ45B 48, referencia 929 121
RS-485	BLITZDUCTOR XT, Basic Unit BXT BAS, referencia 920 300
	BLITZDUCTOR XT, módulo BXT ML2 BD HFS 5, referencia 920 271
RS-232	BLITZDUCTOR XT, Basic Unit BXT BAS, referencia 920 300
	BLITZDUCTOR XT, módulo BXT ML2 BE S 12, referencia 920 222
Entradas digitales +24 V DC	DEHN, Inc., tipo DCO SD2 E 24, referencia 917 988
Salidas digitales +24 V DC y alimentación de sensores	DEHN, Inc., tipo DCO SD2 E 24, referencia 917 988
IO analógicas	DEHN, Inc., tipo DCO SD2 E 12, referencia 917 987
Salidas de relé	ninguna requerida

Tabla A- 4 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
	1 GHz a 3 GHz	< 76dB (µV/m) casi cresta; medido a 10 m

Condiciones ambientales

Tabla A- 5 Transporte y almacenamiento

Condiciones ambientales - Transporte y almacenamiento	
EN 60068-2-2, ensayo Bb, calor seco y EN 60068-2-1, ensayo Ab, frío	De -40 °C a +70 °C
EN 60068230, ensayo Dd, calor húmedo	De 25 °C a 55 °C, 95% de humedad
EN 60068-2-14, ensayo Na, choque de temperatura	De -40 °C a +70 °C, tiempo de permanencia 3 horas, 5 ciclos
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- 6 Condiciones de manejo

Condiciones ambientales - Funcionamiento	
Rango de temperatura ambiente (aire de entrada 25 mm bajo la unidad)	De -20 °C a 60 °C en montaje horizontal De -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	$S0_2$: < 0,5 ppm; H_2S: < 0,1 ppm; RH < 60% sin condensación
	ISA-S71.04 nivel de severidad G1, G2, G3
EN 60068214, ensayo Nb, cambio de temperatura	De 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

Tabla A- 7 Prueba de aislamiento a muy alta tensión

Prueba de aislamiento a muy alta tensión	
Circuitos nominales 24 V DC / 5 V DC	520 V DC (ensayo de tipo de límites de aislamiento óptico)
Circuitos a tierra 115 V AC / 230 V AC	1500 V AC
Circuitos 115 V AC / 230 V AC a circuitos 115 V AC / 230 V AC	1500 V AC
Circuitos 115 V AC / 230 V AC a circuitos 24 V DC / 5 V DC	1500 V AC (ensayo de tipo de 3000 V AC/4242 V DC)
Puerto Ethernet a circuitos de 24 V DC / 5 V DC y tierra[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- 8 Tensiones nominales

Tensión nominal	Tolerancia
24 V DC	De 20,4 V DC a 28,8 V DC
120/230 V AC	85 V AC a 264 V AC, 47 a 63 Hz

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 alimentación de sensores de la CPU, la entrada de alimentación de la bobina de relé de un SM o 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.

Salidas DC

No se suministra circuitería de protección contra cortocircuitos en las salidas DC de las CPU, módulos de señales (SM) y Signal Boards (SB).

Vida útil de los relés

La figura siguiente muestra los datos típicos de rendimiento estimados a partir de ensayos. El rendimiento real puede variar dependiendo de la aplicación. Un circuito de protección externo adaptado a la carga mejorará la vida útil de los contactos. Los contactos normalmente cerrados tienen una vida útil típica de aproximadamente un tercio de los contactos normalmente abiertos en condiciones inductivas y de carga de lámpara.

Un circuito de protección externo aumentará la vida útil de los contactos.

Tabla A- 9 Datos de rendimiento típicos

Datos para la selección de un actuador				
Corriente térmica continua		2 A máx.		
Capacidad de conmutación y vida útil de los contactos				
	Para carga óhmica	Tensión	Inten-sidad	Número de ciclos operativos (típico)
		24 V DC	2,0 A	0,1 millón

Datos para la selección de un actuador				
		24 V DC	1,0 A	0,2 millones
		24 V DC	0,5 A	1,0 millón
		48 V AC	1,5 A	1,5 millones
		60 V AC	1,5 A	1,5 millones
		120 V AC	2,0 A	1,0 millón
		120 V AC	1,0 A	1,5 millones
		120 V AC	0,5 A	2,0 millones
		230 V AC	2,0 A	1,0 millón
		230 V AC	1,0 A	1,5 millones
		230 V AC	0,5 A	2,0 millones
	Para carga inductiva (conforme a IEC 947-5-1 DC13/AC15)	Tensión	Intensidad	Número de ciclos operativos (típico)
		24 V DC	2,0 A	0,05 millones
		24 V DC	1,0 A	0,1 millón
		24 V DC	0,5 A	0,5 millones
		24 V AC	1,5 A	1,0 millón
		48 V AC	1,5 A	1,0 millón
		60 V AC	1,5 A	1,0 millón
		120 V AC	2,0 A	0,7 millones
		120 V AC	1,0 A	1,0 millón
		120 V AC	0,5 A	1,5 millones
		230 V AC	2,0 A	0,7 millones
		230 V AC	1,0 A	1,0 millón
		230 V AC	0,5 A	1,5 millones
Activación de una entrada digital		Posible		
Frecuencia de conmutación				
	Mecánica	Máx. 10 Hz		
	Con carga óhmica	Máx. 1 Hz		
	Con carga inductiva (conforme a IEC 947-5-1 DC13/AC15)	Máx. 0,5 Hz		
	Con carga de lámpara	Máx. 1 Hz		

Retención de la memoria interna de la CPU

- Vida útil de los datos remanentes y de registro: 10 años
- Datos remanentes tras POWER OFF, resistencia de ciclo de escritura: 2 millones de ciclos
- Datos de registro, hasta 2 KB por entrada, resistencia del ciclo de escritura: 500 millones de entradas

---

Nota

Efecto de registros de datos en la memoria interna de CPU

Cada escritura en el registro consume como mínimo 2 KB de memoria. Si el programa escribe pequeñas cantidades de datos frecuentemente, estará consumiendo como mínimo 2 KB de memoria en cada escritura. Sería mejor acumular los pequeños elementos de datos en un bloque de datos (DB), y escribir el bloque de datos en el registro en intervalos menos frecuentes.

Si el programa escribe muchas entradas en el registro, piense en utilizar una tarjeta de memoria SD.

A.2 Módulos CPU

Para obtener una lista más completa de los módulos disponibles para *S7-1200, véase el manual de sistema del controlador programable S7-1200* o la página web de atención al cliente (http://www.siemens.com/tiaportal).

Tabla A- 10 Especificaciones generales

Especificaciones generales		CPU 1211C	CPU 1212C	CPU 1214C	CPU 1215C	CPU 1217C
Referencia	AC/DC/relé	6ES7 211-1BE40-0XB0	6ES7 212-1BE40-0XB0	6ES7 214-1BG40-0XB0	6ES7 215-1BG40-0XB0	--
	DC/DC/relé	6ES7 211-1HE40-0XB0	6ES7 212-1HE40-0XB0	6ES7 214-1HG40-0XB0	6ES7 215-1HG40-0XB0	--
	DC/DC/DC	6ES7 211-1AE40-0XB0	6ES7 212-1AE40-0XB0	6ES7 214-1AG40-0XB0	6ES7 215-1AG40-0XB0	6ES7 217-1AG40-0XB0
Dimensiones An. x Al. x P. (mm)		90 x 100 x 75		110 x 100 x 75	130 x 100 x 75	150 x 100 x 75
Peso	• AC/DC/relé • DC/DC/relé • DC/DC/DC	• 420 gramos • 380 gramos • 370 gramos	• 425 gramos • 385 gramos • 370 gramos	• 475 gramos • 435 gramos • 415 gramos	• 585 gramos • 550 gramos • 520 gramos	- - 530 gramos
Disipación de potencia	• AC/DC/relé • DC/DC/relé • DC/DC/DC	• 10 W • 8 W • 8 W	• 11 W • 9 W • 9 W	• 14 W • 12 W • 12 W	• 14 W • 12 W • 12 W	- - 12 W
Intensidad disponible (5 V DC) para SM y bus CM		750 mA máx.	1000 mA máx.	1600 mA máx.	1600 mA máx.	1600 mA máx.
Intensidad disponible (24 V DC) alimentación de sensores		300 mA máx.	300 mA máx.	400 mA máx.	400 mA máx.	400 mA máx.
Consumo de corriente de las entradas digitales (24 V DC)		4 mA / entrada utilizada	4 mA / entrada utilizada	4 mA / entrada utilizada	4 mA/entradas utilizadas	4 mA / entrada utilizada

Tabla A- 11 Propiedades de la CPU

Propiedades de la CPU		CPU 1211C	CPU 1212C	CPU 1214C	CPU 1215C	CPU 1217C
Memoria de usuario • Memoria de trabajo • Memoria de carga • Memoria remanente		• 50 KB • 1 MB • 10 KB	• 75 KB • 1 MB • 10 KB	• 100 KB • 4 MB • 10 KB	• 125 KB • 4 MB • 10 KB	• 150 KB • 4 MB • 10 KB
E/S digitales integradas Consulte los datos técnicos (Página 395).		6 entradas 4 salidas	8 entradas 6 salidas	14 entradas 10 salidas	14 entradas 10 salidas	14 entradas 10 salidas
E/S analógicas integradas Consulte los datos técnicos (Página 405).		2 entradas	2 entradas	2 entradas	2 entradas 2 salidas	2 entradas 2 salidas
Tamaño de la memoria imagen de proceso • Entradas • Salidas		• 1024 bytes • 1024 bytes	• 1024 bytes • 1024 bytes	• 1024 bytes • 1024 bytes	• 1024 bytes • 1024 bytes	• 1024 bytes • 1024 bytes
Área de marcas (M)		4096 bytes	4096 bytes	8192 bytes	8192 bytes	8192 bytes
Memoria temporal (local)		• 16 KB para arranque y ciclo (incluyendo los FBs y FCs asociados) • 6 KB para cada uno de los niveles de prioridad de alarma restantes (incluidos FBs y FCs)				
Ampliación con SM		Ninguno	2 SM máx.	8 SM máx.	8 SM máx.	8 SM máx.
Ampliación con SB, CB o BB		1 máx.	1 máx.	1 máx.	1 máx.	1 máx.
Ampliación con CM		3 máx.	3 máx.	3 máx.	3 máx.	3 máx.
Con-tadores rápidos	Total	Se han configurado un máximo de 6 para usar cualquier entrada integrada o de SB				
	1 MHz	--	--	--	--	De Ib.2 a Ib.5 (diferencial)
	100/[1]80 kHz	Ia.0 a Ia.5	De Ia.0 a Ia.5	De Ia.0 a Ia.5	De Ia.0 a Ia.5	De Ia.0 a Ia.5
	30/[1]20 kHz	--	De Ia.6 a Ia.7	De Ia.6 a Ib.5	De Ia.6 a Ib.5	De Ia.6 a Ib.1
Salidas de impul-so [2]	Total	Se han configurado un máximo de 4 para usar cualquier salida integrada o de SB				
	1 MHz	--	--	--	--	De Qa.0 a Qa.3 (diferencial)
	100 kHz	De Qa.0 a Qa.3	De Qa.0 a Qa.3	De Qa.0 a Qa.3	De Qa.0 a Qa.3	De Qa.4 a Qb.1
	30 kHz	--	De Qa.4 a Qa.5	De Qa.4 a Qb.1	De Qa.4 a Qb.1	--
Entradas de captura de impulsos		6	8	14	14	14
Alarmas de retardo		4 en total con resolución de 1 ms	4 en total con resolución de 1 ms	4 en total con resolución de 1 ms	4 en total con resolución de 1 ms	4 en total con resolución de 1 ms
Alarmas cíclicas		4 en total con resolución de 1 ms	4 en total con resolución de 1 ms	4 en total con resolución de 1 ms	4 en total con resolución de 1 ms	4 en total con resolución de 1 ms

Propiedades de la CPU	CPU 1211C	CPU 1212C	CPU 1214C	CPU 1215C	CPU 1217C
Alarmas de flanco Con SB opcional	6 crecientes y 6 decrecientes 10 crecientes y 10 decrecientes	8 crecientes y 8 decrecientes 12 crecientes y 12 decrecientes	12 crecientes y 12 decrecientes 16 crecientes y 16 decrecientes	12 crecientes y 12 decrecientes 16 crecientes y 16 decrecientes	12 crecientes y 12 decrecientes 16 crecientes y 16 decrecientes
Reloj en tiempo real • Precisión • Tiempo de retención (capacitador de alto rendimiento sin mantenimiento)	• +/- 60 segundos/mes • 20 días típ./12 días mín. a 40 °C	• +/- 60 segundos/mes • 20 días típ./12 días mín. a 40 °C	• +/- 60 segundos/mes • 20 días típ./12 días mín. a 40 °C	• +/- 60 segundos/mes • 20 días típ./12 días mín. a 40 °C	• +/- 60 segundos/mes • 20 días típ./12 días mín. a 40 °C
Velocidad de ejecución • Booleano • Transferir palabra • Funciones matemáticas con números reales	• 0,08 µs/instrucción • 1,7 µs/instrucción • 2,3 µs/instrucción	• 0,08 µs/instrucción • 1,7 µs/instrucción • 2,3 µs/instrucción	• 0,08 µs/instrucción • 1,7 µs/instrucción • 2,3 µs/instrucción	• 0,08 µs/instrucción • 1,7 µs/instrucción • 2,3 µs/instrucción	• 0,08 µs/instrucción • 1,7 µs/instrucción • 2,3 µs/instrucción

1 La velocidad más lenta se aplica cuando el HSC está configurado para el modo de operación de cuadratura.

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

Tabla A- 12 Comunicación

Datos técnicos	CPU 1211C, CPU 1212C, CPU 1214C	CPU 1215C, CPU 1217C
Comunicación • Transferencia de datos • Aislamiento (señal externa a lógica del PLC) • Tipo de cable	1 puerto Ethernet • 10/100 Mb/s • Aislado por transformador, 1500 V DC • CAT5e apantallado	2 puertos Ethernet • 10/100 Mb/s • Aislado por transformador, 1500 V DC • CAT5e apantallado
Dispositivos	• 4 HMI • 1 PG	• 4 HMI • 1 PG
Conexiones Ethernet[1]	8 (activas o pasivas)	8 (activas o pasivas)
Conexiones S7 de CPU a CPU (GET/PUT)	• 8 (cliente) • 3 (servidor)	• 8 (cliente) • 3 (servidor)

[1] Conexiones para Open User Communications (activas o pasivas): TSEND_C, TRCV_C, TCON, TDISCON, TSEND y TRCV.

Tabla A- 13 Esquema de cableado para CPU 1214C AC/DC/relé

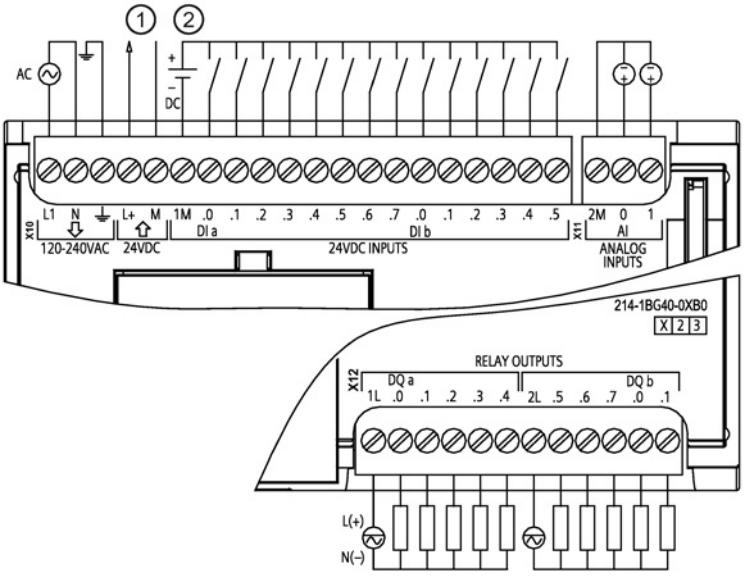
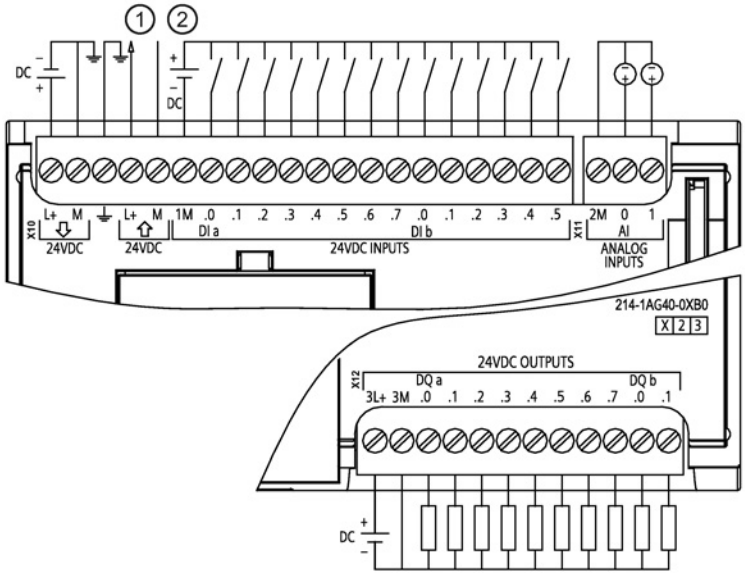
CPU 1214C AC/DC/relé	
	① Salida de alimentación de sensores 24 V DC. Para una inmunidad a interferencias adicional, conecte "M" a masa incluso si no se utiliza la alimentación de sensores. ② Para entradas en sumidero, conecte "-" a "M" (como se indica). Para entradas PNP, conecte "+" a "M". Nota 1: los conectores X11 deben ser de oro. Consulte la referencia en el *Manual de sistema controlador programable S7-1200*, anexo C, Piezas de repuesto. Nota 2: El borne L1 o N (L2) se puede conectar a una fuente de tensión de hasta 240 V AC. El terminal N puede considerarse L2 y no es necesario que esté conectado a tierra. No se requiere polarización para los terminales L1 y N (L2). Nota 3: Consulte el *Manual de sistema del controlador programable S7-1200*, Configuración del dispositivo para más información sobre el puerto Ethernet de la CPU.

Tabla A- 14 Esquema de cableado para CPU 1214C DC/DC/DC

CPU 1214C DC/DC/DC		
	①	Salida de 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 PNP, conecte "+" a "M".
	Nota 1: los conectores X11 deben ser de oro. Consulte la referencia en el *Manual de sistema controlador programable S7-1200*, anexo C, Piezas de repuesto.	
	Nota 2: Consulte el *Manual de sistema del controlador programable S7-1200*, Configuración del dispositivo para más información sobre el puerto Ethernet de la CPU.	

A.3 Módulos de E/S digitales

Para obtener una lista más completa de los módulos disponibles para *S7-1200, véase el manual de sistema del controlador programable S7-1200* o la página web de atención al cliente (http://www.siemens.com/tiaportal).

A.3.1 SB 1221, SB 1222 y SB 1223 entrada/salida digital (DI, DQ y DI/DQ)

Tabla A- 15 SB 1221 con entradas digitales (DI) y SB 1222 con salidas digitales (DQ)

General		SB 1221 DI 4 (200 kHz)	SB 1222 DQ 4 (200 kHz)
Referencia		• 24 V DC: 6ES7 221-3BD30-0XB0 • 5 V DC: 6ES7 221-3AD30-0XB0	• 24 V DC: 6ES7 222-1BD30-0XB0 • 5 V DC: 6ES7 222-1AD30-0XB0
Dimensiones An. x Al. x P. (mm)		38 x 62 x 21	38 x 62 x 21
Peso		35 gramos	35 gramos
Disipación de potencia		• 24 V DC: 1,5 W • 5 V DC: 1,0 W	0,5 W
Consumo de corriente	Bus SM	40 mA	35 mA
	24 V DC	• 24 V DC: 7 mA / entrada + 20 mA • 5 V DC: 15 mA / entrada + 15 mA	15 mA
Entradas/salidas		4 entradas (fuente)	4 salidas (estado sólido - MOSFET)

Tabla A- 16 SB 1223 con combinación de entradas/salidas digitales (DI / DQ)

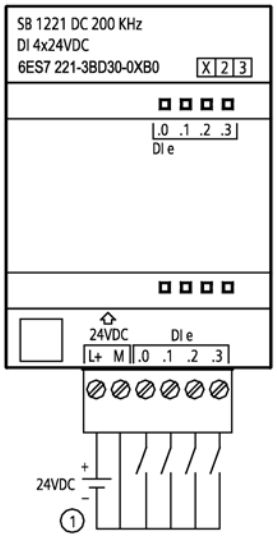
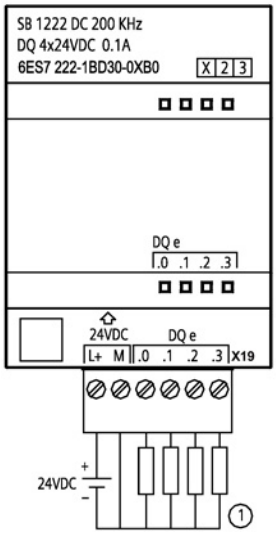
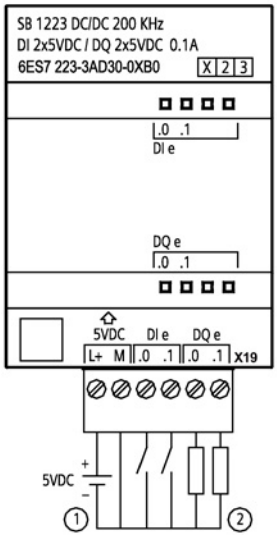
General		SB 1223 DI / DQ (200 kHz)	SB 1223 DI 2 / DQ 2
Referencia		• 24 V DC: 6ES7 223-3BD30-0XB0 • 5 V DC: 6ES7 223-3AD30-0XB0	24 V DC: 6ES7 223-0BD30-0XB0
Dimensiones An. x Al. x P. (mm)		38 x 62 x 21	38 x 62 x 21
Peso		35 gramos	40 gramos
Disipación de potencia		• 24 V DC: 1,0 W • 5 V DC: 0,5 W	24 V DC: 1,0 W
Consumo de corriente	Bus SM	35 mA	50 mA
	24 V DC	• 24 V DC: 7 mA / entrada + 20 mA • 5 V DC: 15 mA / entrada + 15 mA	4 mA / entrada utilizada
Entradas/salidas		2 entradas (fuente) 2 salidas (estado sólido - MOSFET)	2 entradas (CEI tipo 1, sumidero) 2 salidas (estado sólido - MOSFET)

Nota

Las SB de alta velocidad (200 kHz) emplean entradas en fuente. La SB estándar (20 kHz) emplea entradas en sumidero. Consulte los datos técnicos de las entradas y salidas digitales (Página 395).

Las salidas rápidas (200 kHz) (SB 1222 y SB 1223) pueden ser en fuente o en sumidero. Para salidas en fuente, conecte "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 LEDs están en ON cuando por la carga pasa corriente); una salida de sumidero es de lógica negativa (el bit Q y los LEDs están en OFF cuando por la carga pasa corriente). Si el módulo está conectado sin programa de usuario, por defecto estará a 0 V y una carga en sumidero estará ON.

Tabla A- 17 Esquemas de cableado de SB digitales

Módulo de entradas SB 1221	Módulo de salidas SB 1222	Módulo de entradas/salidas SB 1223
SB 1221 DI 4 (200 kHz)	SB 1222 DQ 4 (200 kHz)	SB 1223 DI 2 / DQ 2 (200 kHz)
		
① Soporta únicamente entradas en fuente.	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 LEDs están en ON cuando por la carga pasa corriente); una salida de sumidero es de lógica negativa (el bit Q y los LEDs están en OFF cuando por la carga pasa corriente). Si el módulo está conectado sin programa de usuario, el valor predeterminado será de 0 V para este módulo, lo que significa que una carga en sumidero estará ON.	① Soporta únicamente entradas en fuente. ② 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 LEDs están en ON cuando por la carga pasa corriente); una salida de sumidero es de lógica negativa (el bit Q y los LEDs están en OFF cuando por la carga pasa corriente). Si el módulo está conectado sin programa de usuario, el valor predeterminado será de 0 V para este módulo, lo que significa que una carga en sumidero estará ON.

Nota

Las SB rápidas (200 kHz) (SB 1221 y SB 1223) solo soportan entradas en sumidero. La SB 1223 estándar solo soporta entradas en fuente.

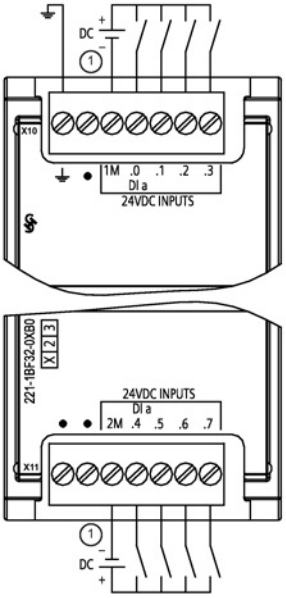
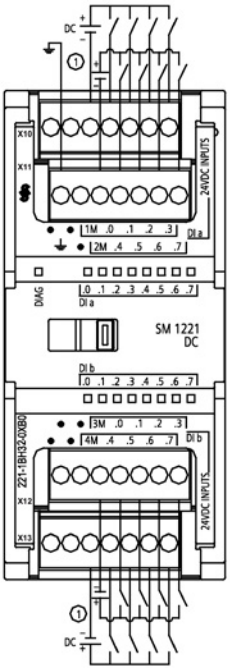
Las salidas rápidas (200 kHz) (SB 1222 y SB 1223) pueden ser en fuente o en sumidero. Para salidas en fuente, conecte "Carga" a "-" (como se indica). Para salidas en sumidero, conecte "Carga" a "+". Dado que la misma circuitería admite configuraciones en sumidero y en fuente, el estado activo de una carga en fuente es opuesto al de una carga en sumidero. Una salida de fuente es de lógica positiva (el bit Q y los LEDs están en ON cuando por la carga pasa corriente); una salida de sumidero es de lógica negativa (el bit Q y los LEDs están en OFF cuando por la carga pasa corriente). Si el módulo está conectado sin programa de usuario, el valor predeterminado será de 0 V para este módulo, lo que significa que una carga en sumidero estará ON.

A.3.2 SM 1221 con entrada digital (DI)

Tabla A- 18 SM 1221 con entrada digital (DI)

Datos técnicos		SM 1221 DI 8 24 VDC	SM 1221 DI 16 24 VDC
Referencia		6ES7 221-1BF32-0XB0	6ES7 221-1BH32-0XB0
Número de entradas (DI) Consulte los datos técnicos (Página 395).		8	16
Dimensiones An. x Al. 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
	24 V DC	4 mA / entrada utilizada	4 mA / entrada utilizada

Tabla A- 19 Esquema de cableado para módulos de entradas digitales SM 1221 (DI)

SM 1221 DI 8 (24 V DC)	SM 1221 DI 16 (24 V DC)
	

① Para entradas NPN, conecte "-" a "M" (como se indica). Para entradas PNP, conecte "+" a "M".

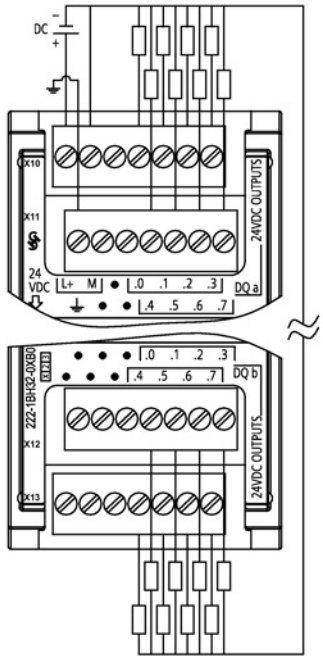
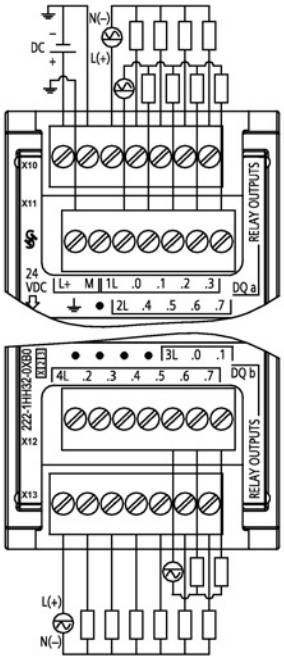
A.3.3 SM 1222 con salida digital (DQ)

Tabla A- 20 SM 1222 con salida digital (DQ)

Datos técnicos	SM 1222 DQ (relé)	SM 1222 DQ (24 VDC)
Referencia	• DQ 8: 6ES7 222-1HF32-0XB0 • DQ 8: inversor: 6ES7 222-1XF32-0XB0 • DQ 16: 6ES7 222-1HH32-0XB0	• DQ 8: 6ES7 222-1BF32-0XB0 • DQ 16: 6ES7 222-1BH32-0XB0
Número de salidas (DQ) Consulte los datos técnicos (Página 395).	• 8 (DQ 8 y DQ 8 inversor) • 16 (DQ 16)	• 8 (DQ 8) • 16 (DQ 16)
Dimensiones An. x Al. x P. (mm)	• DQ 8 y DQ 16: 45 x 100 x 75 • DQ 8 inversor: 70 x 100 x 75	45 x 100 x 75
Peso	• DQ 8: 190 gramos • DQ 8 inversor: 310 gramos • DQ 16: 260 gramos	• DQ 8: 180 gramos • DQ 16: 220 gramos

Datos técnicos		SM 1222 DQ (relé)	SM 1222 DQ (24 VDC)
Disipación de potencia		• DQ 8: 4,5 W • DQ 8 inversor: 5 W • DQ 16: 8,5 W	• DQ 8: 1,5 W • DQ 16: 2,5 W
Consumo de corriente	Bus SM	• DQ 8: 120 mA • DQ 8 inversor: 140 mA • DQ 16: 135 mA	• DQ 8: 120 mA • DQ 16: 140 mA
	24 V DC	• DQ 8 y DQ 16:11 mA / bobina de relé utilizada • DQ 8 inversor: 16,7 mA bobina de relé utilizada	• DQ 8: -- • DQ 16: --

Tabla A- 21 Esquema de cableado para SM 1222 con salida digital (DQ)

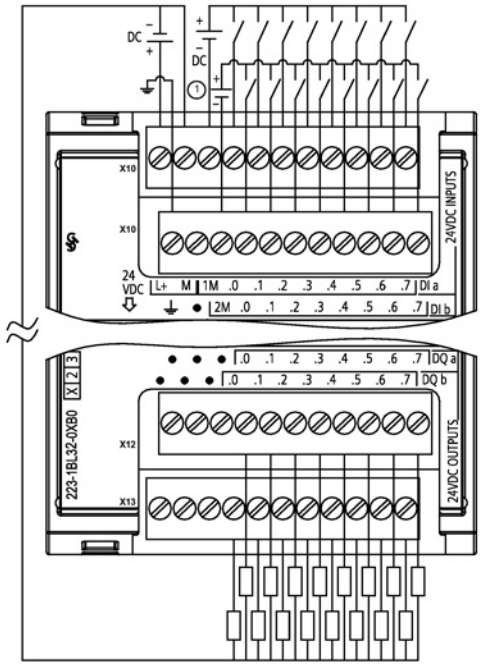
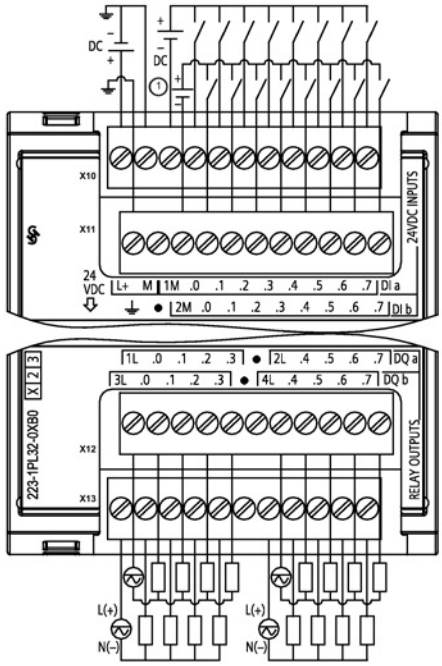
SM 1222 DQ 16, 24 VDC	SM 1222 DQ 16, salida de relé
	

A.3.4 SM 1223 V DC entrada/salida digital (DI / DQ)

Tabla A- 22 SM 1223 combinación de entradas/salidas digitales (DI / DQ)

Datos técnicos		SM 1223 DI (24 VDC) / DQ (relé)	SM 1223 DI (24 VDC) / DQ (24 VDC)
Referencia		DI 8 / DQ 8: 6ES7 223-1PH32-0XB0 DI 16 / DQ 16: 6ES7 223-1PL32-0XB0	DI 8 / DQ 8: 6ES7 223-1BH32-0XB0 DI 8 / DQ 8: 6ES7 223-1BL32-0XB0
Número de entradas / salidas (DI / DQ) Consulte los datos técnicos (Página 395).		• Entradas: 8 o 16 (24 V DC) • Salidas: 8 o 16 (relé)	• Entradas: 8 o 16 (24 V DC) • Salidas: 8 o 16 (24 V DC)
Dimensiones An. x Al. x P. (mm)		• DI 8 / DQ 8: 45 x 100 x 75 • DI 16 / DQ 16: 70 x 100 x 75	• DI 8 / DQ 8: 45 x 100 x 75 • DI 16 / DQ 16: 70 x 100 x 75
Peso		• DI 8 / DQ 8: 230 gramos • DI 16 / DQ 16: 350 gramos	• DI 8 / DQ 8: 210 gramos • DI 16 / DQ 16: 310 gramos
Disipación de potencia		• DI 8 / DQ 8: 5,5 W • DI 16 / DQ 16: 10 W	• DI 8 / DQ 8: 2,5 W • DI 16 / DQ 16: 4,5 W
Consumo de corriente	Bus SM	• DI 8 / DQ 8: 145 mA • DI 16 / DQ 16: 180 mA	• DI 8 / DQ 8: 145 mA • DI 16 / DQ 16: 185 mA
	24 V DC	4 mA / entrada utilizada 11 mA / bobina de relé utilizada	4 mA / entrada utilizada

Tabla A- 23 Esquemas de cableado para módulos SM 1223 con combinación de DI / DQ

SM 1223 DI 16 (24 V DC) / DQ 16 (24 V DC)	SM 1223 DI 16 (24 V DC) / DQ 16 (relé)
	

① Para entradas NPN, conecte "-" a "M" (como se indica). Para entradas PNP, conecte "+" a "M".

A.3.5 SM 1223 120/230 V AC entradas / salidas de relé

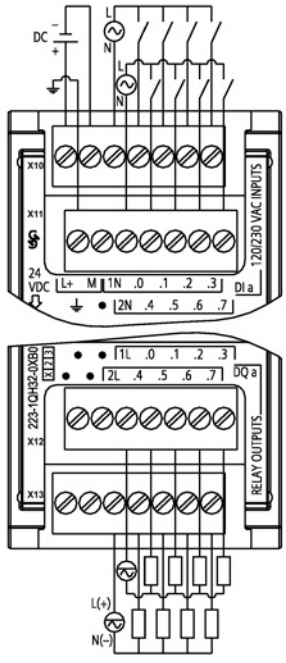
Tabla A- 24 SM 1223 combinación de entradas/salidas digitales V AC (DI / DQ)

Datos técnicos		SM 1223 DI (120/230 V AC) / DQ (relé)
Referencia		DI 8 / DQ 8: 6ES7 223-1QH32-0XB0
Número de entradas / salidas (DI / DQ)		Entradas: 8 (120/230 V AC) Consultar los datos técnicos de las entradas de 120/230 V AC (Página 397). Salidas: 8 (relé) Consultar los datos técnicos de las salidas digitales (Página 398).
Dimensiones An. x Al. x P. (mm)		45 x 100 x 75
Peso		190 gramos
Disipación de potencia		7,5 W
Consumo de corriente	Bus SM	120 mA
	24 V DC	11 mA / bobina de relé utilizada

Nota

El módulo de señales SM 1223 DI 8 x 120/230 V AC, DQ 8 x relé (6ES7 223-1QH32-0XB0) está homologado para ser utilizado en Clase 1, División 2, Grupo de gas A, B, C, D, clase de temperatura T4 Ta = 40 °C.

Tabla A- 25 Esquema de cableado del SM 1223 DI (120/230 V AC) / DQ 8 (relé)

SM 1223 DI 8 (120/230 V AC) / DQ 8 (relé)	
	

A.4 Datos técnicos de las entradas y salidas digitales

A.4.1 Entradas digitales de 24 V DC (DI)

Tabla A- 26 Datos técnicos de las entradas digitales (DI)

Datos técnicos	CPU, SM y SB	SB rápida (200 kHz)
Tipo	• CPU y SM: IEC tipo 1 sumidero (sumidero/fuente) • SB 1223: IEC tipo 1 sumidero (solo sumidero)	SB 1221 200 kHz y SB 1223 200 kHz: fuente
Tensión nominal	24 V DC a 4 mA, nominal	SB 24 V DC: 24 V DC a 7 mA, nominal SB 5 V DC: 5 V DC a 15 mA, nominal
Tensión continua admisible	30 V DC, máx.	SB 24 V DC: 28,8 V DC SB 5 V DC: 6 V DC
Tensión de choque	35 V DC durante 0,5 seg.	SB 24 V DC: 35 V DC durante 0,5 seg. SB 5 V DC: 6 V
Señal 1 lógica (mín.)	15 V DC a 2,5 mA	SB 24 V DC: L+ menos 10 V DC a 2,9 mA SB 5 V DC: L+ menos 2,0 V DC a 5,1 mA
Señal 0 lógica (máx.)	5 V DC a 1 mA	SB 24 V DC: L+ menos 5 V DC a 1,4 mA SB 5 V DC: L+ menos 1,0 V DC a 2,2 mA
Aislamiento (de campo a lógica)	500 V AC durante 1 minuto	500 V AC durante 1 minuto
Grupos de aislamiento	• CPU: 1 • SM 1221 DI 8: 2 • SM 1221 DI 16: 4 • SB 1223 DI 2: 1 • SM 1223: 2	• SB 1221 DI 4: 1 • SB 1223 DI 2: 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)

Datos técnicos	CPU, SM y SB	SB rápida (200 kHz)
Número de entradas ON simultáneamente	• SM 1221 y SM 1223 DI 8: 8 • SM 1221 y SM 1223 DI 16: 16 • SB 1223 DI 2: 2 • CPU 1211C: 6 a 60 °C horizontal o 50 °C vertical • CPU 1212C: 4 (sin puntos adyacentes) a 60 °C horizontal o 50 °C vertical; 8 a 55 °C horizontal o 45 °C vertical • CPU 1214C, CPU 1215C: 7 (sin puntos adyacentes) a 60 °C horizontal o 50 °C vertical; 14 a 55 °C horizontal o 45 °C vertical • CPU 1217C: 5 entradas de sumidero/fuente (sin puntos adyacentes) y 4 entradas diferenciales a 60 °C horizontal o 50 °C vertical; 14 a 55 °C horizontal o 45 °C vertical	• SB 1221 DI 4: 4 • SB 1223 DI 2: 2
Longitud de cable (metros)	• 500 m apantallado, 300 m no apantallado • CPU: 50 m apantallado para HSC	50 m par trenzado apantallado

Nota

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 NPN y fuente PNP
- Cambiar el cable por uno de mayor calidad
- Reduzca los circuitos y componentes de 24 V a 5 V (si el producto está homologado para baja tensión. Consulte las especificaciones detalladas en el Manual de sistema del controlador programable S7-1200)
- Añadir una carga externa a la entrada

Tabla A- 27 Frecuencias de entrada de reloj HSC (máx.)

Datos técnicos	Fase simple	Fase en cuadratura
CPU 1211C	100 kHz	80 kHz
CPU 1212C	100 kHz (Ia.0 a Ia.5) y 30 kHz (Ia.6 a Ia.7)	80 kHz (Ia.0 a Ia.5) y 20 kHz (Ia.6 a Ia.7)
CPU 1214C, CPU 1215C	100 kHz (Ia.0 a Ia.5) y 30 kHz (Ia.6 a Ib.5)	80 kHz (Ia.0 a Ia.5) y 20 kHz (Ia.6 a Ib.5)
CPU 1217C	1 MHz (Ib.2 a Ib.5) 100 kHz (Ia.0 a Ia.5) 30 kHz (Ia.6 a Ib.1)	1 MHz (Ib.2 a Ib.5) 80 kHz (Ia.0 a Ia.5) 20 kHz (Ia.6 a Ib.1)
SB de alta velocidad (200 kHz)	200 kHz	160 kHz
SB de velocidad estándar	30 kHz	20 kHz

[1] Nivel 1 lógico = de 15 a 26 V DC

A.4.2 Entradas digitales 120/230 V AC

Tabla A- 28 Entradas digitales 120/230 V AC

Datos técnicos		SM
Tipo		Tipo 1 IEC
Tensión nominal		120 V AC a 6 mA, 230 V AC a 9 mA
Tensión continua admisible		264 V AC
Tensión de choque		--
Señal 1 lógica (mín.)		79 V AC a 2,5 mA
Señal 0 lógica (máx.)		20 V AC a 1 mA
Corriente de fuga (máx.)		1 mA
Aislamiento (de campo a lógica)		1500 V AC durante 1 minuto
Grupos de aislamiento[1]		4
Tiempos de retardo a la entradas		• Típ. 0,2 a 12,8 ms, seleccionable por el usuario • Máximo: --
Conexión de detector de proximidad a 2 hilos (Bero) (máx.)		1 mA
Longitud del cable	Sin apantallar	300 metros
	Apantallado	500 metros
Número de entradas ON simultáneamente		8

[1] Los canales de un mismo grupo deben ser de la misma fase.

A.4.3 Salidas digitales (DQ)

Tabla A- 29 ¿Cuáles son los datos técnicos de las salidas digitales (DQ)?

Datos técnicos	Relé (CPU y SM)	24 V DC (CPU, SM y SB)	200 kHz 24 V DC (SB)
Tipo	Relé, contacto seco	Estado sólido MOSFET (fuente)	Estado sólido MOSFET (sumidero/fuente)
Rango de tensión	De 5 a 30 V DC o de 5 a 250 V AC	20,4 a 28,8 V DC	De 20,4 a 28,8 V DC [1] De 4,25 a 6,0 V DC [2]
Señal 1 lógica a intensidad máx.	--	20 V DC mín.	L+ menos 1,5 V [1] L+ menos 0,7 V [2]
Señal 0 lógica con carga de 10 KΩ	--	CPU: 20 V DC mín., 0,1 V DC máx. SB: 0,1 V DC máx. SM DC: 0,1 V DC máx.	1,0 V DC, máx. [1] 0,2 V DC, máx. [2]
Intensidad (máx.)	2,0 A	0,5 A	0,1 A
Carga de lámparas	30 W DC / 200 W AC	SB: 5 W	--
Resistencia en estado ON	Máx. 0,2 Ω (si son nuevas)	0,6 Ω máx.	11 Ω máx. [1] o 7 Ω máx. [2]
Resistencia en estado desact.	--	--	6 Ω máx. [1] o 0,2 Ω máx. [2]
Corriente de fuga por salida	--	10 µA máx.	--
Frecuencia de tren de impulsos	CPU: N/A[3]	CPU: 100 kHz máx., 2 Hz mín.[4] SB: 20 kHz máx., 2 Hz mín.[5]	200 kHz máx., 2 Hz mín.
Sobrecorriente momentánea	7 A si están cerrados los contactos	CPU: 8 A durante máx. 100 ms SB: 5 A durante máx. 100 ms SM: 8 A durante máx. 100 ms	0,11 A
Protección contra sobrecargas	No	No	No
Aislamiento (de campo a lógica)	Entre bobina y contacto: 1500 V AC durante 1 minuto Entre bobina y circuito lógico: Ninguno	500 V AC durante 1 minuto	500 V AC durante 1 minuto
Grupos de aislamiento	• CPU 1211C: 1 • CPU 1212C: 2 • CPU 1214C: 2 • CPU 1215C: 2 • SM DQ 8: 2 • SM DQ 8 inversor: 8 • SM DQ 16: 4	• CPU: 1 • SB: 1 • SM (DQ 8): 1 • SM (DQ 16): 1	1[5]
Resistencia de aislamiento	100 MΩ mín. si son nuevas	--	--
Aislamiento entre contactos abiertos	750 V AC durante 1 minuto	--	--
Número de salidas activadas	CPU 1211C: 4 a 60 °C horizontal o 50 °C vertical		--

Datos técnicos	Relé (CPU y SM)	24 V DC (CPU, SM y SB)	200 kHz 24 V DC (SB)
simultáneamente	CPU 1212C: 3 (sin puntos adyacentes) a 60 °C horizontal o 50 °C vertical; 6 a 55 °C horizontal o 45 °C vertical		--
	CPU 1214C: 5 (sin puntos adyacentes) a 60 °C horizontal o 50 °C vertical; 10 a 55 °C horizontal o 45 °C vertical		--
	CPU 1215C: 5 (sin puntos adyacentes) a 60 °C horizontal o 50 °C vertical; 10 a 55 °C horizontal o 45 °C vertical		--
	CPU 1217C: 3 salidas de transistor MOSFET (fuente) (sin puntos adyacentes) y 4 salidas diferenciales a 60 °C horizontal o 50 °C vertical 10 a 55 °C horizontal o 45 °C vertical		--
	SM 1222 DQ8: 8	SM 1222 DQ8: 8	--
	SM1222 DQ 8 inversor: 4 (sin puntos adyacentes) a 60 °C horizontal o 50 °C vertical, 8 a 55 °C horizontal o 45 °C vertical		
	SM 1223 DI 8/DQ 8 relé: 8		
	SM 1222 DQ16: 8 (sin puntos adyacentes) a 60 °C horizontal o 50 °C vertical; 16 a 55 °C horizontal o 45 °C vertical	SM 1222 DQ16: 16	--
	SM 1223 DI 16/DQ 16 relé: 8 (sin puntos adyacentes) a 60 °C horizontal o 50 °C vertical; 16 a 55 °C horizontal o 45 °C vertical	SM 1223 DI 8/DQ 8: 8	
	SM 1223 DI 8 x 120/230 VAC/DQ 8 relé: 4x SM 1223 DI 16/DQ 16 relé: 4 (sin puntos adyacentes) a 60 °C horizontal o 50 °C vertical; 8 a 55 °C horizontal o 45 °C vertical	SM 1223 DI 16/DQ 16: 16	
		SB 1223 DI 1 DQ 2, SM 1223 DI2 DQ2: 2	SB 1222 DQ 4: 2 (sin puntos adyacentes) a 60 °C horizontal o 50 °C vertical; 4 a 55 °C horizontal o 45 °C vertical
			SB 1222 DQ 4 x 5 V DC: 4
			SB 1223 D I 2/DQ 2: 2
			SB 1223 DI 2/DQ 2: 2

Datos técnicos	Relé (CPU y SM)	24 V DC (CPU, SM y SB)	200 kHz 24 V DC (SB)
Intensidad por neutro	SM relé: • SM 1222 DQ 8 y DQ 16: 10 A • SM 1222 DQ 8 inversor: 2A • SM 1223 DI 8/DQ 8:10 A • SM 1223 DI 16/DQ 16: 8 A • SM 1223 DI 8x120/230 VAC/DQ 8 relé: 10	SM 24 V DC • SM 1222 DQ 16: 8 A • SM 1223 DI 8/DQ 8: 4 A • SM 1223 DI 16/DQ 16: 8 A	--
Tensión de corte inductiva	--	L+ menos 48 V, Disipación de 1 W	Ninguno
Frecuencia máxima de conmutación de relé	1 Hz	--	--
Retardo de conmutación	10 ms máx.	CPU: • De Qa.0 a Qa.3: 1,0 µs máx., OFF a ON 3,0 µs máx., ON a OFF • De Qa.4 a Qb.1: 50 µs máx., OFF a ON 200 µs máx., ON a OFF SB: 2 µs máx. OFF a ON; 10 µs máx. ON a OFF SM: 50 µs máx., OFF a ON 200 µs máx., ON a OFF	1,5 µs + 300 ns asc. [1] 1,5 µs + 300 ns desc. [1] 200 ns + 300 ns asc.[2] 200 ns + 300 ns desc. [2]
Vida útil mecánica (sin carga)	Relé: 10.000.000 ciclos abiertos/cerrados	--	--
Vida útil de los contactos bajo carga nominal	Relé: 100.000 ciclos abiertos/cerrados	--	--
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)
Longitud de cable (metros)	500 m apantallado, 150 m no apantallado	500 m apantallado, 150 m no apantallado	50 m par trenzado apantallado

1 24 V DC 200 kHz SB

2 5 V DC 200 kHz SB

3 Para modelos de CPU con salidas de relé se debe instalar una Signal Board (SB) digital para emplear los generadores de impulsos.

4 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 perturbaciones.

5 SB 1223 200 kHz DI 2 / DQ 2: Sin aislamiento hacia entradas

A.5 Módulos de E/S analógicas

Para obtener una lista más completa de los módulos disponibles para *S7-1200, véase el manual de sistema del controlador programable S7-1200* o la página web de atención al cliente (http://www.siemens.com/tiaportal).

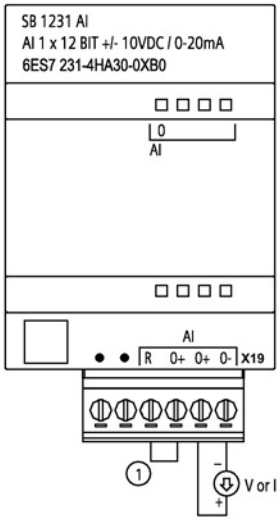
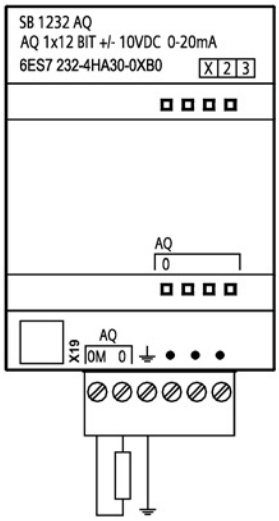
A.5.1 SB 1231 y SB 1232 con entrada analógica (AI) y salida analógica (AQ)

Tabla A- 30 Especificaciones generales

Datos técnicos	SB 1231 AI 1 x12 bits [1]	SB 1232 AQ 1 x 12 bits
Referencia	6ES7 231-4HA30-0XB0	6ES7 232-4HA30-0XB0
Dimensiones An. x Al. x P. (mm)	38 x 62 x 21 mm	38 x 62 x 21 mm
Peso	35 gramos	40 gramos
Disipación de potencia	0,4 W	1,5 W
Consumo de corriente (bus SM)	55 mA	15 mA
Consumo de corriente (24 V DC)	Ninguno	40 mA (sin carga)
Número de entradas / salidas	1	1
Tipo	Tensión o intensidad (diferencial)	Tensión o intensidad

[1] Para poder utilizar la SB 1231 AI 1 x entrada analógica, el firmware de la CPU tiene que ser de la versión 2.0 o superior.

Tabla A- 31 Esquemas de cableado de SB analógicas

SB 1231 AI 1 x12 bits	SB 1232 AQ 1 x 12 bits
	

① Conecte "R" y "0+" para corriente.

A.5.2 SM 1231 con entrada analógica (AI)

Tabla A- 32 SM 1231 con entradas analógicas (AI)

Datos técnicos	SM 1231 AI 4 x 13 bits	SM 1231 AI 8 x 13 bits	SM 1231 AI 4 x 16 bits
Referencia (MLFB)	6ES7 231-4HD32-0XB0	6ES7 231-4HF32-0XB0	6ES7 231-5ND32-0XB0
Número de entradas	4 entradas (AI)	8 entradas (AI)	4 entradas
Tipo	Tensión o intensidad (diferencial), seleccionable en grupos de 2	Tensión o intensidad (diferencial), seleccionable en grupos de 2	Tensión o intensidad (diferencial)
Dimensiones An. x Al. 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	1,5 W	1,5 W	1,8 W
Consumo de corriente (bus SM)	80 mA	90 mA	80 mA
Consumo de corriente (24 V DC)	45 mA	45 mA	65 mA

A.5.3 SM 1232 con salida analógica (AQ)

Tabla A- 33 SM 1232 con salidas analógicas (AQ)

Datos técnicos	SM 1232 AQ 2 x 14 bits	SM 1232 AQ 4 x 14 bits
Referencia (MLFB)	6ES7 232-4HB32-0XB0	6ES7 232-4HD32-0XB0
Número y tipo de salidas	2 salidas (AQ)	4 salidas (AQ)
Dimensiones An. x Al. x P. (mm)	45 x 100 x 75	45 x 100 x 75
Peso	180 gramos	180 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)	45 mA (sin carga)	45 mA (sin carga)

A.5.4 SM 1234 con entrada/salida analógica (AI/AQ)

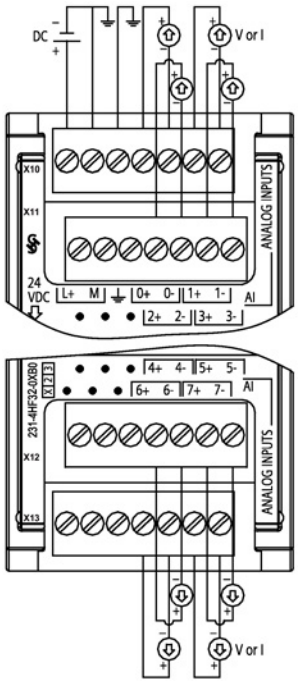
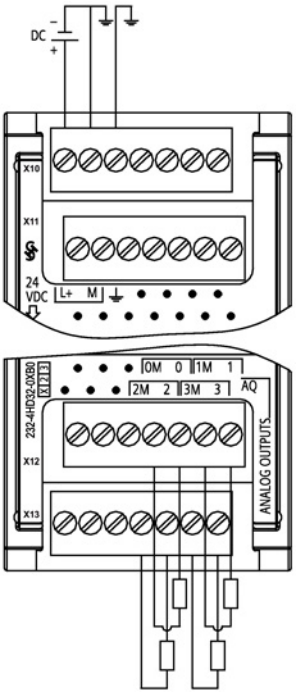
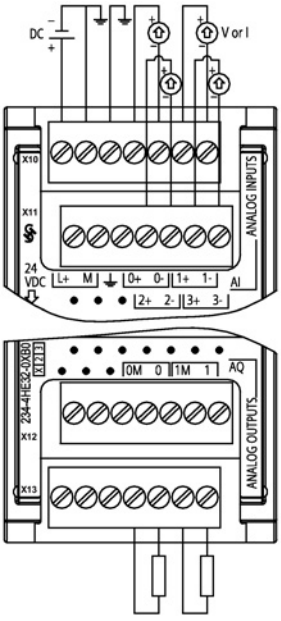
Tabla A- 34 SM 1234 con combinación de entrada/salida analógicas (AI/AQ)

Datos técnicos	SM 1234 AI 4 x 13 bits / AQ 2 x 14 bits
Referencia (MLFB)	6ES7 234-4HE32-0XB0
Número de entradas Tipo	4 entradas (AI) Tensión o intensidad (diferencial), seleccionable en grupos de 2
Número de salidas Tipo	2 salidas (AQ) Tensión o intensidad (diferencial)
Dimensiones An. x Al. x P. (mm)	45 x 100 x 75
Peso	220 gramos

Datos técnicos	SM 1234 AI 4 x 13 bits / AQ 2 x 14 bits
Disipación de potencia	2,0 W
Consumo de corriente (bus SM)	80 mA
Consumo de corriente (24 V DC)	60 mA (sin carga)

A.5.5 Diagramas de cableado de SM 1231 (AI), SM 1232 (AQ) y SM 1234 (AI/AQ)

Tabla A- 35 Diagramas de cableado de SMs analógicos

SM 1231 AI 8 x 13 bits	SM 1232 AQ 4 x 13 bits	SM 1234 AI 4 x 13 bits / AQ2 x 14 bits
		

Nota

Los canales de entrada de tensión que no se utilicen deben cortocircuitarse.

Los canales de entrada de tensión que no se utilicen deben ponerse a entre 0 y 20 mA y/o debe deshabilitarse la notificación de error por rotura de hilo.

Las entradas configuradas para el modo de intensidad no conducirán corriente en bucle a no ser que el módulo esté configurado y reciba alimentación.

Los canales de entrada de intensidad no funcionarán a no ser que el transmisor reciba alimentación externa.

A.6 Battery Board BB 1297

Battery Board BB 1297

Tabla A- 36 Especificaciones generales

Datos técnicos	BB 1297
Referencia	6ES7 297-0AX30-0XA0
Dimensiones An. x Al. x P. (mm)	38 x 62 x 21
Peso	28 gramos
Tiempo de mantenimiento del reloj en tiempo real	Aproximadamente 1 año
Tipo de batería	CR1025[1]
LED "Maint" de la CPU	Indica que es necesario sustituir la batería
Programa de usuario	La aplicación o el sistema puede evaluar el estado de la batería

[1] Véase el *Manual de sistema del controlador programable S7-1200*, capítulo 2, Instalación, para más información sobre la instalación de la BB 1297 o la sustitución de la batería de la BB.

La Battery Board BB 1297 se utiliza para aplicaciones en las que el tiempo de retención del reloj en tiempo real es superior a un mes. Las funciones de la Battery Board BB 1297 son las siguientes:

- Mantiene el reloj en tiempo real durante el apagado del PLC. La CPU S7-1200, junto con la Battery Board BB 1297, es compatible con la retención del reloj en tiempo real durante un período de apagado de la aplicación de hasta un año.
- No se puede utilizar más de una Battery Board BB 1297 u otra SB a la vez.
- No se permite el intercambio bajo tensión ni el enchufe/desenchufe en caliente. La Battery Board BB 1297 solo puede intercambiarse o enchufarse cuando la CPU está apagada. Cuando la CPU esté apagada y la BB 1297 se extraiga para cambiar la batería, el capacitador interno de alto rendimiento mantendrá la hora actual mientras el usuario sustituye la batería.
- El LED "Maint" de la CPU indica que es necesario sustituir la batería.
- El programa de usuario permite supervisar o comprobar el estado de la batería y de la Battery Board, y también permite visualizar un mensaje de usuario en un HMI o un servidor web.

A.7 Datos técnicos de E/S analógicas

A.7.1 Datos técnicos de entradas analógicas (CPU, SM y SB)

Tabla A- 37 Datos técnicos de entradas analógicas (AI)

Datos técnicos	CPU	SB	SM
Tipo	Tensión (asimétrica)	Tensión o intensidad (diferencial)	Tensión o intensidad (diferencial), seleccionable en grupos de 2
Rango	De 0 a 10 V	±10 V, ±5 V, ±2,5, De 0 a 20 mA o De 4 mA a 20 mA	±10 V, ±5 V, ±2,5 V, De 0 a 20 mA o De 4 mA a 20 mA
Resolución	10 bits	11 bits + bit de signo	12 bits + bit de signo
Rango total (palabra de datos)	De 0 a 27648	De -27.648 a 27.648	De -27.648 a 27.648
Precisión (25 °C / de -20 a 60 °C)	3,0% / 3,5% de rango máximo	±0,3% / ±0,6% de rango máximo	±0,1% / ±0,2% de rango máximo
Rango de sobreimpulso/subimpulso (palabra de datos) (V. nota 1)	Tensión: de 27.649 a 32.511	Tensión: de 32.511 a 27.649 / de -27.649 a -32.512	Tensión: de 32.511 a 27.649 / de -27.649 a -32.512
	Intensidad: N/A	Intensidad: de 32.511 a 27.649 / de 0 a -4864	Intensidad: de 32.511 a 27.649 / de 0 a -4864
Rebase por exceso/defecto (palabra de datos) (V. nota 1)	Tensión: de 32.512 a 32.767	Tensión: de 32.767 a 32.512 / de -32.513 a -32.768	Tensión: de 32.767 a 32.512 / de -32.513 a -32.768
	Intensidad: N/A	Intensidad: de 32.767 a 32.512 / de -4865 a -32.768	Intensidad: de 32.767 a 32.512 / de -4865 a -32.768
Tensión/intensidad soportada máxima	35 V DC (tensión)	±35 V / ±40 mA	±35 V / ±40 mA
Filtrado (V. nota 2)	Ninguno, débil, medio o fuerte	Ninguno, débil, medio o fuerte	Ninguno, débil, medio o fuerte
Supresión de ruido (V. nota 2)	10, 50 o 60 Hz	400, 60, 50 o 10 Hz	400, 60, 50 o 10 Hz
Principio de medición	Conversión de valor real	Conversión de valor real	Conversión de valor real
Rechazo en modo común	Ninguno	40 dB, DC a 60 Hz	40 dB, DC a 60 Hz
Rango de señales de servicio (señal y tensión en modo común)	Inferior a +12 V y superior a 0 V	Inferior a +35 V y superior a -35 V	Inferior a +12 V y superior a -12 V
Impedancia de carga	Asimétrica: ≥100 KΩ	Diferencial: 220 KΩ (tensión), 250 Ω (intensidad) Modo común: 55 KΩ (tensión), 55 KΩ (intensidad)	Diferencial: 9 MΩ (tensión), 250 Ω (intensidad) Modo común: 4,5 MΩ (tensión), 4,5 MΩ (intensidad)

Datos técnicos	CPU	SB	SM
Aislamiento (de campo a lógica)	Ninguno	Ninguno	Ninguno
Longitud de cable (metros)	100 m, par trenzado apantallado	100 m, trenzado y apantallado	100 m, trenzado y apantallado
Diagnóstico	Rebase por exceso/defecto	Rebase por exceso/defecto	Rebase por exceso/defecto 24 V DC, baja tensión
Nota 1: Consulte los rangos de medición de tensión e intensidad (Página 406) de las entradas analógicas para determinar los rangos de sobreimpulso/subimpulso y rebase por exceso/defecto. Nota 2: Consulte los tiempos de respuesta a un escalón (Página 408) para determinar los valores de filtrado y supresión de ruido.			

A.7.2 Rangos de medida de entradas (AI) de tensión e intensidad

Tabla A- 38 Representación de entradas analógicas de tensión (SB y SM)

Sistema		Rango de medida de tensión				
Decimal	Hexadecimal	±10 V	±5 V	±2,5 V	±1,25 V	
32767	7FFF[1]	11,851 V	5,926 V	2,963 V	1,481 V	Rebase por exceso
32512	7F00					
32511	7EFF	11,759 V	5,879 V	2,940 V	1,470 V	Rango de sobreimpulso
27649	6C01					
27648	6C00	10 V	5 V	2,5 V	1,250 V	Rango nominal
20736	5100	7,5 V	3,75 V	1,875 V	0,938 V	
1	1	361,7 µV	180,8 µV	90,4 µV	45,2 µV	
0	0	0 V	0 V	0 V	0 V	
-1	FFFF					
-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	

[1] 7FFF puede ser devuelto por una de las siguiente razones: rebase por exceso (como se indica en esta tabla), antes de que haya valores válidos disponibles (por ejemplo inmediatamente después del arranque) o si se detecta rotura de hilo.

Tabla A- 39 Representación de entradas analógicas de intensidad (SB y SM)

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			Rango de subimpulso
-4864	ED00	-3,52 mA	1,185 mA	
-4865	ECFF			Rebase por defecto
-32768	8000			

Tabla A- 40 Representación de entradas analógicas para tensión (CPU 1215C y CPU 1217C)

Sistema		Rango de medida de tensión	
Decimal	Hexadecimal	De 0 a 10 V	
32767	7FFF	11,851 V	Rebase por exceso
32512	7F00		
32511	7EFF	11,759 V	Rango de sobreimpulso
27649	6C01		
27648	6C00	10 V	Rango nominal
20736	5100	7,5 V	
34	22	12 mV	
0	0	0 V	
Valores negativos		Los valores negativos no se soportan	

A.7.3 Respuesta a un escalón de las entradas analógicas (AI)

En la tabla siguiente, se muestran los tiempos de respuesta a un escalón de las entradas analógicas (AI) de la CPU, SB y SM.

Tabla A- 41 Respuesta a un escalón (ms) para las entradas analógicas

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)
Sin (1 ciclo): No se promedia	CPU	N/A	63	65	130
	SB	4,5	18,7	22,0	102
	SM	4	18	22	100
Débil (4 ciclos): 4 muestreos	CPU	N/A	84	93	340
	SB	10,6	59,3	70,8	346
	SM	9	52	63	320
Medio (16 ciclos): 16 muestreos	CPU	N/A	221	258	1210
	SB	33,0	208	250	1240
	SM	32	203	241	1200
Intenso (32 ciclos): 32 muestreos	CPU	N/A	424	499	2410
	SB	63,0	408	490	2440
	SM	61	400	483	2410
Frecuencia de muestreo	CPU	N/A	4,17	5	25
	SB	0,156	1,042	1,250	6,250

A.7.4 Tiempo de muestreo y tiempos de actualización de entradas analógicas

Tabla A- 42 Tiempo de muestreo y tiempo de actualización de SM y CPU

Supresión de frecuencias (tiempo de integración)	Tiempo de muestreo	Tiempo de actualización de todos los canales		
		SM de 4 canales	SM de 8 canales	CPU AI
400 Hz (2,5 ms)	0,625 ms [1]	2,5 ms	10 ms	N/A ms
60 Hz (16,6 ms)	4,170 ms	4,17 ms	4,17 ms	4,17 ms
50 Hz (20 ms)	5,000 ms	5 ms	5 ms	5 ms
10 Hz (100 ms)	25,000 ms	25 ms	25 ms	25 ms

[1] La tasa de muestreo de la SM de 8 canales es de 1,250 ms.

Tabla A- 43 Tiempo de muestreo y tiempo de actualización de SB

Supresión de frecuencias (tiempo de integración)	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.7.5 Datos técnicos de las salidas analógicas

Tabla A- 44 Datos técnicos de salidas analógicas (SB y SM)

Datos técnicos	SB	SM
Tipo	Tensión o intensidad	Tensión o intensidad
Rango	±10 V, de 0 a 20 mA o de 4 a 20 mA	±10 V, de 0 a 20 mA o de 4 a 20 mA
Resolución	Tensión: 12 bits Intensidad: 11 bits	Tensión: 14 bits Intensidad: 13 bits
Rango total (palabra de datos) (V. nota 1)	Tensión: De -27.648 a 27.648 Intensidad: De 0 a 27.648	Tensión: De -27.648 a 27.648 Intensidad: De 0 a 27.648
Precisión (25 °C / de -20 a 60 °C)	±0,5% / ±1% de rango máximo	±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)	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	Tensión: ≥ 1000 Ω Intensidad: ≤ 600 Ω
Reacción al cambiar de RUN a STOP	Último valor o valor sustitutivo (valor predeterminado: 0)	Último valor o valor sustitutivo (valor predeterminado: 0)
Aislamiento (de campo a lógica)	Ninguno	Ninguno
Longitud de cable (metros)	100 m, trenzado y apantallado	100 m, trenzado y apantallado
Diagnóstico	• Rebase por exceso/defecto • Cortocircuito a tierra (solo en modo de tensión) • Rotura de hilo (solo en modo de intensidad)	• Rebase por exceso/defecto • Cortocircuito a tierra (solo en modo de tensión) • Rotura de hilo (solo en modo de intensidad) • 24 V DC, baja tensión
Nota 1: Consulte los rangos de salidas de tensión e intensidad (Página 410) para obtener el rango máximo.		

A.7.6 Rangos de medición de salidas (AQ) de tensión e intensidad

Tabla A- 45 Representación de salidas analógicas para tensión (SB y SM)

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 condición de rebase por exceso o defecto, las salidas analógicas adoptarán el valor sustitutivo del estado STOP.

Tabla A- 46 Representación de salidas analógicas para intensidad (SB y SM)

Sistema		Rango de salida de intensidad		
Decimal	Hexadecimal	De 0 mA a 20 mA	De 4 mA a 20 mA	
32767	7FFF	V. nota 1	V. nota 1	Rebase por exceso
32512	7F00	V. nota 1	V. nota 1	
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	4mA	
-1	FFFF		De 4 mA a 578,7 nA	Rango de subimpulso
-6912	E500		0 mA	
-6913	E4FF			No es posible. Valor de salida limitado a 0 mA.
-32512	8100			
-32513	80FF	V. nota 1	V. nota 1	Rebase por defecto
-32768	8000	V. nota 1	V. nota 1	

[1] En condición de rebase por exceso o defecto, las salidas analógicas adoptarán el valor sustitutivo del estado STOP.

Tabla A- 47 Representación de salidas analógicas para intensidad (CPU 1215C y CPU 1217C)

Sistema		Rango de salida de intensidad	
Decimal	Hexadecimal	De 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	
34	22	0,0247 mA	
0	0	0 mA	
Valores negativos		Los valores negativos no se soportan	

[1] En una situación de rebase por exceso, 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.8 Módulos RTD y termopar

Los módulos de termopar (TC) (SB 1231 TC y SM 1231 TC) miden el valor de la tensión suministrada a las entradas analógicas. Este valor puede ser tanto la temperatura de un termopar como voltios.

- Si se trata de voltios, el valor máximo del rango nominal serán 27648 decimales.
- Si se trata de temperatura, el valor se expresará en grados multiplicados por diez (p. ej. 25,3 grados se expresarán como 253 decimales).

Los módulos RTD (TC) (SB 1231 RTD y SM 1231 RTD) miden el valor de la resistencia conectada a las entradas analógicas. Este valor puede ser tanto temperatura como resistencia.

- Si se trata de resistencia, el valor máximo del rango nominal serán 27648 decimales.
- Si se trata de temperatura, el valor se expresará en grados multiplicados por diez (p. ej. 25,3 grados se expresarán como 253 decimales).

Los módulos RTD soportan mediciones con conexiones de 2, 3 y 4 hilos que van al sensor de resistencia.

Nota

Los módulos RTD y TC indicarán 32767 en cualquier 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.

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 de 2 hilos dará lugar a un error de lectura en el sensor y, por ello, no se garantiza la exactitud.

Nota

Después de aplicar tensión al módulo, éste lleva a cabo una calibración interna del convertidor 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 duración 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.

A.8.1 Datos técnicos de SB 1231 RTD y SB 1231 TC

Nota

Para utilizar estas SB TC y RTD, el firmware de la CPU tiene que ser de la versión 2.0 o superior.

Tabla A- 48 Especificaciones generales

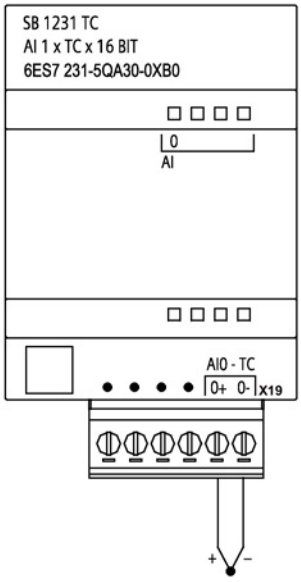
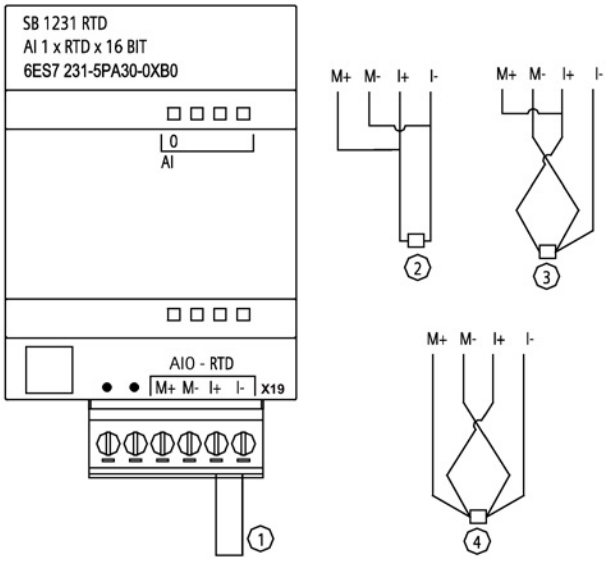
Datos técnicos	SB 1231 AI 1 x 16 bits TC	SB 1231 AI 1 x 16 bits RTD
Referencia	6ES7 231-5QA30-0XB0	6ES7 231-5PA30-0XB0
Dimensiones An. x Al. x P. (mm)	38 x 62 x 21 mm	38 x 62 x 21 mm
Peso	35 gramos	35 gramos
Disipación de potencia	0,5 W	0,7 W
Consumo de corriente (bus SM)	5 mA	5 mA
Consumo de corriente (24 V DC)	20 mA	25 mA
Número de entradas (Página 418) Tipo	1 TC flotante y mV	1 RTD y Ω indicado por el módulo
Diagnóstico	• Rebase por exceso/defecto [1, 2] • Rotura de hilo [3]	• Rebase por exceso/defecto [1, 2] • Rotura de hilo [3]

[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] RTD: 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 señalizar valores aleatorios.

Tabla A- 49 Diagramas de cableado para SB 1231 TC y RTD

SB 1231 AI 1 x 16 bits TC	SB 1231 AI 1 x 16 bits RTD
	

① Entrada loop-back RTD no utilizada

② RTD a 2 hilos ③ RTD a 3 hilos ④ RTD a 4 hilos

A.8.2 Datos técnicos de SM 1231 RTD

Tabla A- 50 Especificaciones generales

Datos técnicos	SM 1231 AI 4 x RTD x 16 bits	SM 1231 AI 8 x RTD x 16 bits
Referencia	6ES7 231-5PD32-0XB0	6ES7 231-5PF32-0XB0
Dimensiones An. x Al. 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 [1] (24 V DC)	40 mA	40 mA
Número de entradas (Página 418) Tipo	4 RTD y Ω indicado por el módulo	8 RTD y Ω indicado por el módulo
Diagnóstico	• Rebase por exceso/defecto [2,3] • Baja tensión 24 V DC[2] • Rotura de hilo (solo en modo de intensidad) [4]	• Rebase por exceso/defecto [2,3] • Baja tensión 24 V DC[2] • Rotura de hilo (solo en modo de intensidad) [4]

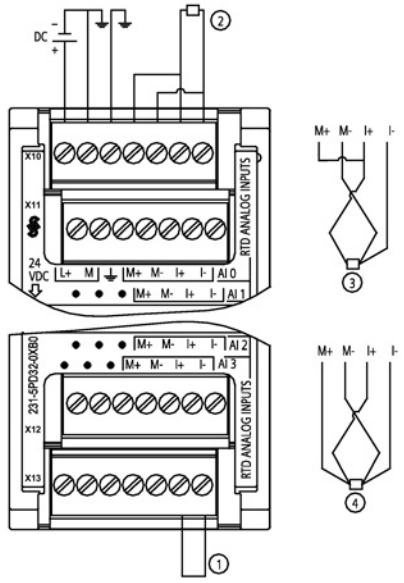
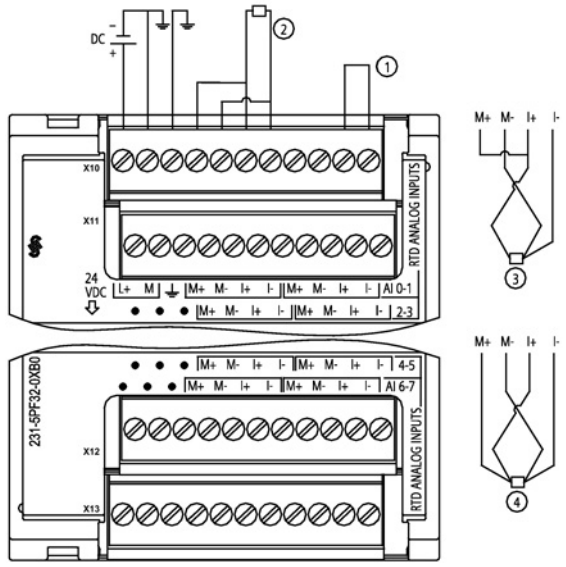
1 20,4 a 28,8 V DC (clase 2, potencia limitada o alimentación de sensor de la CPU)

2 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.

3 La detección de rebase por defecto nunca está habilitada para los rangos de resistencia.

4 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 señalizar valores aleatorios.

Tabla A- 51 Diagramas de cableado de SMs RTD

SM 1231 RTD 4 x 16 bits	SM 1231 RTD 8 x 16 bits
	

① Entradas de bucle RTD no utilizadas

② RTD a 2 hilos

③ RTD a 3 hilos

④ RTD a 4 hilos

Nota: los conectores deben ser de oro. Consulte el *Manual de sistema Controlador programable S7-1200*, anexo C.

A.8.3 Datos técnicos de SM 1231 TC

Tabla A- 52 Especificaciones generales

Modelo	SM 1231 AI 4 x 16 bits TC	SM 1231 AI 8 x 16 bits TC
Referencia	6ES7 231-5QD32-0XB0	6ES7 231-5QF32-0XB0
Dimensiones An. x Al. x P. (mm)	45 x 100 x 75	45 x 100 x 75
Peso	180 gramos	xxx gramos
Disipación de potencia	1,5 W	1,5 W
Consumo de corriente (bus SM)	80 mA	80 mA
Consumo de corriente [1] (24 V DC)	40 mA	40 mA

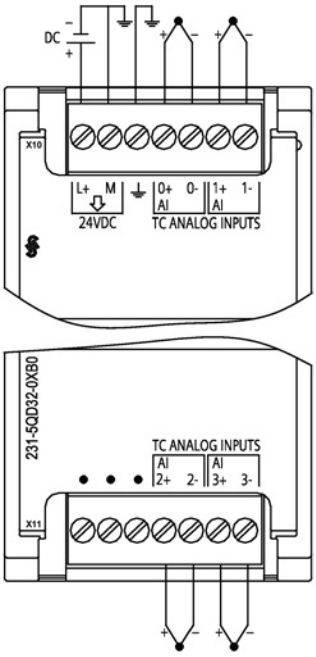
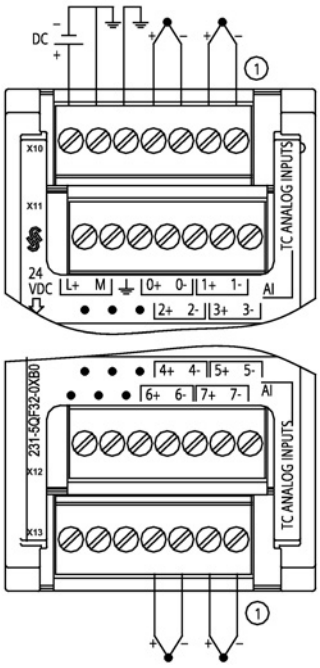
Modelo	SM 1231 AI 4 x 16 bits TC	SM 1231 AI 8 x 16 bits TC
Número de entradas (Página 418) Tipo	4 TC flotante y mV	8 TC flotante y mV
Diagnóstico	• Rebase por exceso/defecto [2] • Baja tensión 24 V DC[2] • Rotura de hilo (solo en modo de intensidad) [3]	• Rebase por exceso/defecto [2] • Baja tensión 24 V DC[2] • Rotura de hilo (solo en modo de intensidad) [3]

[1] 20,4 a 28,8 V DC (clase 2, potencia limitada o alimentación de sensor de la CPU)

[2] 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.

[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 señalizar valores aleatorios.

Tabla A- 53 Diagramas de cableado de SMs TC

SM 1231 AI 4 x 16 bits TC	SM 1231 AI 8 x 16 bits TC
	

① SM 1231 AI 8 TC: Para mejorar la claridad, TC 2, 3, 4 y 5 no se muestran conectados.

A.8.4 Datos técnicos de entradas analógicas para RTD y TC (SM y SB)

Tabla A- 54 Entradas analógicas para módulos RTD y TC (SB y SM)

Datos técnicos		RTD y termopar (TC)
Número de entradas		1 (SB), 4 u 8 (SM)
Tipo		• RTD: RTD y Ω indicado por el módulo • TC: TC flotante y mV
Rango • Rango nominal (palabra de datos) • Rango de sobreimpulso/subimpulso (palabra de datos) • Rebase por exceso/defecto (palabra de datos)		Consultar las tablas de tipo RTD/TC: • RTD (Página 420) • TC (Página 419)
Resolución	Temperatura	0,1 °C / 0,1 °F
	Resistencia / 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 o 400 Hz)
Rechazo en modo común		> 120 dB a 120 V AC
Impedancia		≥ 10 MΩ
Aislamiento	De campo a lógica	500 V AC
	De campo a 24 V DC	SM RTD y SM TC: 500 V AC (no aplicable a SB RTD y SB TC)
	24 V DC a circuito lógico	SM RTD y SM TC: 500 V AC (no aplicable a SB RTD y SB TC)
Aislamiento entre canales		• SM RTD: Ninguno (no aplicable a SB RTD) • SM TC: 120 V AC (no aplicable a SB TC)
Precisión (25 °C / de -20 a 60 °C)		Consultar las tablas de tipo RTD/TC: • RTD (Página 420) • TC (Página 419)
Repetitividad		±0,05% FS
Disipación máxima del sensor		• RTD: 0,5 mW • TC: No aplicable
Principio de medición		Integrador
Tiempo de actualización del módulo		Consultar las tablas de selección de filtro RTD/TC: • RTD (Página 422) • TC (Página 420)
Error de unión fría		• RTD: No aplicable • TC: ±1,5 °C

Datos técnicos	RTD y termopar (TC)
Longitud de cable (metros)	100 metros hasta el sensor (máx.)
Resistencia del cable	• RTD: 20 Ω, 2,7 Ω para 10 Ω RTD máx. • TC: 100 Ω máx.

A.8.5 Tipo de termopar

Tabla A- 55 Tabla de selección de termopares (rangos y precisión)

Tipo	Rango de saturación mínimo[1]	Rango nominal límite inferior	Rango nominal límite superior	Rango de saturación máximo[2]	Precisión de rango [3, 4] normal a 25 °C	Precisión de rango [3, 4] normal entre -20 °C y 60 °C
J	-210,0 °C	-150,0 °C	1.200,0 °C	1.450,0 °C	±0,3 °C	±0,6 °C
K	-270,0 °C	-200,0 °C	1.372,0 °C	1.622,0 °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	1.000,0 °C	1.200,0 °C	±0,3 °C	±0,6 °C
R & S	-50,0 °C	100,0 °C	1.768,0 °C	2.019,0 °C	±1,0 °C	±2,5 °C
B	0,0 °C	200,0 °C	800,0 °C	--	±2,0 °C	±2,5 °C
	--	800,0 °C	1.820,0 °C	1.820,0 °C	±1,0 °C	±2,3 °C
N	-270,0 °C	-200,0 °C	1.300,0 °C	1.550,0 °C	±1,0 °C	±1,6 °C
C	0,0 °C	100,0 °C	2.315,0 °C	2.500,0 °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 -80 mV	27648 80 mV	32511	±0,05%	±0,1%

1 Los valores de termopares inferiores al valor del rango de saturación mínimo se notifican como -32768.

2 Los valores de termopares 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 Solo para SM TC de 4 canales: En presencia de radiofrecuencia radiada de 970 MHz a 990 MHz, la precisión puede verse reducida.

Nota

Canal de termopar

Cada canal del módulo de señales de termopar puede configurarse con un tipo de termopar diferente (seleccionable en el software durante la configuración del módulo).

A.8.6 Selección de filtros de termopares y tiempos de actualización

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.

Tabla A- 56 Selección de filtros de termopares y tiempos de actualización

Supresión de frecuencias (Hz)	Tiempo de integración (ms)	Tiempo de actualización (segundos)		
		SB de 1 canal	SM de 4 canales	SM de 8 canales
10	100	0,301	1,225	2,450
50	20	0,061	0,263	0,525
60	16,67	0,051	0,223	0,445
400[1]	10	0,031	0,143	0,285

[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.

A.8.7 Tabla de selección de tipos de sensor RTD

Tabla A- 57 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]	Precisión de rango normal a 25 °C	Precisión de rango normal entre -20 °C y 60 °C
Pt 0,003850 ITS90 DIN EN 60751	Pt 100 Climat.	-145,00 °C	-120,00 °C	145,00 °C	155,00 °C	±0,20 °C	±0,40 °C
	Pt 10	-243,0 °C	-200,0 °C	850,0 °C	1.000,0 °C	±1,0 °C	±2,0 °C
	Pt 50	-243,0 °C	-200,0 °C	850,0 °C	1.000,0 °C	±0,5 °C	±1,0 °C
	Pt 100						
	Pt 200						
	Pt 500						
	Pt 1000						
Pt 0,003902 Pt 0,003916 Pt 0,003920	Pt 100	-243,0 °C	-200,0 °C	850,0 °C	1.000,0 °C	±0,5 °C	±1,0 °C
	Pt 200	-243,0 °C	-200,0 °C	850,0 °C	1.000,0 °C	±0,5 °C	±1,0 °C
	Pt 500						
	Pt 1000						
Pt 0,003910	Pt 10	-273,2 °C	-240,0 °C	1.100,0 °C	1.295 °C	±1,0 °C	±2,0 °C
	Pt 50	-273,2 °C	-240,0 °C	1.100,0 °C	1.295 °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						

Coeficiente de temperatura	Tipo de RTD	Rango de saturación mínimo[1]	Rango nominal límite inferior	Rango nominal límite superior	Rango de saturación máximo[2]	Precisión de rango normal a 25 °C	Precisión de rango normal entre -20 °C y 60 °C
	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 +32767.

Tabla A- 58 Resistencia

Rango	Rango de saturación mínimo	Rango nominal límite inferior	Rango nominal límite superior	Rango de saturación máximo[1]	Precisión de rango normal a 25 °C	Precisión de rango normal entre -20 °C y 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.

A.8.8 Selección de filtros RTD y tiempos de actualización

Tabla A- 59 Selección de filtros y tiempos de actualización

Supresión de frecuencias de ruido (Hz)	Tiempo de integración (ms)	Tiempo de actualización (segundos)		
		SB de 1 canal	SM de 4 canales	SM de 8 canales
10	100	4/2 hilos: 0,301 3 hilos: 0,601	4/2 hilos: 1,222 3 hilos: 2,445	4/2 hilos: 2,445 3 hilos: 4,845
50	20	4/2 hilos: 0,061 3 hilos: 0,121	4/2 hilos: 0,262 3 hilos: .505	4/2 hilos: 0,525 3 hilos: 1,015
60	16,67	4/2 hilos: 0,051 3 hilos: 0,101	4/2 hilos: 0,222 3 hilos: 0,424	4/2 hilos: 0,445 3 hilos: 0,845
400[1]	10	4/2 hilos: 0,031 3 hilos: 0,061	4/2 hilos: 0,142 3 hilos: 0,264	4/2 hilos: 0,285 3 hilos: 0,525

[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.

Nota

El módulo notifica 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.

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.

A.9 Interfaces de comunicación

Para obtener una lista más completa de los módulos disponibles para *S7-1200, véase el manual de sistema del controlador programable S7-1200* o la página web de atención al cliente (http://www.siemens.com/tiaportal).

A.9.1 Maestro/esclavo PROFIBUS

A.9.1.1 CM 1242-5 ESCLAVO PROFIBUS DP

Tabla A- 60 Datos técnicos del CM 1242-5

Datos técnicos	
Referencia	6GK7 242-5DX30-0XE0
Interfaces	
Conexión a PROFIBUS	Conector hembra sub-D de 9 pines
Consumo de corriente máxima en la interfaz PROFIBUS cuando se conectan componentes de red (por ejemplo, componentes de red ópticos)	15 mA a 5 V (solo para cierre del bus) *)
Condiciones ambientales admisibles	
Temperatura ambiente • durante el almacenamiento • durante el transporte • durante la operación con instalación vertical (perfil DIN simétrico horizontal) • durante la operación con instalación horizontal (perfil DIN simétrico vertical)	 • De -40 °C a 70 °C • De -40 °C a 70 °C • De 0 °C a 55 °C • De 0 °C a 45 °C
Humedad relativa a 25 °C durante el funcionamiento, sin condensación, máxima	95 %
Grado de protección	IP20
Fuente de alimentación, consumo de corriente y pérdidas	
Tipo de fuente de alimentación	DC
Alimentación eléctrica del bus de fondo	5 V
Consumo de corriente (típico)	150 mA
Disipación efectiva (típica)	0,75 W
Aislamiento galvánico • Interfaz PROFIBUS conectada a tierra • Interfaz PROFIBUS conectada a circuito interno	710 V DC durante 1 minuto
Dimensiones y peso	
• Ancho • Altura • Profundidad	• 30 mm • 100 mm • 75 mm

Datos técnicos	
Peso	
• Peso neto	• 115 g
• Peso con embalaje	• 152 g

*)La carga de corriente de un consumidor externo conectado entre VP (pin 6) y DGND (pin 5) no debe exceder un máximo de 15 mA (a prueba de cortocircuito) para el cierre del bus.

A.9.1.2 Asignación de pines del conector sub-D del CM 1242-5

Interfaz PROFIBUS

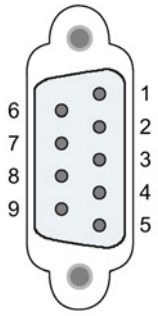


Tabla A- 61 Asignación de pines del conector sub-D

Pin	Descripción	Pin	Descripción
1	- no utilizado -	6	P5V2: alimentación de +5V
2	- no utilizado -	7	- no utilizado -
3	RxD/TxD-P: línea de datos B	8	RxD/TxD-N: línea de datos A
4	RTS	9	- no utilizado -
5	M5V2: potencial de referencia de datos (tierra DGND)	Carcasa	Conector de tierra

A.9.1.3 CM 1243-5 maestro PROFIBUS DP

Tabla A- 62 Datos técnicos del CM 1243-5

Datos técnicos	
Referencia	6GK7 243-5DX30-0XE0
Interfaces	
Conexión a PROFIBUS	Conector hembra sub-D de 9 pines
Consumo de corriente máxima en la interfaz PROFIBUS cuando se conectan componentes de red (por ejemplo, componentes de red ópticos)	15 mA a 5 V (solo para cierre del bus) *)
Condiciones ambientales admisibles	
Temperatura ambiente • durante el almacenamiento • durante el transporte • durante la operación con instalación vertical (perfil DIN simétrico horizontal) • durante la operación con instalación horizontal (perfil DIN simétrico vertical)	 • De -40 °C a 70 °C • De -40 °C a 70 °C • De 0 °C a 55 °C • De 0 °C a 45 °C
Humedad relativa a 25 °C durante el funcionamiento, sin condensación, máxima	95 %
Grado de protección	IP20
Fuente de alimentación, consumo de corriente y pérdidas	
Tipo de fuente de alimentación	DC
Alimentación eléctrica / externa • mínimo • máximo	24 V • 19,2 V • 28,8 V
Consumo de corriente (típico) • de 24 V DC • del bus de fondo del S7-1200	 • 100 mA • 0 mA
Disipación efectiva (típica) • de 24 V DC • del bus de fondo del S7-1200	 • 2,4 W • 0 W
Alimentación eléctrica 24 V DC / externa • Sección de cable mín. • Sección de cable máx. • Par de apriete de los terminales atornillados	 • mín.: 0,14 mm^2 (AWG 25) • máx.: 1,5 mm^2 (AWG 15) • 0,45 Nm (4 lb-in)
Aislamiento galvánico • Interfaz PROFIBUS conectada a tierra • Interfaz PROFIBUS conectada a circuito interno	710 V DC durante 1 minuto

Datos técnicos	
Dimensiones y peso	
• Ancho • Altura • Profundidad	• 30 mm • 100 mm • 75 mm
Peso • Peso neto • Peso con embalaje	 • 134 g • 171 g

*)La carga de corriente de un consumidor externo conectado entre VP (pin 6) y DGND (pin 5) no debe exceder un máximo de 15 mA (a prueba de cortocircuito) para el cierre del bus.

Nota

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

A.9.1.4 El maestro PROFIBUS (CM 1243-5) requiere alimentación de 24 V DC de la CPU

Nota

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

A.9.1.5 Asignación de pines del conector sub-D del CM 1243-5

Interfaz PROFIBUS

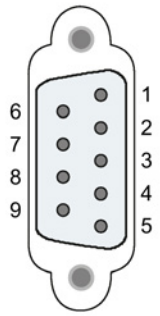


Tabla A- 63 Asignación de pines del conector sub-D

Pin	Descripción	Pin	Descripción
1	- no utilizado -	6	VP: alimentación de +5 V solo para resistencias terminadoras del bus; no para suministrar dispositivos externos
2	- no utilizado -	7	- no utilizado -
3	RxD/TxD-P: línea de datos B	8	RxD/TxD-N: línea de datos A
4	CNTR-P: RTS	9	- no utilizado -
5	DGND: tierra para señales de datos y VP	Carcasa	Conector de tierra

Cable PROFIBUS

Nota

Conectar la pantalla del cable PROFIBUS

La pantalla del cable PROFIBUS debe conectarse.

Para ello, pele el aislamiento del extremo del cable PROFIBUS y conecte la pantalla a la tierra funcional.

A.9.2 CP GPRS

Nota

El CP 1242-7 no está aprobado para aplicaciones marítimas

El CP 1242-7 no dispone de homologación marítima

Nota

Para utilizar estos módulos el firmware de la CPU debe tener la versión 2.0 o superior.

A.9.2.1 CP 1242-7 GPRS

Tabla A- 64 Datos técnicos del CP 1242-7 GPRS V2

Datos técnicos	
Referencia	6GK7 242-7KX3-0XE0
Interfaz inalámbrica	
Conector de antena	Conector SMA
Impedancia nominal	50 ohmios
Conexión inalámbrica	
Potencia de transmisión máxima	• GSM 850, clase 4: +33 dBm ±2dBm • GSM 900, clase 4: +33 dBm ±2dBm • GSM 1800, clase 1: +30 dBm ±2dBm • GSM 1900, clase 1: +30 dBm ±2dBm
GPRS	Multislot clase 10 dispositivo clase B esquema de codificación 1...4 (GMSK)
SMS	Modo de salida: MO Servicio: punto a punto
Condiciones ambientales admisibles	
Temperatura ambiente • durante el almacenamiento • durante el transporte • durante la operación con instalación vertical (perfil DIN simétrico horizontal) • durante la operación con instalación horizontal (perfil DIN simétrico vertical)	 • De -40 °C a 70 °C • De -40 °C a 70 °C • De 0 °C a 55 °C • De 0 °C a 45 °C
Humedad relativa a 25 °C durante el funcionamiento, sin condensación, máxima	95 %
Grado de protección	IP20
Fuente de alimentación, consumo de corriente y pérdidas	
Tipo de fuente de alimentación	DC

Datos técnicos	
Alimentación eléctrica / externa • mínimo • máximo	24 V • 19,2 V • 28,8 V
Consumo de corriente (típico) • de 24 V DC • del bus de fondo del S7-1200	 • 100 mA • 0 mA
Disipación efectiva (típica) • de 24 V DC • del bus de fondo del S7-1200	 • 2,4 W • 0 W
Alimentación de 24 V DC • Sección de cable mín. • Sección de cable máx. • Par de apriete de los terminales atornillados	 • mín.: 0,14 mm^2 (AWG 25) • máx.: 1,5 mm^2 (AWG 15) • 0,45 Nm (4 lb-in)
Aislamiento galvánico Unidad de alimentación a circuito interno	710 V DC durante 1 minuto
Dimensiones y peso	
• Ancho • Altura • Profundidad	• 30 mm • 100 mm • 75 mm
Peso • Peso neto • Peso con embalaje	 • 133 g • 170 g

A.9.2.2 Antena GSM/GPRS ANT794-4MR

Datos técnicos de la antena ANT794-4MR GSM/GPRS

ANT794-4MR	
Referencia	6NH9860-1AA00
Redes inalámbricas móviles	GSM/GPRS
Rangos de frecuencia	• 824 a 960 MHz (GSM 850, 900) • 1 710 a 1 880 MHz (GSM 1 800) • 1 900 a 2 200 MHz (GSM / UMTS)
Características	Omnidireccional
Ganancia de la antena	0 dB
Impedancia	50 ohmios
Relación de onda estacionaria (SWR)	< 2,0
Potencia máx.	20 W
Polaridad	Lineal vertical

ANT794-4MR	
Conector	SMA
Longitud del cable de la entena	5 m
Material externo	PVC duro, resistente a los rayos UV
Grado de protección	IP20
Condiciones ambientales admisibles • Temperatura de servicio • Temperatura de transporte/almacenamiento • Humedad relativa	 • De -40 °C a +70 °C • De -40 °C a +70 °C • 100 %
Material externo	PVC duro, resistente a los rayos UV
Construcción	Antena con 5 m de cable fijo y un conector macho SMA
Dimensiones (d x al) en mm	25 x 193
Peso • Antena incluido el cable • Boquillas	 • 310 g • 54 g
Instalación	Con soporte suministrado

A.9.2.3 Antena plana ANT794-3M

Datos técnicos de la antena plana ANT794-3M

ANT794-3M		
Referencia	6NH9870-1AA00	
Redes inalámbricas móviles	**GSM 900**	**GSM 1800/1900**
Rangos de frecuencia	890 - 960 MHz	1710 - 1990 MHz
Relación de onda estacionaria (VSWR)	≤ 2:1	≤ 1,5:1
Coeficiente de reflexión (Tx)	≈ 10 dB	≈ 14 dB
Ganancia de la antena	0 dB	
Impedancia	50 ohmios	
Potencia máx.	10 W	
Cable de antena	Cable HF RG 174 (fijo) con conector macho SMA	
Longitud del cable	1,2 m	
Grado de protección	IP64	
Rango de temperatura admisible	De -40°C a +75°C	
Inflamabilidad	UL 94 V2	
Material externo	ABS Polylac PA-765, gris claro (RAL 7035)	
Dimensiones (an x la x al) en mm	70,5 x 146,5 x 20,5	
Peso	130 g	

A.9.3 Teleservice (TS)

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

Para obtener más información sobre este producto y consultar la documentación del producto, visite la página web del catálogo de producto del TS Adapter (https://eb.automation.siemens.com/mall/en/de/Catalog/Search?searchTerm=TS%20Adapter%20IE%20basic&tab=).

A.9.4 Comunicación RS485, RS232 y RS422

A.9.4.1 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- 65 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

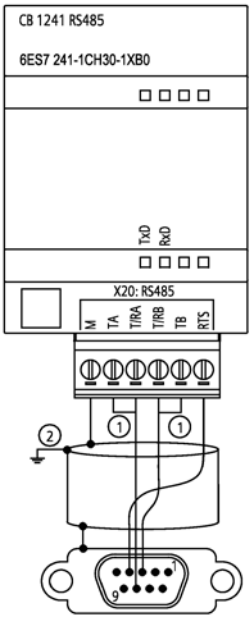
Tabla A- 66 Transmisor y receptor

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 Ω 1,5 V mín. a R_L = 54 Ω
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 127 Ω, se conecta al pin 3 RS485 Conexión del pin TA al pin T/RA, la impedancia de terminación efectiva es de 127 Ω, se conecta al pin 4 RS485
Impedancia de entrada del receptor	5,4K Ω mín. incluyendo terminación

Datos técnicos	CB 1241 RS485
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- 67 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 solo deben finalizarse los dispositivos terminadores.)	
② Utilice un cable de par trenzado apantallado y conéctelo a tierra.	

Solo se terminan los dos extremos de la red RS485. Los dispositivos que están entre los dos dispositivos terminadores no se terminan ni polarizan. Consulte el artículo "Polarizar y terminar un conector de red RS485"

Tabla A- 68 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	--
Car-casa		7 - M

A.9.4.2 Datos técnicos del CM 1241 RS422/485

Datos técnicos del CM 1241 RS422/485

Tabla A- 69 Especificaciones generales

Datos técnicos	CM 1241 RS422/485
Referencia	6ES7 241-1CH32-0XB0
Dimensiones A x A x P (mm)	30 x 100 x 75
Peso	155 gramos

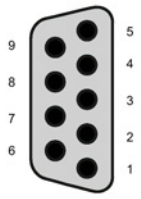
Tabla A- 70 Transmisor y receptor

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 Ω 1,5 V mín. a R_L = 54 Ω
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 y Configurar RS422 y RS485 en el Manual de sistema del controlador programable S7-1200.
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	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- 71 Alimentación eléctrica

Datos técnicos	CM 1241 RS422/485
Pérdida de potencia (disipación)	1,1 W
De +5 V DC	220 mA

Tabla A- 72 Conector RS485 o RS422 (hembra)

Pin	Descripción	Conector (hembra)	Pin	Descripción
1	Masa lógica o de comunicación		6 PWR	+5 V 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.9.4.3 Datos técnicos del CM 1241 RS232

Tabla A- 73 Especificaciones generales

Datos técnicos	CM 1241 RS232
Referencia	6ES7 241-1AH32-0XB0
Dimensiones (mm)	30 x 100 x 75
Peso	150 gramos

Tabla A- 74 Transmisor y receptor

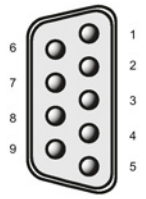
Datos técnicos	CM 1241 RS232
Tipo	RS232 (dúplex completo)
Tensión de salida del transmisor	+/- 5 V mín. a R_L = 3K Ω
Tensión de salida del transmisor	+/- 15 V DC máx.
Impedancia de entrada del receptor	3 K Ω mín.
Umbral/sensibilidad del receptor	0,8 V mín. bajo, 2,4 máx. alto histéresis típica: 0,5 V
Tensión de entrada del receptor	+/- 30 V DC máx.
Aislamiento Señal RS 232 a conexión a masa Señal RS 232 a lógica de la CPU	500 V AC durante 1 minuto
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

Datos técnicos	CM 1241 RS232
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	Hardware, software
Tiempo de espera	0 a 65535 ms

Tabla A- 75 Alimentación eléctrica

Datos técnicos	CM 1241 RS232
Pérdida de potencia (disipación)	1 W
De +5 V DC	200 mA

Tabla A- 76 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.10 Módulos tecnológicos

A.10.1 SM 1278 4 maestros IO-Link SM

A.10.1.1 Especificaciones del módulo de señales SM 1278 4 maestros IO-Link

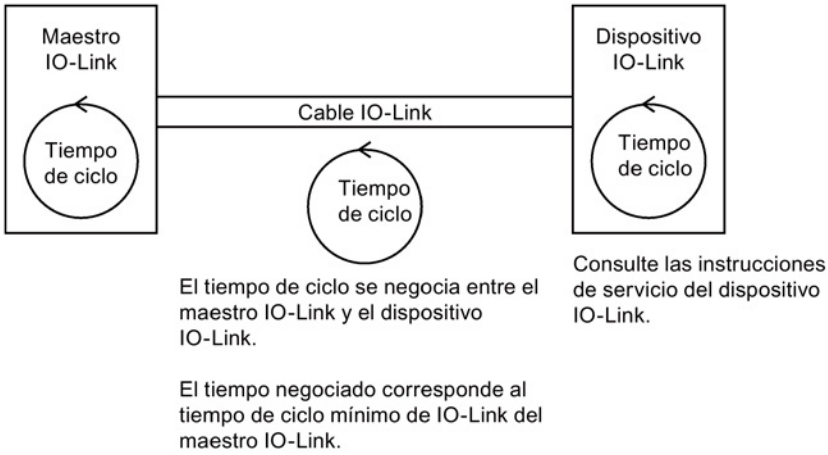
Tabla A- 77 Especificaciones generales

Datos técnicos		Módulo de señales SM 1278 4xIO-Link Master
Referencia		6ES7 278-4BD32-0XB0
Dimensiones An. x Al. x P. (mm)		45 x 100 x 75
Peso		150 gramos
Información general		
	Datos I&M	Sí; de IM0 a IM3
Tensión de alimentación		
	Tensión nominal (DC)	24 V DC
	Límite inferior del rango válido (DC)	19,2 V; 20,5 V si se utiliza IO-Link (la tensión de alimentación para los dispositivos IO-Link en el maestro debe ser como mínimo de 20 V)
	Límite superior del rango válido (DC)	28,8 V DC
	Protección contra inversión de polaridad	Sí
Intensidad de entrada		
	Consumo de corriente	65 mA; sin carga
Alimentación de encoders		
	Número de salidas	4
	Intensidad de salida, valor nominal	200 mA
Disipación		
	Disipación, típ.	1 W, excluida carga de puerto
Entradas/salidas digitales		
	Longitud de cable (metros)	20 m, sin apantallar, máx.
SDLC		
	Longitud de cable (metros)	20 m, sin apantallar, máx.
IO-Link		
	Número de puertos	4
	Número de puertos que pueden controlarse simultáneamente	4
	Protocolo IO-Link 1.0	Sí
	Protocolo IO-Link 1.1	Sí
Estado operativo		
	IO-Link	Sí
	DI	Sí
	DQ	Sí; máx. 100 mA

Datos técnicos			Módulo de señales SM 1278 4xIO-Link Master
Conexión de dispositivos IO-Link			
	Puerto tipo A		Sí
		Velocidad de transferencia	4,8 kBd (COM1)
			38,4 kBd (COM2)
			230,4 kBd (COM3)
	Tiempo de ciclo, mín.		2 ms; dinámico, depende de la longitud de los datos de usuario
	Tamaño de datos de proceso, entrada por puerto		32 bytes; máx.
	Tamaño de datos de proceso, entrada por módulo		32 bytes
	Tamaño de datos de proceso, salida por puerto		32 bytes; máx.
	Tamaño de datos de proceso, salida por módulo		32 bytes
	Memoria para parámetros de dispositivo		2 KB
	Longitud del cable sin apantallar, máx. (metros)		20 m
Información de alarmas/diagnóstico/estado			
	Indicador de estado		Sí
Alarmas			
	Alarma de diagnóstico		Sí; el diagnóstico de puerto solo está disponible en el modo IO-Link
Alarmas de diagnóstico			
	Diagnóstico		
		Vigilancia de la tensión de alimentación	Sí
		Cortocircuito	Sí
LED indicador de diagnóstico			
	Vigilancia de la tensión de alimentación		Sí; LED DIAG intermitente en rojo
	Indicador de estado de canal		Sí; un LED verde por canal para estado de canal Qn (modo SIO) y estado de PORT Cn (modo IO-Link)
	Para diagnóstico de canal		Sí; LED Fn rojo
	Para diagnóstico de módulo		Sí; LED DIAG verde/rojo
Aislamiento galvánico			
	Aislamiento galvánico canales		
		Entre canales	No
		Entre los canales y el bus de fondo	Sí
Diferencia de potencial permitida			
	Entre los diferentes circuitos		75 V DC / 60 V AC (aislamiento básico)
Aislamiento			
	Aislamiento ensayado con		707 V DC (ensayo de tipo)
Condiciones ambientales			
	Temperatura de servicio		
		Mín.	-20 °C
		Máx.	60 °C
		Montaje horizontal, mín.	-20 °C
		Montaje horizontal, máx.	60 °C
		Montaje vertical, mín.	-20 °C

Datos técnicos			Módulo de señales SM 1278 4xIO-Link Master
		Montaje vertical, máx.	50 °C

Sinopsis del tiempo de respuesta



A.10.1.2 Diagramas de cableado del maestro SM 1278 4xIO-Link

Tabla A- 78 Diagrama de cableado del maestro SM 1278 IO-Link

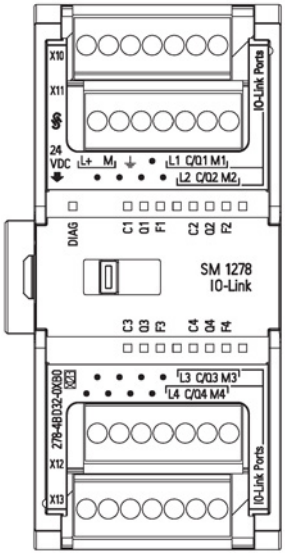
Maestro SM 1278 IO-Link (6ES7 278-4BD32-0XB0)	
	

Tabla A- 79 Asignaciones de pines de conectores para el maestro SM 1278 IO-Link (6ES7 278-4BD32-0XB0)

Pin	X10	X11	X12	X13
1	L+ / 24 V DC	Sin conexión	Sin conexión	Sin conexión
2	M / 24 V DC	Sin conexión	Sin conexión	Sin conexión
3	Tierra funcional	Sin conexión	Sin conexión	Sin conexión
4	Sin conexión	Sin conexión	Sin conexión	Sin conexión
5	L_1	L_2	L_3	L_4
6	C/Q_1	C/QL_2	C/Q_3	C/QL_4
7	ML_1	ML_2	M_3	ML_4

A.11 Productos adicionales

A.11.1 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 332-1SH71-4AA0

Para obtener más información sobre este producto y consultar la documentación del producto, visite la página web del catálogo de productos del PM 1207 (https://eb.automation.siemens.com/mall/en/de/Catalog/Product/6AG1332-1SH71-4AA0).

A.11.2 Módulo de conmutación compacto CSM 1277

El CSM1277 es un Compact switch module 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 del catálogo de productos del CSM 1277 (https://eb.automation.siemens.com/mall/en/de/Catalog/Search?searchTerm=csm%201277&tab=).

A.11.3 Módulo CM CANopen

El módulo CM CANopen es un módulo que se enchufa entre el PLC SIMATIC S7-1200 y cualquier dispositivo que esté ejecutando CANopen. El CM CANopen puede configurarse como maestro o como esclavo. Hay dos CM CANopen modules: el módulo CANopen (referencia 021620-B) y el módulo CANopen (Ruggedized) (referencia 021730-B).

El módulo CANopen ofrece las funciones siguientes:

- posibilidad de conectar 3 módulos por CPU
- conexión de hasta 16 nodos esclavos CANopen
- 256 bytes de entrada y 256 bytes de salida por módulo
- 3 LED que dan información de diagnóstico sobre el módulo, la red y el estado E/S
- almacenamiento de la configuración de red CANopen en el PLC
- capacidad de integración en el catálogo de hardware del paquete de configuración del TIA Portal
- configuración de CANopen mediante CANopen Configuration Studio (incluido) u otra herramienta de configuración externa para CANopen
- cumplimiento de los perfiles de comunicación de CANopen CiA 301 rev. 4.2 y CiA 302 rev. 4.1
- soporte de CAN 2.0A transparente para una gestión de protocolos personalizada
- bloques de función preelaborados disponibles para cualquier programación de PLC en el TIA Portal
- Los módulos CM CANopen incluyen: DSUB con terminales de tornillo para subred, CD con CM CANopen Configuration y cable de configuración USB

Para obtener más información sobre este producto y consultar la documentación correspondiente, visite la página web del catálogo de productos CM CANopen.

Reemplazar una CPU V3.0 por una CPU V4.1

B

B.1 Reemplazar una CPU V3.0 por una CPU V4.1

Existe la posibilidad de reemplazar una CPU V3.0 por una CPU V4.1 (Página 81) y utilizar el proyecto de STEP 7 que se ha diseñado para la CPU V3.0. No es posible actualizar una CPU V3.0 a una CPU V4.1 actualizando el firmware; es preciso sustituir el hardware. Al reemplazar una CPU V3.0 por una CPU V4.1 también es posible comprobar si hay actualizaciones de firmware (Página 358) para los módulos de comunicación y señales conectados.

Nota

No es posible sustituir dispositivos de la versión V4.1 por dispositivos de la versión V3.0

Es posible cambiar una CPU V3.0 por una CPU V4.1, pero no cambiar una CPU V4.1 por una CPU V3.0 una vez se ha descargado la configuración. Si desea ver o utilizar de cualquier otra forma el proyecto existente de STEP 7 V3.0, haga una copia de seguridad del mismo antes de cambiar el dispositivo.

Si todavía no ha descargado la configuración del dispositivo reemplazado puede deshacerla. Sin embargo, una vez descargada no podrá deshacer el cambio de V3.0 a V4.1.

Deben tenerse en cuenta algunos cambios en la configuración y el funcionamiento entre las dos versiones de CPU:

Bloques de organización

Con la versión V4.1 es posible configurar que la ejecución de los OB sea con o sin interrupciones (Página 62). Para proyectos de CPU V3.0 anteriores, STEP 7 determina por defecto que todos los OB sean sin interrupciones.

STEP 7 pone todas las prioridades de los OB (Página 62) a los valores que tenían en el proyecto de STEP 7 de la CPU V3.0.

Es posible cambiar la configuración de las interrupciones o la prioridad más adelante.

La información de arranque del OB de alarma de diagnóstico se refiere al submódulo como un todo si no hay eventos de diagnóstico pendientes.

Protección por contraseña de la CPU

STEP 7 ajusta el nivel de protección por contraseña (Página 91) para la CPU V4.1 al nivel equivalente al de la CPU V3.0 y asigna la contraseña de V3.0 a la contraseña "Acceso completo (sin protección)" de la CPU V4.1:

Nivel de protección V3.0	Nivel de acceso V4.1
Sin protección	Acceso completo (sin protección)
Protección contra escritura	Acceso de lectura
Protección contra escritura/lectura	Acceso HMI

Tenga en cuenta que el nivel de acceso V4.1 "Sin acceso (protección completa)" no existía en la versión V3.0.

Servidor web

Si usa páginas web definidas por el usuario en su proyecto V3.0, guárdelas en la carpeta de instalación del proyecto, dentro de la subcarpeta "UserFiles\Webserver" antes de actualizar el proyecto. Si guarda las páginas definidas por el usuario en este emplazamiento, cuando guarde el proyecto de STEP 7 también se guardarán las páginas web definidas por el usuario.

Si sustituye una CPU V3.0 por una CPU V4.1, la configuración de proyecto de servidor web para la activación del servidor web y la configuración HTTPS serán las mismas que en V3.0. A continuación puede configurar usuarios, permisos, contraseñas (Página 261) e idiomas según las necesidades del servidor web. Si no configura usuarios con privilegios adicionales estará limitado a lo que se ve en las páginas web estándar (Página 262). La CPU S7-1200 V4.1 no soporta el usuario y la contraseña "admin" preconfigurados en versiones anteriores.

La página Data Log del servidor web de S7-1200 V3.0 ofrecía una operación de "descarga y limpieza". La página Navegador de archivos (Página 262) del servidor web V4.1, desde la que se accede a los registros de datos, ya no soporta esta función. En su lugar, el servidor web ofrece la posibilidad de descargar, cambiar el nombre y borrar archivos de registros de datos.

Incompatibilidad de la tarjeta de transferencia

No se puede utilizar una tarjeta de transferencia (Página 65) V3.0 para transferir un programa V3.0 a una CPU V4.1. Hay que abrir el proyecto V3.0 en STEP 7, sustituir el dispositivo por una CPU V4.1 (Página 81) y cargar el proyecto de STEP 7 en la CPU V4.1. Después de haber migrado el proyecto a V4.1, se puede realizar una tarjeta de transferencia V4.1 para posteriores transferencias del programa.

Comunicación GET/PUT

La comunicación GET/PUT estaba activada por defecto en V3.0. Al sustituir una CPU V3.0 por una CPU V4.1 (Página 81) aparece un aviso en la sección de información de compatibilidad indicando que GET/PUT está activado.

Compatibilidad con Motion Control

Las CPU S7-1200 V4.1 no soportan las librerías de control de movimiento de V1.0 y V2.0. Si se cambia un dispositivo para un proyecto de STEP 7 con librerías de control de movimiento de V1.0 o V2.0, durante la compilación se reemplazarán las instrucciones de la librería de control de movimiento de V1.0 o V2.0 por instrucciones de Motion Control (Página 317) compatibles de la versión V3.0.

Si se sustituye una CPU V3.0 por una CPU V4.1 para un proyecto de STEP 7 que contenga dos versiones diferentes de instrucciones de Motion Control (V3.0 y V5.0), durante la compilación se reemplazarán las instrucciones de Motion Control (Página 317) compatibles con V5.0.

Al sustituir una CPU V3.por una CPU V4.1, la versión del objeto tecnológico (TO) de Motion Control no cambia automáticamente de la versión V3.0 a la V5.0. Si desea actualizar a versiones más actuales, vaya al árbol de instrucciones y seleccione la versión necesaria de S7-1200 Motion Control para el proyecto tal y como se muestra en la tabla siguiente:

Versión de CPU	Versiones de Motion control admisibles
V4.1 (motion control V5.0)	V5.0, V4.0 o V3.0
V4.0 (motion control V4.0)	V4.0 o V3.0
V3.0 (motion control V3.0)	V3.0

La estructura TO de las versiones de control de movimiento V3.0 y V5.0 difiere. Todos los bloques asociados también cambian. Las interfaces de bloques, las tablas de observación y los seguimientos se actualizan a la nueva estructura de control de movimiento V5.0. Encontrará las diferencias entre los parámetros de motion control de la CPU V3.0 y la CPU V4.1 para ejes en las dos tablas siguientes:

CPU V3.0 (Motion control V3.0)	CPU V4.1 (Motion control V5.0)
Config.General.LengthUnit	Units.LengthUnit
Config.Mechanics.PulsesPerDriveRevolution	Actor.DriveParameter.PulsesPerDriveRevolution
Config.Mechanics.LeadScrew	Mechanics.LeadScrew
Config.Mechanics.InverseDirection	Actor.InverseDirection
Config.DynamicLimits.MinVelocity	DynamicLimits.MinVelocity
Config.DynamicLimits.MaxVelocity	DynamicLimits.MaxVelocity
Config.DynamicDefaults.Acceleration	DynamicDefaults.Acceleration
Config.DynamicDefaults.Deceleration	DynamicDefaults.Deceleration
Config.DynamicDefaults.EmergencyDeceleration	DynamicDefaults.EmergencyDeceleration
Config.DynamicDefaults.Jerk	DynamicDefaults.Jerk
Config.PositionLimits_SW.Active	PositionLimitsSW.Active
Config.PositionLimits_SW.MinPosition	PositionLimitsSW.MinPosition
Config.PositionLimits_SW.MaxPosition	PositionLimitsSW.MaxPosition
Config.PositionLimits_HW.Active	PositionLimitsHW.Active
Config.PositionLimits_HW.MinSwitchedLevel	PositionLimitsHW.MinSwitchLevel
Config.PositionLimits_HW.MaxSwitchedLevel	PositionLimitsHW.MaxSwitchLevel
Config.Homing.AutoReversal	Homing.AutoReversal
Config.Homing.Direction	Homing.ApproachDirection

CPU V3.0 (Motion control V3.0)	CPU V4.1 (Motion control V5.0)
Config.Homing.SideActiveHoming	Sensor[1].ActiveHoming.SideInput
Config.Homing.SidePassiveHoming	Sensor[1].PassiveHoming.SideInput
Config.Homing.Offset	Sensor[1].ActiveHoming.HomePositionOffset
Config.Homing.FastVelocity	Homing.ApproachVelocity
Config.Homing.SlowVelocity	Homing.ReferencingVelocity
MotionStatus.Position	Posición
MotionStatus.Velocity	Velocidad
MotionStatus.Distance	StatusPositioning.Distance
MotionStatus.TargetPosition	StatusPositioning.TargetPosition
StatusBits.SpeedCommand	StatusBits.VelocityCommand
StatusBits.Homing	StatusBits.HomingCommand

El único parámetro de la tabla de comandos cuyo nombre cambia es la matriz con los comandos:

V3.0	V4.1
Config.Command[]	Command[]

Nota: La matriz "Command[]" es un UDT del tipo "TO_CmdTab_Config_Command" en V3.0 y "TO_Struct_Command" en V4.1.

Cambios en las instrucciones

Las instrucciones siguientes presentan cambios en los parámetros o en su comportamiento:

- RDREC y WRREC (Página 154)
- CONV (Página 116)

Comunicación con HMI Panels

Si tenía uno o varios HMI Panels (Página 21) conectados a la CPU S7-1200 V3.0, la comunicación con la CPU S7-1200 V4.1 dependerá del tipo de comunicación que utilice y de la versión de firmware del HMI panel. Vuelva a compilar el proyecto y descárguelo en la CPU y el HMI y/o actualice el firmware HMI.

Requisitos para volver a compilar bloques de programa

Después de sustituir una CPU V3.0 por una CPU V4.1 es necesario volver a compilar todos los bloques de programa antes de poder cargarlos en la CPU V4.1. Además, si alguno de los bloques tiene protección de know-how (Página 94) o protección de copia ligada al número de serie del PLC (Página 95), deberá eliminar la protección antes de compilar y cargar los bloques. (Sin embargo, no es necesario desactivar la protección de copia ligada a una Memory Card.) Una vez que la compilación ha finalizado correctamente, puede volver a configurar la protección de know-how y/o la protección de copia vinculada al número de serie del PLC. Tenga en cuenta que si el proyecto incluye bloques con protección de know-how suministrados por un OEM (Original Equipment Manufacturer) deberá ponerse en contacto con el OEM para que le suministre versiones V4.1 de dichos bloques.

Por norma general, Siemens recomienda volver a compilar la configuración hardware y el software en STEP 7 y cargarlos en todos los dispositivos del proyecto después de cambiar un dispositivo. Corrija todos los errores que se encuentren al compilar el proyecto y vuelva a compilarlo hasta que no haya errores. Seguidamente, puede descargar el proyecto a la CPU V4.1.

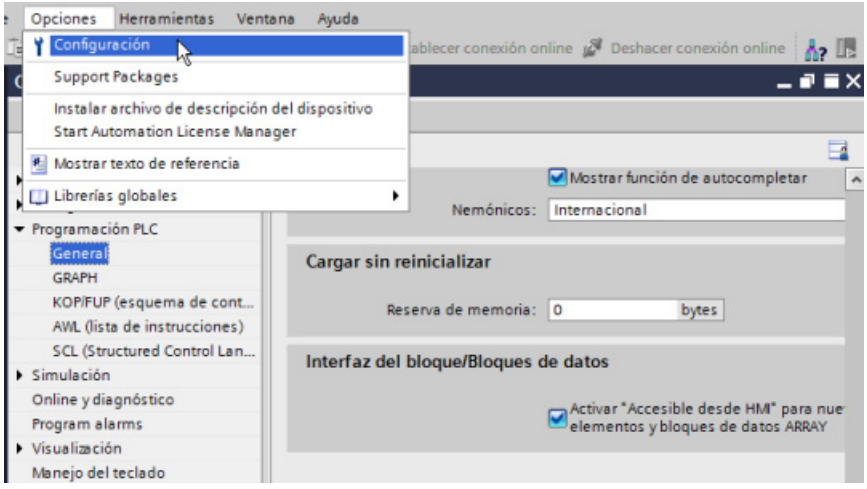
Los proyectos S7-1200 V3.0 pueden no ser aptos para CPU S7-1200 V4.1

En la S7-1200 V4.0 se ha agregado un área reservada de 100 bytes para cada DB para permitir la descarga sin reinicializar.

Se puede retirar el área de reserva de 100 bytes de los DB antes de intentar descargar un proyecto V3.0 en una CPU V4.1.

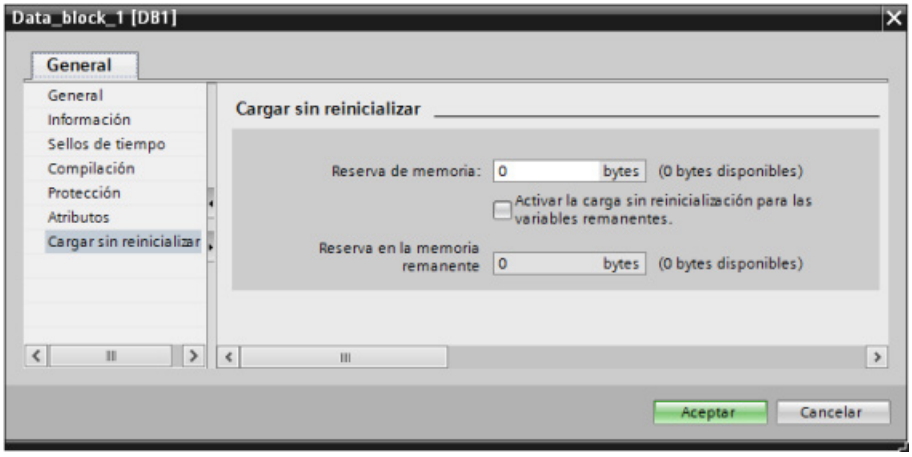
Para eliminar el área de reserva de 100 bytes siga los pasos siguientes antes de cambiar el dispositivo:

1. Desde el menú principal del TIA Portal, elija el comando Opciones > Configuración.
2. Desde el árbol de navegación, abra Programación PLC > Nodo general.
3. En el área "Cargar sin reinicializar", ajuste la reserva de memoria a 0 bytes.



Si ya ha sustituido el dispositivo, tiene que eliminar la reserva de 100 bytes individualmente de cada bloque:

1. Desde el árbol del proyecto, haga clic con el botón derecho del ratón en un bloque de datos de la carpeta Bloques de programa y elija Propiedades en el menú contextual.
2. En el cuadro de diálogo de las propiedades del bloque de datos, elija el nodo "Cargar sin reinicializar".
3. Ajuste la reserva de memoria a 0 bytes.
4. Repita el procedimiento para cada bloque de datos del proyecto.



Consulte el *Manual de sistema del controlador programable S7-1200* para conocer más detalles de las funciones V4.1.

Índice alfabético

A

Actualización de firmware
  Desde STEP 7, 358
Actualización de una CPU V3.0 a la versión V4.1, 443
Actualización del firmware
  Desde STEP 7, 358
Actualizar páginas web definidas por el usuario, 267, 267
Agregar dispositivo
  CPU sin especificar, 79, 360
  Detectar hardware existente, 79
Agregar entradas o salidas a instrucciones KOP o FUP, 33
Ajuste de válvulas PID, 209
Alarmas
  Latencia de alarmas, 62
  Vista general, 61
Ambiental
  condiciones de manejo, 377
  Condiciones de transporte y almacenamiento, 377
Ampliar las prestaciones del S7-1200, 19
Analizador lógico, 364
Aprobación C-Tick, 373
Aprobación marina, 374
Áreas de memoria
  Acceso inmediato, 69
  Bloque de datos, 69
  Direccionamiento de valores booleanos o de bit, 70
  Imagen de proceso, 69
  Memoria global, 69
  Memoria temporal, 69
AS-i
  Agregar esclavo AS-i, 173
  Agregar módulo maestro AS-i CM1243-2, 172
  Dirección, 173
  Maestro AS-i CM 1243-2, 172

B

Barra de herramientas Favoritos, 33
Battery Board, BB 1297, 404
Bloque de administración de datos (DHB), 102
Bloque de datos
  Bloque de datos global, 69, 102
  Bloque de datos instancia, 69
  Bloques de organización (OB), 99
  Obtener valores, 353, 353
  Restablecer los valores iniciales, 353, 353
Bloque de datos global, 69, 102
Bloque de datos instancia, 69
Bloque de función (FB)
  Bloque de datos instancia, 101
  Parámetros de salida, 101
  Protección de know-how, 94
  Valor inicial, 101
Bloque de organización
  Clases de prioridad, 61
  Configurar el funcionamiento, 100
  Crear, 100
  De ciclo, varios, 100
  Función, 61
  Llamada, 61
  Procesar, 99, 99
  Protección de know-how, 94
Bloque de programa
  Crear, 103
  Primeros pasos, 43
Bloque lógico, 97
  Alarmas, 18
  Bloques de organización (OB), 18, 99
  Contadores (requisitos de memoria y cantidad), 18
  DB (bloque de datos), 102
  Enlazar a una CPU, Memory Card o contraseña, 95
  FB (bloque de función), 101
  FC (función), 101
  Llamar un bloque, 105
  Número de bloques lógicos, 18
  Número de OB, 18
  Observar, 18
  Profundidad de anidamiento, 18
  Protección anticopia, 95
  Protección de know-how, 94
  Tamaño del programa de usuario, 18
  Temporizadores (requisitos de memoria y cantidad), 18
Bloques
  Alarmas, 18, 62
  Bloques de datos (DBs), 60
  Bloques de función (FBs), 60
  Bloques de organización (OB), 18, 60, 61, 62
  comprobación de coherencia, 133
  Contadores (requisitos de memoria y cantidad), 18
  Copiar bloques desde una CPU online, 354

Eventos, 62
Funciones (FCs), 60
Llamar otro bloque lógico, 105
Número de bloques lógicos, 18
Número de OB, 18, 62
OB de arranque, 62
Observar, 18
Primeros pasos, 103
Profundidad de anidamiento, 18
Protección por contraseña, 94
Tamaño del programa de usuario, 18
Temporizadores (requisitos de memoria y cantidad), 18
Tipos, 60
Botones RUN/STOP, 35
Byte de marcas de ciclo, 88
Byte de marcas de sistema, 87

C

CALCULATE (calcular), 118
Escalar valores analógicos, 51
Uso en ecuaciones complejas, 50
Cambiar dispositivo, 81
Cambiar los ajustes de STEP 7, 36
Cambio de dispositivo
CPU V3.0 por una CPU V4.1, 443
Procedimiento, 81
Carga
Determinar, 360
Cargar
Copiar bloques desde una CPU online, 354
Programa de usuario, 354
Cargar en estado operativo RUN
Iniciar desde STEP 7, 363
Vista general, 361
CB 1241 RS485, 432
CEIL (redondear número en coma flotante a entero superior), 117
Ciclo
Operación de forzado permanente, 351, 351
Clase de protección, 378
Colas de espera, 62
Columnas y encabezados en Task Cards, 37
Communication Board (CB)
Agregar módulos, 82
CB 1241 RS485, 432
Indicadores LED, 192
RS485, 192
Tabla de comparación, 19
Vista general, 20
Compact switch module CSM 1277, 441
Compact switch module, CSM 1277, 441
Comparar y sincronizar CPU online/offline, 355
Compatibilidad electromagnética, 376
Compatibilidad electromagnética (CEM), 375
Comprobación de coherencia, 133
Comprobar el programa, 131
Comunicación
Activa/pasiva, 161, 163, 178
Conexión de red, 148
Configuración, 161, 163, 178
Dirección AS-i, 173
Dirección IP, 89
Dirección PROFIBUS, 170
ID de conexión, 157
Número de conexiones (PROFINET/PROFIBUS), 152
Parámetros, 161
PROFINET y PROFIBUS, 149
Red, 147
TCON_Param, 161
Comunicación activa/pasiva
Configurar los interlocutores, 163, 178
ID de conexión, 157
Parámetros, 161
Comunicación de red, 147
Comunicación pasiva/activa
Configurar los interlocutores, 163, 178
ID de conexión, 157
Parámetros, 161
Comunicación PtP, 190
Comunicación punto a punto, 190
Comunicación S7
Configurar la conexión, 164
Comunicación serie, 190
Comunicación TCP/IP, 147, 156
Conector de bus, 20
Conexión activa/pasiva, 163
Conexión de red
Conexión de dispositivos, 148
Dispositivos HMI, 53
Conexión local/interlocutor, 163
Conexiones
Conexión de red, 53
Conexión HMI, 53
Conexión S7, 177
Configuración, 161
ID de conexión, 157
Interlocutores, 163, 178
Número de conexiones (PROFINET/PROFIBUS), 152
Protocolos Ethernet, 177
Servidor web, 265

Tipos de comunicación, 149
Tipos, conexiones multinodo, 177
Conexiones HTTP, servidor web, 265
Conexiones multinodo
Protocolos Ethernet, 177
Tipos de conexión, 177
Conexiones servidor web máximas, 265
Configuración, 36
Agregar módulos, 82
AS-i, 173
Conexión de red, 148
Determinar, 79, 360
Dirección IP, 89
Dirección PROFIBUS, 170
HSC (contador rápido), 146
Instrucciones PID_Compact y PID_3Step, 231
Instrucciones PID_Temp, 234
páginas web definidas por el usuario, 269
Parámetros de arranque, 84
PROFIBUS, 170
PROFINET, 89
Puerto Industrial Ethernet, 89
Configuración de dispositivos, 77
Agregar módulos, 82
Agregar un dispositivo nuevo, 80
AS-i, 173
Cambiar un tipo de dispositivo, 81
Conexión de red, 148
Configurar la CPU, 84, 88
Configurar los módulos, 84, 88
Determinar, 79, 360
Módulos no enchufados, 41
PROFIBUS, 170
PROFINET, 89
Puerto AS-i, 173
Puerto Ethernet, 89
Configuración de parámetros de transmisión, 163, 178
Configuración futura (control de configuración), 83
Configuración hardware, 77
Agregar módulos, 82
Agregar un dispositivo nuevo, 80
AS-i, 173
Conexión de red, 148
Configurar la CPU, 84, 88
Configurar los módulos, 84, 88
Determinar, 79, 360
PROFIBUS, 170
PROFINET, 89
Puerto AS-i, 173
Puerto Ethernet, 89
Configurar parámetros
CPU, 84, 88
Módulos, 84, 88
PROFINET, 89
Puerto Ethernet, 89
Contactos
Programación, 46
Contador rápido
Configuración, 146
Funcionamiento, 137
HSC, 136
No se puede forzar permanentemente, 351
Contadores
Cantidad, 18
Configuración del HSC, 146
HSC (contador rápido), 136
Operación HSC, 137
Tamaño, 18
Control de configuración (configuración futura), 83
Control de movimiento
Configurar el eje, 280, 290
Referenciar (secuencia de referenciación activa), 315
sincronoización, 277
Vista general, 271
CONV (convertir), 116
Copiar bloques desde una CPU online, 354
CPU
Agregar módulos, 82
Agregar un dispositivo nuevo, 80
Botones RUN/STOP, 35
búfer de diagnóstico, 356
Communication Boards (CB), 20
comparar y sincronizar bloques, 355
Conectar online, 345
Conexión de red, 148
Configuración de dispositivos, 77
Configuración del HSC, 146
Configurar la comunicación con HMI, 147
Configurar parámetros, 84, 88
Copiar bloques desde una CPU online, 354
CPU 1211C, 382
CPU 1212C, 382
CPU 1214C, 382
CPU 1215C, 382
CPU 1217C, 382
CPU sin especificar, 79, 360
Dirección AS-i, 173
Dirección IP, 89
Dirección PROFIBUS, 170
Ejecución del programa, 57
Estados operativos, 59
Forzado permanente, 351
Forzar permanentemente, 350

Llamar un bloque, 105
Niveles de seguridad, 91, 91
Número de conexiones de comunicación, 152
Obtener valores de un DB, 353
Online, 356
Panel de operador, 35, 59, 346
Parámetros de arranque, 84
Procesamiento del arranque, 84
Procesar los OB, 99
PROFINET, 89
Protección de acceso, 91
Protección de know-how, 94
Protección por contraseña, 91
Puerto Ethernet, 89
Restablecer los ajustes de fábrica, 357
Restablecer los valores iniciales de un DB, 353
Signal Boards (SB), 20
Tabla de comparación, 17
Tablas de observación, 348
Tiempos de respuesta a un escalón para entradas analógicas, 408
Tipos de comunicación, 149
Vigilancia online, 347
Vista general, 15
Zona de disipación, 23, 26
CPU sin especificar, 79, 360, 360
Creación de páginas web definidas por el usuario, 266
Crear DB de páginas web definidas por el usuario, 269
Crear una conexión de red
Entre PLC y HMI, 53
Entre PLCs, 148
Crear una conexión HMI, 53
Customer support, 5

D

Datos técnicos, 371
CB 1241 RS485, 432
CM 1241 RS232, 435
CM 1241 RS422/485, 434
Compatibilidad electromagnética (CEM), 375
Condiciones ambientales, 377
CPU 1211C, 382
CPU 1212C, 382
CPU 1214C, 382
CPU 1215C, 382
CPU 1217C, 382
datos técnicos generales, 371
entornos industriales, 374
Homologaciones, 371
Representación de entradas analógicas (intensidad), 407
Representación de entradas analógicas (tensión), 406
Representación de salidas analógicas (intensidad), 410
Representación de salidas analógicas (tensión), 410
SB 1221 4 DI 24 VDC 200 kHz, 387
SB 1221 4 DI 5 VDC 200 kHz, 387
SB 1222 4 DQ 24 VDC 200 kHz, 387
SB 1222 4 DQ 5 VDC 200 kHz, 387
SB 1223 2 DI/2 DQ 24 VDC, 387
SB 1223 DI/DQ 24 VDC 200 kHz, 387
SB 1223 DI/DQ 5 VDC 200 kHz, 387
SB 1231 AI 1 x 12 bits, 401
SB 1231 AI 1 x 16 bits RTD, 413
SB 1231 AI 1 x 16 bits TC, 413
SB 1232 AQ 1 x 12 bits, 401
SM 1221 DI 16 24 VDC, 390
SM 1221 DI 8 24 VDC, 390
SM 1222 DQ 16 24 VDC, 391
SM 1222 DQ 16 relé, 391
SM 1222 DQ 8 24 VDC, 391
SM 1222 DQ 8 relé, 391
SM 1222 DQ 8 relé inversor, 391
SM 1223 DI 120/230 VAC/DQ relé, 394
SM 1223 DI 16/DQ 16 relé, 393
SM 1223 DI 8/DQ 8, 393, 393
SM 1223 DI 8/DQ 8 relé, 393
SM 1231 AI 4 x 13 bits, 402
SM 1231 AI 4 x 16 bits, 402
SM 1231 AI 4 x 16 bits TC, 416
SM 1231 AI 4 x RTD x 16 bits, 415
SM 1231 AI 8 x 13 bits, 402
SM 1231 AI 8 x 16 bits TC, 416
SM 1231 AI 8 x RTD x 16 bits, 415
SM 1232 AQ 2 x 14 bits, 402
SM 1232 AQ 4 x 14 bits, 402
SM 1234 AI 4 x 13 bits / AQ 2 x 14 bits, 402
SM 1278 4xIO-Link Master, 437
Tensiones nominales, 378
Tiempos de respuesta a un escalón para entradas, 408
DB (bloque de datos), 102
DB de control para páginas web definidas por el usuario
parámetro para instrucción WWW, 270
DBs de fragmentos (páginas web definidas por el usuario)
generar, 269
DC
Salidas, 379
Depuración en modo RUN, 361

Determinar, 360
Determinar para cargar una CPU online, 79
DeviceStates, 134
Diagnóstico
Búfer, 356
DeviceStates, 134
GET_DIAG, 134
Get_IM_Data, 135
Indicador de estado, 87
Instrucción LED, 134
ModuleStates, 135
Diagramas de cableado
CB 1241 RS 485, 433
CPU 1214C AC/DC/relé, 385
CPU 1214C DC/DC/DC, 386
Maestro SM 1278 IO-Link, 439
SB 1221 DI 4 200 kHz, 389
SB 1223 DI 2/DQ 2 200 kHz, 389
SB 1231 AI 1 x 12 bits, 401
SB 1231 AI 1 x 16 bits RTD, 414
SB 1231 AI 1 x 16 bits TC, 414
SB 1232 AQ 1 x 12 bits, 401
SM 1221 DI 16 24 VDC, 391
SM 1221 DI 8 24 VDC, 391
SM 1222 DQ 16 24 VDC, 392
SM 1222 DQ 16 relé, 392
SM 1223 DI 16 VDC / DQ 16 relé, 393
SM 1223 DI 16 VDC/ DQ 16 24 VDC, 393
SM 1223 DI 8 120/230 VAC/DQ 8 relé, 394
SM 1231 AI 8 x 13 bits, 403
SM 1231 RTD 4 x 16 bits, 416
SM 1231 RTD 8 x 16 bits, 416
SM 1232 AQ 4 x 13 bits, 403
SM 1234 AI 4 x 13 bits / AQ 2 x 14 bits, 403
Diferencias
en las instrucciones de comunicación punto a punto, 191
Dirección IP, 89, 89
configurar CPU online, 356
Dirección IP del router, 89
Dirección MAC, 89
Direccionamiento
Áreas de memoria, 69
Bloque de datos, 69
Entradas (I) o salidas (Q) individuales, 70
Imagen de proceso, 69
Memoria global, 69
Memoria temporal, 69
Valores booleanos o de bit, 70
Directrices
Instalación, 24
Directrices de cableado
Espacio libre para ventilación y refrigeración, 26
Diseñar un sistema PLC, 60, 97
Dispositivos HMI
Conexión de red, 148
Vista general, 21
Dividir el área de editores
Primeros pasos, 44, 48, 48
Documentación, 4
Drag & Drop entre editores, 38

E

E/S
Direccionamiento, 71
observar estado en KOP, 348
Operación de forzado permanente, 351
Representación de entradas analógicas (intensidad), 407
Representación de entradas analógicas (tensión), 406
Representación de salidas analógicas (intensidad), 410
Representación de salidas analógicas (tensión), 410
Tablas de observación, 348
Tiempos de respuesta a un escalón para entradas analógicas, 408
E/S analógicas
Conversión a unidades de ingeniería, 51
Representación de entradas (intensidad), 407
Representación de entradas (tensión), 406
Representación de salidas (intensidad), 410
Representación de salidas (tensión), 410
Tiempos de respuesta a un escalón para entradas, 408
Editar en estado operativo RUN, (Cargar en estado operativo RUN)
Editor de programas
estado, 348
observar, 348
Obtener valores de un DB, 353
Restablecer los valores iniciales de un DB, 353
Ejecución de eventos, 62
Ejecución del programa
Estructura de bloque, 60
Vista general, 57
Ejecutar comandos de eje como secuencia de movimientos (MC_CommandTable), 336
Ejemplos, comunicación
Comunicación de CPU con conexiones de transmisión y recepción separadas, 158

Comunicación de CPU con una conexión de transmisión y recepción común, 159
Comunicación de CPU mediante conexiones TSEND_C o TRCV_C, 160
Direccionamiento de esclavo AS-i, 174
Protocolos de comunicación PROFINET, 156
Telecontrol, 185
Ejemplos, control de movimiento
Características de velocidad del referenciado MC, 315
Configuración de velocidad de salida de impulsos de la CPU 1217C, 274
Configuraciones de velocidad de salida de impulsos de CPU 1211C, CPU 1212C, CPU 1214C y CPU 1215C, 275
Configurar una tabla de comandos de movimientos de objetos tecnológicos, 303
Ejemplos, instrucciones
CALCULAR, 50
Ejemplos, PID
PID_3Step, ajustes de configuración, 232
PID_Compact, ajustes de configuración, 231
PID_Temp, ajustes de configuración, 234
Ejemplos, varios
Arrastrar y soltar entre editores, 38
Función de traza y analizador lógico, 364
Segmento de un tipo de datos marcado, 72
Superposición de variable AT, 73
Enlace dinámico, 95
Enlazar a una CPU, Memory Card o contraseña, 95
Entornos industriales
Homologaciones, 374
Entrada principal, 391
Entradas analógicas
Especificaciones para CPUs, SBs y SMs, 405
Especificaciones para SBs y SMs RTD/TC, 418
Entradas digitales
Especificaciones de AC para SMs, 397
Especificaciones de V DC para CPUs, SMs y SBs, 395
Entradas y salidas
observar, 347
Escalar valores analógicos, 51
Espacio libre, ventilación y refrigeración, 26
Estado
Indicadores LED (interfaz de comunicación), 192
Instrucción LED, 134
Estado operativo, 35, 35, 59, 346
Estado operativo RUN, 59, 61
Botones de la barra de herramientas, 35
Ejecución del programa, 57
Panel de operador, 35, 59, 346
Estado operativo STOP, 59
Botones de la barra de herramientas, 35
Panel de operador, 35, 59, 346
Estructura de llamadas, 133
Estructura del programa, 97
Ethernet
Compact switch module CSM 1277, 441
Comunicación, 147
Conexión de red, 148
Dirección IP, 89
GET, 176
ID de conexión, 157
Modo ad hoc, 157
Número de conexiones de comunicación, 152
PUT, 176
Tipos de comunicación, 149
Vista general, 156
Eventos, 356

F

FAQs, 4
FB (bloque de función), 101
FC (función), 101
FLOOR, 117
Forzado permanente
Ciclo, 351
Entradas de periferia, 350, 351
Entradas y salidas, 351
Memoria I, 350, 351
Forzar permanentemente, 350
Frecuencia, bits de reloj, 88
Frecuencias de entrada de reloj HSC, 397
Función (FC), 101
Protección de know-how, 94
Función de trazas, 364
Funciones nuevas, 27
FUP (Diagrama de funciones), 107

G

Generar DB de páginas web definidas por el usuario, 269
GET, 176
GET (leer datos de una CPU remota)
Configurar la conexión, 164
GET_DIAG, 134
Get_IM_Data, 135
Grado de protección, 378

H

Herramientas online y diagnóstico
Cargar en estado operativo RUN, 361
HMI
Conexión de red, 53
Conexión HMI, 53
Configurar la comunicación PROFINET, 147
Pantalla, 54
Primeros pasos, 52, 54
Homologación ATEX, 373
Homologación CE, 371
Homologación coreana, 373
Homologación cULus, 372
Homologación FM, 372
Homologaciones
ATEX, 373
CE, 371
C-Tick, 373
cULus, 372
FM, 372
Homologación coreana, 373
Marítima, 374
Hora
configurar CPU online, 356
Hotline, 5
HSC (contador rápido)
Configuración, 146, 146
Funcionamiento, 136, 137

I

Imagen de proceso
estado, 348, 350
Forzar permanentemente, 350
observar, 348
Operación de forzado permanente, 351
Vigilancia de estado o valor, 347
Indicador de primer ciclo, 87
Indicadores LED
Instrucción LED, 134
Interfaz de comunicación, 192
Información de contacto, 5, 81
Información del programa
En la estructura de llamadas, 133
Iniciar
Trazas, 364
Inmunidad a ondas de choque, 376
Insertar instrucciones
Drag & Drop, 32
Drag & Drop entre editores, 38
Favoritos, 33
Insertar un dispositivo
CPU sin especificar, 79, 360
Instalación
Dimensiones de montaje, 23
Directrices, 24
Espacio, 26
Módulos de señales (SM), 20
Refrigeración, 26
Ventilación, 26
Vista general, 24
Zona de disipación, 23, 26
Instrucción Copiar valor, 114
Instrucción CTRL_PWM, 128
Instrucción Cuadro
Primeros pasos, 49
Instrucción MOVE_BLK (Copiar área), 114
Instrucción TRCV_C, 154
Instrucción TSEND_C, 154
Instrucción UMOVE_BLK (Copiar área sin interrupciones), 114
Instrucciones
Agregar entradas o salidas a instrucciones KOP o FUP, 33
Agregar un parámetro, 49
CALCULAR, 50
CALCULATE (calcular), 118
CEIL (redondear número en coma flotante a entero superior), 117
Columnas y encabezados, 37
Comparación, 114
Contador, 125
CONV (convertir), 116
Copiar área (MOVE_BLK), 114
Copiar área sin interrupciones (UMOVE_BLK), 114
CTRL_PWM, 128
Desplazamiento, 114
DeviceStates, 134
Drag & Drop, 32
Drag & Drop entre editores, 38
Escalar valores analógicos, 51
estado, 348
Estado LED, 134
Favoritos, 33
FLOOR, 117
GET, 176
GET_DIAG, 134
Get_IM_Data, 135
HSC (contador rápido), 136, 137
Insertar, 32
Instrucciones ampliables, 34
lógicas con bits, 111

MC_ChangeDynamic (cambiar la configuración de la dinámica del eje), 339
MC_CommandTable, 336
MC_Halt (detener eje), 325
MC_Home (referenciar eje), 322
MC_MoveAbsolute (posicionamiento absoluto del eje), 327
MC_MoveJog (desplazamiento del eje en modo Jog), 334
MC_MoveRelative (posicionamiento relativo del eje), 329
MC_MoveVelocity (mover el eje a la velocidad predefinida), 331
MC_Power (habilitar/bloquear eje), 318
MC_ReadParam (leer parámetros de un objeto tecnológico), 343
MC_Reset (confirmar error), 321
MC_WriteParam (escribir parámetros de un objeto tecnológico), 341
ModuleStates, 135
Motion Control, 317
NORM_X (normalizar), 117
observar, 348
Operación de forzado permanente, 351
PID_Compact (controlador PID universal con ajuste integrado), 203
PID_Temp (regulador PID universal que permite manejar el control de temperatura), 218
Primeros pasos, 48, 49
PUT, 176
ROUND, 116
SCALE_X (escalar), 117
TRCV_C, 154
TRUNC (truncar), 116
TSEND_C, 154
Versiones de instrucciones, 37
Vigilancia de estado o valor, 347
WWW, 270
Instrucciones ampliables, 34
Instrucciones con contadores, 125
Instrucciones de comparación, 114
Instrucciones de Motion Control, 317
Instrucciones Ethernet
TRCV_C, 154
TSEND_C, 154
Instrucciones lógicas con bits, 111
Interfaces de comunicación
Agregar módulos, 82
CB 1241 RS485, 432
CM 1241 RS232, 435
RS232 y RS485, 192
Tabla de comparación de módulos, 19
Interfaz de usuario
Vistas del proyecto y del portal de STEP 7, 31
Interfaz PROFINET
Propiedades de direcciones Ethernet, 89
IO-Link
Arranque, 366
Componentes, 365
Configurar, 367
Datos, 366
Perfil de dispositivo, 367
Sinopsis de la tecnología, 365
ISO on TCP
Configuración de la conexión, 163
ID de conexión, 157
Modo ad hoc, 157
Parámetros, 161

K

KOP (esquema de contactos)
editor de programas, 348
estado, 348
observar, 348
KOP (Esquema de contactos)
estado, 350
Vigilancia de estado o valor, 347
Vista general, 106

L

Latencia, 62
LED (Obtención estado LED), 134
Librería del protocolo USS, 194
Librería global
USS, 194
Limitaciones
Páginas web definidas por el usuario, 268
Servidor web, 265

Ll

Llamada de bloque
Principios básicos, 60

M

Maestro IO-Link
Diagrama de bloques, 369
Ejemplos de conexión, 370
Ilustración, 368
Manuales, 4
Máscara de subred, 89
Matemáticas, 50, 118
MC_ChangeDynamic (cambiar la configuración de la dinámica del eje), 339
MC_CommandTable, 336
MC_Halt (detener eje), 325
MC_Home (referenciar eje), 322
MC_MoveAbsolute (posicionamiento absoluto del eje), 327
MC_MoveJog (desplazamiento del eje en modo Jog), 334
MC_MoveRelative (posicionamiento relativo del eje), 329
MC_MoveVelocity (mover el eje a la velocidad predefinida), 331
MC_Power (habilitar/bloquear eje), 318
MC_ReadParam (leer parámetros de un objeto tecnológico), 343
MC_Reset (confirmar error), 321
MC_WriteParam (escribir parámetros de un objeto tecnológico), 341
Memoria
Direcciones de entradas de periferia (tabla de forzado permanente), 350
Marcas de ciclo, 86
Marcas de sistema, 86
Memoria de carga, 65
Memoria de trabajo, 65
Memoria remanente, 65
Memoria temporal (L), 69
Memoria de carga, 17, 65
Páginas web definidas por el usuario, 268
Memoria de trabajo, 17, 65
Memoria global, 69
Memoria I
Direcciones de entradas de periferia (tabla de forzado permanente), 350
Forzar permanentemente, 350
observar, 347
observar KOP, 348
Operación de forzado permanente, 351, 351
Tabla de forzado permanente, 350
tabla de observación, 347
Memoria remanente, 17, 65
Memoria temporal (L), 69
Memory Card
Memoria de carga, 65
Modbus
Versiones, 37
Modificar
estado del editor de programas, 348
Modo ad hoc, TCP e ISO on TCP, 157
Modo ARRANQUE
Ejecución del programa, 57
Operación de forzado permanente, 351
Modo RUN
Operación de forzado permanente, 351
Modo STOP
Operación de forzado permanente, 351
ModuleStates, 135
Módulo de alimentación
PM1207, 441
Módulo de alimentación PM 1207, 441
Módulo de comunicación (CM)
Agregar módulo CM 1243-5 (maestro DP), 169
Agregar módulo maestro AS-i CM1243-2, 172
Agregar módulos, 82
Agregar un dispositivo nuevo, 80
CM 1241 RS232, 435
CM 1241 RS422/RS485, 434
Configuración de dispositivos, 77
Indicadores LED, 192
RS232 y RS485, 192
Tabla de comparación, 19
Vista general, 20
Módulo de comunicaciones (CM), librería USS, 194
Módulo de señales (SM)
Tiempos de respuesta a un escalón para entradas analógicas, 408
Módulo de señales IO-Link Master, 437
Módulo tecnológico, SM 1278 4xIO-Link Master, 437
Módulos
Communication Boards (CB), 20
Configurar parámetros, 84, 88
Módulo de comunicación (CM), 20
Módulos de señales (SM), 20
Procesador de comunicaciones (CP), 20
Signal Board (SB), 20
Tabla de comparación, 19
Zona de disipación, 23, 26
Módulos CANopen
021620-B, 021630-B, 442
Módulos de comunicación RS232 y RS485, 192
Módulos de señales (SM)
Agregar módulos, 82
Agregar un dispositivo nuevo, 80
Configuración de dispositivos, 77

Representación de entradas analógicas (intensidad), 407
Representación de entradas analógicas (tensión), 406
Representación de salidas analógicas (intensidad), 410
Representación de salidas analógicas (tensión), 410
SM 1221 DI 16 24 VDC, 390
SM 1221 DI 8 24 VDC, 390
SM 1222 DQ 16 24 VDC, 391
SM 1222 DQ 16 relé, 391
SM 1222 DQ 8 24 VDC, 391
SM 1222 DQ 8 relé, 391
SM 1222 DQ 8 relé inversor, 391
SM 1223 DI 1223 DI 120/230 VAC/DQ relé, 394
SM 1223 DI 16/DQ 16 relé, 393
SM 1223 DI 8/ DQ 8/DQ relé, 393
SM 1223 DI 8/DQ 8, 393, 393
SM 1231 AI 4 x 13 bits, 402
SM 1231 AI 4 x 16 bits, 402
SM 1231 AI 4 x 16 bits TC, 416
SM 1231 AI 4 x RTD x 16 bits, 415
SM 1231 AI 8 x 13 bits, 402
SM 1231 AI 8 x 16 bits TC, 416
SM 1231 AI 8 x RTD x 16 bits, 415
SM 1232 AQ 2 x 14 bits, 402
SM 1232 AQ 4 x 14 bits, 402
SM 1234 AI 4 x 13 bits / AQ 2 x 14 bits, 402
SM 1278 4xIO-Link Master, 437
Vista general, 20
Módulos de señales analógicos
SM 1231, 402
SM 1231 de termopar, 416
SM 1231 RTD, 415
SM 1232, 402
SM 1234, 402
Módulos de señales digitales
SM 1221, 390
SM 1222, 391
SM 1223, 393, 394
Módulos no enchufados, 41
Montaje
Dimensiones, 23
Directrices, 24
Espacio, 26
Refrigeración, 26
Ventilación, 26
Zona de disipación, 23, 26
Motion Control
Finales de carrera por software y hardware, 308
MC_ChangeDynamic (cambiar la configuración de la dinámica del eje), 339
MC_CommandTable, 336
MC_Halt (detener eje), 325
MC_Home (referenciar eje), 322
MC_MoveAbsolute (posicionamiento absoluto del eje), 327
MC_MoveJog (desplazamiento del eje en modo Jog), 334
MC_MoveRelative (posicionamiento relativo del eje), 329
MC_MoveVelocity (mover el eje a la velocidad predefinida), 331
MC_Power (habilitar/bloquear eje), 318
MC_ReadParam (leer parámetros de un objeto tecnológico), 343
MC_Reset (confirmar error), 321
MC_WriteParam (escribir parámetros de un objeto tecnológico), 341
Parámetros de configuración de referenciación, 313
Referenciar el eje, 311
MRES, panel de operador, 35, 59, 346
My Documentation Manager, 4

N

Nivel de protección
Bloque lógico, 94
CPU, 91
Enlazar a una CPU, Memory Card o contraseña, 95
NORM_X (normalizar), 117
Nuevas funciones, 27
Nuevo proyecto
Agregar un dispositivo HMI, 52
Conexión de red, 53
Conexión HMI, 53
Pantalla HMI, 54
Primeros pasos, 43
Números de puerto
Asignación a interlocutores, 156

O

OB, (Bloque de organización)
Objetos tecnológicos
Motion Control, 279
PID, 201
Observar
Estado KOP, 348
Estado KOP y uso de la tabla de observación, 347
Instrucción LED, 134

Obtener valores de un DB, 353
Operación de forzado permanente, 351
Restablecer los valores iniciales de un DB, 353
Tabla de forzado permanente, 350
Observar el programa, 131
Obtención estado LED, 134
Obtener valores de un DB online, 353
Online
Botones RUN/STOP, 35
comparar y sincronizar, 355
Conectar online, 345
Determinar, 360
Dirección IP, 356
estado, 348
Forzar permanentemente, 350
hora, 356
Obtener valores de un DB, 353
Operación de forzado permanente, 351
Panel de operador, 35, 59, 346
Restablecer los valores iniciales de un DB, 353
tabla de observación, 347, 348
Vigilancia de estado o valor, 347
Vigilancia de la carga de memoria, 346
Vigilancia del tiempo de ciclo, 346
OPC, configuración, 186

P

Páginas HTML, definidas por el usuario, 266
Actualizar, 267
Desarrollar, 266
ubicaciones de páginas, 269
Páginas web
Servicio técnico, soporte y documentación de STEP 7, 4
Páginas web de STEP 7, 4
Páginas web definidas por el usuario, 261, 266
activar con instrucción WWW, 270
Actualizar, 267
Configurar, 269
Crear con editor HTML, 266
generar bloques de programa, 269
limitaciones de memoria de carga, 268
programación en STEP 7, 270
Páginas web estándar, 261, 262
Páginas web, definidas por el usuario, 266
Panel de operador, 35, 59, 346
Paneles (HMI), 21
Paneles Basic (HMI), 21
Paneles de operador, 21
Parametrización, 101
Parámetros de arranque, 84
Parámetros de salida, 101
PID
Algoritmo PID_3Step, 200
Algoritmo PID_Compact, 200
PID_3STEP (controlador PID con ajuste de válvulas), 209
PID_Compact (controlador PID universal con ajuste integrado), 203
PID_Temp (regulador PID universal que permite manejar el control de temperatura), 218
Puesta en servicio, 248
Vista general, 200
PLC
Agregar módulos, 82
comparar y sincronizar, 355
Configuración del HSC, 146
Copiar bloques desde una CPU online, 354
Forzar permanentemente, 350
Instrucciones, 48
Llamar un bloque, 105
observar, 347
Operación de forzado permanente, 351
Primeros pasos, 43
Protección de know-how, 94
Usar bloques, 60, 97
Variables, 44, 48
Vista general de la CPU, 15
Podcasts, 4
Primeros pasos
Bloque de programa, 103
Bloque lógico, 103
Conexión de red, 53
Conexión HMI, 53
Contactos, 46
CPU, 43
Direccionamiento, 48
Dividir el área de editores, 44, 48, 48
HMI, 52, 54
Instrucción Cuadro, 49
Instrucciones, 48
Instrucciones matemáticas, 49
Nuevo PLC, 43
Programa KOP, 46, 49
Proyecto, 43
Variables, 44, 48
Variables PLC, 44, 48
Prioridad
Clase de prioridad, 61
Prioridad de procesamiento, 62
Procesador de comunicaciones (CP)
Agregar módulos, 82

Tabla de comparación, 19
Vista general, 20
PROFIBUS
Agregar esclavo DP, 169
Agregar módulo CM 1243-5 (maestro DP), 169
Conexión de red, 148
Conexión S7, 177
Dirección, 170
Dirección, configurar, 170
Esclavo, 166
GET, 176
Maestro, 166
Módulo CM 1242-5 (esclavo DP), 166
Módulo CM 1243-5 (maestro DP), 166
Número de conexiones de comunicación, 152
PUT, 176
PROFINET, 147
Comprobar una red, 90
Conexión de red, 148
Conexión S7, 177
Dirección IP, 89
GET, 176
ID de conexión, 157
Modo ad hoc, 157
Número de conexiones de comunicación, 152
PUT, 176
Tipos de comunicación, 149
Vista general, 156
PROFINET RT, 156
Programa
Bloques de organización (OB), 99
Clase de prioridad, 61
Copiar bloques desde una CPU online, 354
Ejemplo de segmento, 46, 49
Enlazar a una CPU, Memory Card o contraseña, 95
Instrucciones matemáticas, 49
Llamar un bloque, 105
Obtener valores de un DB, 353
Primeros pasos, 46, 49
Protección por contraseña, 94
Restablecer los valores iniciales de un DB, 353
Programa de usuario
Agregar entradas o salidas a instrucciones KOP o FUP, 33
Bloques de organización (OB), 99
Copiar bloques desde una CPU online, 354
Drag & Drop entre editores, 38
Enlazar a una CPU, Memory Card o contraseña, 95
Favoritos, 33
Insertar instrucciones, 32
Instrucciones ampliables, 34
Protección por contraseña, 94
Programación
Agregar entradas o salidas a instrucciones KOP o FUP, 33
Algoritmo PID_3Step, 200
Algoritmo PID_Compact, 200
Clase de prioridad, 61
comparar y sincronizar bloques lógicos, 355
CPU sin especificar, 79, 360
Drag & Drop entre editores, 38
Enlazar a una CPU, Memory Card o contraseña, 95
Estructurada, 97
Favoritos, 33
FUP (Diagrama de funciones), 107
Insertar instrucciones, 32
Instrucciones ampliables, 34
KOP (Esquema de contactos), 106
Lineal, 97
Módulos no enchufados, 41
PID_3STEP (controlador PID con ajuste de válvulas), 209
PID_Compact (controlador PID universal con ajuste integrado), 203
PID_Temp (regulador PID universal que permite manejar el control de temperatura), 218
Primeros pasos, 48
SCL (Structured Control Language), 108, 109
Vista general de PID, 200
Programación con STEP 7
páginas web definidas por el usuario, 270
Programación estructurada, 97, 97
Programación lineal, 97
Propiedades de la CPU, para páginas web definidas por el usuario, 269
Protección anticopia
Enlazar a una CPU, Memory Card o contraseña, 95
Protección contra inversión de polaridad, 379
Protección de acceso, CPU, 91
Protección de know-how
Protección por contraseña, 94
Protección por contraseña
Acceso a la CPU, 91
Bloque lógico, 94
CPU, 91
Enlazar a una CPU, Memory Card o contraseña, 95
Protección anticopia, 95
Protocolo
Comunicación, 190
Freeport, 190
ISO on TCP, 156
Modbus, 190
PROFINET RT, 156
TCP, 156

UDP, 156
USS, 190
Protocolo Freeport, 190
Protocolo ISO on TCP, 156
Protocolo UDP, 156
Protocolos Ethernet, 156
Conexiones multinodo, 177
Proyecto
Agregar un dispositivo HMI, 52
comparar y sincronizar, 355
Conexión de red, 53
Conexión HMI, 53
Enlazar a una CPU, Memory Card o contraseña, 95
Pantalla HMI, 54
Primeros pasos, 43
Programa, 48
Protección de acceso, 91
Proteger un bloque lógico, 94
Restringir el acceso a la CPU, 91
Variables, 44, 48
Prueba de aislamiento a muy alta tensión, 378
PTO (tren de impulsos), 128
No se puede forzar permanentemente, 351
PUT, 176
PUT (escribir datos en una CPU remota)
Configurar la conexión, 164
PWM
Instrucción CTRL_PWM, 128
PWM (Modulación del ancho de impulso)
No se puede forzar permanentemente, 351

R

Recursos de información, 4
Reemplazar una CPU V3.0 por una CPU V4.1, 443
Referencia cruzada para mostrar la utilización, 132
Refrigeración, 26
Registro
Sinopsis de los registros, 129
Restablecer los ajustes de fábrica, 357
Restablecer los valores iniciales de un DB, 353
Restablecimiento de los ajustes de fábrica, 357
ROUND, 116
Router IP, 89

S

Salidas digitales
Especificaciones para relé, CPUs, SMs y SBs de 24 V DC, 398
SCALE_X (escalar), 117
SCL (Structured Control Language)
Algoritmo PID_3Step, 200
Algoritmo PID_Compact, 200
CEIL (redondear número en coma flotante a entero superior), 117
CONV (convertir), 116
DeviceStates, 134
Editor de programas, 109
Estado LED, 134
FLOOR, 117
GET_DIAG, 134
Get_IM_Data, 135
MC_ChangeDynamic (cambiar la configuración de la dinámica del eje), 339
MC_CommandTable, 336
MC_Halt (detener eje), 325
MC_Home (referenciar eje), 322
MC_MoveAbsolute (posicionamiento absoluto del eje), 327
MC_MoveJog (desplazamiento del eje en modo Jog), 334
MC_MoveRelative (posicionamiento relativo del eje), 329
MC_MoveVelocity (mover el eje a la velocidad predefinida), 331
MC_Power (habilitar/bloquear eje), 318
MC_ReadParam (leer parámetros de un objeto tecnológico), 343
MC_Reset (confirmar error), 321
MC_WriteParam (escribir parámetros de un objeto tecnológico), 341
ModuleStates, 135
NORM_X (normalizar), 117
PID_3STEP (controlador PID con ajuste de válvulas), 209
PID_Compact (controlador PID universal con ajuste integrado), 203
PID_Temp (regulador PID universal que permite manejar el control de temperatura), 218
redondear, 116
SCALE_X (escalar), 117
Sección Var, 109
Truncar, 116
Vista general, 108
Vista general de PID, 200
Secuencia de movimientos (MC_CommandTable), 336
Segmento
Conexión de red, 53
Segmento (de un tipo de datos marcado), 71
Segmento, programación KOP, 46
Seguridad
CPU, 91

Enlazar a una CPU, Memory Card o contraseña, 95
Protección anticopia, 95
Protección de acceso, 91
Protección de know-how de un bloque lógico, 94
Servicio y asistencia, 5
Servidor web, 261
Conexiones HTTP máximas, 265
Limitaciones, 265
Signal Board (SB)
Tiempos de respuesta a un escalón para entradas analógicas, 408
Signal Boards (SB)
Agregar módulos, 82
Configuración de dispositivos, 77
Representación de entradas (intensidad), 407
Representación de entradas (tensión), 406
Representación de salidas analógicas (intensidad), 410
Representación de salidas analógicas (tensión), 410
SB 1221 4 DI 24 VDC 200 kHz, 387
SB 1221 4 DI 5 VDC 200 kHz, 387
SB 1222 4 DQ 24 VDC 200 kHz, 387
SB 1222 4 DQ 5 VDC 200 kHz, 387
SB 1223 2 DI/2 DQ 24 VDC, 387
SB 1223 DI/DQ 24 VDC 200 kHz, 387
SB 1223 DI/DQ 5 VDC 200 kHz, 387
SB 1231 AI 1 x 12 bits, 401
SB 1231 AI 1 x 16 bits RTD, 413
SB 1231 AI 1 x 16 bits TC, 413
SB 1231 RTD, 413
SB 1231 TC, 413
SB 1232 AQ 1 x 12 bits, 401
Vista general, 20
Signal Boards (SM)
Agregar un dispositivo nuevo, 80
Signal Boards analógicas
SB 1231, 401
SB 1232, 401
Signal Boards digitales
SB 1221, 387
SB 1222, 387
SB 1223, 387
Sincronización, 277
Sinopsis de módulos de termopar, 412
Sinopsis de módulos RTD, 412
SM y SB
Tabla de comparación, 19
SMS, 185
Soporte técnico de Siemens, 5
STEP 7
Agregar entradas o salidas a una instrucción KOP o FUP, 33
Agregar módulos, 82
Agregar un dispositivo nuevo, 80
AS-i, 173
Botones RUN/STOP, 35
búfer de diagnóstico, 356
Cambiar la configuración, 36
Clase de prioridad (OB), 61
comparar y sincronizar, 355
Conexión de red, 148
Configuración de dispositivos, 77
Configuración del HSC, 146
Configurar la CPU, 84, 88
Configurar los módulos, 84, 88
Copiar bloques desde una CPU online, 354
Drag & Drop entre editores, 38
Entradas o salidas ampliables, 34
Favoritos, 33
Forzar permanentemente, 350
Insertar instrucciones, 32
Módulos no enchufados, 41
observar, 347, 348
Obtener valores de un DB, 353
Operación de forzado permanente, 351
Panel de operador, 35, 59, 346
PROFIBUS, 170
PROFINET, 89
Protección por contraseña, 94
Puerto Ethernet, 89
Restablecer los valores iniciales de un DB, 353
Vista del portal y Vista del proyecto, 31
Superposición de variable AT, 73
Support, 5
Sustitución de módulos, 41

T

Tabla de comparación
Dispositivos HMI, 21
Modelos de CPU, 17
Módulos, 19
Tabla de forzado permanente
direccionar entradas de periferia, 350
Forzar permanentemente, 350
Operación de forzado permanente, 351
Tabla de observación
Forzado permanente, 131
observar, 347
Tablas de observación, 348
Tarjeta de programa, 65

Task Cards
Columnas y encabezados, 37
TCON
Configuración, 163
ID de conexión, 157
Parámetros de la conexión, 161
TCON_Param, 161
TCP
Configuración de la conexión, 163, 163
ID de conexión, 157
Modo ad hoc, 157
Parámetros, 161
Protocolo, 156
Technical support, 5
Telecontrol, 181
TeleService vía GPRS, 181
Temporizadores
Cantidad, 18
Tamaño, 18
Tensiones nominales, 378, 378
TIA Portal
Agregar un dispositivo nuevo, 80
Configuración de dispositivos, 77
Configurar la CPU, 84, 88
Configurar los módulos, 88
PROFINET, 89
TIA Portal, Vista del portal y Vista del proyecto, 31
Tipo de datos DTL, 67
Tipo de datos DTL (Data and Time Long), 67
Tipos de datos, 66
DTL, 67
TRCV
ID de conexión, 157
TRCV (recibir datos vía Ethernet (TCP))
Modo ad hoc, 157
TRCV_C
Modo ad hoc, 157
Parámetros de la conexión, 161
TRCV_C (recibir datos vía Ethernet (TCP))
ID de conexión, 157
TRCV_C (recibir datos vía Ethernet (TCP))
Configuración, 163
Tren de impulsos (PTO), 128
TRUNC (truncar), 116
TS Adapter, 19
TSAP (Transport Service Access Points), 165
Instrucciones para la asignación a dispositivos, 156
TSEND
ID de conexión, 157
TSEND_C
Parámetros de la conexión, 161
TSEND_C (enviar datos vía Ethernet (TCP))
Configuración, 163
ID de conexión, 157
TURCV
Parámetros de la conexión, 161
TURCV (recibir datos vía Ethernet (UDP))
Configuración, 163
TUSEND
Parámetros, 161
TUSEND (enviar datos vía Ethernet (UDP))
Configuración, 163

U

UDP
Configuración de la conexión, 163
Parámetros, 161

V

Valores booleanos o de bit, 70
Valores iniciales
Obtener y restablecer los valores iniciales de un DB, 353
Variable
Operación de forzado permanente, 351
Segmento, 71
Superposición, 73
Vigilancia de estado o valor, 347
Variables
Primeros pasos, 44, 48
Variables PLC
Primeros pasos, 44, 48
Ventilación, 26
Versiones de instrucciones, 37
Vida útil de los relés, 379
Vigilancia de la carga de memoria, online, 346
Vigilancia del tiempo de ciclo, 346
Vista del portal, 31
Agregar un dispositivo nuevo, 80
Configurar el puerto Ethernet, 89
Configurar la CPU, 84, 88
Configurar los módulos, 84, 88
PROFINET, 89
Vista del proyecto, 31, 31
Agregar un dispositivo nuevo, 80
Configuración de dispositivos, 77
Configurar el puerto Ethernet, 89
Configurar los módulos, 84, 88
Configurar los parámetros de la CPU, 84, 88
PROFINET, 89

Visualización, dispositivos HMI, 21

W

WWW, 270

Z

Zona de disipación, 23, 26