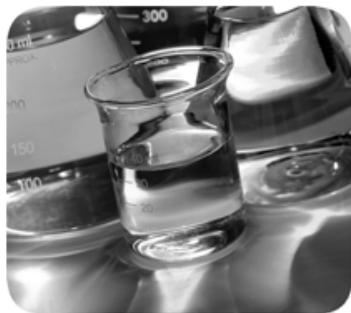




# Instrucciones generales de los controladores LOGIX5000 Manual de referencia

1756 ControlLogix, 1756 GuardLogix, 1769 CompactLogix, 1769 Compact GuardLogix, 1789 SoftLogix, 5069 CompactLogix, Emulate 5570





# Información importante para el usuario

Lea este documento y los documentos incluidos en la sección de recursos adicionales acerca de la instalación, configuración y operación de este equipo antes de instalar, configurar, operar o mantener este producto. Es necesario que los usuarios se familiaricen con la instalación y las instrucciones de cableado además de los requisitos de todos los códigos, leyes y estándares aplicables.

Actividades como la instalación, ajustes, puesta en servicio, uso, montaje, desmontaje y mantenimiento deben ser llevadas a cabo por personal formado adecuadamente según el código de prácticas aplicable. Si se utiliza este equipo de algún modo no especificado por el fabricante, la protección ofrecida por el equipo puede verse perjudicada.

Bajo ningún concepto será Rockwell Automation, Inc. responsable de los daños indirectos o emergentes que resulten del uso o de la aplicación de estos equipos.

Los ejemplos y diagramas de este manual se incluyen únicamente con carácter ilustrativo. Debido a la gran cantidad de variables y requisitos asociados a cualquier instalación en particular, Rockwell Automation Inc. no puede asumir responsabilidad alguna por el uso basado en dichos ejemplos y diagramas.

Rockwell Automation, Inc. no asume responsabilidad alguna por patentes con respecto al uso de la información, los circuitos, los equipos o el software descritos en este manual.

Queda prohibida la reproducción total o parcial del contenido de este manual sin el permiso por escrito de Rockwell Automation, Inc.

A lo largo del manual, hacemos uso de notas para advertirle de consideraciones de seguridad cuando es necesario.



**WARNING:** Identifica información sobre prácticas o circunstancias que pueden provocar una explosión en un entorno peligroso, las cuales pueden producir lesiones o la muerte, daños materiales o pérdida económica.



**ATENCIÓN:** Identifica información sobre prácticas o circunstancias que pueden producir lesiones o la muerte, daños materiales o pérdida económica. Estos mensajes le ayudan a identificar un peligro, a evitarlo y a reconocer las consecuencias.

---

**Importante:** Identifica información crítica para aplicar y comprender correctamente el producto.

---

Estas etiquetas también pueden encontrarse en el exterior o en el interior del equipo para proporcionar precauciones específicas.



**RIESGO DE DESCARGA:** Estas etiquetas pueden encontrarse en el exterior o en el interior de los equipos, por ejemplo, en variadores o motores, para alertar a las personas de una tensión peligrosa.



**RIESGO DE QUEMADURAS:** Estas etiquetas pueden encontrarse en el exterior o en el interior de los equipos, por ejemplo, en variadores o motores, para alertar a las personas de que las superficies pueden alcanzar temperaturas peligrosas.

---



**RIESGO DE ARCO ELÉCTRICO** Estas etiquetas pueden encontrarse en el exterior o en el interior de los equipos, por ejemplo, en centro de control del motor, para advertir a las personas sobre un posible arco eléctrico. Un arco eléctrico puede provocar graves lesiones o la muerte. Utilice equipamiento de protección personal (PPE) adecuado. Observe TODOS los requisitos normativos de prácticas de seguridad laboral y de equipamiento de protección personal (PPE).

---

Allen-Bradley, Rockwell Software, Rockwell Automation y TechConnect son marcas comerciales de Rockwell Automation, Inc.

Las marcas comerciales que no pertenecen a Rockwell Automation son propiedad de las empresas correspondientes.

# Resumen de cambios

---

Este manual incluye información nueva y actualizada. Utilice estas tablas de referencia para localizar la información modificada.

## Cambios generales

Esta tabla identifica los cambios que se aplican a toda la información relativa a una cuestión del manual y el motivo del cambio. Por ejemplo, la adición de nuevo hardware compatible, un cambio en el diseño del software o más material de referencia, daría como resultado cambios en todos los temas relacionados con esa cuestión.

Cuestión	Motivo
Todos los temas de instrucción.	Se han añadido los nuevos controladores de seguridad 5580 y 5380 a la lista de controladores aplicables.
Todos los temas de instrucción	Se han actualizado el Diagrama de escalera, la Señal de restablecimiento y las imágenes de los Ejemplos para reflejar la interfaz de usuario actualizada.
Todos los temas de instrucción	En los Fallos mayores/menores y en la sección Consulte también, se ha reemplazado la referencia cruzada de Atributos comunes con una nueva referencia cruzada con el tema Índice a través de matrices.
Todos los temas de los títulos de los capítulos de la instrucción	Añadida una nueva tabla de enlaces con enlaces a cada instrucción en el capítulo

## Características nuevas o mejoradas

Esta tabla contiene una lista de los temas modificados en esta versión, el motivo de cambio y un enlace al tema que contiene la información modificada.

Nombre del tema	Motivo
<a href="#">Un impulso (ONS)</a> en la página 148	Se ha movido la sección Descripción al párrafo introductorio.
	En la sección Operandos, se ha añadido una nota Importante.
	Se ha actualizado la columna descripción en la tabla Diagrama de escalera.
	Se ha actualizado la columna de acción realizada en la sección Ejecución.
<a href="#">Un impulso en flanco descendente (OSF)</a> en la página 151	Se ha movido la sección Descripción al párrafo introductorio.

Nombre del tema	Motivo
	En la sección Operandos, se ha añadido una nota Importante.
	Se ha actualizado la columna descripción en la tabla Diagrama de escalera.
	Se ha actualizado la columna de acción realizada en la sección Ejecución.
<a href="#">Un impulso en flanco ascendente (OSR)</a> en la página 157	Se ha movido la sección Descripción al párrafo introductorio.
	En la sección Operandos, se ha añadido una nota Importante.
	Se ha actualizado la columna descripción en la tabla Diagrama de escalera.
	Se ha actualizado la columna de acción realizada en la sección Ejecución.
<a href="#">Conteo descendente (CTD)</a> en la página 172	En la sección Operandos se ha añadido una nota importante y se ha actualizado la tabla de Diagrama de escalera y la tabla Estructura de COUNTER.
	Se ha actualizado la columna de acción realizada en la tabla Diagrama de escalera en la sección Ejecución.
<a href="#">Conteo ascendente (CTU)</a> en la página 177	En la sección Operandos se ha añadido una nota importante y se ha actualizado la tabla de Diagrama de escalera y la tabla Estructura de COUNTER.
	Se ha actualizado la columna de acción realizada en la tabla Diagrama de escalera en la sección Ejecución.
<a href="#">Restablecer (RES)</a> en la página 187	En la sección Operandos, se ha añadido una nota Importante.
	En la sección Descripción, se ha actualizado la columna de borrados de instrucciones.
	Se ha actualizado la columna de acción realizada en la sección Ejecución.
	Se ha actualizado la sección Ejemplo de restablecimiento.
<a href="#">Temporizador retentivo activado (RTO)</a> en la página 191	En la sección Operandos se ha añadido una nota importante y se ha actualizado la tabla de Diagrama de escalera y la tabla Estructura de COUNTER.
	Se ha actualizado la sección Descripción

Nombre del tema	Motivo
	Se ha actualizado la columna de acción realizada en la tabla Diagrama de escalera en la sección Ejecución.
<a href="#">Temporizador de retardo a la desconexión (TOF)</a> en la página 201	En la sección Operandos se ha añadido una nota importante y se ha actualizado la tabla de Diagrama de escalera y la tabla Estructura de COUNTER.
	Se ha actualizado la sección Descripción.
	Se ha actualizado la columna de acción realizada en la tabla Diagrama de escalera en la sección Ejecución.
<a href="#">Temporizador de retardo a la conexión (TON)</a> en la página 211	En la sección Operandos se ha añadido una nota importante y se ha actualizado la tabla de Diagrama de escalera y la tabla Estructura de COUNTER.
	Se ha actualizado la sección Descripción.
	Se ha actualizado la columna de acción realizada en la tabla Diagrama de escalera en la sección Ejecución.
<a href="#">Comparar (CMP)</a> en la página 370	Se ha movido la sección Descripción a la introducción.
	Se ha añadido una nota importante en la sección Operandos.
	Se ha eliminado la tabla Operadores válidos y se ha añadido una referencia cruzada al tema Operadores válidos en la sección Consulte también.
	Se ha actualizado la columna de acción realizada en la sección Ejecución.
<a href="#">Mayor que (GRT)</a> en la página 380	En la sección Comparación de cadenas, se ha actualizado la lista de controladores.
	Se ha actualizado la columna de acción realizada en la sección Ejecución.
<a href="#">Límite (LIM)</a> en la página 404	Se ha actualizado la columna de acción realizada en la sección Ejecución.
<a href="#">Valor absoluto (ABS)</a> en la página 426	Se ha movido la sección Descripción a la introducción.

Nombre del tema	Motivo
	En la sección Texto estructurado, se ha añadido un consejo.
	Se ha añadido una nota importante en la sección Operandos.
	Se ha actualizado la columna de acción realizada en la sección Ejecución.
	En la sección Ejemplos, se ha actualizado un ejemplo de Texto estructurado.
<a href="#">Sumar (ADD)</a> en la página 430	Se ha movido la sección Descripción al párrafo introductorio.
	Se ha añadido un consejo en Texto estructurado.
	Se ha añadido una nota importante en la sección Operandos.
	Se ha actualizado la lista de controladores en la sección Afectar a las marcas de estado matemático.
	Se ha actualizado la columna de Acción realizada en la sección Ejecución.
	Se ha actualizado el ejemplo de Texto estructurado.
<a href="#">Dividir (DIV)</a> en la página 439	Se ha movido la sección Descripción al párrafo introductorio.
	Se ha añadido un consejo en Texto estructurado.
	Se ha añadido una nota importante en la sección Operandos.
	Se ha actualizado la lista de controladores en la sección Afectar a las marcas de estado matemático.
	Se ha actualizado la columna de Acción realizada en la sección Ejecución.
	Se ha actualizado el ejemplo de Texto estructurado
<a href="#">Mayor o igual que (GEQ)</a> en la página 386	En la sección Comparación de cadenas, se ha actualizado el Consejo.
	Se ha actualizado la columna de acción realizada en la sección Ejecución.
<a href="#">Menor que (LES)</a> en la página 392	En la sección Comparación de cadenas, se ha actualizado el Consejo.
	Se ha actualizado la columna de acción realizada en la sección Ejecución.
<a href="#">Menor o igual que (LEQ)</a> en la página 398	En la sección Comparación de cadenas, se ha actualizado el Consejo.
	Se ha actualizado la columna de acción realizada en la sección Ejecución.

Nombre del tema	Motivo
<a href="#">Máscara igual a (MEQ)</a> en la página 410	En la sección Comparación de cadenas, se ha actualizado el Consejo.
	Se ha actualizado la columna de acción realizada en la sección Ejecución.
<a href="#">No igual a (NEQ)</a> en la página 415	En la sección Comparación de cadenas, se ha actualizado el Consejo.
	Se ha actualizado la columna de acción realizada en la sección Ejecución.
<a href="#">Calcular (CPT)</a> en la página 434	<p>Se ha movido la sección Descripción a la introducción.</p> <p>Se ha movido la tabla de Operadores válidos al tema Operadores válidos.</p> <p>Se han actualizado las secciones Afectar a las marcas de estado matemático y Ejecución.</p>
<a href="#">Módulo (MOD)</a> en la página 443	<p>En la sección Texto estructurado, se ha añadido un consejo.</p> <p>Se ha añadido una nota importante en la sección Operandos.</p>
	Se ha actualizado la sección Descripción.
	En la sección Afectar a las marcas de estado matemático, se ha actualizado la lista de controladores.
	Se ha actualizado la columna de acción realizada en la sección Ejecución y añadido un nuevo consejo.
	En la sección Ejemplo, se ha añadido un ejemplo de Texto estructurado.
<a href="#">Negación (NEG)</a> en la página 452	<p>Se ha movido la sección Descripción a la introducción.</p> <p>Se ha añadido una nota importante en la sección Operandos.</p>
	Se ha actualizado la columna de acción realizada en la sección Ejecución y añadido un nuevo consejo.
	En la sección Ejemplo, se ha añadido un ejemplo de Texto estructurado.
<a href="#">Raíz cuadrada (SQR/SQRT)</a> en la página 456	<p>En la sección Texto estructurado, se ha añadido un consejo.</p> <p>Se ha añadido una nota importante en la sección Operandos.</p>
	En la sección Descripción, se ha añadido un segundo ejemplo y se ha actualizado la lista de controladores en la tabla.

Nombre del tema	Motivo
	Se ha actualizado la columna de acción realizada en la sección Ejecución.
	En la sección Ejemplo, se ha añadido un ejemplo de Texto estructurado.
<a href="#">Y a nivel de bits (AND)</a> en la página 477	En la sección Texto estructurado, se ha añadido un consejo.
	Se ha añadido una nota importante en la sección Operandos.
	En la sección Diagrama de escalera, se han añadido consejos a la columna Descripción y un consejo después de la tabla.
	Se ha actualizado la sección Descripción.
	Se ha actualizado la columna de acción realizada en la sección Ejecución.
	En la sección Ejemplo, se ha añadido un ejemplo de Texto estructurado.
<a href="#">O exclusivo a nivel de bits (XOR)</a> en la página 481	En la sección Texto estructurado, se ha añadido un nuevo consejo.
	En la sección Operandos, se ha añadido una nota Importante.
	En la tabla Diagrama de escalera, se han actualizado las descripciones.
	Se ha añadido un nuevo consejo debajo de la tabla Diagrama de escalera.
	Se ha actualizado la lista de controladores en la sección Afectar a las marcas de estado matemático.
	Se ha actualizado la columna de acción realizada en la sección Ejecución.
<a href="#">NO a nivel de bits (NOT)</a> en la página 486	En la sección Texto estructurado, se ha añadido un consejo.
	En la sección Operandos, se ha añadido una nota Importante.
	En la tabla Diagrama de escalera, se han actualizado las descripciones. Se ha añadido un nuevo consejo debajo de la tabla Diagrama de escalera.
	En la sección Afectar a las marcas de estado matemático, se ha añadido a la lista de controladores.
	Se ha actualizado la columna de acción realizada en la sección Ejecución.

Nombre del tema	Motivo	
<a href="#">O a nivel de bits (OR)</a> en la página 490	En la sección Texto estructurado, se ha añadido un nuevo consejo.	
	En la sección Operandos, se ha añadido una nota Importante.	
	En la tabla Diagrama de escalera, se han actualizado las descripciones.	
	Se ha actualizado la lista de controladores en la sección Afectar a las marcas de estado matemático.	
	Se ha actualizado la columna de acción realizada en la sección Ejecución.	
<a href="#">Borrar (CLR)</a> en la página 507	En la sección Operandos, se ha añadido una nota Importante.	
	Se ha actualizado la lista de controladores en la sección Afectar a las marcas de estado matemático.	
	Se ha actualizado la columna de acción realizada en la sección Ejecución.	
	<a href="#">Mover con máscara (MVM)</a> en la página 509	Se ha movido la sección Descripción a la introducción.
		Se ha añadido una nota importante en la sección Operandos.
		Se ha actualizado la lista de controladores en la sección Marcas de estado matemático.
		Se ha actualizado la lista de controladores en la sección Fallos menores/mayores.
<a href="#">Intercambiar byte (SWPB)</a> en la página 520	Se ha movido la sección Descripción a la introducción.	
	Se ha añadido una nota importante en la sección Operandos.	
	Se han actualizado las tablas Diagrama de escalera y Texto estructurado en la sección Operandos.	
<a href="#">Copiar archivo (COP), Copiar archivo sincrónico (CPS)</a> en la página 528	Se ha movido la sección Descripción a la introducción.	
	Se ha añadido una nota importante en la sección Operandos.	

Nombre del tema	Motivo
	Se ha actualizado la lista de controladores en la sección Marcas de estado matemático.
	Se ha actualizado la lista de controladores en la sección Fallos menores/mayores.
<a href="#">Búsqueda y comparación de archivos (FSC)</a> en la página 563	Se ha añadido una cadena a la lista Tipos de datos para el operando Expresión.
	Se ha actualizado la Tabla de operadores válidos.
<a href="#">Desplazamiento de bit a la derecha (BSR)</a> en la página 609	Se ha movido la sección Descripción al párrafo introductorio.
<a href="#">Restar (SUB)</a> en la página 461	Se ha movido la sección Descripción al párrafo introductorio.
	Se ha añadido un consejo en Texto estructurado.
	Se ha añadido una nota importante en la sección Operandos.
	Se ha actualizado la lista de controladores en la sección Afectar a las marcas de estado matemático.
	Se ha actualizado la columna de Acción realizada en la sección Ejecución.
	Se ha actualizado el ejemplo de Texto estructurado.
<a href="#">Tamaño en elementos (SIZE)</a> en la página 590	Se ha movido la sección Descripción al párrafo introductorio.
	Se ha añadido una nota importante en la sección Operandos.
	Se ha actualizado la columna de Acción realizada en la sección Ejecución.
<a href="#">Multiplicar (MUL)</a> en la página 448	Se ha movido la sección Descripción al párrafo introductorio.
	Se ha añadido un consejo en Texto estructurado.
	Se ha añadido una nota importante en la sección Operandos.

Nombre del tema	Motivo
	Se ha actualizado la columna de Acción realizada en la sección Ejecución. Se ha actualizado el ejemplo de Texto estructurado.
<a href="#">Saltar a subrutina (JSR), Subrutina (SBR) y Retorno (RET)</a> en la página 672	Se ha movido la sección Descripción al párrafo introductorio. En la sección Operandos, se ha añadido una nota importante, una nota de advertencia y se ha actualizado la tabla Diagrama de escalera.
<a href="#">Aritmética y lógica de archivo (FAL)</a> en la página 536	Se ha movido la sección Descripción al párrafo introductorio. En la sección Operandos, se ha añadido una nota importante, una nota de advertencia y se ha actualizado la tabla Diagrama de escalera.
	Se ha actualizado la lista de controladores en la sección Marcas de estado matemático.
<a href="#">Llenar archivo (FLL)</a> en la página 560	Se ha movido la sección Descripción al párrafo introductorio.
	En la sección Operandos, se ha añadido una nota importante, una nota de advertencia y se ha actualizado la tabla Diagrama de escalera.



## Localizador de instrucciones

---

Utilice este localizador para buscar cada instrucción en el manual de instrucciones de controladores Logix5000 aplicable.

Manual de referencia Instrucciones generales de los controladores Logix5000 1756-RM003	Manual de referencia Logix5000 Controllers Advanced Process Control and Drives and Equipment Phase and Sequence Instructions 1756-RM006	Manual de referencia Logix5000 Controllers Motion Instructions MOTION-RM002
Valor absoluto (ABS)	Alarma (ALM)	Control coordinado accionado maestro (MDCC)
Sumar (ADD)	Conectar a fase de equipo (PATT)	Aplicar ajustes a eje de movimiento (MAAT)
Alarma analógica (ALMA)	Conectar a secuencia de equipo (SATT)	Aplicar diagnósticos de conexión de movimiento (MAHD)
Instrucción siempre falso (AFI)	Control coordinado (CC)	Leva de salida de armado de movimiento (MAOC)
Arco coseno (ACS, ACOS)	Circuito multivibrador D (DFF)	Registro de armado de movimiento (MAR)
Arco seno (ASN, ASIN)	Tiempo muerto (DEDT)	Supervisión de armado de movimiento (MAW)
Arco tangente (ATN, ATAN)	Derivada (DERV)	Restablecimiento de fallo de eje de movimiento (MAFR)
Caracteres ASCII en el búfer (ACB)	Desconectar de fase de equipo (PDET)	Engranaje de eje de movimiento (MAG)
Borrar ASCII búfer (ACL)	Desconectar de secuencia de equipo (SDET)	Posición inicial de eje de movimiento (MAH)
Líneas de handshake ASCII (AHL)	Dispositivos de 3 estados discreto (D3SD)	Impulso de eje de movimiento (MAJ)
Lectura ASCII (ARD)	Dispositivo de 2 estados discreto (D2SD)	Mover eje de movimiento (MAM)
Lectura ASCII de línea (ARL)	PID mejorado (PIDE)	Leva de posición de eje de movimiento (MAPC)
Prueba ASCII para línea de búfer (ABL)	Selección mejorada (ESEL)	Paro de eje de movimiento (MAS)
Escritura ASCII (AWT)	Fallo de borrado de fase de equipo (PCLF)	Leva de tiempo de eje de movimiento (MATC)
Escritura ASCII con anexo (AWA)	Comando de fase de equipo (PCMD)	Desactivación de eje de movimiento (MASD)
Distribuir campo de bits con receptor (BTD)	Solicitud externa de fase de equipo (PXRQ)	Restablecer desactivación de eje de movimiento (MASR)
Distribuir campo de bits con receptor (BTDT)	Fallo de fase de equipo (PFL)	Perfil de leva de cálculo de movimiento (MCCP)
Desplazamiento de bit a la izquierda (BSL)	Parámetros de nueva fase de equipo (PRNP)	Movimiento de ruta de movimiento coordinado (MCPM)
Desplazamiento de bit a la derecha (BSR)	Comando de anulación de fase de equipo (POVR)	Valores esclavos de cálculo de movimiento (MCSV)

<b>Manual de referencia Instrucciones generales de los controladores Logix5000 1756-RM003</b>	<b>Manual de referencia Logix5000 Controllers Advanced Process Control and Drives and Equipment Phase and Sequence Instructions 1756-RM006</b>	<b>Manual de referencia Logix5000 Controllers Motion Instructions MOTION-RM002</b>
Y a nivel de bits (AND)	Fase de equipo en pausa (PPD)	Transformada de movimiento coordinado con orientación (MCTO)
NO a nivel de bits (NOT)	Secuencia de equipo asigna identificador de secuencia (SASI)	Posición de transformada de cálculo de movimiento (MCTP)
O a nivel de bits (OR)	Fallo al borrar la secuencia de equipo (SCLF)	Posición de transformada de cálculo de movimiento con orientación (MCTPO)
Y booleano (BAND)	Comando de secuencia de equipo (SCMD)	Dinámica de cambio de movimiento (MCD)
O exclusivo booleano (BXOR)	Anular secuencia de equipo (SOVR)	Dinámica de cambio coordinado de movimiento (MCCD)
NO booleano (BNOT)	Function Generator (FGEN)	Movimiento circular coordinado de movimiento (MCCM)
O booleano (BOR)	Filtro de paso alto (HPF)	Movimiento lineal de coordenadas de movimiento (MCLM)
Interrupción (BRK)	Límite alto/bajo (HLL)	Desactivación de coordenadas de movimiento (MCSD)
Puntos de interrupción (BPT)	Integrador (INTG)	Restablecimiento de desactivación de coordenadas de movimiento (MCSR)
Borrar (CLR)	Control de modelo interno (IMC)	Paro de coordenadas de movimiento (MCS)
Comparar (CMP)	Circuito multivibrador JK (JKFF)	Transformada de coordenadas de movimiento (MCT)
Convertir en BCD (TOD)	Adelanto-retardo (LDLG)	Variador directo de movimiento desactivado (MDF)
Convertir en entero (FRD)	Filtro de paso bajo (LPF)	Variador directo de movimiento activado (MDO)
Copiar archivo (COP), Copiar archivo sincrónico (CPS)	Captura máxima (MAXC)	Iniciar movimiento directo (MDS)
Coseno (COS)	Captura mínima (MINC)	Leva de salida de desactivación de movimiento (MDOC)
Calcular (CPT)	Control modular de varias variables (MMC)	Registro de desarme de movimiento (MDR)
Conteo descendente (CTD)	Promedio de movimiento (MAVE)	Observar desarme de movimiento (MDW)
Conteo ascendente (CTU)	Desviación estándar de movimiento (MSTD)	Desactivación de grupo de movimiento (MGSD)
Conteo ascendente/descendente CTUD	Multiplexor (MUX)	Restablecimiento de desactivación de grupo de movimiento (MGSR)
Transición de datos (DTR)	Filtro de muesca (NTCH)	Paro de grupo de movimiento (MGS)
Grados (DEG)	Estado de fase completo (PSC)	Posición de estroboscopio de grupo de movimiento (MGSP)
Detección de diagnóstico (DDT)	Posición proporcional (POSP)	Posición de redefinir movimiento (MRP)

<b>Manual de referencia Instrucciones generales de los controladores Logix5000 1756-RM003</b>	<b>Manual de referencia Logix5000 Controllers Advanced Process Control and Drives and Equipment Phase and Sequence Instructions 1756-RM006</b>	<b>Manual de referencia Logix5000 Controllers Motion Instructions MOTION-RM002</b>
Alarma digital (ALMD)	Proporcional + Integral (PI)	Ajuste de eje de movimiento de marcha (MRAT)
DINT en cadena (DTOS)	Multiplicador de impulso (PMUL)	Diagnósticos de conexión de movimiento de marcha (MRHD)
Dividir (DIV)	Rampa/estabilización (RMPS)	Servo de movimiento desactivado (MSF)
Fin de transición (EOT)	Limitador de régimen (RLIM)	Servo de movimiento activado (MSO)
Igual a (EQU)	Restablecimiento dominante (RESD)	
Aritmética y lógica de archivo (FAL)	Escala (SCL)	
Comparación de bits de archivo (FBC)	Curva en S (SCRV)	
Carga FIFO (FFL)	Controlador de segundo orden (SOC)	
Descarga FIFO (FFU)	Adelanto-retardo de segundo orden (LDL2)	
Promedio de archivo (AVE)	Seleccionar (SEL)	
Desviación estándar de archivo (STD)	Rechazo seleccionado (SNEG)	
Llenar archivo (FLL)	Sumador seleccionado (SSUM)	
Clasificación de archivo (SRT)	Establecimiento dominante (SETD)	
Encontrar cadena (FIND)	Proporcional de tiempo de rango dividido (SRTP)	
Instrucción FOR (FOR)	Totalizador (TOT)	
Búsqueda y comparación de archivos (FSC)	Acumulador progresivo/regresivo (UPDN)	
Obtener valor del sistema (GSV) y Establecer valor del sistema (SST)		
Mayor o igual que (GEQ)		
Mayor que (GRT)		
Insertar cadena (INSERT)		
Salida inmediata (IOT)		
Saltar a etiqueta (JMP) y Etiqueta (LBL)		
Saltar a subrutina (JSR), Subrutina (SBR) y Retorno (RET)		
Saltar a subrutina externa (JXR)		
Menor que (LES)		
Menor o igual que (LEQ)		

<b>Manual de referencia Instrucciones generales de los controladores Logix5000 1756-RM003</b>	<b>Manual de referencia Logix5000 Controllers Advanced Process Control and Drives and Equipment Phase and Sequence Instructions 1756-RM006</b>	<b>Manual de referencia Logix5000 Controllers Motion Instructions MOTION-RM002</b>
Carga LIFO (LFL)		
Descarga LIFO (LFU)		
Validación de licencia (LV)		
Límite (LIM)		
Logaritmo base (LOG)		
Minúsculas (LOWER)		
Mover con máscara (MVM)		
Mover con máscara con receptor (MVTM)		
Restablecimiento de control maestro (MCR)		
Máscara igual que (MEQ)		
Mensaje (MSG)		
Cadena central (MID)		
Módulo (MOD)		
Mover (MOV)		
Multiplicar (MUL)		
Logaritmo natural (LN)		
Cambiar signo (NEG)		
Diferente de (NEQ)		
Sin operación (NOP)		
Un impulso (ONS)		
Un impulso en flanco descendente(OSF)		
Un impulso en flanco descendente con entrada (OSFI)		
Un impulso en flanco ascendente (OSR)		
Un impulso en flanco ascendente con entrada (OSRI)		
Activación de salida (OTE)		
Enclavamiento de salida (OTL)		
Desenclavamiento de salida (OTU)		
PID mejorado (PID)		
Radianes (RAD)		
Real en cadena (RTOS)		
Restablecer (RES)		
Restablecer SFC (SFR)		
Retorno (RET)		

<b>Manual de referencia Instrucciones generales de los controladores Logix5000 1756-RM003</b>	<b>Manual de referencia Logix5000 Controllers Advanced Process Control and Drives and Equipment Phase and Sequence Instructions 1756-RM006</b>	<b>Manual de referencia Logix5000 Controllers Motion Instructions MOTION-RM002</b>
Temporizador retentivo activado (RTO)		
Temporizador retentivo activado con restablecimiento (RTOR)		
Pausa SFC (SFP)		
Tamaño en elementos (SIZE)		
Secuenciador de entrada (SQI)		
Carga de secuenciador (SQL)		
Secuenciador de salida (SQO)		
Seno (SIN)		
Raíz cuadrada (SQR/SQRT)		
Concatenar cadenas (CONCAT)		
Eliminar cadena (DELETE)		
Cadena en DINT (STOD)		
Cadena en REAL (STOR)		
Intercambiar byte (SWPB)		
Restar (SUB)		
Tangente (TAN)		
Temporizador de retardo a la desconexión (TOF)		
Temporizador de retardo a la desconexión con restablecimiento (TOFR)		
Temporizador de retardo a la conexión (TON)		
Temporizador de retardo a la conexión con restablecimiento (TONR)		
Fin temporal (TND)		
Puntos de rastreo (TPT)		
Desencadenar tarea de evento (EVENT)		
Truncar (TRN)		
Instrucción desconocida (UNK)		
Mayúsculas (UPPER)		
Inhabilitación de interrupción de usuario (UID) / Habilitación de interrupción de usuario (UIE)		
X a la potencia de Y (XPY)		
Examinar si cerrado (XIC)		
Examinar si abierto (XIO)		
O exclusivo a nivel de bits (XOR)		



## Tabla de contenido

---

### **Resumen de cambios**

Entorno de Studio 5000 .....	30
Recursos adicionales.....	30
Avisos legales .....	30

### **Localizador de instrucciones**

### **Prefacio Instrucciones de alarma**

## **Capítulo 1**

Instrucciones de alarma.....	35
Alarma analógica (ALMA): Bloque de funciones .....	36
Alarma analógica (ALMA) – Diagrama de escalera .....	60
Alarma analógica (ALMA): Texto estructurado .....	86
Alarma digital (ALMD): Bloque de funciones .....	108
Alarma digital (ALMD) - Diagrama de escalera .....	120
Alarma digital (ALMD) - Texto estructurado .....	131

## **Capítulo 2**

### **Instrucciones de bit**

Instrucciones de bit.....	143
Examinar si cerrado (XIC) .....	144
Examinar si abierto (XIO).....	146
Un impulso (ONS) .....	148
Un impulso en flanco descendente(OSF) .....	151
Un impulso en flanco descendente con entrada (OSFI).....	154
Un impulso en flanco ascendente (OSR) .....	157
Un impulso en flanco ascendente con entrada (OSRI).....	161
Activación de salida (OTE).....	164
Enclavamiento de salida (OTL) .....	166
Desbloqueo de salida (OTU).....	168

## **Capítulo 3**

### **Instrucciones de temporizador y contador**

Instrucciones de temporizador y contador.....	171
Conteo descendente (CTD) .....	172
Conteo ascendente (CTU) .....	177
Conteo ascendente/descendente (CTUD) .....	182
Restablecer (RES).....	187
Temporizador retentivo activado (RTO) .....	191
Temporizador retentivo activado con restablecimiento (RTOR) .....	196
Temporizador de retardo a la desconexión (TOF) .....	201
Temporizador de retardo a la desconexión con restablecimiento (TOFR) ..	206
Temporizador de retardo a la conexión (TON) .....	211
Temporizador de retardo a la conexión con restablecimiento (TONR) .....	216

## Capítulo 4

### Entrada/salida:

Instrucciones de entrada/salida .....	223
Mensaje (MSG) .....	224
Ejemplos de configuración de MSG .....	234
Códigos y tipos de fallos mayores .....	235
Códigos y tipos de fallos menores .....	241
Códigos de error de mensaje .....	244
Códigos de error .....	245
Códigos de error extendidos .....	246
Códigos de error PLC y SLC (.ERR) .....	249
Códigos de error en la transferencia en bloque .....	252
Especificación de detalles de comunicación .....	253
Especificar mensajes SLC .....	263
Especificar mensajes de transferencia en bloques .....	263
Obtener valor del sistema (GSV) y Establecer valor del sistema (SSV) .....	264
Salida inmediata (IOT) .....	268
Acceso a los valores del sistema .....	271
Determinar la información de la memoria del controlador .....	272
Códigos de estado de DeviceNet .....	275
Obtener y establecer valores del sistema .....	278
Ejemplo de programación GSV/SSV .....	280
Objetos GSV/SSV .....	284
Acceso al objeto AddOnInstructionDefinition .....	286
Acceso al objeto ALARMBUFFER .....	286
Acceso al objeto Axis .....	289
Acceso al objeto Controller .....	300
Acceso al objeto ControllerDevice .....	302
Acceso al objeto CoordinateSystem .....	304
Acceso al objeto MotionGroup .....	306
Acceso al objeto Message .....	307
Acceso al objeto CST .....	308
Acceso al objeto Datalog .....	309
Acceso al objeto DF1 .....	312
Acceso al objeto FaultLog .....	315
Acceso al objeto HardwareStatus .....	316
Acceso al objeto Message .....	318
Acceso al objeto Module .....	318
Acceso al objeto Routine .....	321
Acceso al objeto Redundancy .....	321
Acceso al objeto Program .....	325
Acceso al objeto de Seguridad .....	326
Acceso al objeto SerialPort .....	328
Acceso al objeto Task .....	329
Acceso al objeto TimeSynchronize .....	331
Acceso al objeto WallClockTime .....	335
Objetos de seguridad GSV/SSV .....	336

Marcas de estado de monitor .....	342
Seleccionar del tipo de mensaje .....	342
Fallos de módulo: 16#0000 - 16#00ff.....	344
Fallos de módulo: 16#0100 - 16#01ff.....	346
Fallos de módulo: 16#0200 - 16#02ff.....	351
Fallos de módulo: 16#0300 - 16#03ff.....	352
Fallos de módulo: 16#0800 - 16#08ff.....	355
Fallos de módulo: 16#fd00 - 16#fdff.....	356
Fallos de módulo: 16#fe00 - 16#feff.....	357
Fallos de módulo: 16#ff00 - 16#ffff.....	359
Especificar mensajes CIP .....	360
Especificar mensajes PLC-3 .....	366
Especificar mensajes PLC-5 .....	367
Especificar mensajes PLC-2 .....	368

## Capítulo 5

### Comparar instrucciones

Comparar instrucciones.....	369
Comparar (CMP) .....	370
Igual a (EQU) .....	374
Mayor que (GRT) .....	380
Mayor o igual que (GEQ) .....	386
Menor que (LES).....	392
Menor o igual a (LEQ).....	398
Límite (LIM).....	404
Máscara igual a (MEQ) .....	410
No igual a (NEQ) .....	415
Operadores válidos.....	422
¿Qué es el relleno de ceros?.....	423

## Capítulo 6

### Instrucciones de cálculo/matemáticas

Instrucciones de cálculo/matemáticas.....	425
Valor absoluto (ABS) .....	426
Sumar (ADD) .....	430
Calcular (CPT).....	434
Dividir (DIV).....	439
Módulo (MOD) .....	443
Multiplicar (MUL) .....	448
Cambiar signo (NEG) .....	452
Raíz cuadrada (SQR/SQRT).....	456
Restar (SUB) .....	461

## Instrucciones de movimiento/lógicas

### Capítulo 7

Instrucciones de movimiento/lógicas .....	467
Distribuir campo de bits con receptor (BTD) .....	468
Distribuir campo de bits con receptor (BTDT) .....	472
Y a nivel de bits (AND) .....	477
O exclusivo a nivel de bits (XOR).....	481
NO a nivel de bits (NOT) .....	486
O a nivel de bits (OR) .....	490
Y booleano (BAND) .....	495
O exclusivo booleano (BXOR).....	498
NO booleano (BNOT).....	501
O booleano (BOR) .....	503
Borrar (CLR) .....	507
Mover con máscara (MVM).....	509
Mover con máscara con receptor (MVMT) .....	512
Mover (MOV) .....	517
Intercambiar byte (SWPB).....	520

## Matriz (Archivo)/Instrucciones misceláneas

### Capítulo 8

Matriz (Archivo)/Instrucciones misceláneas .....	527
Copiar archivo (COP), Copiar archivo sincrónico (CPS) .....	528
Aritmética y lógica de archivo (FAL) .....	536
Promedio de archivo (AVE).....	556
Llenar archivo (FLL) .....	560
Búsqueda y comparación de archivos (FSC).....	563
Clasificación de archivo (SRT).....	580
Desviación estándar de archivo (STD) .....	585
Tamaño en elementos (SIZE).....	590
Modo Todos.....	595
Diagrama de flujo del modo Todos (FSC).....	596
Modo Numérico.....	597
Diagrama de flujo del modo Numérico (FSC) .....	599
Modo Incremental .....	599
Diagrama de flujo del modo Incremental (FSC) .....	601
Etiqueta de matriz .....	601
Desviación estándar .....	601

## Instrucciones de Matriz (Archivo)/Desplazamiento

### Capítulo 9

Instrucciones de Matriz (Archivo)/Desplazamiento .....	603
Desplazamiento de bit a la izquierda (BSL) .....	604
Desplazamiento de bit a la derecha (BSR).....	609
Carga FIFO (FFL) .....	614
Descarga FIFO (FFU) .....	621
Carga LIFO (LFL) .....	628
Descarga LIFO (LFU).....	635

## Instrucciones de secuenciador

### Capítulo 10

Instrucciones de secuenciador .....	645
Secuenciador de entrada (SQI).....	646
Carga de secuenciador (SQL) .....	650
Secuenciador de salida (SQO) .....	654

## Instrucciones de control de programa

### Capítulo 11

Instrucciones de control de programa.....	660
Instrucción siempre falso (AFI) .....	662
Fin de transición (EOT) .....	663
Saltar a subrutina externa (JXR) .....	666
Saltar a etiqueta (JMP) y Etiqueta (LBL) .....	669
Saltar a subrutina (JSR), Subrutina (SBR) y Retorno (RET) .....	672
Restablecimiento de control maestro (MCR) .....	681
Diagrama de flujo de MCR (Falso).....	685
Sin operación (NOP) .....	686
Pausa SFC (SFP).....	688
Restablecer SFC (SFR).....	690
Fin temporal (TND) .....	693
Desencadenar tarea de evento (EVENT) .....	695
Inhabilitación de interrupción de usuario (UID) / Habilitación de interrupción de usuario (UIE).....	701
Instrucción desconocida (UNK).....	704

## Instrucciones Para/Dividir

### Capítulo 12

Instrucciones Para/Dividir.....	705
Interrupción (BRK).....	705
Instrucción FOR (FOR).....	707
Saltar a subrutina (JSR), Subrutina (SBR) y Retorno (RET).....	712

## Capítulo 13

### Instrucciones especiales

Instrucciones especiales .....	721
Transición de datos (DTR) .....	722
Detección de diagnóstico (DDT) .....	725
Comparación de bits de archivo (FBC) .....	734
PID mejorado (PID) .....	742
Usar instrucciones PID .....	749
Bloqueo anti-restablecimiento y transferencia sin perturbaciones de manual a automático (PID) .....	753
Reinicio sin perturbaciones (PID) .....	754
Lazos en cascada (PID) .....	755
Control de relación (PID) .....	756
Uniformidad de la derivada (PID) .....	757
Prealimentación o polarización de salida (PID) .....	757
Temporización de la instrucción PID .....	758
Establecer la banda muerta (PID) .....	761
Usar el límite de salida (PID) .....	762

## Capítulo 14

### Instrucciones trigonométricas

Instrucciones trigonométricas .....	764
Arcocoseno (ACS, ACOS) .....	765
Arcoseno (ASN, ASIN) .....	769
Arcotangente (ATN, ATAN) .....	773
Coseno (COS) .....	777
Seno (SIN) .....	781
Tangente (TAN) .....	785

## Capítulo 15

### Matemática avanzada

Instrucciones matemáticas avanzadas .....	791
Logaritmo de base 10 (LOG) .....	792
Logaritmo natural (LN) .....	796
X a la potencia de Y (XPY) .....	800

## Capítulo 16

### Instrucciones de conversión matemática

Instrucciones de conversión matemática .....	805
Convertir en BCD (TOD) .....	806
Convertir en entero (FRD) .....	810
Grados (DEG) .....	814
Radianes (RAD) .....	817
Truncar (TRN) .....	821

**Instrucciones de puerto serial ASCII****Capítulo 17**

Instrucciones de puerto serial ASCII .....	827
Carácteres ASCII en el búfer (ACB) .....	829
Borrar ASCII búfer (ACL) .....	832
Líneas de handshake ASCII (AHL) .....	836
Lectura ASCII (ARD) .....	840
Lectura ASCII de línea (ARL) .....	845
Prueba ASCII para línea de búfer (ABL) .....	851
Escritura ASCII (AWT) .....	855
Escritura ASCII con anexo (AWA) .....	861
Tipos de cadena .....	867
Códigos de error ASCII .....	868

**Instrucciones de cadena ASCII**

Instrucciones de cadena ASCII .....	871
Encontrar cadena (FIND) .....	872
Insertar cadena (INSERT) .....	875
Cadena central (MID) .....	879
Concatenar cadenas (CONCAT) .....	882
Eliminar cadena (DELETE) .....	887

**Instrucciones de conversión ASCII**

Instrucciones de conversión ASCII .....	892
DINT en cadena (DTOS) .....	894
Minúsculas (LOWER) .....	897
REAL en cadena (RTOS) .....	899
Cadena en DINT (STOD) .....	902
Cadena en REAL (STOR) .....	905
Mayúsculas (UPPER) .....	909

**Instrucciones de depuración**

Instrucciones de depuración .....	913
Puntos de interrupción (BPT) .....	914
Puntos de rastreo (TPT) .....	918

**Atributos comunes para instrucciones generales**

Atributos comunes .....	923
Indicadores de estado matemáticos .....	923
Valores inmediatos .....	926
Conversiones de datos .....	927

Tipos de datos .....	931
Tipos de datos LINT .....	933
Valores de punto flotante .....	934
Índice a través de matrices .....	936
Direccionamiento de bit .....	936

## **Capítulo 22**

### **Atributos del bloque de funciones**

Elegir los elementos del bloque de funciones.....	939
Enclavamiento de datos .....	940
Orden de ejecución .....	941
Respuestas del bloque de funciones a las condiciones de desbordamiento .....	946
Modos de temporización .....	946
Control de programa/operador.....	950

## **Capítulo 23**

### **Programación de texto estructurado**

Sintaxis de texto estructurado .....	955
Componentes de texto estructurado: Comentarios.....	957
Componentes de texto estructurado: Asignaciones .....	958
Componentes de texto estructurado: Expresiones .....	960
Usar operadores y funciones aritméticos .....	962
Usar operadores a nivel de bits.....	963
Usar operadores lógicos.....	963
Usar operadores relacionales .....	964
Componentes de texto estructurado: Instrucciones .....	966
Componentes de texto estructurado: Construcciones .....	967
Literales de cadena de caracteres .....	968
Tipos de cadena .....	969
CASE_OF .....	970
FOR_DO .....	972
IF_THEN .....	975
REPEAT_UNTIL .....	978
WHILE_DO .....	981

## **Índice**

Este manual proporciona a los programadores detalles sobre el conjunto disponible de instrucciones de General, Movimiento, Proceso y Variadores para un controlador basado en Logix.

Si diseña, programa o soluciona aplicaciones de seguridad que utilizan controladores GuardLogix, consulte el [Manual de referencia de seguridad GuardLogix Safety Application Instruction Set](#), publicación [1756-RM095](#).

Este manual forma parte de un conjunto de manuales relacionados que muestran procedimientos comunes para programar y operar controladores LOGIX 5000™.

Para ver una lista completa de manuales de procedimientos comunes, consulte el [LOGIX 5000 Controllers Common Procedures Programming Manual](#), publicación [1756-PM001](#).

El término controlador LOGIX 5000 se refiere a cualquier controlador que esté basado en el sistema operativo LOGIX 5000.

## Entorno de Studio 5000

El Studio 5000 Automation Engineering & Design Environment® combina los elementos de ingeniería y diseño en un entorno común. El primer elemento es la aplicación Studio 5000 Logix Designer®. La aplicación Logix Designer es el nuevo nombre de marca asignado al software RSLogix 5000® y continuará siendo el producto para programar los controladores LOGIX 5000™ en soluciones discretas, de proceso, de lotes, de movimiento, de seguridad y basadas en variadores.



El entorno Studio 5000® constituye la base para el futuro de las herramientas y capacidades de diseño de ingeniería de Rockwell Automation®. Este entorno Studio 5000 es el lugar donde los ingenieros de diseño desarrollan todos los elementos de sus sistemas de control.

## Recursos adicionales

Estos documentos contienen información adicional sobre los productos relacionados de Rockwell Automation.

Recurso	Descripción
<a href="#">Pautas de cableado y conexión a tierra de equipos de automatización industrial</a> , publicación 1770-4.1	Proporciona instrucciones generales de instalación de un sistema industrial de Rockwell Automation.
Página web de certificaciones de productos, disponible en <a href="http://ab.rockwellautomation.com">http://ab.rockwellautomation.com</a>	Proporciona declaraciones de conformidad, certificados y otros datos de certificación.

Puede ver o descargar publicaciones en <http://www.rockwellautomation.com/literature>. Para solicitar copias impresas de la documentación técnica, póngase en contacto con su distribuidor o representante de ventas de Rockwell Automation local.

## Avisos legales

### Aviso de copyright

Copyright © 2018 < RA > Technologies, Inc. Reservados todos los derechos.  
Impreso en EE. UU.

Este documento y los productos Rockwell Software correspondientes tienen copyright de Rockwell Automation Technologies, Inc. Queda prohibida cualquier reproducción o distribución sin previo acuerdo por escrito de Rockwell Automation Technologies, Inc. Consulte el acuerdo de licencia para obtener más información.

### **Contrato de licencia para el usuario final (EULA)**

Puede ver el Acuerdo de licencia del usuario final ("EULA") de Rockwell Automation abriendo el archivo License.rtf que se encuentra en la carpeta de instalación del producto, en su disco duro.

### **Otras licencias**

El software incluido en este producto contiene software con copyright autorizado bajo una o más licencias de código abierto. En el software se incluyen copias de estas licencias. El correspondiente código fuente de los paquetes de fuente abierta incluidos en este producto se encuentran en sus respectivas páginas web.

De forma alternativa, puede obtener el código fuente correspondiente contactando con Rockwell Automation a través de nuestro formulario Contacto en el sitio web de Rockwell Automation:

<http://www.rockwellautomation.com/global/about-us/contact/contact.page> .

Incluya "Código abierto" como parte del texto de solicitud.

El siguiente código abierto se utiliza como parte de este producto:

<b>Software</b>	<b>Copyright</b>	<b>Nombre de licencia</b>	<b>Texto de licencia</b>
AngularJS	Copyright 2010-2017 Google, Inc.	MIT License	Licencia AngularJS 1.5.9
Bootstrap	Copyright 2011-2017 Twitter, Inc. Copyright 2011-2017 The Bootstrap Authors	MIT License	Licencia Bootstrap 3.3.7
jQuery	Copyright 2005, 2014 JS Foundation y otros contribuidores	MIT License	Licencia jQuery 2.1.1
OpenSans	Copyright 2017 Google, Inc.	Apache License, Versión 2.0	Licencia OpenSans

### **Aviso de marcas registradas**

Allen-Bradley, ControlBus, ControlFLASH, Compact GuardLogix, Compact I/O, ControlLogix, CompactLogix, DCM, DH+, Data Highway Plus, DriveLogix, DPI, DriveTools, Explorer, FactoryTalk, FactoryTalk Administration Console, FactoryTalk Alarms and Events, FactoryTalk Batch, FactoryTalk Directory, FactoryTalk Security, FactoryTalk Services Platform, FactoryTalk View, FactoryTalk View SE, FLEX Ex, FlexLogix, FLEX I/O, Guard I/O, High Performance Drive, Integrated Architecture, Kinetix, Logix5000, LOGIX 5000, Logix5550, MicroLogix, DeviceNet, EtherNet/IP, PLC-2, PLC-3, PLC-5, PanelBuilder, PowerFlex, PhaseManager, POINT I/O, PowerFlex, Rockwell

Automation, RSBizWare, Rockwell Software, RSEmulate, Historian, RSFieldbus, RSLinx, RSLogix, RSNetWorx for DeviceNet, RSNetWorx for EtherNet/IP, RSMACC, RSView, RSView32, Rockwell Software Studio 5000 Automation Engineering & Design Environment, Studio 5000 View Designer, SCANport, SLC, SoftLogix, SMC Flex, Studio 5000, Ultra 100, Ultra 200, VersaView, WINtelligent, XM, SequenceManager son marcas comerciales de Rockwell Automation, Inc.

Cualquier logotipo, software o hardware propiedad de Rockwell Automation que no se mencione en este documento también es una marca, registrada o no, de Rockwell Automation, Inc.

### Otras marcas registradas

CmFAS Assistant, CmDongle, CodeMeter, CodeMeter Control Center y WIBU son marcas de WIBU-SYSTEMS AG registradas en Estados Unidos u otros países. Microsoft es una marca de Microsoft Corporation registrada en Estados Unidos u otros países. ControlNet es una marca de ControlNet International. DeviceNet es una marca de la Open DeviceNet Vendors Association (ODVA). Ethernet/IP es una marca de ControlNet International con licencia de ODVA.

Cualesquiera otras marcas registradas pertenecen a sus respectivos propietarios y, por la presente, este hecho queda reconocido.

### Garantía

Este producto tiene la garantía de la licencia del producto. El rendimiento del producto puede resultar afectado por la configuración del sistema, la aplicación llevada a cabo, el control del operador, el mantenimiento y otros factores. Rockwell Automation no es responsable de estos factores que intervienen. Las instrucciones de este documento no cubren todos los detalles o todas las variaciones en el equipo, el procedimiento o el proceso descrito, ni ofrecen indicaciones para satisfacer cada contingencia posible durante la instalación, la operación o el mantenimiento. La implementación de este producto puede variar entre los usuarios.

Este documento es actual en el momento de la versión del producto. Sin embargo, el software correspondiente puede haber sufrido modificaciones desde el lanzamiento de la versión. Rockwell Automation, Inc. se reserva el derecho para cambiar cualquier información contenida en este documento o el software en cualquier momento sin previo aviso. Es su responsabilidad obtener la información más actual disponible en Rockwell al instalar o utilizar este producto.

### Conformidad medioambiental

Rockwell Automation mantiene información medioambiental actual del producto en su sitio web en  
<http://www.rockwellautomation.com/rockwellautomation/about-us/sustainability-ethics/product-environmental-compliance.page>

**Póngase en contacto con Rockwell**

Teléfono del servicio técnico: 1.440.646.3434

Soporte en línea: <http://www.rockwellautomation.com/support/>



## Instrucciones de alarma

### Instrucciones de alarma

Se utiliza las instrucciones de ALMD y ALMA para supervisar y controlar las condiciones de alarma.

Las instrucciones de alarma basadas en Logix están disponibles en diagrama de escalera, texto estructurado y diagrama de bloque de funciones. Estas instrucciones integran la alarma entre las aplicaciones RSView® SE y los controladores LOGIX 5000™.

#### Instrucciones disponibles

##### Diagrama de escalera



##### Bloque de funciones



##### Texto estructurado



Si desea:	Utilice esta instrucción:
Proporcionar alarma para cualquier valor booleano discreto para un diagrama de escalera.	ALMD
Proporcionar alarma para cualquier valor booleano discreto para un diagrama bloque de funciones.	ALMD
Proporcionar alarma para cualquier valor booleano discreto para texto estructurado.	ALMD
Proporcionar nivel e índice de cambio de alarma para cualquier señal analógica para un diagrama de escalera.	ALMA

Proporcionar nivel e índice de cambio de alarma para cualquier señal analógica para un diagrama de bloque de funciones.	ALMA
Proporcionar nivel e índice de cambio de alarma para cualquier señal analógica para texto estructurado.	ALMA

### Consulte también

[Matriz \(Archivo\)/Instrucciones misceláneas](#) en la página 527

[Instrucciones de conversión ASCII](#) en la página 892

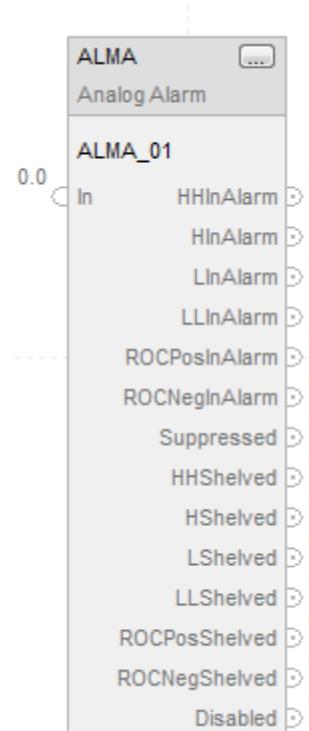
## Alarma analógica (ALMA): Bloque de funciones

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

La instrucción ALMA proporciona el nivel y el índice de cambio de alarma para cualquier señal analógica.

### Idioma disponible

### Bloque de funciones



## Operandos

### Bloque de funciones

Estos operandos se encuentran en la instrucción.

Operando	Tipo	Formato	Descripción
ALMA tag	ALARM_ANALOG	estructura	Estructura de ALMA

## Ejecución

### Bloque de funciones

Condición/estado	Acción realizada
Pre-escaneado	La condición de entrada de reglón se borra a falsa. Se han borrado todos los parámetros de la estructura ALMA. Se han confirmado todas las condiciones de alarma. Se han borrado todas las peticiones del operador. Se han borrado todas las marcas de tiempo. Se han borrado todas las marcas de entrega.
La condición de entrada de reglón es falsa	La condición de entrada de reglón se borra a falsa.
La condición de entrada de reglón es verdadera	La condición de entrada de reglón se establece en verdadera. La instrucción se ejecuta.
Post-escaneado	La condición de entrada de reglón se borra a falsa.

Estos parámetros de entrada son comunes para el bloque de funciones.

Parámetro de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Si no está en falso, la instrucción no se ejecuta y las salidas no se actualizan. Está establecido de forma predeterminada.
In	REAL	Valor de entrada de alarma, que se compara con los límites de alarma para detectar la condición de alarma. Valor predeterminado = 0,0. Diagrama de escalera: Copiado del operando de instrucción. Texto estructurado: Copiado del operando de instrucción.

Parámetro de entrada	Tipo de datos	Descripción
InFault	BOOL	Indicador de estado incorrecto para la entrada. La aplicación de usuario puede establecer InFault para indicar que hay un error en la señal de entrada. Cuando se establece, la instrucción establece InFaulted (Status.1). Cuando se borra y se establece en falso, la instrucción desactiva InFaulted (Status.1). En cualquier caso, la instrucción sigue evaluando las condiciones de alarma de In. El valor predeterminado está en falso (buen estado).
HHEnabled	BOOL	Detección de condición de alarma alta alta. Establecer en verdadero para habilitar la detección de la condición de alarma alta alta. Borrar y establecer en falso para hacer que la detección no esté disponible para la condición de alarma alta alta. Está establecido de forma predeterminada.
HEnabled	BOOL	Detección de condición de alarma alta. Establecer en verdadero para habilitar la detección de la condición de alarma alta. Borrar y establecer en falso para hacer que la detección no esté disponible para la condición de alarma alta. Está establecido de forma predeterminada.
LEnabled	BOOL	Detección de la condición de alarma baja. Establecer en verdadero para habilitar la detección de la condición de alarma baja. Borrar y establecer en falso para hacer que la detección no esté disponible para la condición de alarma baja. Está establecido de forma predeterminada.
LLEnabled	BOOL	Detección de la condición de alarma baja baja. Establecer en verdadero para habilitar la detección de la condición de alarma baja baja. Borrar y establecer en falso para hacer que la detección no esté disponible para la condición de alarma baja baja. Está establecido de forma predeterminada.
AckRequired	BOOL	Especifica si es necesaria la confirmación de la alarma. Cuando se establece en verdadero, se necesita confirmación. Cuando se borra y se establece en falso, no se necesita confirmación y HHAcked, HAcked, LAcked, LLAcked, ROCPosAcked y ROCNegAcked están siempre establecidos en verdadero. El valor predeterminado es verdadero.
ProgAckAll	BOOL	El programa del usuario lo establece en verdadero para confirmar todas las condiciones de alarma. Solo surte efecto si no se confirma alguna condición de alarma. Requiere una transición de falso a verdadero. El valor predeterminado es falso. Diagrama de escalera: Copiado del operando de instrucción. Texto estructurado: Copiado del operando de instrucción.
OperAckAll	BOOL	La interfaz de operador lo establece en verdadero para confirmar todas las condiciones de alarma. Solo surte efecto si no se confirma alguna condición de alarma. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso.

Parámetro de entrada	Tipo de datos	Descripción
HHProgAck	BOOL	Confirmación del programa de alarma alta alta. Establecido en verdadero por el programa de usuario para confirmar una condición alta alta. Solo surte efecto si no se confirma la condición de alarma. Requiere una transición de falso a verdadero. El valor predeterminado es falso.
HHOperAck	BOOL	Confirmación del operador de alarma alta alta. Establecido en verdadero por la interfaz de operador para confirmar la condición alta alta. Solo surte efecto si no se confirma la condición de alarma. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso.
HProgAck	BOOL	Confirmación del programa de alarma alta. Establecido en verdadero por el programa de usuario para confirmar una condición alta. Solo surte efecto si no se confirma la condición de alarma. Requiere una transición de falso a verdadero. El valor predeterminado es falso.
HOperAck	BOOL	Confirmación del operador de alarma alta. Establecido en verdadero por la interfaz de operador para confirmar la condición alta. Solo surte efecto si no se confirma la condición de alarma. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso.
LProgAck	BOOL	Confirmación del programa de alarma baja. Establecido en verdadero por el programa de usuario para confirmar una condición baja. Solo surte efecto si no se confirma la condición de alarma. Requiere una transición de falso a verdadero. El valor predeterminado es falso.
LOperAck	BOOL	Confirmación del operador de alarma baja. Establecido en verdadero por la interfaz de operador para confirmar la condición baja. Solo surte efecto si no se confirma la condición de alarma. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso.
LLProgAck	BOOL	Confirmación del programa de alarma baja baja. Establecido en verdadero por el programa de usuario para confirmar una condición baja baja. Solo surte efecto si no se confirma la condición de alarma. Requiere una transición de falso a verdadero. El valor predeterminado es falso.
LLOperAck	BOOL	Confirmación del operador de alarma baja baja. Establecido en verdadero por la interfaz de operador para confirmar la condición baja baja. Solo surte efecto si no se confirma la condición de alarma. La instrucción de alarma borra este parámetro en falso. El valor predeterminado es falso.
ROCPosProgAck	BOOL	Confirmación del programa de índice de cambio positivo. Establecido en verdadero por el programa del usuario para confirmar una condición de índice de cambio positivo. Necesita una transición de falso a verdadero mientras no se confirme la condición de alarma. El valor predeterminado es falso.

Parámetro de entrada	Tipo de datos	Descripción
ROCPoPosOperAck	BOOL	Confirmación del operador de índice de cambio positivo. Establecido en verdadero por la interfaz de operador para confirmar una condición de índice de cambio positivo. Necesita una transición de falso a verdadero mientras no se confirme la condición de alarma. La instrucción de alarma establece este parámetro en falso. El valor predeterminado es falso.
ROCNegProgAck	BOOL	Confirmación del programa de índice de cambio negativo. Establecido en verdadero por el programa del usuario para confirmar una condición de índice de cambio negativo. Necesita una transición de falso a verdadero mientras no se confirme la condición de alarma. El valor predeterminado es falso.
ROCNegOperAck	BOOL	Confirmación del operador de índice de cambio negativo. Establecido en verdadero por la interfaz de operador para confirmar una condición de índice de cambio negativo. Necesita una transición de falso a verdadero mientras no se confirme la condición de alarma. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso.
ProgSuppress	BOOL	El programa del usuario lo establece en verdadero para suprimir la alarma. Está borrado de forma predeterminada.
OperSuppress	BOOL	La interfaz de operador lo establece en verdadero para suprimir la alarma. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso.
ProgUnsuppress	BOOL	El programa del usuario lo establece en verdadero para anular la supresión de la alarma. Tiene prioridad sobre los comandos de supresión. El valor predeterminado es falso.
OperUnsuppress	BOOL	La interfaz de operador lo establece en verdadero para anular la supresión de la alarma. Tiene prioridad sobre los comandos de supresión. La instrucción de alarma establece este parámetro en falso. El valor predeterminado es falso.
HHOperShelve	BOOL	Archivado del operador alto alto. Establecido en verdadero por la interfaz de operador para archivar o rearchivar la condición alta alta. Requiere una transición desde falso en un escaneado del programa a verdadero en el siguiente escaneado del programa. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso. Los comandos de desarchivado tienen prioridad sobre los de archivado. El archivado de una alarma pospone el procesamiento de alarma. Es como suprimir una alarma, excepto que ese archivado tiene un tiempo limitado. Si se confirma una alarma mientras se está archivando, permanece confirmada incluso si vuelve a estar activa de nuevo. Se convierte en no confirmada cuando finaliza la duración del archivado.

Parámetro de entrada	Tipo de datos	Descripción
HOperShelve	BOOL	Archivado del operador alto. Establecido en verdadero por la interfaz de operador para archivar o rearchivar la condición alta. Requiere una transición desde falso en un escaneado del programa a verdadero en el siguiente escaneado del programa. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso. Los comandos de desarchivado tienen prioridad sobre los de archivado.
LOperShelve	BOOL	Archivado del operador bajo. Establecido en verdadero por la interfaz de operador para archivar o rearchivar la condición baja. Requiere una transición desde falso en un escaneado del programa a verdadero en el siguiente escaneado del programa. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso. Los comandos de desarchivado tienen prioridad sobre los de archivado.
LLOperShelve	BOOL	Archivado del operador bajo bajo. Establecido en verdadero por la interfaz de operador para archivar o rearchivar la condición baja baja. Requiere una transición desde falso en un escaneado del programa a verdadero en el siguiente escaneado del programa. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso. Los comandos de desarchivado tienen prioridad sobre los de archivado.
ROCPosOperShelve	BOOL	Archivado del operador de índice de cambio positivo. Establecido en verdadero por la interfaz de operador para archivar o rearchivar una condición de índice de cambio positivo. Requiere una transición desde falso en un escaneado del programa a verdadero en el siguiente escaneado del programa. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso. Los comandos de desarchivado tienen prioridad sobre los de archivado.
ROCNegOperShelve	BOOL	Archivado del operador de índice de cambio negativo. Establecido en verdadero por la interfaz de operador para archivar o rearchivar una condición de índice de cambio negativo. Requiere una transición desde falso en un escaneado del programa a verdadero en el siguiente escaneado del programa. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso. Los comandos de desarchivado tienen prioridad sobre los de archivado.
ProgUnshelveAll	BOOL	Establecido en verdadero por el programa de usuario para desarchivar todas las condiciones de esta alarma. Si se establece en verdadero tanto archivar como desarchivar, los comandos de desarchivado tienen prioridad sobre los de archivado. El valor predeterminado es falso.

Parámetro de entrada	Tipo de datos	Descripción
HHOperUnshelve	BOOL	Desarchivado del operador alto alto. Establecido en verdadero por la interfaz de operador para desarchivar la condición alta alta. La instrucción de alarma cambia este parámetro a falso. Si se establece en verdadero tanto archivar como desarchivar, los comandos de desarchivado tienen prioridad sobre los de archivado. El valor predeterminado es falso.
HOperUnshelve	BOOL	Desarchivado del operador alto. Establecido en verdadero por la interfaz de operador para desarchivar la condición alta. La instrucción de alarma cambia este parámetro a falso. Si se establece en verdadero tanto archivar como desarchivar, los comandos de desarchivado tienen prioridad sobre los de archivado. El valor predeterminado es falso.
LOperUnshelve	BOOL	Desarchivado del operador bajo. Establecido en verdadero por la interfaz de operador para desarchivar la condición baja. La instrucción de alarma cambia este parámetro a falso. Si se establece en verdadero tanto archivar como desarchivar, los comandos de desarchivado tienen prioridad sobre los de archivado. El valor predeterminado es falso.
LLOperUnshelve	BOOL	Desarchivado del operador bajo bajo. Establecido en verdadero por la interfaz de operador para desarchivar la condición baja baja. La instrucción de alarma cambia este parámetro a falso. Si se establece en verdadero tanto archivar como desarchivar, los comandos de desarchivado tienen prioridad sobre los de archivado. El valor predeterminado es falso.
ROCPosOperUnshelve	BOOL	Desarchivado del operador de índice de cambio positivo. Establecido en verdadero por la interfaz de operador para desarchivar una de índice de cambio positivo. La instrucción de alarma cambia este parámetro a falso. Si se establece tanto archivar como desarchivar, los comandos de desarchivado tienen prioridad sobre los de archivado. El valor predeterminado es falso.
ROCNegOperUnshelve	BOOL	Desarchivado del operador de índice de cambio negativo. Establecido en verdadero por la interfaz de operador para desarchivar una de índice de cambio negativo. La instrucción de alarma cambia este parámetro a falso. Si se establece en verdadero tanto archivar como desarchivar, los comandos de desarchivado tienen prioridad sobre los de archivado. El valor predeterminado es falso.
ProgDisable	BOOL	El programa del usuario lo establece en verdadero para deshabilitar la alarma. El valor predeterminado es falso. Diagrama de escalera: Copiado del operando de instrucción. Texto estructurado: Copiado del operando de instrucción.

Parámetro de entrada	Tipo de datos	Descripción
OperDisable	BOOL	<p>La interfaz de operador lo establece en verdadero para deshabilitar la alarma. La instrucción de alarma cambia este parámetro a falso.</p> <p>El valor predeterminado es falso.</p>
ProgEnable	BOOL	<p>El programa del usuario lo establece en verdadero para habilitar la alarma. Tiene prioridad sobre un comando de inhabilitación.</p> <p>El valor predeterminado es falso.</p> <p>Diagrama de escalera:</p> <p>Copiado del operando de instrucción.</p> <p>Texto estructurado:</p> <p>Copiado del operando de instrucción.</p>
OperEnable	BOOL	<p>La interfaz de operador lo establece en verdadero para habilitar la alarma. Tiene prioridad sobre el comando de inhabilitación. La instrucción de alarma borra este parámetro en falso.</p> <p>El valor predeterminado es falso.</p>
AlarmCountReset	BOOL	<p>Establecido en verdadero por el programa de usuario para restablecer los conteos de alarma para todas las condiciones.</p> <p>El valor predeterminado es falso.</p>
HHMinDurationEnable	BOOL	<p>Habilitación de la duración mínima de alarma alta-alta.</p> <p>Establecer en verdadero para habilitar el temporizador de duración mínima cuando se detecta la condición alta alta.</p> <p>El valor predeterminado es verdadero.</p>
HMinDurationEnable	BOOL	<p>Habilitación de la duración mínima de alarma alta.</p> <p>Establecer en verdadero para habilitar el temporizador de duración mínima cuando se detecta la condición alta.</p> <p>El valor predeterminado es verdadero.</p>
LMinDurationEnable	BOOL	<p>Habilitación de la duración mínima de alarma baja.</p> <p>Establecer en verdadero para habilitar el temporizador de duración mínima cuando se detecta la condición baja.</p> <p>El valor predeterminado es verdadero.</p>
LLMinDurationEnable	BOOL	<p>Habilitación de la duración mínima de alarma baja-baja.</p> <p>Establecer en verdadero para habilitar el temporizador de duración mínima cuando se detecta la condición baja baja.</p> <p>El valor predeterminado es verdadero.</p>
HHLimit	REAL	<p>Límite de alarma alta alta.</p> <p>Valor válido = HLlimit &lt; HHLimit &lt; punto flotante positivo máximo.</p> <p>Valor predeterminado = 0,0.</p>
HHSeverity	DINT	<p>Gravedad de la condición de alarma alta alta. Esto no afecta al procesamiento de las alarmas por el controlador pero puede usarse para clasificar y filtrar las funciones del suscriptor de alarma.</p> <p>Valor válido = del 1 al 1000 (1000 = la más grave; 1 = la menos grave).</p> <p>Valor predeterminado = 500.</p>

Parámetro de entrada	Tipo de datos	Descripción
HLimit	REAL	Límite de alarma alta. Valor válido = LLimit < HLimit < HHLimit. Valor predeterminado = 0,0.
HSeverity	DINT	Gravedad de la condición de alarma alta. Esto no afecta al procesamiento de las alarmas por el controlador pero puede usarse para clasificar y filtrar las funciones del suscriptor de alarma. Valor válido = del 1 al 1000 (1000 = la más grave; 1 = la menos grave). Valor predeterminado = 500.
LLimit	REAL	Límite de alarma baja. Valor válido = LLLimit < LLimit < HLimit. Valor predeterminado = 0,0.
LSeverity	DINT	Gravedad de la condición de alarma baja. Esto no afecta al procesamiento de las alarmas por el controlador pero puede usarse para clasificar y filtrar las funciones del suscriptor de alarma. Valor válido = del 1 al 1000 (1000 = la más grave; 1 = la menos grave). Valor predeterminado = 500.
LLLlimit	REAL	Límite de alarma baja baja. Valor válido = punto flotante negativo máximo < LLLlimit < LLimit. Valor predeterminado = 0,0.
LLSeverity	DINT	Gravedad de la condición de alarma baja baja. Esto no afecta al procesamiento de las alarmas por el controlador pero puede usarse para clasificar y filtrar las funciones del suscriptor de alarma. Valor válido = del 1 al 1000 (1000 = la más grave; 1 = la menos grave). Valor predeterminado = 500.
MinDurationPRE	DINT	Duración mínima preseleccionada (en milisegundos) para que una condición de nivel de alarma se mantenga como verdadera antes de que la condición se marque como InAlarm y se envíe la notificación de alarma a los clientes. El controlador recopila los datos de alarma tan pronto como se detecta la condición de alarma para que no se pierdan datos mientras se espera hasta alcanzar la duración mínima. No se aplica a los límites del índice de cambio o a condiciones para las que la detección de la duración mínima está deshabilitada. MinDurationPRE sólo se aplica a la primera excursión desde el estado normal en cualquier dirección. Por ejemplo, una vez que la condición alta excede el tiempo de espera, la condición alta alta se activa inmediatamente, al mismo tiempo que la condición baja espera a que llegue el tiempo de espera.  Valor válido = 0...2147483647. Valor predeterminado = 0.
ShelveDuration	DINT	Duración del tiempo (en minutos) en la que se archivará una alarma archivada. El tiempo mínimo es un minuto. El tiempo máximo lo define MaxShelveDuration.

Parámetro de entrada	Tipo de datos	Descripción
MaxShelveDuration	DINT	Duración máxima del tiempo (en minutos) en la que se puede archivar una alarma.
Deadband	REAL	<p>Banda muerta para detectar que los niveles de alarma alta alta, alta, baja y baja han vuelto a normal.</p> <p>Una banda muerta distinta a cero puede reducir la vibración de la condición de alarma si el valor In cambia continuamente pero permanece cerca del umbral de condición de nivel. El valor de banda muerta no afecta a la transición al estado InAlarm (activo). Una vez que la condición de nivel está activa, pero antes de que la condición vuelva al estado inactivo (normal), el valor In debe:</p> <ul style="list-style-type: none"> <li>Caiga por debajo del umbral menos la banda muerta (para condiciones alta y alta alta)</li> <li>O</li> <li>esté por encima del umbral más la banda muerta (para condiciones baja y baja baja).</li> </ul> <p>La banda muerta no se utiliza para condicionar la medición de tiempo de duración mínima.</p> <p>Valor válido = 0 = Banda muerta &lt; Abarca desde la primera alarma baja habilitada hasta la primera alarma alta habilitada.</p> <p>Valor predeterminado = 0,0.</p>
ROCPoSLimit	REAL	<p>Límite para un índice de cambio en aumento en unidades por segundo. La detección está habilitada para cualquier valor &gt; 0,0 si ROCPeriod también es &gt; 0,0.</p> <p>Valor válido = 0,0...32767,0.</p> <p>Valor predeterminado = 0,0.</p>
ROCPoSSeverity	DINT	<p>Gravedad de la condición de índice de cambio en aumento. Esto no afecta al procesamiento de las alarmas por el controlador pero puede usarse para clasificar y filtrar las funciones del suscriptor de alarma.</p> <p>Valor válido = 1...32767,0.</p> <p>Valor predeterminado = 500.</p>
ROCNegLimit	REAL	<p>Límite para un índice de cambio en descenso en unidades por segundo. La detección está habilitada para cualquier valor &gt; 0,0 si ROCPeriod también es &gt; 0,0.</p> <p>Valor válido = de 0,0 al punto flotante posible máximo.</p> <p>Valor predeterminado = 0,0.</p>
ROCNegSeverity	DINT	<p>Gravedad de la condición de índice de cambio en descenso. Esto no afecta al procesamiento de las alarmas por el controlador pero puede usarse para clasificar y filtrar las funciones del suscriptor de alarma.</p> <p>Valor válido = del 1 al 1000 (1000 = la más grave; 1 = la menos grave).</p> <p>Valor predeterminado = 500.</p>

Parámetro de entrada	Tipo de datos	Descripción
ROCPeriod	REAL	Período de tiempo en segundos para calcular (intervalo de muestreo) el valor del índice de cambio. Cada vez que caduque el intervalo de muestreo, se almacena una nueva muestra de In y se vuelve a calcular el valor ROC. La detección del índice de cambio está activada para cualquier valor > 0,0. Válido = 0,0...32767,0 Valor predeterminado = 0,0.

### Conecitar un botón a la etiqueta OperShelve

Para evitar que la alarma se vuelva a archivar sin que lo desee, la instrucción de alarma solo procesa la etiqueta OperShelve si hace una transición desde falso a verdadero entre un escaneado del programa y el siguiente. Si un operador presiona un botón para archivar la alarma mientras está establecida la etiqueta ProgUnshelve, la alarma no se archiva porque la etiqueta ProgUnshelve tiene prioridad. Sin embargo, dado que los escaneados del programa se completan en milisegundos, el operador puede seguir pulsando el botón de pulsador, por lo que la etiqueta OperShelve sigue establecida sobre varios escaneados del programa incluso aunque la etiqueta ProgUnshelve estuviera establecida en falso. Esto significa que la alarma no está archivada.

Para archivar la alarma, el operador puede soltar y volver a presionar el botón.

### Parámetros de salida

Estos parámetros de salida son comunes para el bloque de funciones.

Parámetro de salida	Tipo de datos	Descripción
AnyInAlarmUnack	BOOL	Combinación de estado activo de alarma y estado de confirmación. Establecido en verdadero cuando se detecta cualquier condición de alarma y no se ha confirmado. Borrado y establecido en falso cuando ninguna condición de alarma esté activa, confirmada o ambas.
HHInAlarm	BOOL	Estado de condición de alarma alta alta. Establecido en verdadero cuando una condición alta alta se activa. Borrado y establecido en falso cuando no hay ninguna condición alta alta.
HInAlarm	BOOL	Estado de condición de alarma alta. Establecido en verdadero cuando una condición alta se activa. Borrado y establecido en falso cuando no hay ninguna condición alta.
LInAlarm	BOOL	Estado de condición de alarma baja. Establecido en verdadero cuando una condición baja se activa. Borrado y establecido en falso cuando no hay ninguna condición baja.

Parámetro de salida	Tipo de datos	Descripción
LLInAlarm	BOOL	Estado de la condición de alarma baja baja. Establecido en verdadero cuando una condición baja baja se activa. Borrado y establecido en falso cuando no hay ninguna condición baja baja.
ROCPoInAlarm	BOOL	Estado de condición de alarma de índice de cambio positivo. Establecido en verdadero cuando haya una condición de índice de cambio positivo. Borrado y establecido en falso cuando no hay ninguna condición positiva de índice de cambio.
ROCNegInAlarm	BOOL	Estado de condición de alarma de índice de cambio negativo. Establecido en verdadero cuando hay una condición negativa de índice de cambio. Borrado y establecido en falso cuando no hay ninguna condición negativa de índice de cambio.
ROC	REAL	Índice de cambio calculado del valor In. Este valor se actualiza cuando se escanea la instrucción después de cada ROCPeriod transcurrido. Se utiliza el valor ROC para evaluar las condiciones de ROCPoInAlarm y ROCNegInAlarm. ROC = (muestra actual de In – muestra anterior de In)/ROCPeriod.
HHAcked	BOOL	Estado confirmado de condición alta alta. Establecido en verdadero cuando una condición alta alta se confirma. Siempre establecido en verdadero cuando AckRequired se borra y establece en falso. Borrado y establecido en false cuando no se confirma ninguna condición alta alta.
HAcked	BOOL	Estado confirmado de condición alta. Establecido en verdadero cuando una condición alta se confirma. Siempre establecido en verdadero cuando AckRequired se borra y establece en falso. Borrado y establecido en falso cuando no se confirma ninguna condición alta.
LAcked	BOOL	Estado confirmado de condición baja. Establecido en verdadero cuando una condición baja se confirma. Siempre establecido en verdadero cuando AckRequired se borra y establece en falso. Borrado y establecido en falso cuando no se confirma ninguna condición baja.
LLAcked	BOOL	Estado confirmado de condición baja baja. Establecido en verdadero cuando una condición baja baja se confirma. Siempre establecido en verdadero cuando AckRequired se borra y establece en falso. Borrado y establecido en falso cuando no se confirma ninguna condición baja baja.
ROCPoAcked	BOOL	Estado confirmado de condición de índice de cambio positivo. Establecido en verdadero cuando se confirma una condición positiva de índice de cambio. Siempre establecido en verdadero cuando AckRequired se borra y establece en falso. Borrado y establecido en falso cuando no se confirma ninguna condición positiva de índice de cambio.

Parámetro de salida	Tipo de datos	Descripción
ROCNegAcked	BOOL	Estado confirmado de condición de índice de cambio negativo. Establecido en verdadero cuando se confirma una condición negativa de índice de cambio. Siempre establecido en verdadero cuando AckRequired se borra y establece en falso. Borrado y establecido en falso cuando no se confirma ninguna condición negativa de índice de cambio.
HHInAlarmUnack	BOOL	Combinación de estado activo de condición alta alta y estado de no confirmación. Establecido en verdadero cuando la condición alta alta está activa (HHInAlarm es verdadero) y no confirmada. Se borra a falso cuando la condición alta alta no está activa, está confirmada o ambas.
HInAlarmUnack	BOOL	Combinación de estado activo de condición alta y estado de no confirmación. Establecido en verdadero cuando la condición alta está activa (HInAlarm es verdadero) y no confirmada. Se borra y se establece en falso cuando la condición alta no está activa, está confirmada o ambas.
LInAlarmUnack	BOOL	Combinación de estado activo de condición baja y estado de no confirmación. Establecido en verdadero cuando la condición baja está activa (LInAlarm es verdadero) y no confirmada. Se borra y se establece en falso cuando la condición baja no está activa, está confirmada o ambas.
LLInAlarmUnack	BOOL	Combinación de estado activo de condición baja baja y estado de no confirmación. Establecido en verdadero cuando la condición baja baja está activa (LLInAlarm es verdadero) y no confirmada. Se borra y se establece en falso cuando la condición baja baja no está activa, está confirmada o ambas.
ROCPoSInAlarmUnack	BOOL	Combinación de estado activo de condición de índice de cambio positivo y estado de no confirmación. Establecido en verdadero cuando la condición de índice de cambio positivo está activa (ROCPoSInAlarm es verdadero) y no confirmada. Se borra y se establece en falso cuando la condición de índice de cambio positivo no está activa, está confirmada o ambas.
ROCNegInAlarmUnack	BOOL	Combinación de estado activo de condición de índice de cambio negativo y estado de no confirmación. Establecido en verdadero cuando la condición de índice de cambio negativo está activa (ROCNegInAlarm es verdadero) y no confirmada. Se borra y se establece en falso cuando la condición de índice de cambio negativo no está activa, está confirmada o ambas.
Suppressed	BOOL	Estado de alarma suprimida. Establecido en verdadero cuando se haya suprimido la alarma. Se borra y se establece en falso cuando no se ha suprimido la alarma.
HHShelved	BOOL	Estado de archivado de la condición alta alta. Establecido en verdadero cuando una condición alta alta se archiva. Se borra y se establece en falso cuando la condición alta alta se desarchiva.

Parámetro de salida	Tipo de datos	Descripción
HShelved	BOOL	Estado de archivado de la condición alta. Establecido en verdadero cuando una condición alta se archiva. Se borra y se establece en falso cuando la condición alta se desarchiva.
LShelved	BOOL	Estado de archivado de la condición baja. Establecido en verdadero cuando una condición baja se archiva. Se borra y se establece en falso cuando la condición baja se desarchiva.
LLShelved	BOOL	Estado de archivado de la condición baja baja. Establecido en verdadero cuando una condición baja baja se archiva. Se borra y se establece en falso cuando la condición baja baja se desarchiva.
ROCPoSShelved	BOOL	Estado archivado de condición de índice de cambio positivo. Establecido en verdadero cuando se archiva una condición de índice de cambio positivo. Se borra y se establece en falso cuando la condición de índice de cambio positivo se desarchiva.
ROCNegShelved	BOOL	Estado archivado de condición de índice de cambio negativo. Establecido en verdadero cuando se archiva una condición de índice de cambio negativo. Se borra y se establece en falso cuando la condición de índice de cambio negativo se desarchiva.
Disabled	BOOL	Estado de alarma deshabilitada. Establecido en verdadero cuando la alarma no está disponible (deshabilitada). Se borra y se establece en falso cuando se haya habilitado la alarma.
HHInAlarmTime	LINT	Marca de tiempo cuando la instrucción ALMA detecta que el valor In ha excedido el límite de la condición alta para la transición más reciente al estado activo.
HHAlarmCount	DINT	Número de veces que se ha activado la condición alta alta. Si se alcanza el valor máximo, el contador deja el valor en el valor máximo de conteo.
HInAlarmTime	LINT	Marca de tiempo cuando la instrucción ALMA detecta que el valor In ha excedido el límite de la condición alta para la transición más reciente al estado activo.
HAlarmCount	DINT	Número de veces que se ha activado la condición alta. Si se alcanza el valor máximo, el contador deja el valor en el valor máximo de conteo.
LInAlarmTime	LINT	Marca de tiempo cuando la instrucción ALMA detecta que el valor In ha excedido el límite de la condición baja para la transición más reciente al estado activo.
LAlarmCount	DINT	Número de veces que se ha activado la condición baja. Si se alcanza el valor máximo, el contador deja el valor en el valor máximo de conteo.
LLInAlarmTime	LINT	Marca de tiempo cuando la instrucción ALMA detecta que el valor In ha excedido el límite de la condición baja para la transición más reciente al estado activo.
LLAlarmCount	DINT	Número de veces que se ha activado la condición baja baja. Si se alcanza el valor máximo, el contador deja el valor en el valor máximo de conteo.

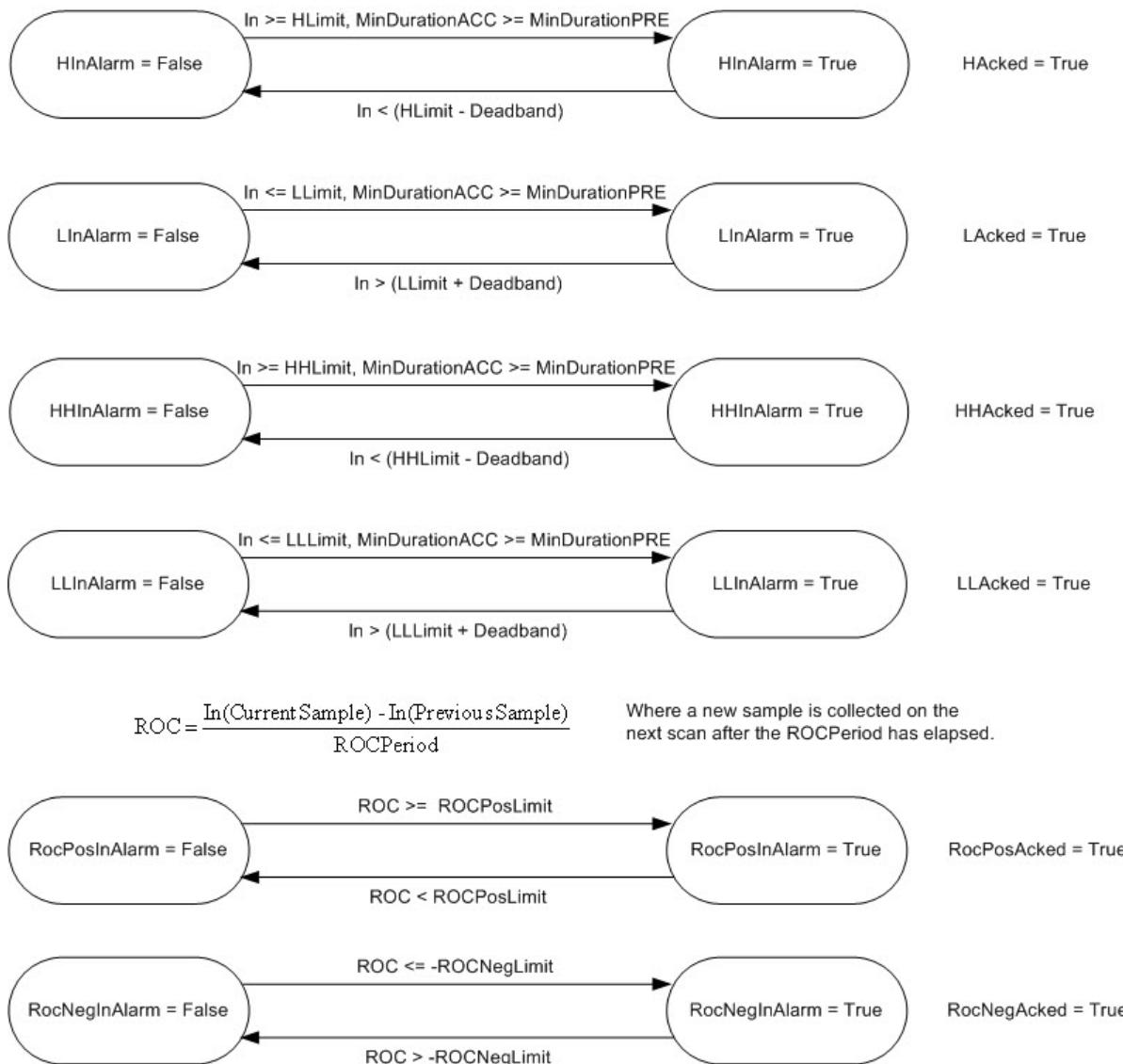
Parámetro de salida	Tipo de datos	Descripción
ROCPoInAlarmTime	LINT	Marca de tiempo cuando la instrucción ALMA detecta que el valor In ha excedido el límite de la condición de índice de cambio positivo para la transición más reciente al estado activo.
ROCPoInAlarmCount	DINT	Número de veces que se ha activado la condición de índice de cambio positivo. Si se alcanza el valor máximo, el contador deja el valor en el valor máximo de conteo.
ROCNegInAlarmTime	LINT	Marca de tiempo cuando la instrucción ALMA detecta que el valor In ha excedido el límite de la condición de índice de cambio negativo para la transición más reciente al estado activo.
ROCNegAlarmCount	DINT	Número de veces que se ha activado la condición de índice de cambio negativo. Si se alcanza el valor máximo, el contador deja el valor en el valor máximo de conteo.
AckTime	LINT	Marca de tiempo de la confirmación de la condición más reciente. Si la alarma no requiere confirmación, esta marca de tiempo es igual a la hora de alarma de la condición más reciente.
RetToNormalTime	LINT	Marca de tiempo de alarma que vuelve a un estado normal.
AlarmCountResetTime	LINT	Marca de tiempo que indica cuándo se ha restablecido el conteo de alarmas.
ShelveTime	LINT	La marca de tiempo indica cuando una condición de alarma se ha archivado por última vez. Establecido por el controlador cuando una condición de alarma se archiva. Todas las condiciones de alarma se pueden archivar y desarchivar varias veces. Cada vez que la condición de alarma se archiva la marca de tiempo se establece en la hora actual.
UnshelveTime	LINT	Marca de tiempo que indica cuando todas las condiciones de alarma se van a desarchivar. El valor solo se establece cuando no se ha archivado todavía ninguna condición de alarma. La marca de tiempo viene determinado por la suma del periodo de tiempo de ShelveDuration y la hora actual. Si una condición de alarma se desarchiva de forma programática o por un operador y no se archiva ninguna otra condición de alarma, entonces el valor se establece en la hora actual.
Status	DINT	Indicadores de estado combinado: Status.0 = InstructFault Status.1 = InFaulted Status.2 = SeverityInv Status.3 = AlarmLimitsInv Status.4 = DeadbandInv Status.5 = ROCPoLimitInv Status.6 = ROCNegLimitInv Status.7 = ROCPeriodInv Status.8 = Overflow

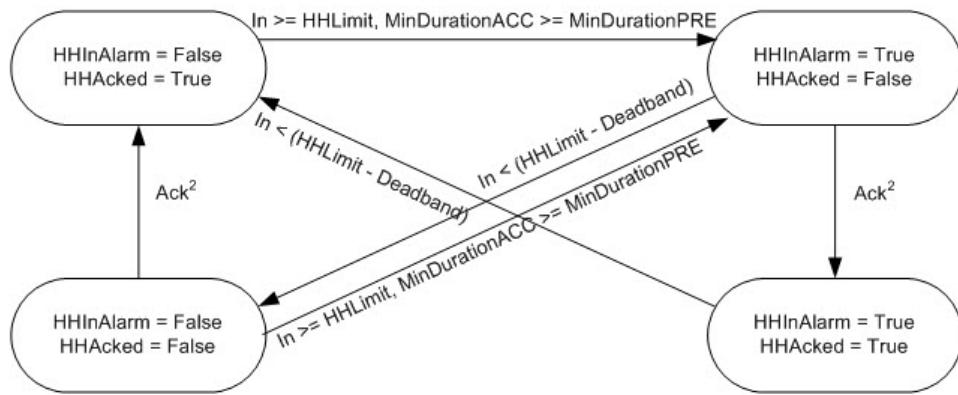
Parámetro de salida	Tipo de datos	Descripción
InstructFault (Status.0)	BOOL	Existen condiciones de error de instrucción No se trata de un error mayor o menor del controlador. Se comprueba los bits de estado restantes para determinar lo que ha ocurrido.
InFaulted (Status.1)	BOOL	El programa de usuario ha establecido InFault para indicar la existencia de datos de entrada de mala calidad. La alarma sigue evaluando In para las condiciones de alarma.
SeverityInv (Status.2)	BOOL	La configuración de gravedad de alarma no es válida. Si Gravedad < 1, la instrucción utiliza Gravedad = 1. Si la gravedad > 1000, la instrucción utiliza Severity = 1000.
AlarmLimitsInv (Status.3)	BOOL	La configuración de límite de alarma no es válida (por ejemplo, LLimit < LLLimit). Si no es válido, la instrucción desactiva todos los bits activos de las condiciones de nivel. Hasta que no se haya eliminado el fallo, no se detectará ninguna condición nueva de nivel.
DeadbandInv (Status.4)	BOOL	La configuración de banda muerta no es válida. Si no es válida, la instrucción utiliza Banda muerta = 0,0. Valor válido = 0 = Banda muerta < Abarca desde la primera alarma baja habilitada hasta la primera alarma alta habilitada.
ROCPoSLimitInv (Status.5)	BOOL	Límite de índice de cambio positivo no válido. Si no es válido, la instrucción utilizará ROCPoSLimit = 0,0 y la detección de índice de cambio positivo no estará disponible.
ROCNegLimitInv (Status.6)	BOOL	Límite de índice de cambio negativo no válido. Si no es válido, la instrucción utilizará ROCNegLimit = 0,0 y la detección de índice de cambio negativo no estará disponible.
ROCPeriodInv (Status.7)	BOOL	Período de índice de cambio no válido. Si no es válido, la instrucción utilizará ROCPeriod = 0,0 y la detección de índice de cambio no estará disponible.
Overflow (Status.8)	BOOL	Desbordamiento. El bit de desbordamiento se establece en verdadero cuando se detecta una condición de desbordamiento. El bit de desbordamiento se borra y se establece en falso cuando la condición de desbordamiento se ha corregido.

### Diagramas de estado de la alarma analógica

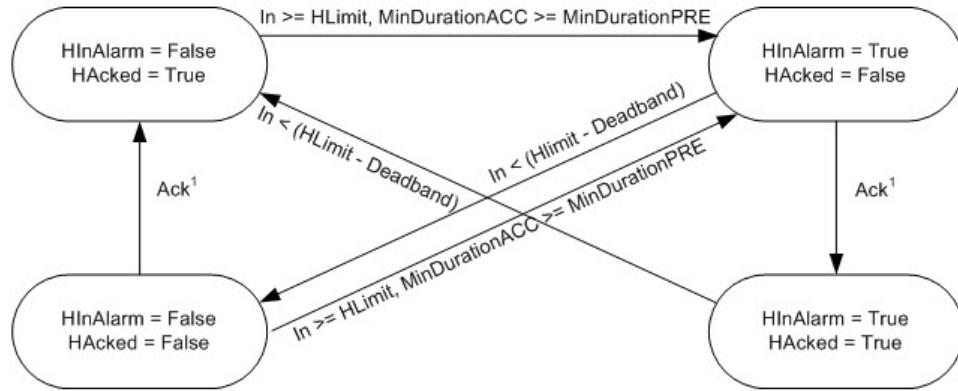
Estas ilustraciones muestran la forma en la que la alarma analógica responde a los cambios en las condiciones de alarma y los comandos del operador.

#### AckRequired = False

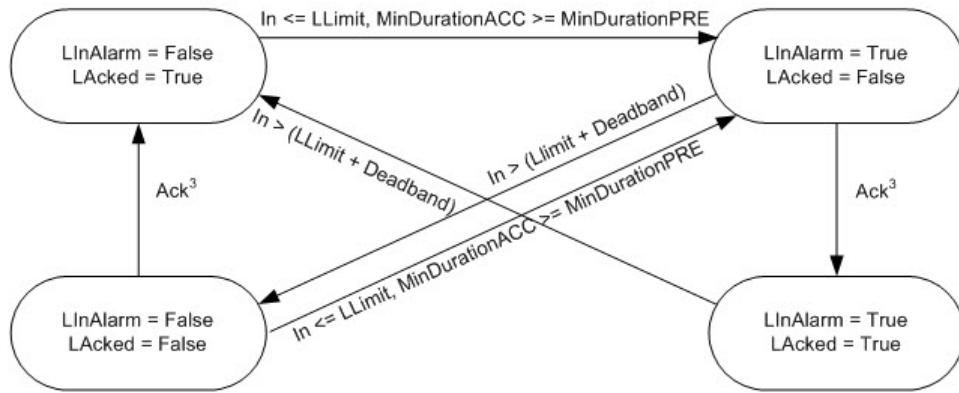


**AckRequired = True**

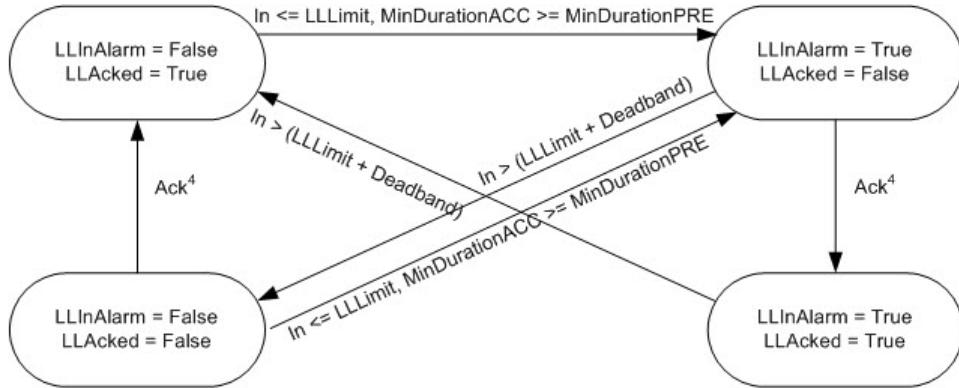
<sup>2</sup> HH alarm condition can be acked by several different ways: HHProgAck, HHOperAck, ProgAckAll, OperAckAll, clients (RSLogix 5000, RSview)



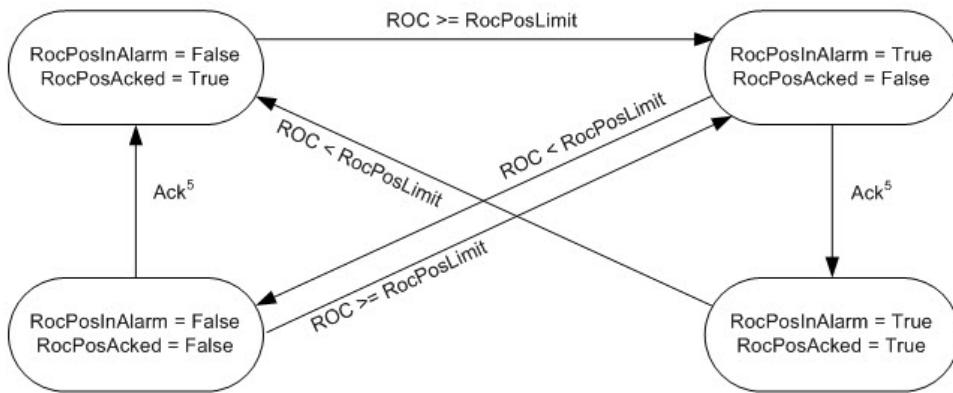
<sup>1</sup> H alarm condition can be acked by several different ways: HProgAck, HOperAck, ProgAckAll, OperAckAll, clients (RSLogix 5000, RSview)



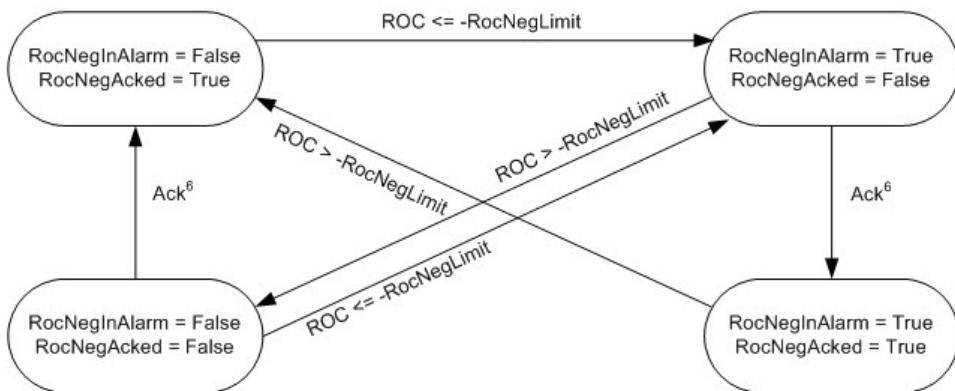
<sup>3</sup> L alarm condition can be acked by several different ways: LProgAck, LOperAck, ProgAckAll, OperAckAll, clients (RSLogix 5000, RSview)



<sup>4</sup> LL alarm condition can be acked by several different ways: LLProgAck, LLOperAck, ProgAckAll, OperAckAll, clients (RSLogix 5000, RSview)



<sup>5</sup> RocPos alarm condition can be acked by several different ways: RocPosProgAck, RocPosOperAck, ProgAckAll, OperAckAll, clients (RSLogix 5000, RSview)

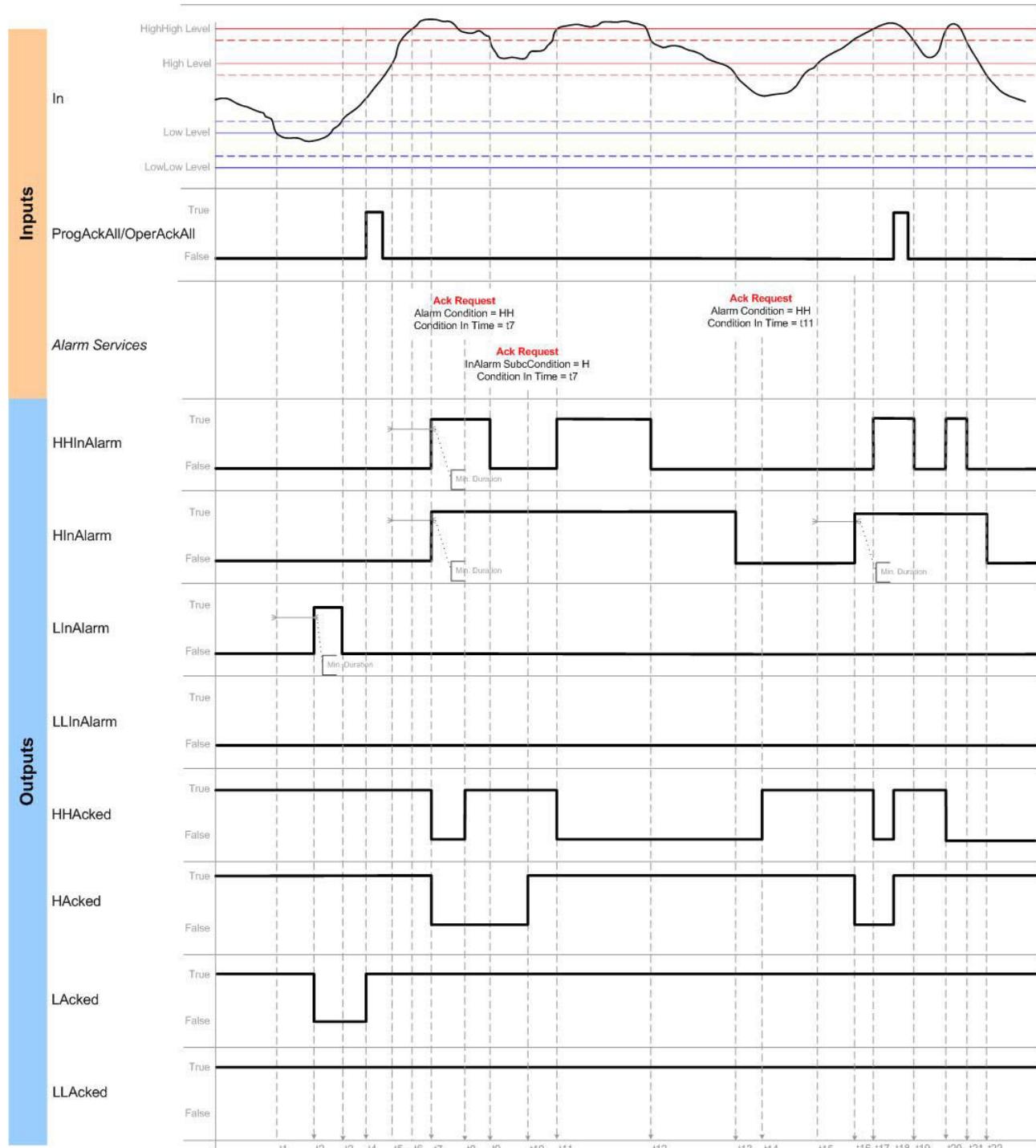


<sup>6</sup> RocNeg alarm condition can be acked by several different ways: RocNegProgAck, RocNegOperAck, ProgAckAll, OperAckAll, clients (RSLogix 5000, RSview)

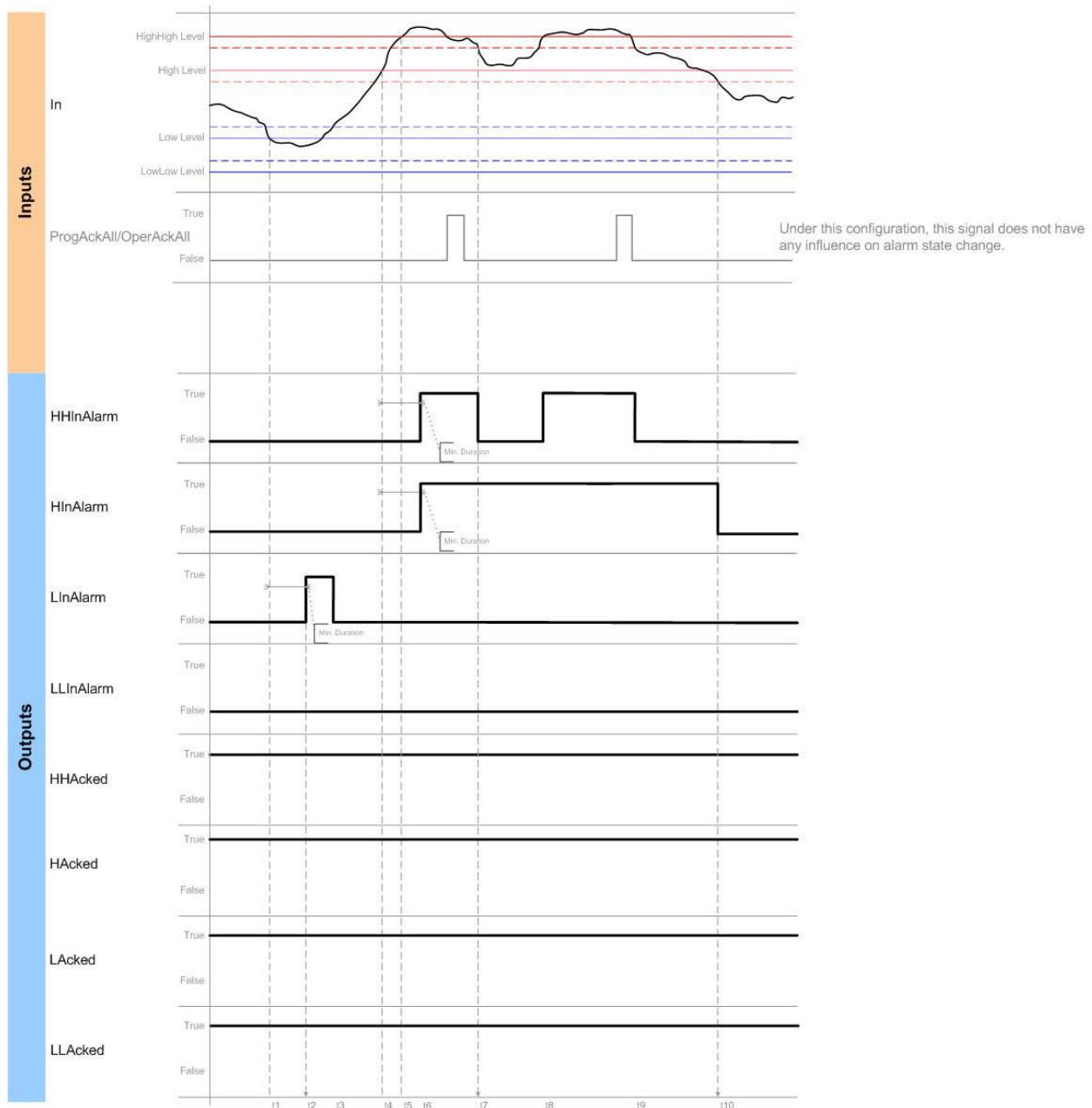
### Diagramas de temporización de la alarma analógica

Estos diagramas de temporización muestran la secuencia de las operaciones de la alarma analógica.

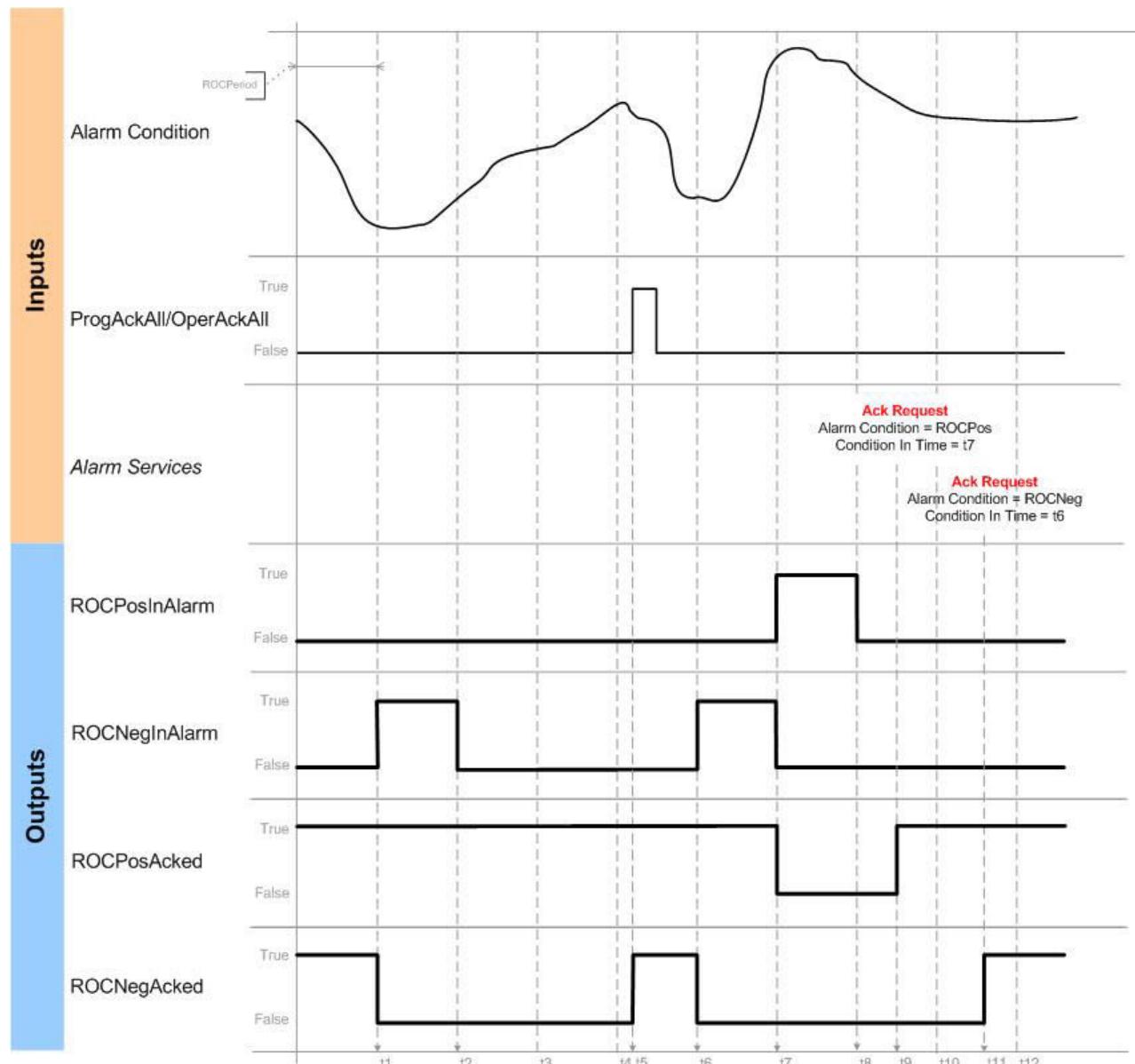
### Confirmación de comportamiento del nivel de condiciones



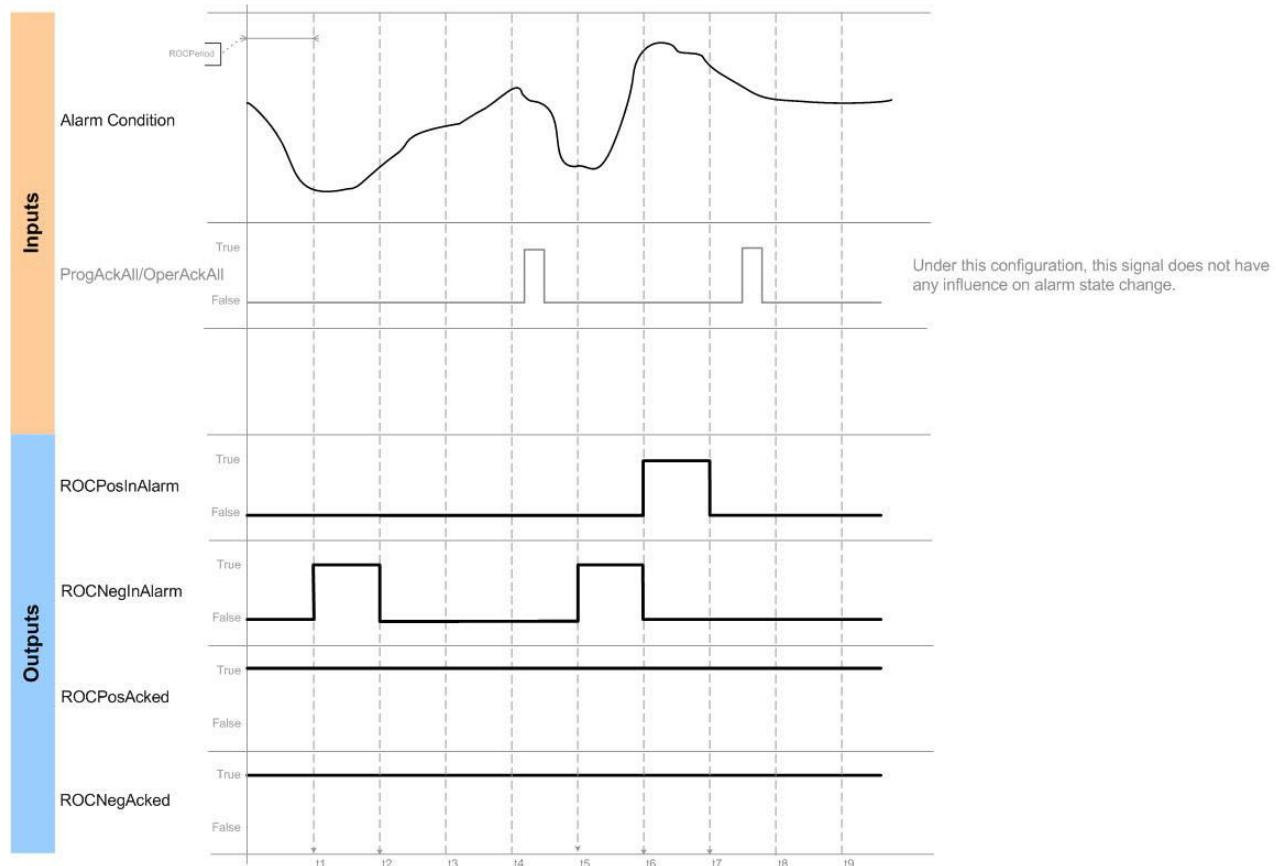
### Sin confirmación de comportamiento del nivel de condiciones



### Confirmación de comportamiento de las condiciones ROC



### Sin confirmación de comportamiento de las condiciones ROC



### Afectar a las marcas de estado matemático

No

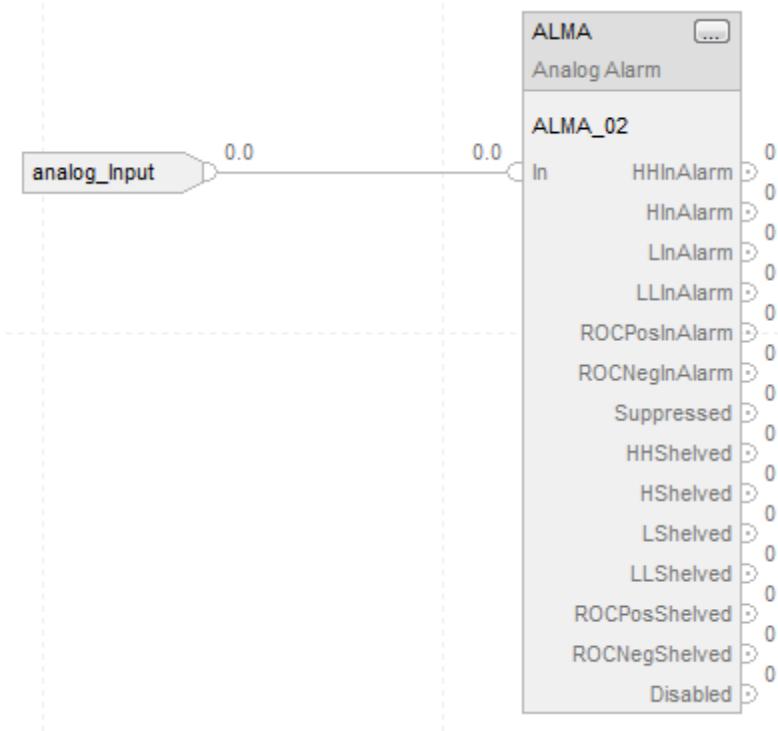
### Fallos mayores/menores

No relacionado para esta instrucción. Consulte los Atributos comunes para fallos relacionados con el operando.

### Ejemplo: Alarma analógica

#### Bloque de funciones

A continuación se muestra un ejemplo de una instrucción de ALMA en bloque de funciones. En este ejemplo, se supervisa el nivel de transmisión del Tanque 32 (Tank32LT) para las condiciones de alarma. Se puede usar la etiqueta Tank32LevelAck para confirmar todas las condiciones de esta alarma.



#### Consulte también

[Alarma analógica \(ALMA\): Lógica de escalera](#) en la página 60

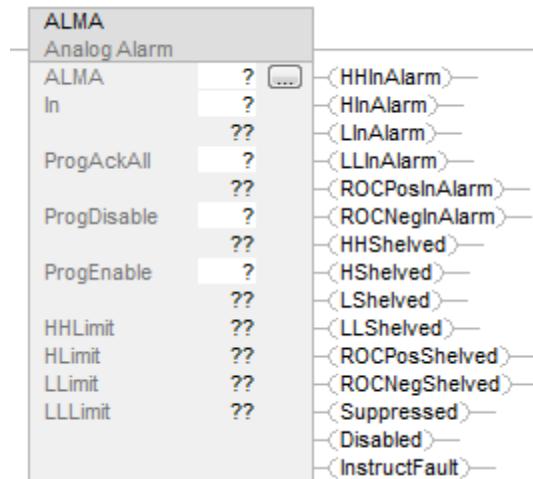
[Alarma analógica \(ALMA\): Texto estructurado](#) en la página 86

[Atributos comunes](#) en la página 923

## Alarma analógica (ALMA) – Diagrama de escalera

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

La instrucción ALMA proporciona el nivel y el índice de cambio de alarma para cualquier señal analógica.



## Operando

Operando	Tipo	Formato	Descripción
ALMA	ALARM_ANALOG	Estructura	Estructura de ALMA
In	REAL DINT INT SINT	Etiqueta Inmediato	Valor de entrada de alarma, que se compara con los límites de alarma para detectar la condición de alarma.
ProgAckAll	BOOL	Etiqueta Inmediato	En la transición de Falso a Verdadero, confirmar todas las condiciones de alarma que requieran confirmación.
ProgDisable	BOOL	Etiqueta Inmediato	Cuando sea Verdadero, desactiva la alarma (no anula los comandos de habilitación).
ProgEnable	BOOL	Etiqueta Inmediato	Cuando sea Verdadero, activa la alarma (tiene prioridad sobre los comandos de deshabilitación).

## Parámetros de entrada

Estos parámetros de entrada son comunes para la lógica de escalera.

Parámetro de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Corresponde al estado de reglón. Si no está en falso, la instrucción no se ejecuta y las salidas no se actualizan.
In	REAL	Copiado del operando de instrucción.

Parámetro de entrada	Tipo de datos	Descripción
InFault	BOOL	Indicador de estado incorrecto para la entrada. La aplicación de usuario puede establecer InFault para indicar que hay un error en la señal de entrada. Cuando se establece, la instrucción establece InFaulted (Status.1). Cuando se borra y se establece en falso, la instrucción desactiva InFaulted (Status.1). En cualquier caso, la instrucción sigue evaluando las condiciones de alarma de In. El valor predeterminado está en falso (buen estado).
HHEnabled	BOOL	Detección de condición de alarma alta alta. Establecer en verdadero para habilitar la detección de la condición de alarma alta alta. Borrar y establecer en falso para hacer que la detección no esté disponible para la condición de alarma alta alta. Está establecido de forma predeterminada.
HEnabled	BOOL	Detección de condición de alarma alta. Establecer en verdadero para habilitar la detección de la condición de alarma alta. Borrar y establecer en falso para hacer que la detección no esté disponible para la condición de alarma alta. Está establecido de forma predeterminada.
LEnabled	BOOL	Detección de la condición de alarma baja. Establecer en verdadero para habilitar la detección de la condición de alarma baja. Borrar y establecer en falso para hacer que la detección no esté disponible para la condición de alarma baja. Está establecido de forma predeterminada.
LLEnabled	BOOL	Detección de la condición de alarma baja baja. Establecer en verdadero para habilitar la detección de la condición de alarma baja baja. Borrar y establecer en falso para hacer que la detección no esté disponible para la condición de alarma baja baja. Está establecido de forma predeterminada.
AckRequired	BOOL	Especifica si es necesaria la confirmación de la alarma. Cuando se establece en verdadero, se necesita confirmación. Cuando se borra y se establece en falso, no se necesita confirmación y HHAcked, HAcked, LAcked, LLAcked, ROCPosAcked y ROCNegAcked están siempre establecidos en verdadero. El valor predeterminado es verdadero.
ProgAckAll	BOOL	Copiado del operando de instrucción.
OperAckAll	BOOL	La interfaz de operador lo establece en verdadero para confirmar todas las condiciones de alarma. Solo surte efecto si no se confirma alguna condición de alarma. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso.
HHProgAck	BOOL	Confirmación del programa de alarma alta alta. Establecido en verdadero por el programa de usuario para confirmar una condición alta alta. Solo surte efecto si no se confirma la condición de alarma. Requiere una transición de falso a verdadero. El valor predeterminado es falso.

Parámetro de entrada	Tipo de datos	Descripción
HHOperAck	BOOL	Confirmación del operador de alarma alta alta. Establecido en verdadero por la interfaz de operador para confirmar la condición alta alta. Solo surte efecto si no se confirma la condición de alarma. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso.
HProgAck	BOOL	Confirmación del programa de alarma alta. Establecido en verdadero por el programa de usuario para confirmar una condición alta. Solo surte efecto si no se confirma la condición de alarma. Requiere una transición de falso a verdadero. El valor predeterminado es falso.
HOperAck	BOOL	Confirmación del operador de alarma alta. Establecido en verdadero por la interfaz de operador para confirmar la condición alta. Solo surte efecto si no se confirma la condición de alarma. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso.
LProgAck	BOOL	Confirmación del programa de alarma baja. Establecido en verdadero por el programa de usuario para confirmar una condición baja. Solo surte efecto si no se confirma la condición de alarma. Requiere una transición de falso a verdadero. El valor predeterminado es falso.
LOperAck	BOOL	Confirmación del operador de alarma baja. Establecido en verdadero por la interfaz de operador para confirmar la condición baja. Solo surte efecto si no se confirma la condición de alarma. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso.
LLProgAck	BOOL	Confirmación del programa de alarma baja baja. Establecido en verdadero por el programa de usuario para confirmar una condición baja baja. Solo surte efecto si no se confirma la condición de alarma. Requiere una transición de falso a verdadero. El valor predeterminado es falso.
LLOperAck	BOOL	Confirmación del operador de alarma baja baja. Establecido en verdadero por la interfaz de operador para confirmar la condición baja baja. Solo surte efecto si no se confirma la condición de alarma. La instrucción de alarma borra este parámetro en falso. El valor predeterminado es falso.
ROCPosProgAck	BOOL	Confirmación del programa de índice de cambio positivo. Establecido en verdadero por el programa del usuario para confirmar una condición de índice de cambio positivo. Necesita una transición de falso a verdadero mientras no se confirme la condición de alarma. El valor predeterminado es falso.

Parámetro de entrada	Tipo de datos	Descripción
ROCPoOperAck	BOOL	Confirmación del operador de índice de cambio positivo. Establecido en verdadero por la interfaz de operador para confirmar una condición de índice de cambio positivo. Necesita una transición de falso a verdadero mientras no se confirme la condición de alarma. La instrucción de alarma establece este parámetro en falso. El valor predeterminado es falso.
ROCNegProgAck	BOOL	Confirmación del programa de índice de cambio negativo. Establecido en verdadero por el programa del usuario para confirmar una condición de índice de cambio negativo. Necesita una transición de falso a verdadero mientras no se confirme la condición de alarma. El valor predeterminado es falso.
ROCNegOperAck	BOOL	Confirmación del operador de índice de cambio negativo. Establecido en verdadero por la interfaz de operador para confirmar una condición de índice de cambio negativo. Necesita una transición de falso a verdadero mientras no se confirme la condición de alarma. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso.
ProgSuppress	BOOL	El programa del usuario lo establece en verdadero para suprimir la alarma. Está borrado de forma predeterminada.
OperSuppress	BOOL	La interfaz de operador lo establece en verdadero para suprimir la alarma. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso.
ProgUnsuppress	BOOL	El programa del usuario lo establece en verdadero para anular la supresión de la alarma. Tiene prioridad sobre los comandos de supresión. El valor predeterminado es falso.
OperUnsuppress	BOOL	La interfaz de operador lo establece en verdadero para anular la supresión de la alarma. Tiene prioridad sobre los comandos de supresión. La instrucción de alarma establece este parámetro en falso. El valor predeterminado es falso.
HHOperShelve	BOOL	Archivado del operador alto alto. Establecido en verdadero por la interfaz de operador para archivar o rearchivar la condición alta alta. Requiere una transición de falso a verdadero. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso. Los comandos de desarchivado tienen prioridad sobre los de archivado. El archivado de una alarma pospone el procesamiento de alarma. Es como suprimir una alarma, excepto que ese archivado tiene un tiempo limitado. Si se confirma una alarma mientras se está archivando, permanece confirmada incluso si vuelve a estar activa de nuevo. Se convierte en no confirmada cuando finaliza la duración del archivado.

Parámetro de entrada	Tipo de datos	Descripción
HOperShelve	BOOL	<p>Archivado del operador alto. Establecido en verdadero por la interfaz de operador para archivar o rearchivar la condición alta. Requiere una transición desde falso en un escaneado del programa a verdadero en el siguiente escaneado del programa. La instrucción de alarma cambia este parámetro a falso.</p> <p>El valor predeterminado es falso.</p> <p>Los comandos de desarchivado tienen prioridad sobre los de archivado.</p>
LOperShelve	BOOL	<p>Archivado del operador bajo. Establecido en verdadero por la interfaz de operador para archivar o rearchivar la condición baja. Requiere una transición desde falso en un escaneado del programa a verdadero en el siguiente escaneado del programa. La instrucción de alarma cambia este parámetro a falso.</p> <p>El valor predeterminado es falso.</p> <p>Los comandos de desarchivado tienen prioridad sobre los de archivado.</p>
LLOperShelve	BOOL	<p>Archivado del operador bajo bajo. Establecido en verdadero por la interfaz de operador para archivar o rearchivar la condición baja baja. Requiere una transición desde falso en un escaneado del programa a verdadero en el siguiente escaneado del programa. La instrucción de alarma cambia este parámetro a falso.</p> <p>El valor predeterminado es falso.</p> <p>Los comandos de desarchivado tienen prioridad sobre los de archivado.</p>
ROCPosOperShelve	BOOL	<p>Archivado del operador de índice de cambio positivo. Establecido en verdadero por la interfaz de operador para archivar o rearchivar una condición de índice de cambio positivo. Requiere una transición desde falso en un escaneado del programa a verdadero en el siguiente escaneado del programa. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso.</p> <p>Los comandos de desarchivado tienen prioridad sobre los de archivado.</p>
ROCNegOperShelve	BOOL	<p>Archivado del operador de índice de cambio negativo. Establecido en verdadero por la interfaz de operador para archivar o rearchivar una condición de índice de cambio negativo. Requiere una transición desde falso en un escaneado del programa a verdadero en el siguiente escaneado del programa. La instrucción de alarma cambia este parámetro a falso.</p> <p>El valor predeterminado es falso.</p> <p>Los comandos de desarchivado tienen prioridad sobre los de archivado.</p>
ProgUnshelveAll	BOOL	<p>Establecido en verdadero por el programa de usuario para desarchivar todas las condiciones de esta alarma. Si se establece en verdadero tanto archivar como desarchivar, los comandos de desarchivado tienen prioridad sobre los de archivado.</p> <p>El valor predeterminado es falso.</p>

Parámetro de entrada	Tipo de datos	Descripción
HHOperUnshelve	BOOL	Desarchivado del operador alto alto. Establecido en verdadero por la interfaz de operador para desarchivar la condición alta alta. La instrucción de alarma cambia este parámetro a falso. Si se establece en verdadero tanto archivar como desarchivar, los comandos de desarchivado tienen prioridad sobre los de archivado. El valor predeterminado es falso.
HOperUnshelve	BOOL	Desarchivado del operador alto. Establecido en verdadero por la interfaz de operador para desarchivar la condición alta. La instrucción de alarma cambia este parámetro a falso. Si se establece en verdadero tanto archivar como desarchivar, los comandos de desarchivado tienen prioridad sobre los de archivado. El valor predeterminado es falso.
LOperUnshelve	BOOL	Desarchivado del operador bajo. Establecido en verdadero por la interfaz de operador para desarchivar la condición baja. La instrucción de alarma cambia este parámetro a falso. Si se establece en verdadero tanto archivar como desarchivar, los comandos de desarchivado tienen prioridad sobre los de archivado. El valor predeterminado es falso.
LLOperUnshelve	BOOL	Desarchivado del operador bajo bajo. Establecido en verdadero por la interfaz de operador para desarchivar la condición baja baja. La instrucción de alarma cambia este parámetro a falso. Si se establece en verdadero tanto archivar como desarchivar, los comandos de desarchivado tienen prioridad sobre los de archivado. El valor predeterminado es falso.
ROCPosOperUnshelve	BOOL	Desarchivado del operador de índice de cambio positivo. Establecido en verdadero por la interfaz de operador para desarchivar una de índice de cambio positivo. La instrucción de alarma cambia este parámetro a falso. Si se establece tanto archivar como desarchivar, los comandos de desarchivado tienen prioridad sobre los de archivado. El valor predeterminado es falso.
ROCNegOperUnshelve	BOOL	Desarchivado del operador de índice de cambio negativo. Establecido en verdadero por la interfaz de operador para desarchivar una de índice de cambio negativo. La instrucción de alarma cambia este parámetro a falso. Si se establece en verdadero tanto archivar como desarchivar, los comandos de desarchivado tienen prioridad sobre los de archivado. El valor predeterminado es falso.
ProgDisable	BOOL	Copiado del operando de instrucción.
OperDisable	BOOL	La interfaz de operador lo establece en verdadero para deshabilitar la alarma. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso.
ProgEnable	BOOL	Copiado del operando de instrucción.

Parámetro de entrada	Tipo de datos	Descripción
OperEnable	BOOL	La interfaz de operador lo establece en verdadero para habilitar la alarma. Tiene prioridad sobre el comando de inhabilitación. La instrucción de alarma borra este parámetro en falso. El valor predeterminado es falso.
AlarmCountReset	BOOL	Establecido en verdadero por la interfaz de operador para restablecer los conteos de alarma para todas las condiciones. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso.
HHMinDurationEnable	BOOL	Habilitación de la duración mínima de alarma alta-alta. Establecer en verdadero para habilitar el temporizador de duración mínima cuando se detecta la condición alta alta. El valor predeterminado es verdadero.
HMinDurationEnable	BOOL	Habilitación de la duración mínima de alarma alta. Establecer en verdadero para habilitar el temporizador de duración mínima cuando se detecta la condición alta. El valor predeterminado es verdadero.
LMinDurationEnable	BOOL	Habilitación de la duración mínima de alarma baja. Establecer en verdadero para habilitar el temporizador de duración mínima cuando se detecta la condición baja. El valor predeterminado es verdadero.
LLMinDurationEnable	BOOL	Habilitación de la duración mínima de alarma baja-baja. Establecer en verdadero para habilitar el temporizador de duración mínima cuando se detecta la condición baja baja. El valor predeterminado es verdadero.
HHLimit	REAL	Límite de alarma alta alta. Valor válido = HLlimit < HHLimit < punto flotante positivo máximo. Valor predeterminado = 0,0.
HHSeverity	DINT	Gravedad de la condición de alarma alta alta. Esto no afecta al procesamiento de las alarmas por el controlador pero puede usarse para clasificar y filtrar las funciones del suscriptor de alarma. Valor válido = del 1 al 1000 (1000 = la más grave; 1 = la menos grave). Valor predeterminado = 500.
HLimit	REAL	Límite de alarma alta. Valor válido = LLlimit < HLlimit < HHLimit. Valor predeterminado = 0,0.
HSeverity	DINT	Gravedad de la condición de alarma alta. Esto no afecta al procesamiento de las alarmas por el controlador pero puede usarse para clasificar y filtrar las funciones del suscriptor de alarma. Valor válido = del 1 al 1000 (1000 = la más grave; 1 = la menos grave). Valor predeterminado = 500.
LLimit	REAL	Límite de alarma baja. Valor válido = LLLimit < LLlimit < HLlimit. Valor predeterminado = 0,0.

Parámetro de entrada	Tipo de datos	Descripción
LSeverity	DINT	Gravedad de la condición de alarma baja. Esto no afecta al procesamiento de las alarmas por el controlador pero puede usarse para clasificar y filtrar las funciones del suscriptor de alarma. Valor válido = del 1 al 1000 (1000 = la más grave; 1 = la menos grave). Valor predeterminado = 500.
LLLlimit	REAL	Límite de alarma baja baja. Valor válido = punto flotante negativo máximo < LLLimit < LLlimit. Valor predeterminado = 0,0.
LLSeverity	DINT	Gravedad de la condición de alarma baja baja. Esto no afecta al procesamiento de las alarmas por el controlador pero puede usarse para clasificar y filtrar las funciones del suscriptor de alarma. Valor válido = del 1 al 1000 (1000 = la más grave; 1 = la menos grave). Valor predeterminado = 500.
MinDurationPRE	DINT	Duración mínima preseleccionada (en milisegundos) para que una condición de nivel de alarma se mantenga como verdadera antes de que la condición se marque como InAlarm y se envíe la notificación de alarma a los clientes. El controlador recopila los datos de alarma tan pronto como se detecta la condición de alarma para que no se pierdan datos mientras se espera hasta alcanzar la duración mínima. No se aplica a los límites del índice de cambio o a condiciones para las que la detección de la duración mínima está deshabilitada. MinDurationPRE sólo se aplica a la primera excursión desde el estado normal en cualquier dirección. Por ejemplo, una vez que la condición alta excede el tiempo de espera, la condición alta alta se activa inmediatamente, al mismo tiempo que la condición baja espera a que llegue el tiempo de espera. Valor válido = 0...2147483647. Valor predeterminado = 0.
ShelveDuration	DINT	Duración del tiempo (en minutos) en la que se archivará una alarma archivada. El tiempo mínimo es un minuto. El tiempo máximo lo define MaxShelveDuration.
MaxShelveDuration	DINT	Duración máxima del tiempo (en minutos) en la que se puede archivar una alarma.

Parámetro de entrada	Tipo de datos	Descripción
Deadband	REAL	<p>Banda muerta para detectar que los niveles de alarma alta alta, alta, baja y baja han vuelto a normal.</p> <p>Una banda muerta distinta a cero puede reducir la vibración de la condición de alarma si el valor In cambia continuamente pero permanece cerca del umbral de condición de nivel. El valor de banda muerta no afecta a la transición al estado InAlarm (activo). Una vez que la condición de nivel está activa, pero antes de que la condición vuelva al estado inactivo (normal), el valor In debe:</p> <ul style="list-style-type: none"> <li>Caiga por debajo del umbral menos la banda muerta (para condiciones alta y alta alta)</li> <li>O</li> <li>esté por encima del umbral más la banda muerta (para condiciones baja y baja baja).</li> </ul> <p>La banda muerta no se utiliza para condicionar la medición de tiempo de duración mínima.</p> <p>Valor válido = 0 = Banda muerta &lt; Abarca desde la primera alarma baja habilitada hasta la primera alarma alta habilitada.</p> <p>Valor predeterminado = 0,0.</p>
ROCPoSLimit	REAL	<p>Límite para un índice de cambio en aumento en unidades por segundo. La detección está habilitada para cualquier valor &gt; 0,0 si ROCPeriod también es &gt; 0,0.</p> <p>Valor válido = de 0,0 al punto flotante posible máximo.</p> <p>Valor predeterminado = 0,0.</p>
ROCPoSSeverity	DINT	<p>Gravedad de la condición de índice de cambio en aumento. Esto no afecta al procesamiento de las alarmas por el controlador pero puede usarse para clasificar y filtrar las funciones del suscriptor de alarma.</p> <p>Valor válido = del 1 al 1000 (1000 = la más grave; 1 = la menos grave).</p> <p>Valor predeterminado = 500.</p>
ROCNegLimit	REAL	<p>Límite para un índice de cambio en descenso en unidades por segundo. La detección está habilitada para cualquier valor &gt; 0,0 si ROCPeriod también es &gt; 0,0.</p> <p>Valor válido = de 0,0 al punto flotante posible máximo.</p> <p>Valor predeterminado = 0,0.</p>
ROCNegSeverity	DINT	<p>Gravedad de la condición de índice de cambio en descenso. Esto no afecta al procesamiento de las alarmas por el controlador pero puede usarse para clasificar y filtrar las funciones del suscriptor de alarma.</p> <p>Valor válido = del 1 al 1000 (1000 = la más grave; 1 = la menos grave).</p> <p>Valor predeterminado = 500.</p>

Parámetro de entrada	Tipo de datos	Descripción
ROCPeriod	REAL	Período de tiempo en segundos para calcular (intervalo de muestreo) el valor del índice de cambio. Cada vez que caduque el intervalo de muestreo, se almacena una nueva muestra de In y se vuelve a calcular el valor ROC. En vez de un bit de habilitación como otros estados en la alarma analógica, la detección del índice de cambio se habilita poniendo cualquier valor distinto de cero en el ROCPeriod. Válido = 0,0...32767,0 Valor predeterminado = 0,0.

### Parámetros de salida

Estos parámetros de salida son comunes para la lógica de escalera.

Parámetro de salida	Tipo de datos	Descripción
AnyInAlarmUnack	BOOL	Combinación de estado activo de alarma y estado de confirmación. Establecido en verdadero cuando se detecta cualquier condición de alarma y no se ha confirmado. Borrado y establecido en falso cuando ninguna condición de alarma esté activa, confirmada o ambas.
HHInAlarm	BOOL	Estado de condición de alarma alta alta. Establecido en verdadero cuando una condición alta alta se activa. Borrado y establecido en falso cuando no hay ninguna condición alta alta.
HInAlarm	BOOL	Estado de condición de alarma alta. Establecido en verdadero cuando una condición alta se activa. Borrado y establecido en falso cuando no hay ninguna condición alta.
LInAlarm	BOOL	Estado de condición de alarma baja. Establecido en verdadero cuando una condición baja se activa. Borrado y establecido en falso cuando no hay ninguna condición baja.
LLInAlarm	BOOL	Estado de la condición de alarma baja baja. Establecido en verdadero cuando una condición baja se activa. Borrado y establecido en falso cuando no hay ninguna condición baja.
ROCPoSInAlarm	BOOL	Estado de condición de alarma de índice de cambio positivo. Establecido en verdadero cuando haya una condición de índice de cambio positivo. Borrado y establecido en falso cuando no hay ninguna condición positiva de índice de cambio.
ROCNegInAlarm	BOOL	Estado de condición de alarma de índice de cambio negativo. Establecido en verdadero cuando haya una condición de índice de cambio negativo. Borrado y establecido en falso cuando no hay ninguna condición negativa de índice de cambio.

Parámetro de salida	Tipo de datos	Descripción
ROC	REAL	Índice de cambio calculado del valor In. Este valor se actualiza cuando se escanea la instrucción después de cada ROCPeriod transcurrido. Se utiliza el valor ROC para evaluar las condiciones de ROCPosInAlarm y ROCNegInAlarm. ROC = (muestra actual de In – muestra anterior de In)/ROCPeriod.
HHAcked	BOOL	Estado confirmado de condición alta alta. Establecido en verdadero cuando una condición alta alta se confirma. Siempre establecido en verdadero cuando AckRequired se borra y establece en falso. Borrado y establecido en false cuando no se confirma ninguna condición alta alta.
HAcked	BOOL	Estado confirmado de condición alta. Establecido en verdadero cuando una condición alta se confirma. Siempre establecido en verdadero cuando AckRequired se borra y establece en falso. Borrado y establecido en falso cuando no se confirma ninguna condición alta.
LAcked	BOOL	Estado confirmado de condición baja. Establecido en verdadero cuando una condición baja se confirma. Siempre establecido en verdadero cuando AckRequired se borra y establece en falso. Borrado y establecido en falso cuando no se confirma ninguna condición baja.
LLAcked	BOOL	Estado confirmado de condición baja baja. Establecido en verdadero cuando una condición baja baja se confirma. Siempre establecido en verdadero cuando AckRequired se borra y establece en falso. Borrado y establecido en falso cuando no se confirma ninguna condición baja baja.
ROCPoSAcked	BOOL	Estado confirmado de condición de índice de cambio positivo. Establecido en verdadero cuando se confirma una condición positiva de índice de cambio. Siempre establecido en verdadero cuando AckRequired se borra y establece en falso. Borrado y establecido en falso cuando no se confirma ninguna condición positiva de índice de cambio.
ROCNegAcked	BOOL	Estado confirmado de condición de índice de cambio negativo. Establecido en verdadero cuando se confirma una condición negativa de índice de cambio. Siempre establecido en verdadero cuando AckRequired se borra y establece en falso. Borrado y establecido en falso cuando no se confirma ninguna condición negativa de índice de cambio.
HHInAlarmUnack	BOOL	Combinación de estado activo de condición alta alta y estado de no confirmación. Establecido en verdadero cuando la condición alta alta está activa (HHInAlarm es verdadero) y no confirmada. Se borra a falso cuando la condición alta alta no está activa, está confirmada o ambas.
HInAlarmUnack	BOOL	Combinación de estado activo de condición alta y estado de no confirmación. Establecido en verdadero cuando la condición alta está activa (HInAlarm es verdadero) y no confirmada. Se borra y se establece en falso cuando la condición alta no está activa, está confirmada o ambas.

Parámetro de salida	Tipo de datos	Descripción
LInAlarmUnack	BOOL	Combinación de estado activo de condición baja y estado de no confirmación. Establecido en verdadero cuando la condición baja está activa (LInAlarm es verdadero) y no confirmada. Se borra y se establece en falso cuando la condición baja no está activa, está confirmada o ambas.
LLInAlarmUnack	BOOL	Combinación de estado activo de condición baja baja y estado de no confirmación. Establecido en verdadero cuando la condición baja baja está activa (LLInAlarm es verdadero) y no confirmada. Se borra y se establece en falso cuando la condición baja baja no está activa, está confirmada o ambas.
ROCPoSInAlarmUnack	BOOL	Combinación de estado activo de condición de índice de cambio positivo y estado de no confirmación. Establecido en verdadero cuando la condición de índice de cambio positivo está activa (ROCPoSInAlarm es verdadero) y no confirmada. Se borra y se establece en falso cuando la condición de índice de cambio positivo no está activa, está confirmada o ambas.
ROCNegInAlarmUnack	BOOL	Combinación de estado activo de condición de índice de cambio negativo y estado de no confirmación. Establecido en verdadero cuando la condición de índice de cambio negativo está activa (ROCNegInAlarm es verdadero) y no confirmada. Se borra y se establece en falso cuando la condición de índice de cambio negativo no está activa, está confirmada o ambas.
Suppressed	BOOL	Estado de alarma suprimida. Establecido en verdadero cuando se haya suprimido la alarma. Se borra y se establece en falso cuando no se ha suprimido la alarma.
HHShelved	BOOL	Estado de archivado de la condición alta alta. Establecido en verdadero cuando una condición alta alta se archiva. Se borra y se establece en falso cuando la condición alta alta se desarchiva.
HShelved	BOOL	Estado de archivado de la condición alta. Establecido en verdadero cuando una condición alta se archiva. Se borra y se establece en falso cuando la condición alta se desarchiva.
LShelved	BOOL	Estado de archivado de la condición baja. Establecido en verdadero cuando una condición baja se archiva. Se borra y se establece en falso cuando la condición baja se desarchiva.
LLShelved	BOOL	Estado de archivado de la condición baja baja. Establecido en verdadero cuando una condición baja baja se archiva. Se borra y se establece en falso cuando la condición baja baja se desarchiva.
ROCPoSShelved	BOOL	Estado archivado de condición de índice de cambio positivo. Establecido en verdadero cuando se archiva una condición de índice de cambio positivo. Se borra y se establece en falso cuando la condición de índice de cambio positivo se desarchiva.

Parámetro de salida	Tipo de datos	Descripción
ROCNegShelved	BOOL	Estado archivado de condición de índice de cambio negativo. Establecido en verdadero cuando se archiva una condición de índice de cambio negativo. Se borra y se establece en falso cuando la condición de índice de cambio negativo se desarchiva.
Disabled	BOOL	Estado de alarma deshabilitada. Establecido en verdadero cuando la alarma no está disponible (deshabilitada). Se borra y se establece en falso cuando se haya habilitado la alarma.
Commissioned	BOOL	El bit activado no se utiliza.
MinDurationACC	DINT	No utilizado. El valor siempre es 0.
HHInAlarmTime	LINT	Marca de tiempo cuando la instrucción ALMA detecta que el valor In ha excedido el límite de la condición alta para la transición más reciente al estado activo.
HHAlarmCount	DINT	Número de veces que se ha activado la condición alta. Si se alcanza el valor máximo, el contador deja el valor en el valor máximo de conteo.
HInAlarmTime	LINT	Marca de tiempo cuando la instrucción ALMA detecta que el valor In ha excedido el límite de la condición alta para la transición más reciente al estado activo.
HAlarmCount	DINT	Número de veces que se ha activado la condición alta. Si se alcanza el valor máximo, el contador deja el valor en el valor máximo de conteo.
LInAlarmTime	LINT	Marca de tiempo cuando la instrucción ALMA detecta que el valor In ha excedido el límite de la condición baja para la transición más reciente al estado activo.
LAlarmCount	DINT	Número de veces que se ha activado la condición baja. Si se alcanza el valor máximo, el contador deja el valor en el valor máximo de conteo.
LLInAlarmTime	LINT	Marca de tiempo cuando la instrucción ALMA detecta que el valor In ha excedido el límite de la condición baja para la transición más reciente al estado activo.
LLAlarmCount	DINT	Número de veces que se ha activado la condición baja. Si se alcanza el valor máximo, el contador deja el valor en el valor máximo de conteo.
ROCPoInAlarmTime	LINT	Marca de tiempo cuando la instrucción ALMA detecta que el valor In ha excedido el límite de la condición de índice de cambio positivo para la transición más reciente al estado activo.
ROCPoInAlarmCoun	DINT	Número de veces que se ha activado la condición de índice de cambio positivo. Si se alcanza el valor máximo, el contador deja el valor en el valor máximo de conteo.
ROCNegInAlarmTime	LINT	Marca de tiempo cuando la instrucción ALMA detecta que el valor In ha excedido el límite de la condición de índice de cambio negativo para la transición más reciente al estado activo.
ROCNegAlarmCount	DINT	Número de veces que se ha activado la condición de índice de cambio negativo. Si se alcanza el valor máximo, el contador deja el valor en el valor máximo de conteo.

Parámetro de salida	Tipo de datos	Descripción
AckTime	LINT	Marca de tiempo de la confirmación de la condición más reciente. Si la alarma no requiere confirmación, esta marca de tiempo es igual a la hora de alarma de la condición más reciente.
RetToNormalTime	LINT	Marca de tiempo de alarma que vuelve a un estado normal.
AlarmCountResetTime	LINT	Marca de tiempo que indica cuándo se ha restablecido el conteo de alarmas.
ShelveTime	LINT	La marca de tiempo indica cuando una condición de alarma se ha archivado por última vez. Establecido por el controlador cuando una condición de alarma se archiva. Todas las condiciones de alarma se pueden archivar y desarchivar varias veces. Cada vez que la condición de alarma se archiva la marca de tiempo se establece en la hora actual.
UnshelveTime	LINT	Marca de tiempo que indica cuando todas las condiciones de alarma se van a desarchivar. El valor solo se establece cuando no se ha archivado todavía ninguna condición de alarma. La marca de tiempo viene determinado por la suma del periodo de tiempo de ShelveDuration y la hora actual. Si una condición de alarma se desarchiva de forma programática o por un operador y no se archiva ninguna otra condición de alarma, entonces el valor se establece en la hora actual.

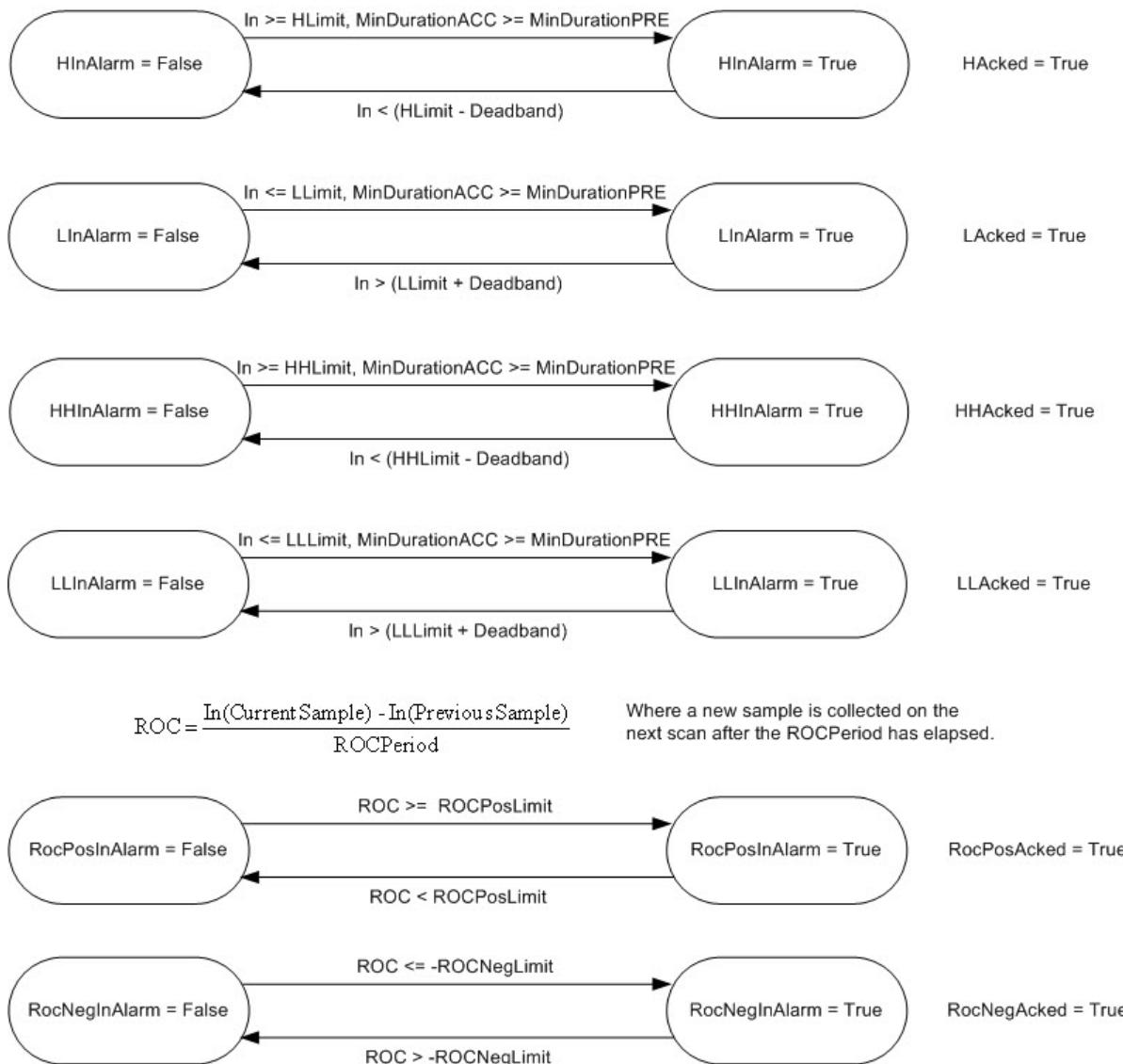
Parámetro de salida	Tipo de datos	Descripción		
Status	DINT	Indicadores de estado combinado: Indicador de estado Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.
		Status.0 = InstructFault	X	X
		Status.1 = InFaulted	X	X
		Status.2 = SeverityInv	X	X
		Status.3 = AlarmLimitsInv	X	X
		Status.4 = DeadbandInv	X	X
		Status.5 = ROCPosLimitInv	X	X
		Status.6 = ROCNegLimitInv	X	X
		Status.7 = ROCPeriodInv	X	X
		Status.8 = Overflow	-	X
InstructFault (Status.0)	BOOL	Existen condiciones de error de instrucción No se trata de un error mayor o menor del controlador. Se comprueba los bits de estado restantes para determinar lo que ha ocurrido.		
InFaulted (Status.1)	BOOL	El programa de usuario ha establecido InFault para indicar la existencia de datos de entrada de mala calidad. La alarma sigue evaluando In para las condiciones de alarma.		
SeverityInv (Status.2)	BOOL	La configuración de gravedad de alarma no es válida. Si Gravedad < 1, la instrucción utiliza Gravedad = 1. Si la gravedad > 1000, la instrucción utiliza Severity = 1000.		
AlarmLimitsInv (Status.3)	BOOL	La configuración de límite de alarma no es válida (por ejemplo, LLimit < LLLimit). Si no es válido, la instrucción desactiva todos los bits activos de las condiciones de nivel. Hasta que no se haya eliminado el fallo, no se detectará ninguna condición nueva de nivel.		

Parámetro de salida	Tipo de datos	Descripción
DeadbandInv (Status.4)	BOOL	La configuración de banda muerta no es válida. Si no es válida, la instrucción utiliza Banda muerta = 0,0. Valor válido = 0 = Banda muerta < Abarca desde la primera alarma baja habilitada hasta la primera alarma alta habilitada.
ROCPoSLimitInv (Status.5)	BOOL	Límite de índice de cambio positivo no válido. Si no es válido, la instrucción utilizará ROCPoSLimit = 0,0 y la detección de índice de cambio positivo no estará disponible.
ROCNegLimitInv (Status.6)	BOOL	Límite de índice de cambio negativo no válido. Si no es válido, la instrucción utilizará ROCNegLimit = 0,0 y la detección de índice de cambio negativo no estará disponible.
ROCPeriodInv (Status.7)	BOOL	Período de índice de cambio no válido. Si no es válido, la instrucción utilizará ROCPeriod = 0,0 y la detección de índice de cambio no estará disponible.
Overflow (Status.8)	BOOL	El bit de desbordamiento se establece en verdadero cuando se detecta una condición de desbordamiento. El bit de desbordamiento se borra y se establece en falso cuando la condición de desbordamiento se ha corregido. Se aplica solamente a Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580..

### Diagramas de estado de la alarma analógica

Estas ilustraciones muestran la forma en la que la alarma analógica responde a los cambios en las condiciones de alarma y los comandos del operador.

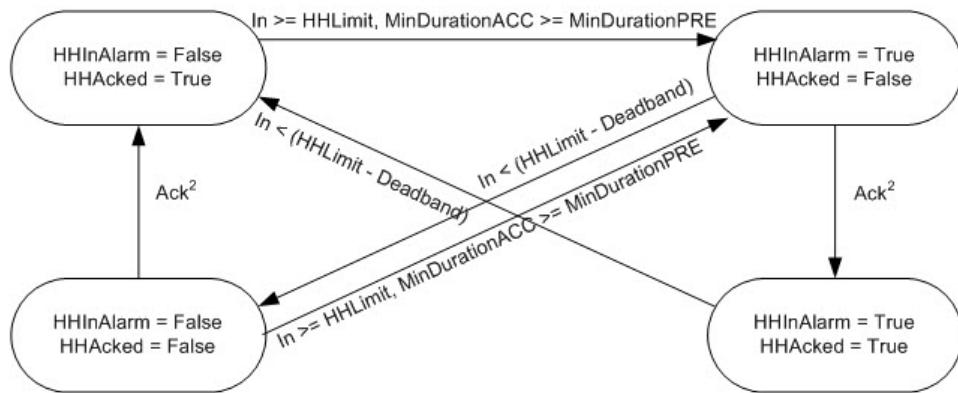
#### AckRequired = False



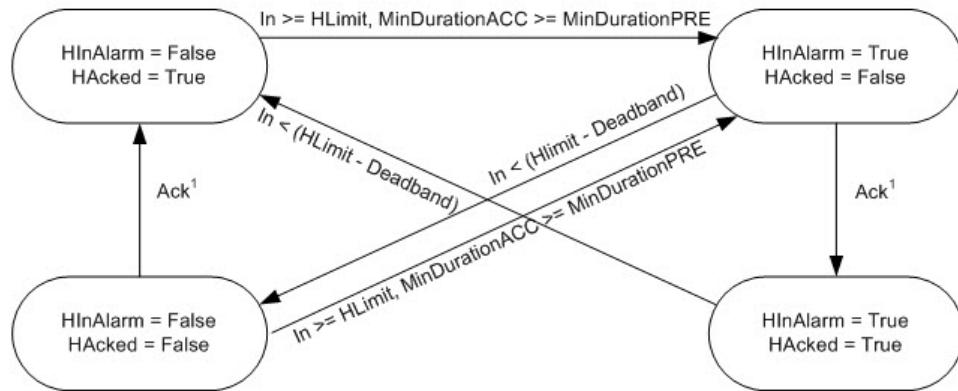
$$\text{ROC} = \frac{\text{In}(Current\ Sample) - \text{In}(Previous\ Sample)}{\text{ROCPeriod}}$$

Where a new sample is collected on the next scan after the ROCPeriod has elapsed.

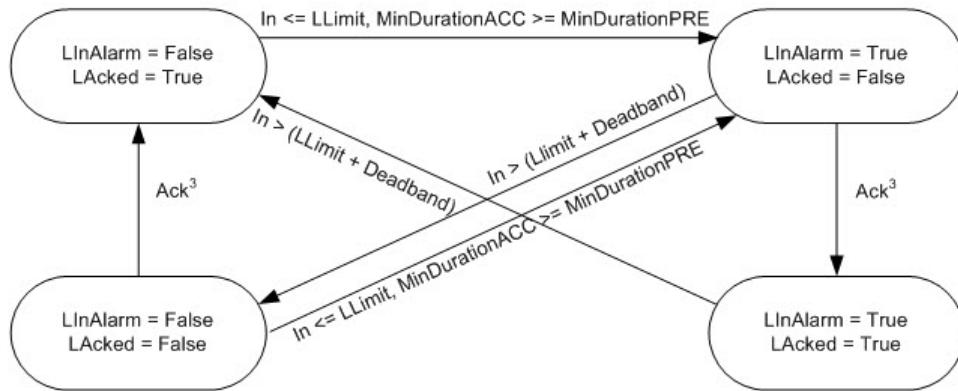
**AckRequired = True**



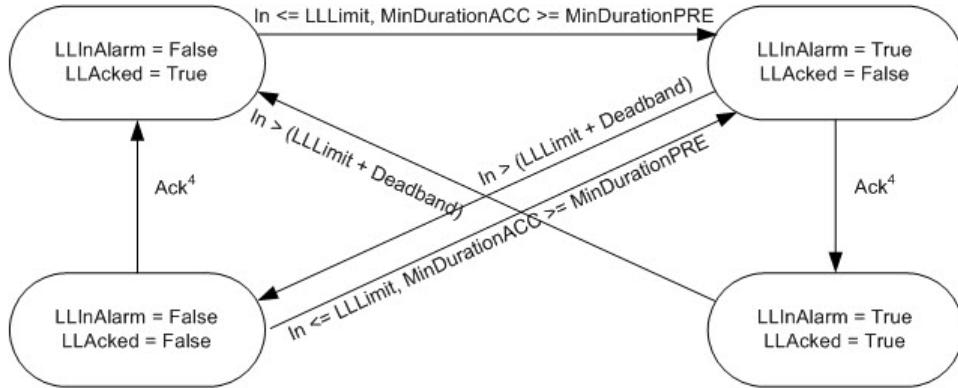
<sup>2</sup> HH alarm condition can be acked by several different ways: HHProgAck, HHOperAck, ProgAckAll, OperAckAll, clients (RSLogix 5000, RSview)



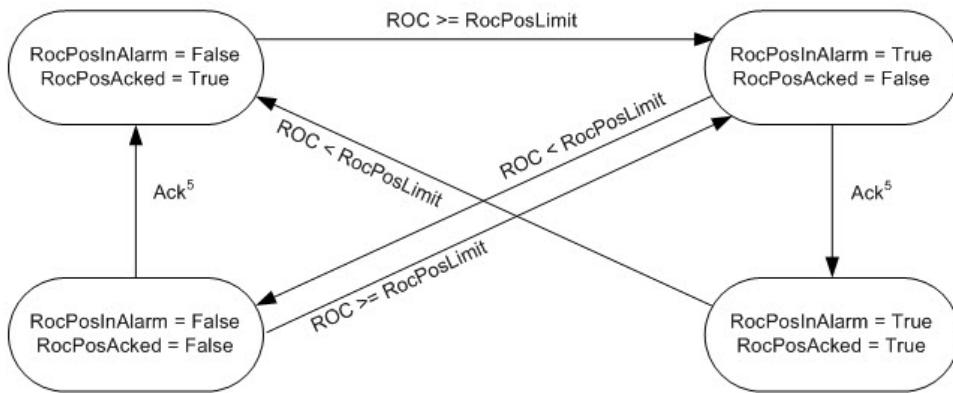
<sup>1</sup> H alarm condition can be acked by several different ways: HProgAck, HOOperAck, ProgAckAll, OperAckAll, clients (RSLogix 5000, RSview)



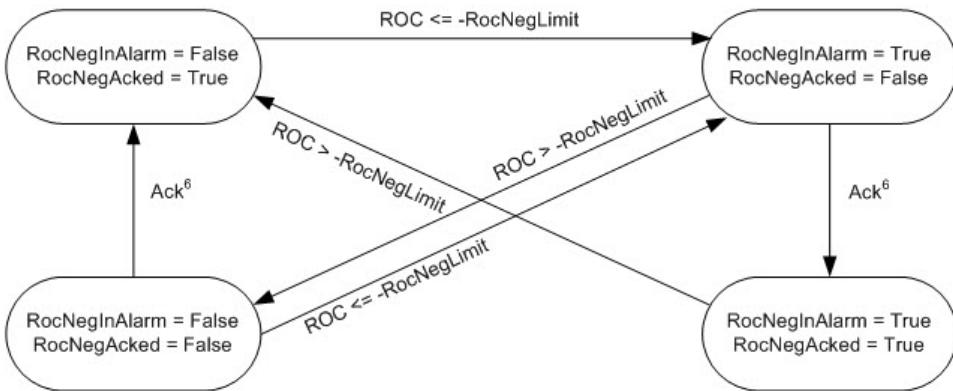
<sup>3</sup> L alarm condition can be acked by several different ways: LProgAck, LOperAck, ProgAckAll, OperAckAll, clients (RSLogix 5000, RSview)



<sup>4</sup> LL alarm condition can be acked by several different ways: LLProgAck, LLOperAck, ProgAckAll, OperAckAll, clients (RSLogix 5000, RSview)



<sup>5</sup> RocPos alarm condition can be acked by several different ways: RocPosProgAck, RocPosOperAck, ProgAckAll, OperAckAll, clients (RSLogix 5000, RSview)

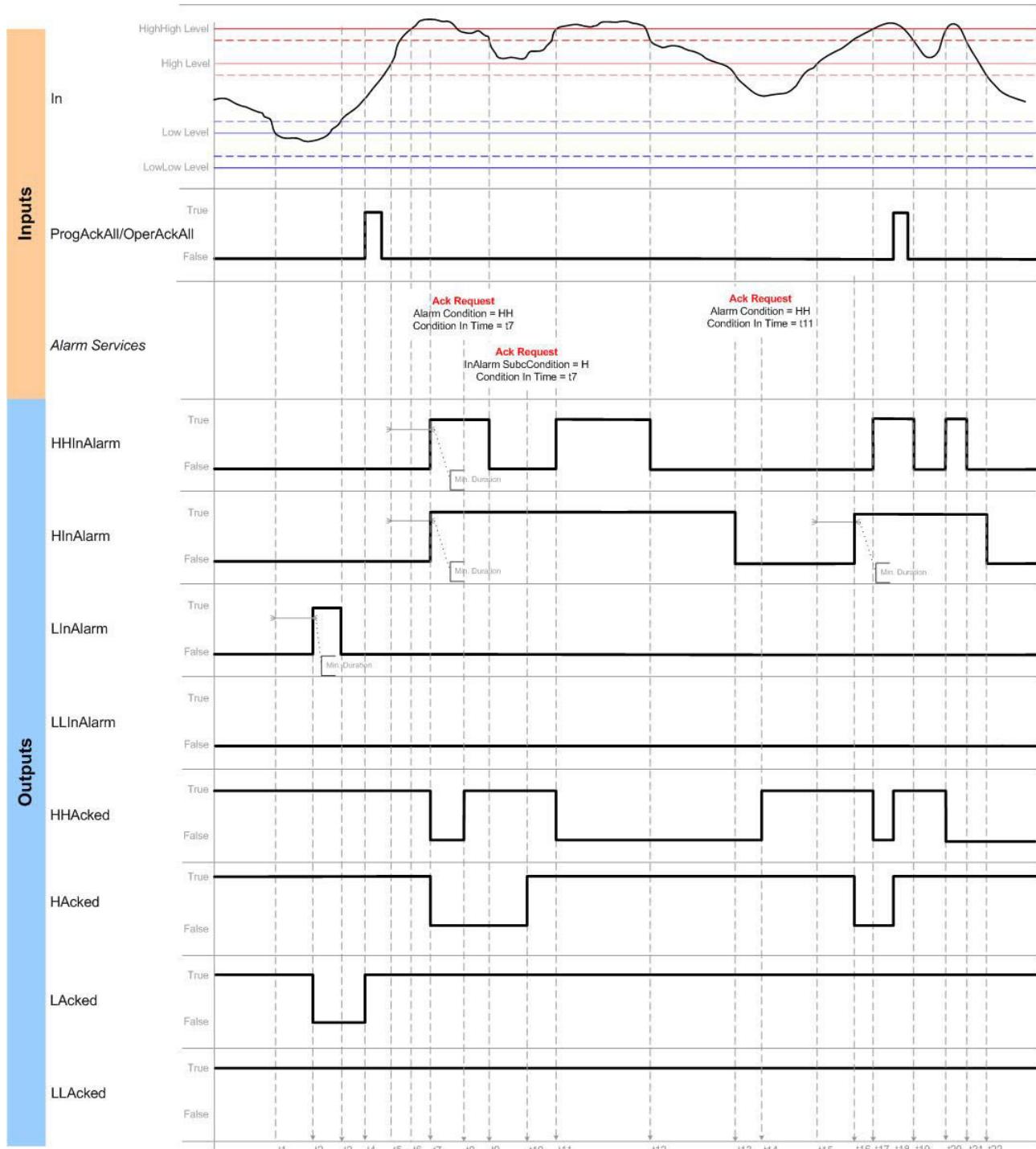


<sup>6</sup> RocNeg alarm condition can be acked by several different ways: RocNegProgAck, RocNegOperAck, ProgAckAll, OperAckAll, clients (RSLogix 5000, RSview)

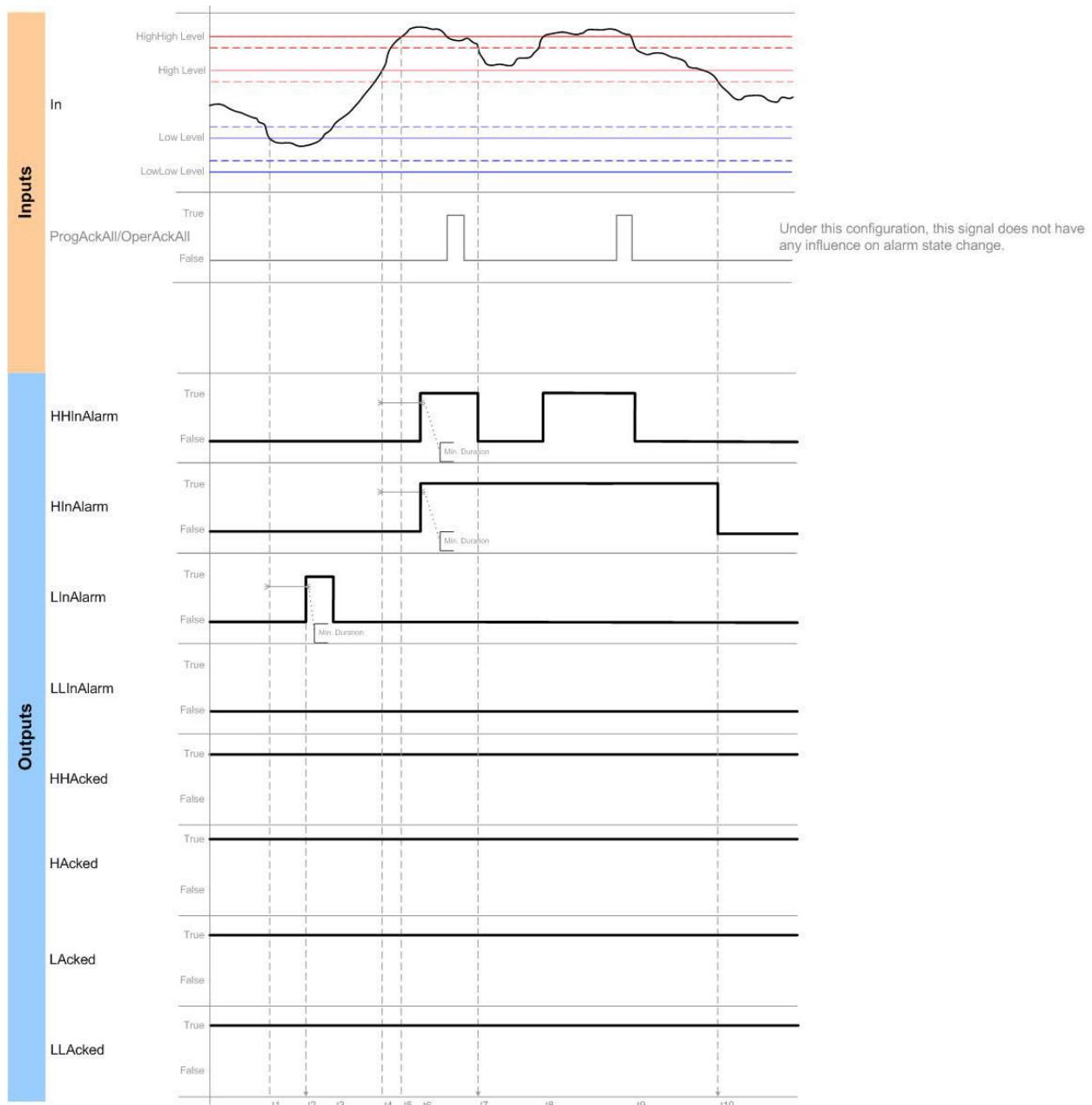
### Diagramas de temporización de la alarma analógica

Estos diagramas de temporización muestran la secuencia de las operaciones de la alarma analógica.

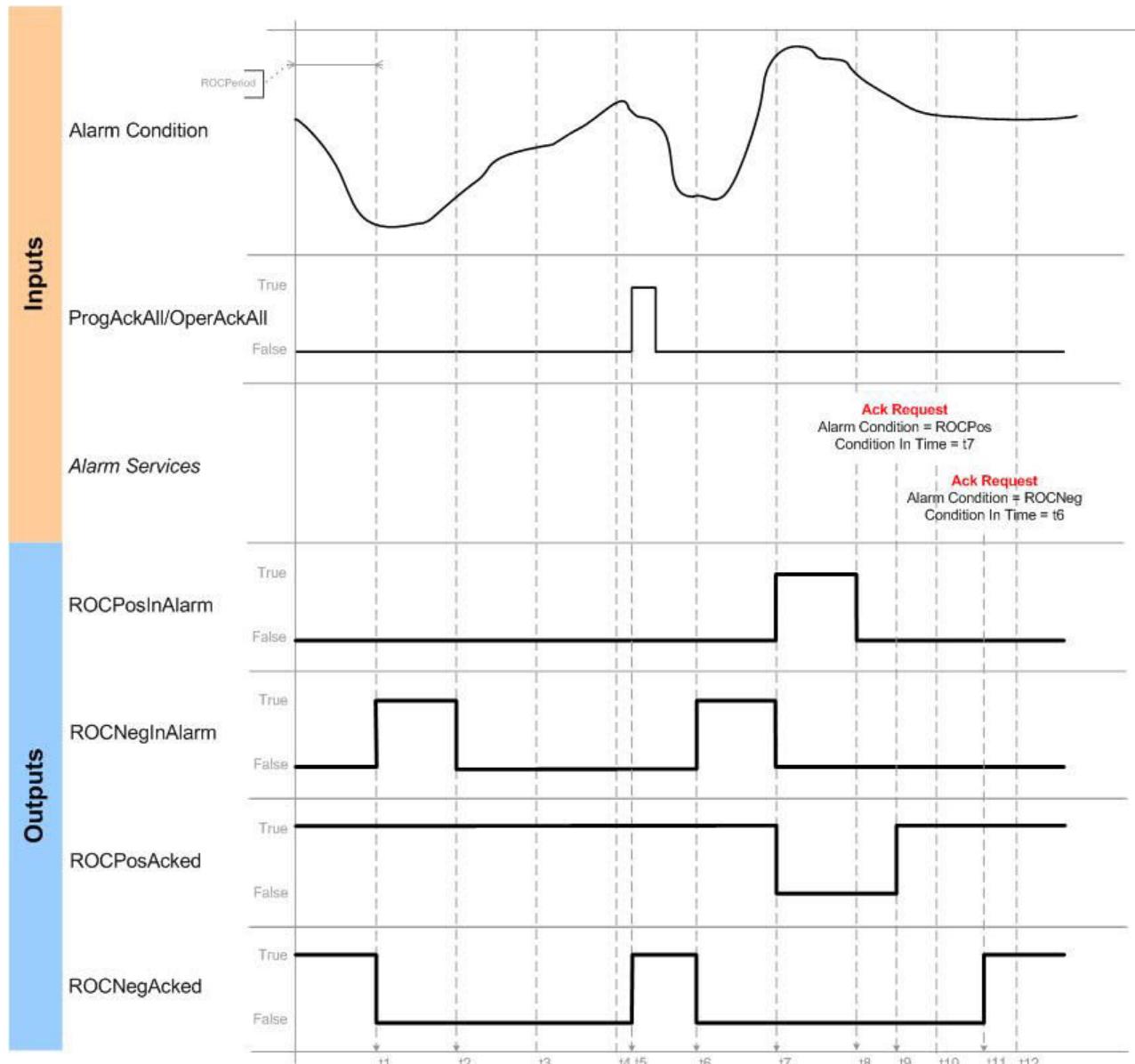
### Confirmación de comportamiento del nivel de condiciones



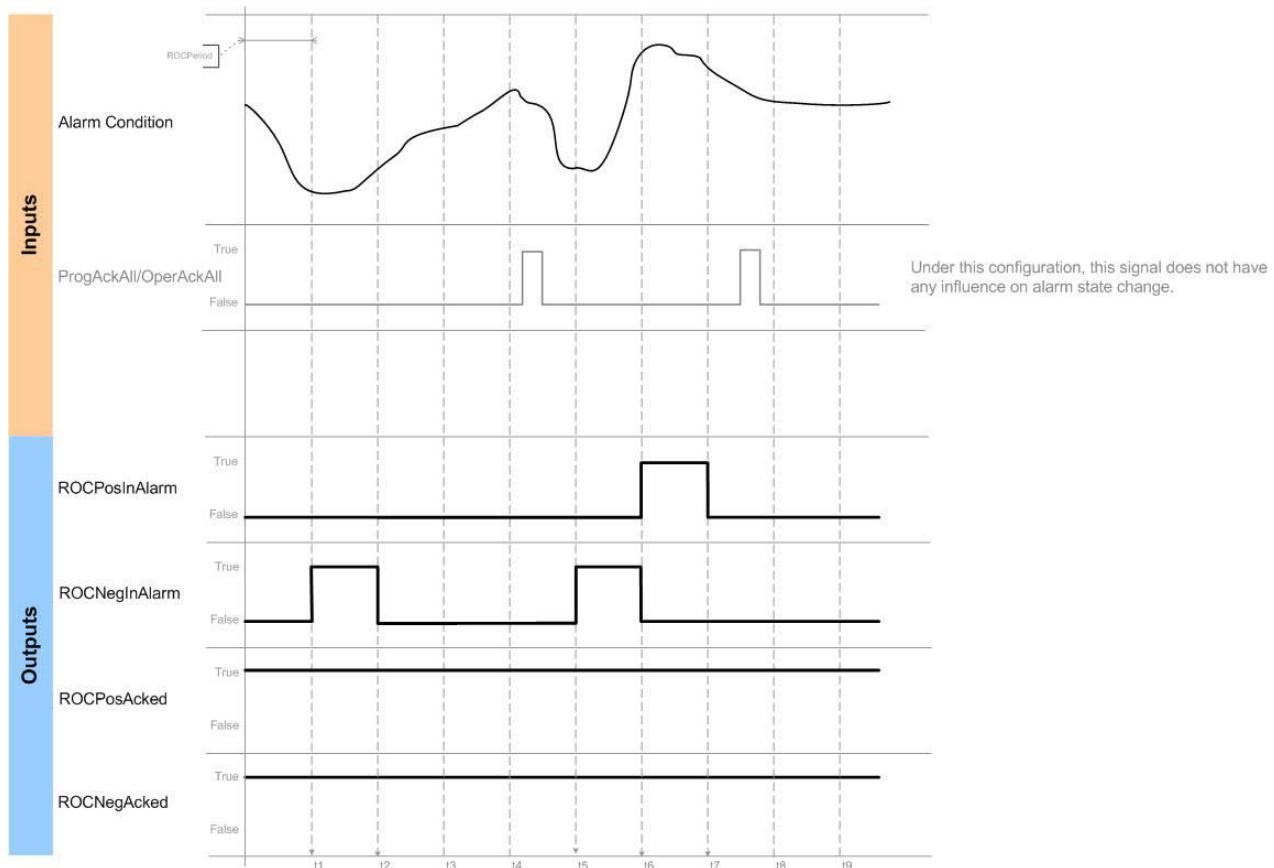
### Sin confirmación de comportamiento del nivel de condiciones



### Confirmación de comportamiento de las condiciones ROC



### Sin confirmación de comportamiento de las condiciones ROC



### Coneectar un botón a la etiqueta OperShelve

La instrucción de alarma solo procesa la etiqueta OperShelve en transición de borrada a establecida, para evitar el rearquivado no deseado de la alarma. Por ejemplo, si un operador presiona un botón de pulsador para archivar la alarma mientras está establecida la etiqueta ProgUnshelve, la alarma no se archiva porque la etiqueta ProgUnshelve tiene prioridad. Para archivar la alarma, el operador puede soltar y volver a presionar el botón una vez borrado ProgUnshelve.

### Afectar a las marcas de estado matemático

Controladores	Marcas de estado matemático afectadas
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Condicional
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí

Ocurrirá un fallo menor si:	Tipo de fallo	Código de fallo
El valor de entrada es INF o NAN solamente para Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570.	4	4

Consulte *Marcas de estado matemático*.

#### Fallos mayores/menores

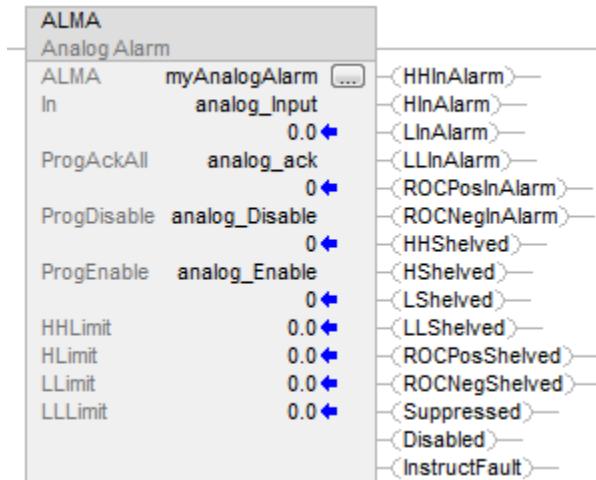
No relacionado para esta instrucción. Consulte los *Atributos comunes* para fallos relacionados con el operando.

#### Ejecución

Condición/estado	Acción realizada
Pre-escaneado	La condición de salida de reglón se borra a falsa. Se han borrado todos los parámetros de la estructura ALMA. Se han confirmado todas las condiciones de alarma. Se han borrado todas las peticiones del operador. Se han borrado todas las marcas de tiempo. Se han borrado todas las marcas de entrega.
La condición de entrada de reglón es falsa	La condición de salida de reglón se borra a falsa.
La condición de entrada de reglón es verdadera	La condición de salida de reglón se establece en verdadera La instrucción se ejecuta
Post-escaneado	La condición de salida de reglón se borra a falsa

## Ejemplos

### Diagrama de escalera



### Consulte también

[Alarma analógica \(ALMA\): Bloque de funciones](#) en la página 36

[Alarma analógica \(ALMA\): Texto estructurado](#) en la página 86

[Atributos comunes](#) en la página 923

## Alarma analógica (ALMA): Texto estructurado

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

La instrucción ALMA proporciona el nivel y el índice de cambio de alarma para cualquier señal analógica.

### Texto estructurado

ALMA (ALMA,In,ProgAckAll,ProgDisable,ProgEnable)

### Operандos

#### Texto estructurado

Estos operandos se encuentran en la instrucción.

Operando	Tipo	Formato	Descripción
ALMA	ALARM_ANALOG	Estructura	Estructura de ALMA
In	REAL DINT INT SINT	Etiqueta Inmediato	Valor de entrada de alarma, que se compara con los límites de alarma para detectar la condición de alarma.
ProgAckAll	BOOL	Etiqueta Inmediato	En la transición de Falso a Verdadero, confirmar todas las condiciones de alarma que requieran confirmación.
ProgDisable	BOOL	Etiqueta Inmediato	Cuando sea Verdadero, desactiva la alarma (no anula los comandos de habilitación).
ProgEnable	BOOL	Etiqueta Inmediato	Cuando sea Verdadero, activa la alarma (tiene prioridad sobre los comandos de deshabilitación).

Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

### Parámetros de entrada

Estos parámetros de entrada son comunes para texto estructurado.

Parámetro de entrada	Tipo de datos	Descripción
EnableIn	BOOL	<p>Diagrama de escalera: Corresponde al estado de reglón. Si no está en falso, la instrucción no se ejecuta y las salidas no se actualizan.</p> <p>Texto estructurado: Si no está en falso, la instrucción no se ejecuta y las salidas no se actualizan.</p> <p>Está establecido de forma predeterminada.</p> <p>Bloque de funciones: Si no está en falso, la instrucción no se ejecuta y las salidas no se actualizan.</p> <p>Está establecido de forma predeterminada.</p>
In	REAL	<p>Valor de entrada de alarma, que se compara con los límites de alarma para detectar la condición de alarma. Valor predeterminado = 0,0.</p> <p>Diagrama de escalera: Copiado del operando de instrucción.</p> <p>Texto estructurado: Copiado del operando de instrucción.</p>
InFault	BOOL	<p>Indicador de estado incorrecto para la entrada. La aplicación de usuario puede establecer InFault para indicar que hay un error en la señal de entrada. Cuando se establece, la instrucción establece InFaulted (Status.1). Cuando se borra y se establece en falso, la instrucción desactiva InFaulted (Status.1). En cualquier caso, la instrucción sigue evaluando las condiciones de alarma de In.</p> <p>El valor predeterminado está en falso (buen estado).</p>

Parámetro de entrada	Tipo de datos	Descripción
HHEnabled	BOOL	Detección de condición de alarma alta alta. Establecer en verdadero para habilitar la detección de la condición de alarma alta alta. Borrar y establecer en falso para hacer que la detección no esté disponible para la condición de alarma alta alta. Está establecido de forma predeterminada.
HEnabled	BOOL	Detección de condición de alarma alta. Establecer en verdadero para habilitar la detección de la condición de alarma alta. Borrar y establecer en falso para hacer que la detección no esté disponible para la condición de alarma alta. Está establecido de forma predeterminada.
LEnabled	BOOL	Detección de la condición de alarma baja. Establecer en verdadero para habilitar la detección de la condición de alarma baja. Borrar y establecer en falso para hacer que la detección no esté disponible para la condición de alarma baja. Está establecido de forma predeterminada.
LLEnabled	BOOL	Detección de la condición de alarma baja baja. Establecer en verdadero para habilitar la detección de la condición de alarma baja baja. Borrar y establecer en falso para hacer que la detección no esté disponible para la condición de alarma baja baja. Está establecido de forma predeterminada.
AckRequired	BOOL	Especifica si es necesaria la confirmación de la alarma. Cuando se establece en verdadero, se necesita confirmación. Cuando se borra y se establece en falso, no se necesita confirmación y HHAcked, HAcked, LAcked, LLAcked, ROCPosAcked y ROCNegAcked están siempre establecidos en verdadero. El valor predeterminado es verdadero.
ProgAckAll	BOOL	El programa del usuario lo establece en verdadero para confirmar todas las condiciones de alarma. Solo surte efecto si no se confirma alguna condición de alarma. Requiere una transición de falso a verdadero. El valor predeterminado es falso. Diagrama de escalera: Copiado del operando de instrucción. Texto estructurado: Copiado del operando de instrucción.
OperAckAll	BOOL	La interfaz de operador lo establece en verdadero para confirmar todas las condiciones de alarma. Solo surte efecto si no se confirma alguna condición de alarma. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso.
HHProgAck	BOOL	Confirmación del programa de alarma alta alta. Establecido en verdadero por el programa de usuario para confirmar una condición alta alta. Solo surte efecto si no se confirma la condición de alarma. Requiere una transición de falso a verdadero. El valor predeterminado es falso.
HHOperAck	BOOL	Confirmación del operador de alarma alta alta. Establecido en verdadero por la interfaz de operador para confirmar la condición alta alta. Solo surte efecto si no se confirma la condición de alarma. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso.

Parámetro de entrada	Tipo de datos	Descripción
HProgAck	BOOL	Confirmación del programa de alarma alta. Establecido en verdadero por el programa de usuario para confirmar una condición alta. Solo surte efecto si no se confirma la condición de alarma. Requiere una transición de falso a verdadero. El valor predeterminado es falso.
HOperAck	BOOL	Confirmación del operador de alarma alta. Establecido en verdadero por la interfaz de operador para confirmar la condición alta. Solo surte efecto si no se confirma la condición de alarma. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso.
LProgAck	BOOL	Confirmación del programa de alarma baja. Establecido en verdadero por el programa de usuario para confirmar una condición baja. Solo surte efecto si no se confirma la condición de alarma. Requiere una transición de falso a verdadero. El valor predeterminado es falso.
LOperAck	BOOL	Confirmación del operador de alarma baja. Establecido en verdadero por la interfaz de operador para confirmar la condición baja. Solo surte efecto si no se confirma la condición de alarma. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso.
LLProgAck	BOOL	Confirmación del programa de alarma baja baja. Establecido en verdadero por el programa de usuario para confirmar una condición baja baja. Solo surte efecto si no se confirma la condición de alarma. Requiere una transición de falso a verdadero. El valor predeterminado es falso.
LLOperAck	BOOL	Confirmación del operador de alarma baja baja. Establecido en verdadero por la interfaz de operador para confirmar la condición baja baja. Solo surte efecto si no se confirma la condición de alarma. La instrucción de alarma borra este parámetro en falso. El valor predeterminado es falso.
ROCPoSProgAck	BOOL	Confirmación del programa de índice de cambio positivo. Establecido en verdadero por el programa del usuario para confirmar una condición de índice de cambio positivo. Necesita una transición de falso a verdadero mientras no se confirme la condición de alarma. El valor predeterminado es falso.
ROCPosOperAck	BOOL	Confirmación del operador de índice de cambio positivo. Establecido en verdadero por la interfaz de operador para confirmar una condición de índice de cambio positivo. Necesita una transición de falso a verdadero mientras no se confirme la condición de alarma. La instrucción de alarma establece este parámetro en falso. El valor predeterminado es falso.
ROCNegProgAck	BOOL	Confirmación del programa de índice de cambio negativo. Establecido en verdadero por el programa del usuario para confirmar una condición de índice de cambio negativo. Necesita una transición de falso a verdadero mientras no se confirme la condición de alarma. El valor predeterminado es falso.
ROCNegOperAck	BOOL	Confirmación del operador de índice de cambio negativo. Establecido en verdadero por la interfaz de operador para confirmar una condición de índice de cambio negativo. Necesita una transición de falso a verdadero mientras no se confirme la condición de alarma. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso.

Parámetro de entrada	Tipo de datos	Descripción
ProgSuppress	BOOL	El programa del usuario lo establece en verdadero para suprimir la alarma. Está borrado de forma predeterminada.
OperSuppress	BOOL	La interfaz de operador lo establece en verdadero para suprimir la alarma. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso.
ProgUnsuppress	BOOL	El programa del usuario lo establece en verdadero para anular la supresión de la alarma. Tiene prioridad sobre los comandos de supresión. El valor predeterminado es falso.
OperUnsuppress	BOOL	La interfaz de operador lo establece en verdadero para anular la supresión de la alarma. Tiene prioridad sobre los comandos de supresión. La instrucción de alarma establece este parámetro en falso. El valor predeterminado es falso.
HHOperShelve	BOOL	Archivado del operador alto alto. Establecido en verdadero por la interfaz de operador para archivar o rearchivar la condición alta alta. Requiere una transición desde falso en un escaneado del programa a verdadero en el siguiente escaneado del programa. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso. Los comandos de desarchivado tienen prioridad sobre los de archivado. El archivado de una alarma pospone el procesamiento de alarma. Es como suprimir una alarma, excepto que ese archivado tiene un tiempo limitado. Si se confirma una alarma mientras se está archivando, permanece confirmada incluso si vuelve a estar activa de nuevo. Se convierte en no confirmada cuando finaliza la duración del archivado.
HOperShelve	BOOL	Archivado del operador alto. Establecido en verdadero por la interfaz de operador para archivar o rearchivar la condición alta. Requiere una transición desde falso en un escaneado del programa a verdadero en el siguiente escaneado del programa. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso. Los comandos de desarchivado tienen prioridad sobre los de archivado.
LOperShelve	BOOL	Archivado del operador bajo. Establecido en verdadero por la interfaz de operador para archivar o rearchivar la condición baja. Requiere una transición desde falso en un escaneado del programa a verdadero en el siguiente escaneado del programa. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso. Los comandos de desarchivado tienen prioridad sobre los de archivado.
LLOperShelve	BOOL	Archivado del operador bajo bajo. Establecido en verdadero por la interfaz de operador para archivar o rearchivar la condición baja baja. Requiere una transición desde falso en un escaneado del programa a verdadero en el siguiente escaneado del programa. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso. Los comandos de desarchivado tienen prioridad sobre los de archivado.

Parámetro de entrada	Tipo de datos	Descripción
ROCPoSOperShelve	BOOL	Archivado del operador de índice de cambio positivo. Establecido en verdadero por la interfaz de operador para archivar o rearchivar una condición de índice de cambio positivo. Requiere una transición desde falso en un escaneado del programa a verdadero en el siguiente escaneado del programa. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso. Los comandos de desarchivado tienen prioridad sobre los de archivado.
ROCNegOperShelve	BOOL	Archivado del operador de índice de cambio negativo. Establecido en verdadero por la interfaz de operador para archivar o rearchivar una condición de índice de cambio negativo. Requiere una transición desde falso en un escaneado del programa a verdadero en el siguiente escaneado del programa. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso. Los comandos de desarchivado tienen prioridad sobre los de archivado.
ProgUnshelveAll	BOOL	Establecido en verdadero por el programa de usuario para desarchivar todas las condiciones de esta alarma. Si se establece en verdadero tanto archivar como desarchivar, los comandos de desarchivado tienen prioridad sobre los de archivado. El valor predeterminado es falso.
HOperUnshelve	BOOL	Desarchivado del operador alto alto. Establecido en verdadero por la interfaz de operador para desarchivar la condición alta alta. La instrucción de alarma cambia este parámetro a falso. Si se establece en verdadero tanto archivar como desarchivar, los comandos de desarchivado tienen prioridad sobre los de archivado. El valor predeterminado es falso.
HOperUnshelve	BOOL	Desarchivado del operador alto. Establecido en verdadero por la interfaz de operador para desarchivar la condición alta. La instrucción de alarma cambia este parámetro a falso. Si se establece en verdadero tanto archivar como desarchivar, los comandos de desarchivado tienen prioridad sobre los de archivado. El valor predeterminado es falso.
LOperUnshelve	BOOL	Desarchivado del operador bajo. Establecido en verdadero por la interfaz de operador para desarchivar la condición baja. La instrucción de alarma cambia este parámetro a falso. Si se establece en verdadero tanto archivar como desarchivar, los comandos de desarchivado tienen prioridad sobre los de archivado. El valor predeterminado es falso.
LLOperUnshelve	BOOL	Desarchivado del operador bajo bajo. Establecido en verdadero por la interfaz de operador para desarchivar la condición baja baja. La instrucción de alarma cambia este parámetro a falso. Si se establece en verdadero tanto archivar como desarchivar, los comandos de desarchivado tienen prioridad sobre los de archivado. El valor predeterminado es falso.
ROCPoSOperUnshelve	BOOL	Desarchivado del operador de índice de cambio positivo. Establecido en verdadero por la interfaz de operador para desarchivar una de índice de cambio positivo. La instrucción de alarma cambia este parámetro a falso. Si se establece tanto archivar como desarchivar, los comandos de desarchivado tienen prioridad sobre los de archivado. El valor predeterminado es falso.

Parámetro de entrada	Tipo de datos	Descripción
ROCNegOperUnshelve	BOOL	Desarchivado del operador de índice de cambio negativo. Establecido en verdadero por la interfaz de operador para desarchivar una de índice de cambio negativo. La instrucción de alarma cambia este parámetro a falso. Si se establece en verdadero tanto archivar como desarchivar, los comandos de desarchivado tienen prioridad sobre los de archivado. El valor predeterminado es falso.
ProgDisable	BOOL	El programa del usuario lo establece en verdadero para deshabilitar la alarma. El valor predeterminado es falso. Diagrama de escalera: Copiado del operando de instrucción. Texto estructurado: Copiado del operando de instrucción.
OperDisable	BOOL	La interfaz de operador lo establece en verdadero para deshabilitar la alarma. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso.
ProgEnable	BOOL	El programa del usuario lo establece en verdadero para habilitar la alarma. Tiene prioridad sobre un comando de inhabilitación. El valor predeterminado es falso. Diagrama de escalera: Copiado del operando de instrucción. Texto estructurado: Copiado del operando de instrucción.
OperEnable	BOOL	La interfaz de operador lo establece en verdadero para habilitar la alarma. Tiene prioridad sobre el comando de inhabilitación. La instrucción de alarma borra este parámetro en falso. El valor predeterminado es falso.
AlarmCountReset	BOOL	Establecido en verdadero por el programa de usuario para restablecer los conteos de alarma para todas las condiciones. El valor predeterminado es falso.
HHMinDurationEnable	BOOL	Habilitación de la duración mínima de alarma alta-alta. Establecer en verdadero para habilitar el temporizador de duración mínima cuando se detecta la condición alta alta. El valor predeterminado es verdadero.
HMinDurationEnable	BOOL	Habilitación de la duración mínima de alarma alta. Establecer en verdadero para habilitar el temporizador de duración mínima cuando se detecta la condición alta. El valor predeterminado es verdadero.
LMinDurationEnable	BOOL	Habilitación de la duración mínima de alarma baja. Establecer en verdadero para habilitar el temporizador de duración mínima cuando se detecta la condición baja. El valor predeterminado es verdadero.
LLMinDurationEnable	BOOL	Habilitación de la duración mínima de alarma baja-baja. Establecer en verdadero para habilitar el temporizador de duración mínima cuando se detecta la condición baja baja. El valor predeterminado es verdadero.
HHLimit	REAL	Límite de alarma alta alta. Valor válido = HLlimit < HHLimit < punto flotante positivo máximo. Valor predeterminado = 0,0.

Parámetro de entrada	Tipo de datos	Descripción
HHSeverity	DINT	Gravedad de la condición de alarma alta alta. Esto no afecta al procesamiento de las alarmas por el controlador pero puede usarse para clasificar y filtrar las funciones del suscriptor de alarma. Valor válido = del 1 al 1000 (1000 = la más grave; 1 = la menos grave). Valor predeterminado = 500.
HLimit	REAL	Límite de alarma alta. Valor válido = LLimit < HLimit < HHLimit. Valor predeterminado = 0,0.
HSeverity	DINT	Gravedad de la condición de alarma alta. Esto no afecta al procesamiento de las alarmas por el controlador pero puede usarse para clasificar y filtrar las funciones del suscriptor de alarma. Valor válido = del 1 al 1000 (1000 = la más grave; 1 = la menos grave). Valor predeterminado = 500.
LLimit	REAL	Límite de alarma baja. Valor válido = LLLimit < LLimit < HLimit. Valor predeterminado = 0,0.
LSeverity	DINT	Gravedad de la condición de alarma baja. Esto no afecta al procesamiento de las alarmas por el controlador pero puede usarse para clasificar y filtrar las funciones del suscriptor de alarma. Valor válido = del 1 al 1000 (1000 = la más grave; 1 = la menos grave). Valor predeterminado = 500.
LLLlimit	REAL	Límite de alarma baja baja. Valor válido = punto flotante negativo máximo < LLLlimit < LLimit. Valor predeterminado = 0,0.
LLSeverity	DINT	Gravedad de la condición de alarma baja baja. Esto no afecta al procesamiento de las alarmas por el controlador pero puede usarse para clasificar y filtrar las funciones del suscriptor de alarma. Valor válido = del 1 al 1000 (1000 = la más grave; 1 = la menos grave). Valor predeterminado = 500.
MinDurationPRE	DINT	Duración mínima preseleccionada (en milisegundos) para que una condición de nivel de alarma se mantenga como verdadera antes de que la condición se marque como InAlarm y se envíe la notificación de alarma a los clientes. El controlador recopila los datos de alarma tan pronto como se detecta la condición de alarma para que no se pierdan datos mientras se espera hasta alcanzar la duración mínima. No se aplica a los límites del índice de cambio o a condiciones para las que la detección de la duración mínima está deshabilitada. MinDurationPRE sólo se aplica a la primera excursión desde el estado normal en cualquier dirección. Por ejemplo, una vez que la condición alta excede el tiempo de espera, la condición alta alta se activa inmediatamente, al mismo tiempo que la condición baja espera a que llegue el tiempo de espera. Valor válido = 0...2147483647. Valor predeterminado = 0.
ShelveDuration	DINT	Duración del tiempo (en minutos) en la que se archivará una alarma archivada. El tiempo mínimo es un minuto. El tiempo máximo lo define MaxShelveDuration.
MaxShelveDuration	DINT	Duración máxima del tiempo (en minutos) en la que se puede archivar una alarma.

Parámetro de entrada	Tipo de datos	Descripción
Deadband	REAL	<p>Banda muerta para detectar que los niveles de alarma alta alta, alta, baja y baja han vuelto a normal.</p> <p>Una banda muerta distinta a cero puede reducir la vibración de la condición de alarma si el valor In cambia continuamente pero permanece cerca del umbral de condición de nivel. El valor de banda muerta no afecta a la transición al estado InAlarm (activo). Una vez que la condición de nivel está activa, pero antes de que la condición vuelva al estado inactivo (normal), el valor In debe:</p> <ul style="list-style-type: none"> <li>Caiga por debajo del umbral menos la banda muerta (para condiciones alta y alta alta)</li> <li>O</li> <li>esté por encima del umbral más la banda muerta (para condiciones baja y baja baja).</li> </ul> <p>La banda muerta no se utiliza para condicionar la medición de tiempo de duración mínima.</p> <p>Valor válido = 0 = Banda muerta &lt; Abarca desde la primera alarma baja habilitada hasta la primera alarma alta habilitada.</p> <p>Valor predeterminado = 0,0.</p>
ROCPosLimit	REAL	<p>Límite para un índice de cambio en aumento en unidades por segundo. La detección está habilitada para cualquier valor &gt; 0,0 si ROCPeriod también es &gt; 0,0.</p> <p>Valor válido = 0,0...32767,0.</p> <p>Valor predeterminado = 0,0.</p>
ROCPosSeverity	DINT	<p>Gravedad de la condición de índice de cambio en aumento. Esto no afecta al procesamiento de las alarmas por el controlador pero puede usarse para clasificar y filtrar las funciones del suscriptor de alarma.</p> <p>Valor válido = 1...32767,0.</p> <p>Valor predeterminado = 500.</p>
ROCNegLimit	REAL	<p>Límite para un índice de cambio en descenso en unidades por segundo. La detección está habilitada para cualquier valor &gt; 0,0 si ROCPeriod también es &gt; 0,0.</p> <p>Valor válido = de 0,0 al punto flotante posible máximo.</p> <p>Valor predeterminado = 0,0.</p>
ROCNegSeverity	DINT	<p>Gravedad de la condición de índice de cambio en descenso. Esto no afecta al procesamiento de las alarmas por el controlador pero puede usarse para clasificar y filtrar las funciones del suscriptor de alarma.</p> <p>Valor válido = del 1 al 1000 (1000 = la más grave; 1 = la menos grave).</p> <p>Valor predeterminado = 500.</p>
ROCPeriod	REAL	<p>Período de tiempo en segundos para calcular (intervalo de muestreo) el valor del índice de cambio. Cada vez que caduque el intervalo de muestreo, se almacena una nueva muestra de In y se vuelve a calcular el valor ROC. La detección del índice de cambio está activada para cualquier valor &gt; 0,0.</p> <p>Valor válido = 0,0...32767,0.</p> <p>Valor predeterminado = 0,0.</p>

### Conectar un botón a la etiqueta OperShelve

Para evitar que la alarma se vuelva a archivar sin que lo deseé, la instrucción de alarma solo procesa la etiqueta OperShelve si hace una transición desde falso a verdadero entre un escaneado del programa y el siguiente. Si un operador presiona

un botón para archivar la alarma mientras está establecida la etiqueta ProgUnshelve, la alarma no se archiva porque la etiqueta ProgUnshelve tiene prioridad. Sin embargo, dado que los escaneados de programa se completan en milisegundos, el operador puede seguir pulsando el botón por lo que la etiqueta OperShelve sigue establecido sobre varios escaneados de programa incluso aunque la etiqueta ProgUnshelve haya sido falso. Esto significa que la alarma no está archivada.

Para archivar la alarma, el operador puede soltar y volver a presionar el botón.

### Parámetros de salida

Estos parámetros de salida son comunes para texto estructurado.

Parámetro de salida	Tipo de datos	Descripción
AnyInAlarmUnack	BOOL	Combinación de estado activo de alarma y estado de confirmación. Establecido en verdadero cuando se detecta cualquier condición de alarma y no se ha confirmado. Borrado y establecido en falso cuando ninguna condición de alarma esté activa, confirmada o ambas.
HHInAlarm	BOOL	Estado de condición de alarma alta alta. Establecido en verdadero cuando una condición alta alta se activa. Borrado y establecido en falso cuando no hay ninguna condición alta alta.
HInAlarm	BOOL	Estado de condición de alarma alta. Establecido en verdadero cuando una condición alta se activa. Borrado y establecido en falso cuando no hay ninguna condición alta.
LInAlarm	BOOL	Estado de condición de alarma baja. Establecido en verdadero cuando una condición baja se activa. Borrado y establecido en falso cuando no hay ninguna condición baja.
LLInAlarm	BOOL	Estado de la condición de alarma baja baja. Establecido en verdadero cuando una condición baja baja se activa. Borrado y establecido en falso cuando no hay ninguna condición baja baja.
ROCPoSInAlarm	BOOL	Estado de condición de alarma de índice de cambio positivo. Establecido en verdadero cuando haya una condición de índice de cambio positivo. Borrado y establecido en falso cuando no hay ninguna condición positiva de índice de cambio.
ROCNegInAlarm	BOOL	Estado de condición de alarma de índice de cambio negativo. Establecido en verdadero cuando hay una condición negativa de índice de cambio. Borrado y establecido en falso cuando no hay ninguna condición negativa de índice de cambio.
ROC	REAL	Índice de cambio calculado del valor In. Este valor se actualiza cuando se escanea la instrucción después de cada ROCPeriod transcurrido. Se utiliza el valor ROC para evaluar las condiciones de ROCPoSInAlarm y ROCNegInAlarm. ROC = (muestra actual de In – muestra anterior de In)/ROCPeriod.

Parámetro de salida	Tipo de datos	Descripción
HHAcked	BOOL	Estado confirmado de condición alta alta. Establecido en verdadero cuando una condición alta alta se confirma. Siempre establecido en verdadero cuando AckRequired se borra y establece en falso. Borrado y establecido en false cuando no se confirma ninguna condición alta alta.
HAcked	BOOL	Estado confirmado de condición alta. Establecido en verdadero cuando una condición alta se confirma. Siempre establecido en verdadero cuando AckRequired se borra y establece en falso. Borrado y establecido en falso cuando no se confirma ninguna condición alta.
LAcked	BOOL	Estado confirmado de condición baja. Establecido en verdadero cuando una condición baja se confirma. Siempre establecido en verdadero cuando AckRequired se borra y establece en falso. Borrado y establecido en falso cuando no se confirma ninguna condición baja.
LLAcked	BOOL	Estado confirmado de condición baja baja. Establecido en verdadero cuando una condición baja baja se confirma. Siempre establecido en verdadero cuando AckRequired se borra y establece en falso. Borrado y establecido en falso cuando no se confirma ninguna condición baja baja.
ROCPoSAcked	BOOL	Estado confirmado de condición de índice de cambio positivo. Establecido en verdadero cuando se confirma una condición positiva de índice de cambio. Siempre establecido en verdadero cuando AckRequired se borra y establece en falso. Borrado y establecido en falso cuando no se confirma ninguna condición positiva de índice de cambio.
ROCNegAcked	BOOL	Estado confirmado de condición de índice de cambio negativo. Establecido en verdadero cuando se confirma una condición negativa de índice de cambio. Siempre establecido en verdadero cuando AckRequired se borra y establece en falso. Borrado y establecido en falso cuando no se confirma ninguna condición negativa de índice de cambio.
HHInAlarmUnack	BOOL	Combinación de estado activo de condición alta alta y estado de no confirmación. Establecido en verdadero cuando la condición alta alta está activa (HHInAlarm es verdadero) y no confirmada. Se borra a falso cuando la condición alta alta no está activa, está confirmada o ambas.
HInAlarmUnack	BOOL	Combinación de estado activo de condición alta y estado de no confirmación. Establecido en verdadero cuando la condición alta está activa (HInAlarm es verdadero) y no confirmada. Se borra y se establece en falso cuando la condición alta no está activa, está confirmada o ambas.
LInAlarmUnack	BOOL	Combinación de estado activo de condición baja y estado de no confirmación. Establecido en verdadero cuando la condición baja está activa (LInAlarm es verdadero) y no confirmada. Se borra y se establece en falso cuando la condición baja no está activa, está confirmada o ambas.
LLInAlarmUnack	BOOL	Combinación de estado activo de condición baja baja y estado de no confirmación. Establecido en verdadero cuando la condición baja baja está activa (LLInAlarm es verdadero) y no confirmada. Se borra y se establece en falso cuando la condición baja baja no está activa, está confirmada o ambas.

Parámetro de salida	Tipo de datos	Descripción
ROCPoInAlarmUnack	BOOL	Combinación de estado activo de condición de índice de cambio positivo y estado de no confirmación. Establecido en verdadero cuando la condición de índice de cambio positivo está activa (ROCPoInAlarm es verdadero) y no confirmada. Se borra y se establece en falso cuando la condición de índice de cambio positivo no está activa, está confirmada o ambas.
ROCNegInAlarmUnack	BOOL	Combinación de estado activo de condición de índice de cambio negativo y estado de no confirmación. Establecido en verdadero cuando la condición de índice de cambio negativo está activa (ROCNegInAlarm es verdadero) y no confirmada. Se borra y se establece en falso cuando la condición de índice de cambio negativo no está activa, está confirmada o ambas.
Suppressed	BOOL	Estado de alarma suprimida. Establecido en verdadero cuando se haya suprimido la alarma. Se borra y se establece en falso cuando no se ha suprimido la alarma.
HHShelved	BOOL	Estado de archivado de la condición alta alta. Establecido en verdadero cuando una condición alta alta se archiva. Se borra y se establece en falso cuando la condición alta alta se desarchiva.
HShelved	BOOL	Estado de archivado de la condición alta. Establecido en verdadero cuando una condición alta se archiva. Se borra y se establece en falso cuando la condición alta se desarchiva.
LShelved	BOOL	Estado de archivado de la condición baja. Establecido en verdadero cuando una condición baja se archiva. Se borra y se establece en falso cuando la condición baja se desarchiva.
LLShelved	BOOL	Estado de archivado de la condición baja baja. Establecido en verdadero cuando una condición baja baja se archiva. Se borra y se establece en falso cuando la condición baja baja se desarchiva.
ROCPoShelved	BOOL	Estado archivado de condición de índice de cambio positivo. Establecido en verdadero cuando se archiva una condición de índice de cambio positivo. Se borra y se establece en falso cuando la condición de índice de cambio positivo se desarchiva.
ROCNegShelved	BOOL	Estado archivado de condición de índice de cambio negativo. Establecido en verdadero cuando se archiva una condición de índice de cambio negativo. Se borra y se establece en falso cuando la condición de índice de cambio negativo se desarchiva.
Disabled	BOOL	Estado de alarma deshabilitada. Establecido en verdadero cuando la alarma no está disponible (deshabilitada). Se borra y se establece en falso cuando se haya habilitado la alarma.
HHInAlarmTime	LINT	Marca de tiempo cuando la instrucción ALMA detecta que el valor In ha excedido el límite de la condición alta alta para la transición más reciente al estado activo.
HHAlarmCount	DINT	Número de veces que se ha activado la condición alta alta. Si se alcanza el valor máximo, el contador deja el valor en el valor máximo de conteo.
HInAlarmTime	LINT	Marca de tiempo cuando la instrucción ALMA detecta que el valor In ha excedido el límite de la condición alta para la transición más reciente al estado activo.

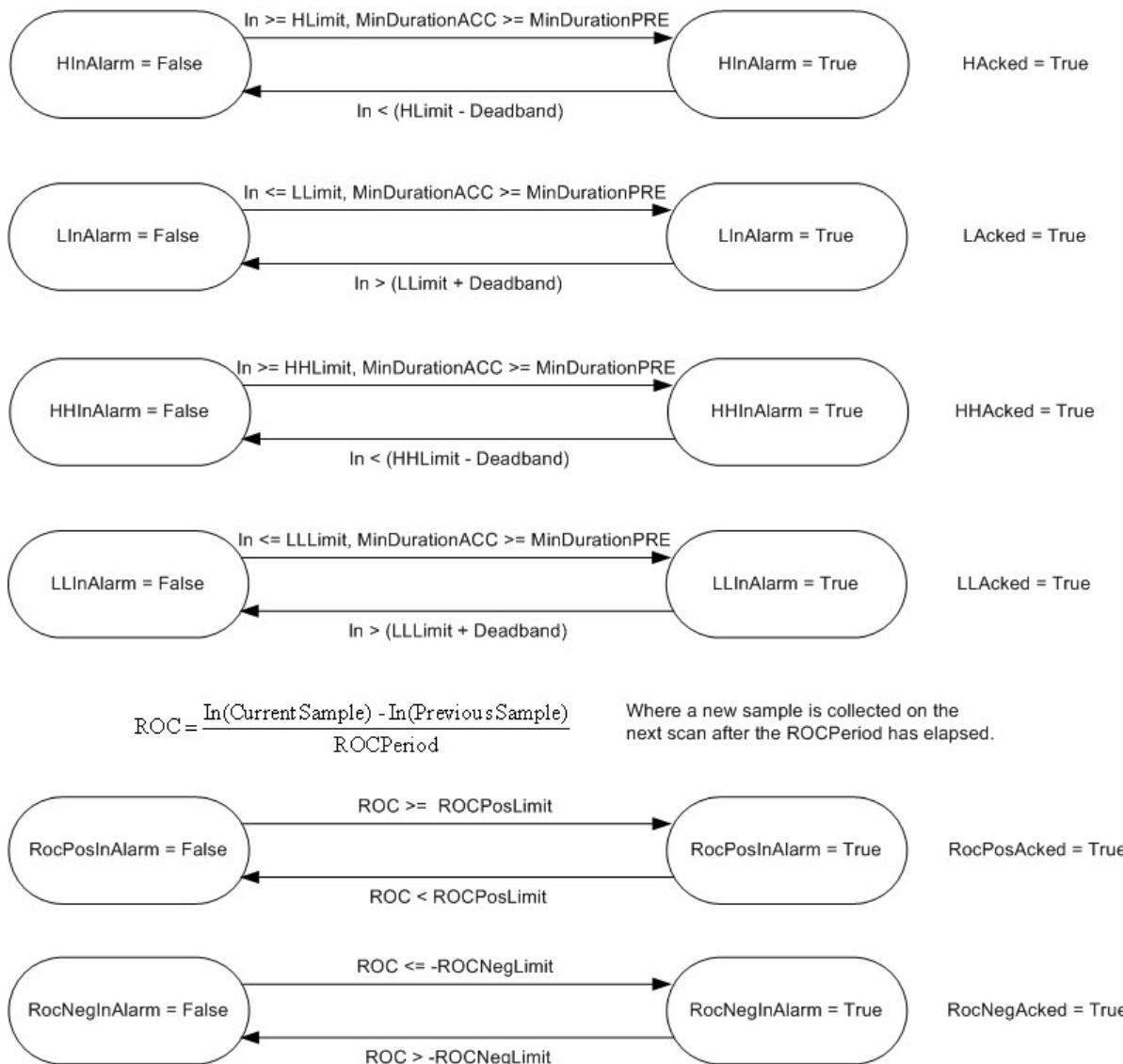
Parámetro de salida	Tipo de datos	Descripción
HAlarmCount	DINT	Número de veces que se ha activado la condición alta. Si se alcanza el valor máximo, el contador deja el valor en el valor máximo de conteo.
LInAlarmTime	LINT	Marca de tiempo cuando la instrucción ALMA detecta que el valor In ha excedido el límite de la condición baja para la transición más reciente al estado activo.
LAlarmCount	DINT	Número de veces que se ha activado la condición baja. Si se alcanza el valor máximo, el contador deja el valor en el valor máximo de conteo.
LLInAlarmTime	LINT	Marca de tiempo cuando la instrucción ALMA detecta que el valor In ha excedido el límite de la condición baja para la transición más reciente al estado activo.
LLAlarmCount	DINT	Número de veces que se ha activado la condición baja. Si se alcanza el valor máximo, el contador deja el valor en el valor máximo de conteo.
ROCPoInAlarmTime	LINT	Marca de tiempo cuando la instrucción ALMA detecta que el valor In ha excedido el límite de la condición de índice de cambio positivo para la transición más reciente al estado activo.
ROCPoInAlarmCount	DINT	Número de veces que se ha activado la condición de índice de cambio positivo. Si se alcanza el valor máximo, el contador deja el valor en el valor máximo de conteo.
ROCNegInAlarmTime	LINT	Marca de tiempo cuando la instrucción ALMA detecta que el valor In ha excedido el límite de la condición de índice de cambio negativo para la transición más reciente al estado activo.
ROCNegAlarmCount	DINT	Número de veces que se ha activado la condición de índice de cambio negativo. Si se alcanza el valor máximo, el contador deja el valor en el valor máximo de conteo.
AckTime	LINT	Marca de tiempo de la confirmación de la condición más reciente. Si la alarma no requiere confirmación, esta marca de tiempo es igual a la hora de alarma de la condición más reciente.
RetToNormalTime	LINT	Marca de tiempo de alarma que vuelve a un estado normal.
AlarmCountResetTime	LINT	Marca de tiempo que indica cuándo se ha restablecido el conteo de alarmas.
ShelveTime	LINT	La marca de tiempo indica cuando una condición de alarma se ha archivado por última vez. Establecido por el controlador cuando una condición de alarma se archiva. Todas las condiciones de alarma se pueden archivar y desarchivar varias veces. Cada vez que la condición de alarma se archiva la marca de tiempo se establece en la hora actual.
UnshelveTime	LINT	Marca de tiempo que indica cuando todas las condiciones de alarma se van a desarchivar. El valor solo se establece cuando no se ha archivado todavía ninguna condición de alarma. La marca de tiempo viene determinado por la suma del periodo de tiempo de ShelveDuration y la hora actual. Si una condición de alarma se desarchiva de forma programática o por un operador y no se archiva ninguna otra condición de alarma, entonces el valor se establece en la hora actual.

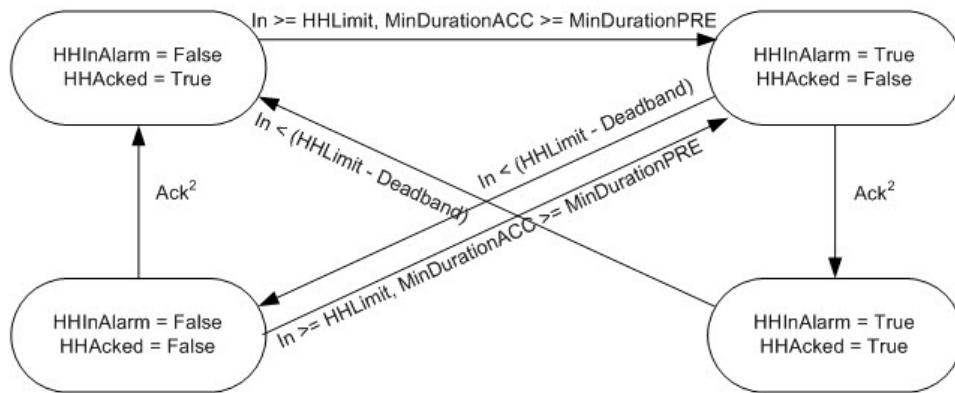
Parámetro de salida	Tipo de datos	Descripción
Status	DINT	Indicadores de estado combinado: Status.0 = InstructFault Status.1 = InFaulted Status.2 = SeverityInv Status.3 = AlarmLimitsInv Status.4 = DeadbandInv Status.5 = ROCPosLimitInv Status.6 = ROCNegLimitInv Status.7 = ROCPeriodInv Status.8 = Overflow
InstructFault (Status.0)	BOOL	Existen condiciones de error de instrucción No se trata de un error mayor o menor del controlador. Se comprueba los bits de estado restantes para determinar lo que ha ocurrido.
InFaulted (Status.1)	BOOL	El programa de usuario ha establecido InFault para indicar la existencia de datos de entrada de mala calidad. La alarma sigue evaluando In para las condiciones de alarma.
SeverityInv (Status.2)	BOOL	La configuración de gravedad de alarma no es válida. Si Gravedad < 1, la instrucción utiliza Gravedad = 1. Si la gravedad > 1000, la instrucción utiliza Severity = 1000.
AlarmLimitsInv (Status.3)	BOOL	La configuración de límite de alarma no es válida (por ejemplo, LLLimit < LLLimit). Si no es válido, la instrucción desactiva todos los bits activos de las condiciones de nivel. Hasta que no se haya eliminado el fallo, no se detectará ninguna condición nueva de nivel.
DeadbandInv (Status.4)	BOOL	La configuración de banda muerta no es válida. Si no es válida, la instrucción utiliza Banda muerta = 0,0. Valor válido = 0 = Banda muerta < Abarca desde la primera alarma baja habilitada hasta la primera alarma alta habilitada.
ROCPosLimitInv (Status.5)	BOOL	Límite de índice de cambio positivo no válido. Si no es válido, la instrucción utilizará ROCPosLimit = 0,0 y la detección de índice de cambio positivo no estará disponible.
ROCNegLimitInv (Status.6)	BOOL	Límite de índice de cambio negativo no válido. Si no es válido, la instrucción utilizará ROCNegLimit = 0,0 y la detección de índice de cambio negativo no estará disponible.
ROCPeriodInv (Status.7)	BOOL	Período de índice de cambio no válido. Si no es válido, la instrucción utilizará ROCPeriod = 0,0 y la detección de índice de cambio no estará disponible.
Overflow (Status.8)	BOOL	Desbordamiento. El bit de desbordamiento se establece en verdadero cuando se detecta una condición de desbordamiento. El bit de desbordamiento se borra y se establece en falso cuando la condición de desbordamiento se ha corregido.

### Diagramas de estado de la alarma analógica

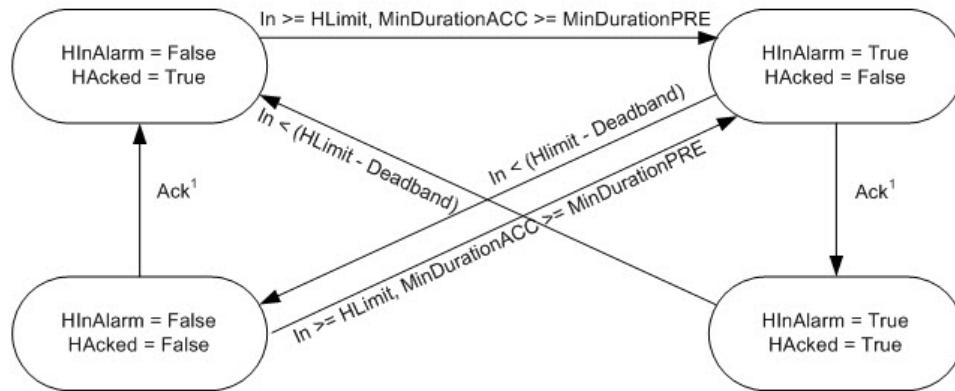
Estas ilustraciones muestran la forma en la que la alarma analógica responde a los cambios en las condiciones de alarma y los comandos del operador.

#### AckRequired = False

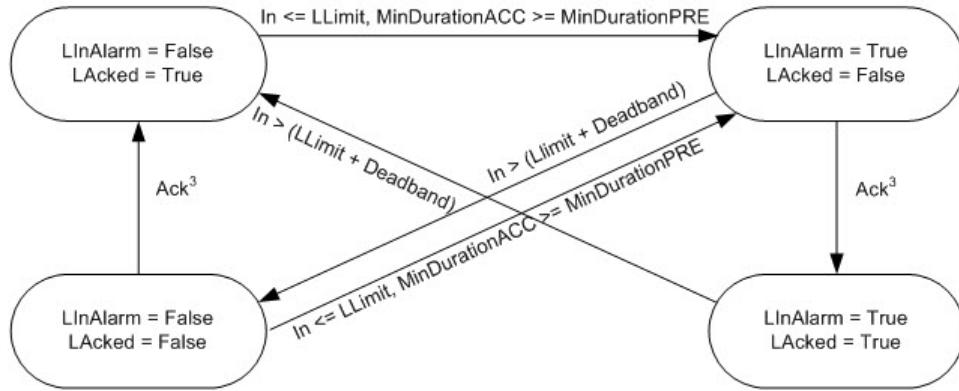


**AckRequired = True**

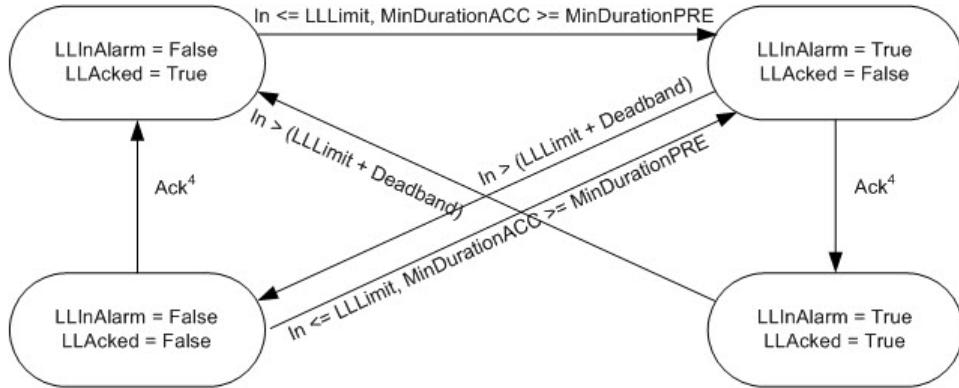
<sup>2</sup> HH alarm condition can be acked by several different ways: HHProgAck, HHOperAck, ProgAckAll, OperAckAll, clients (RSLogix 5000, RSview)



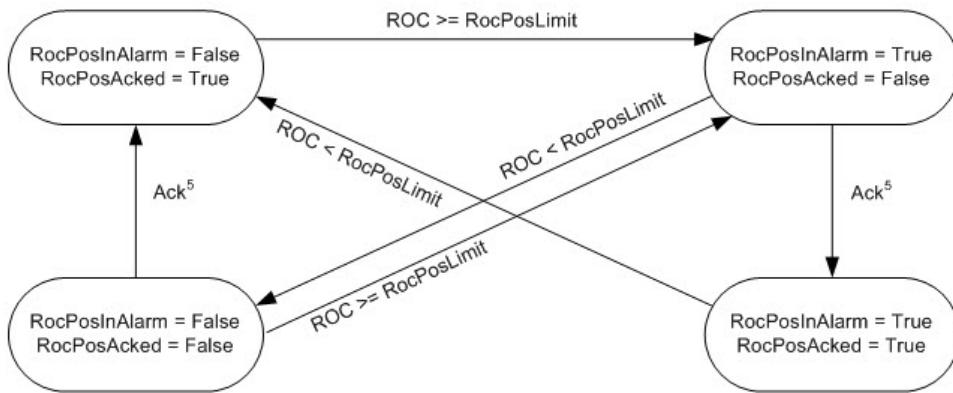
<sup>1</sup> H alarm condition can be acked by several different ways: HProgAck, HOperAck, ProgAckAll, OperAckAll, clients (RSLogix 5000, RSview)



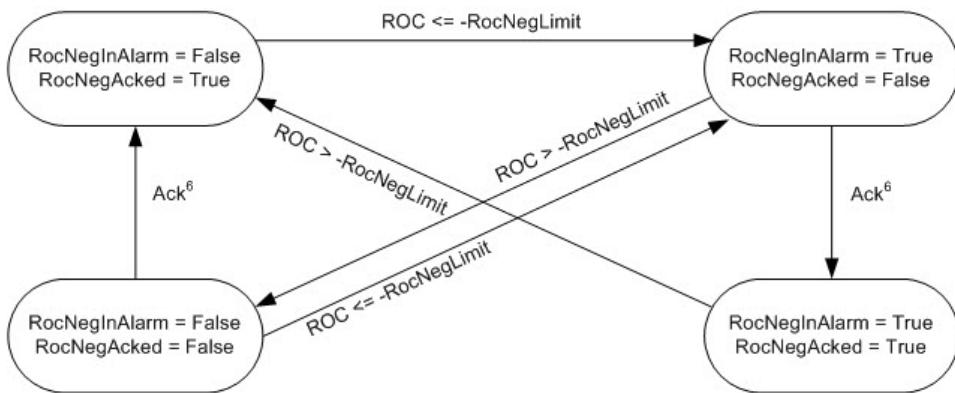
<sup>3</sup> L alarm condition can be acked by several different ways: LProgAck, LOperAck, ProgAckAll, OperAckAll, clients (RSLogix 5000, RSview)



<sup>4</sup> LL alarm condition can be acked by several different ways: LLProgAck, LLOperAck, ProgAckAll, OperAckAll, clients (RSLogix 5000, RSview)



<sup>5</sup> RocPos alarm condition can be acked by several different ways: RocPosProgAck, RocPosOperAck, ProgAckAll, OperAckAll, clients (RSLogix 5000, RSview)

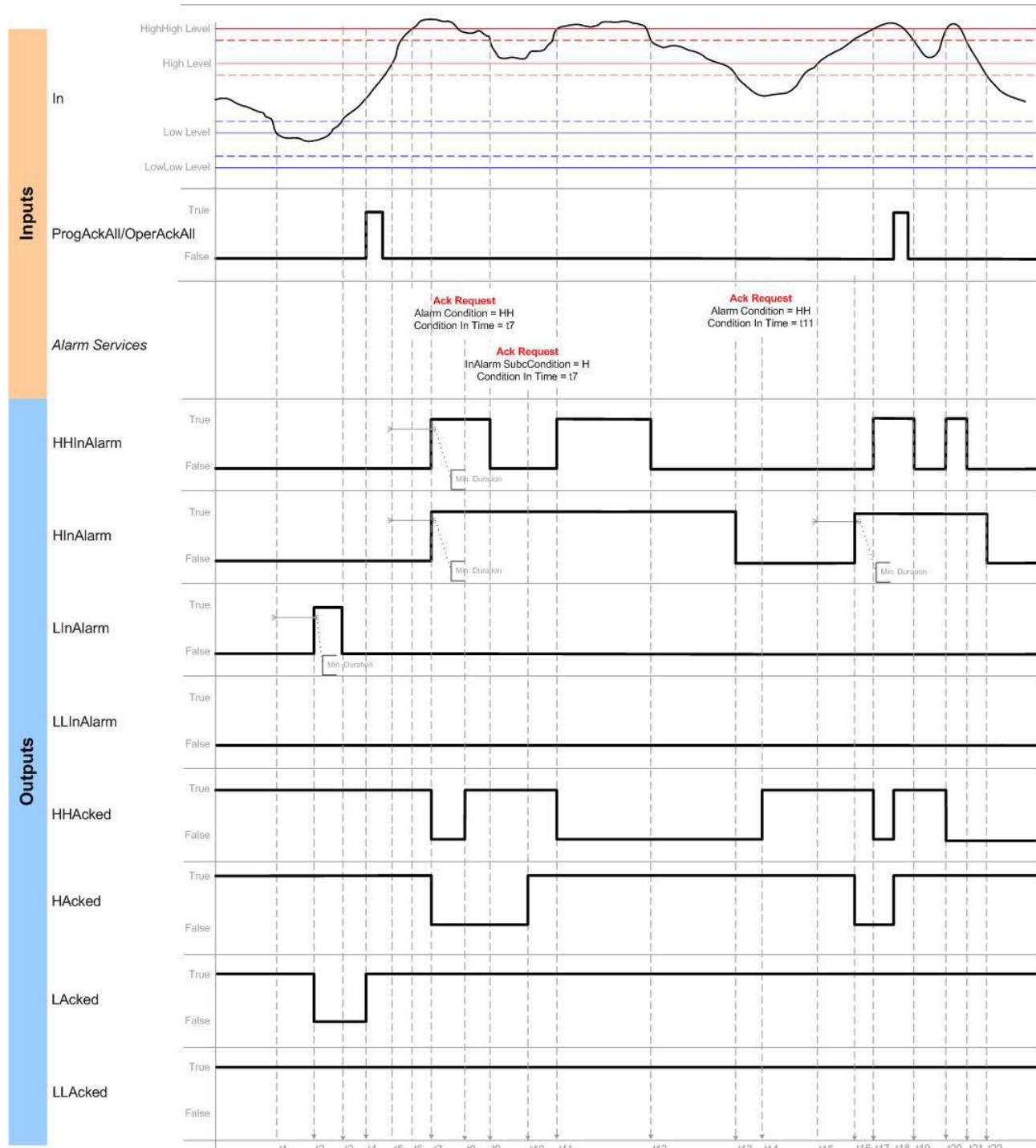


<sup>6</sup> RocNeg alarm condition can be acked by several different ways: RocNegProgAck, RocNegOperAck, ProgAckAll, OperAckAll, clients (RSLogix 5000, RSview)

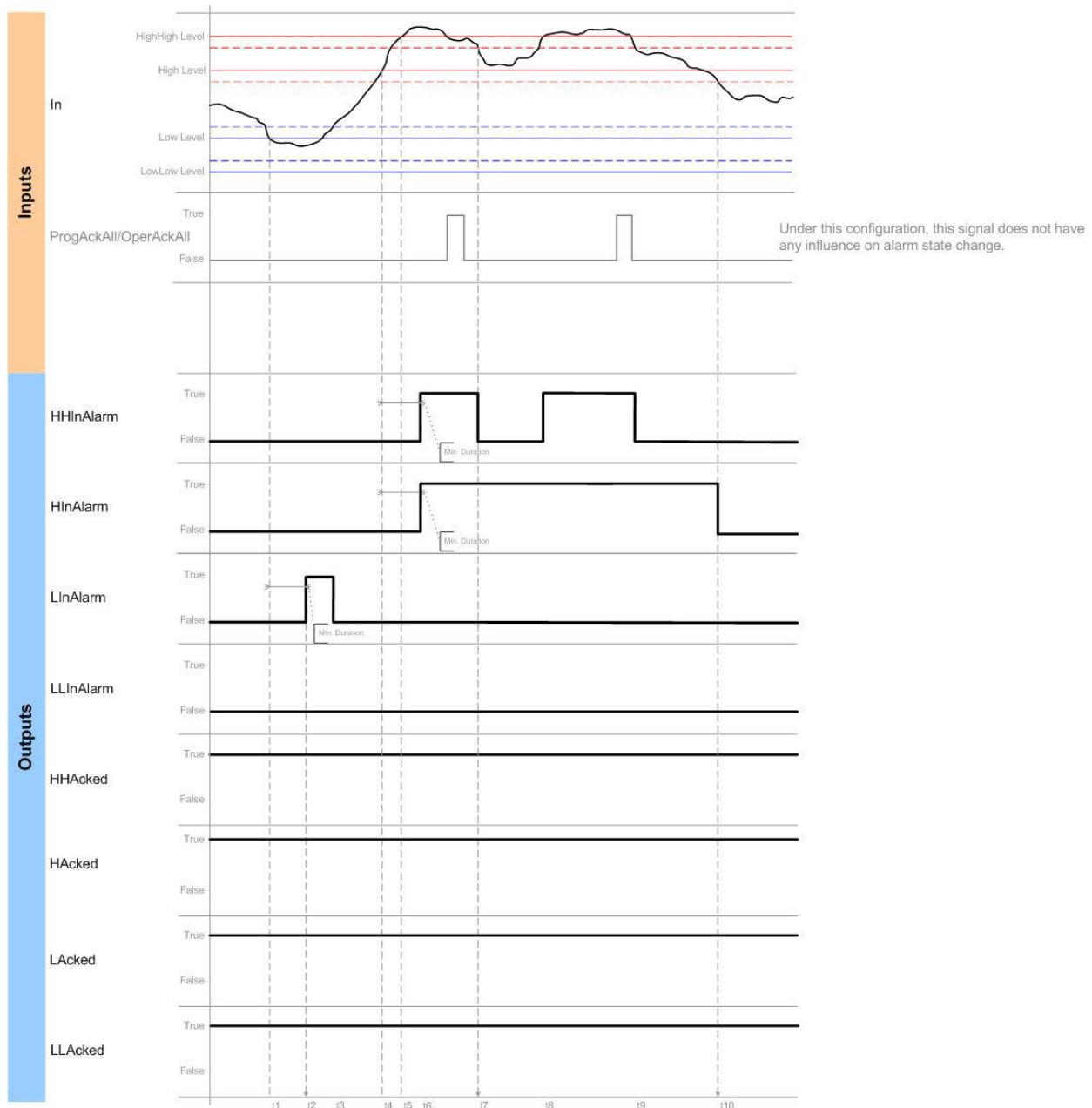
### Diagramas de temporización de la alarma analógica

Estos diagramas de temporización muestran la secuencia de las operaciones de la alarma analógica.

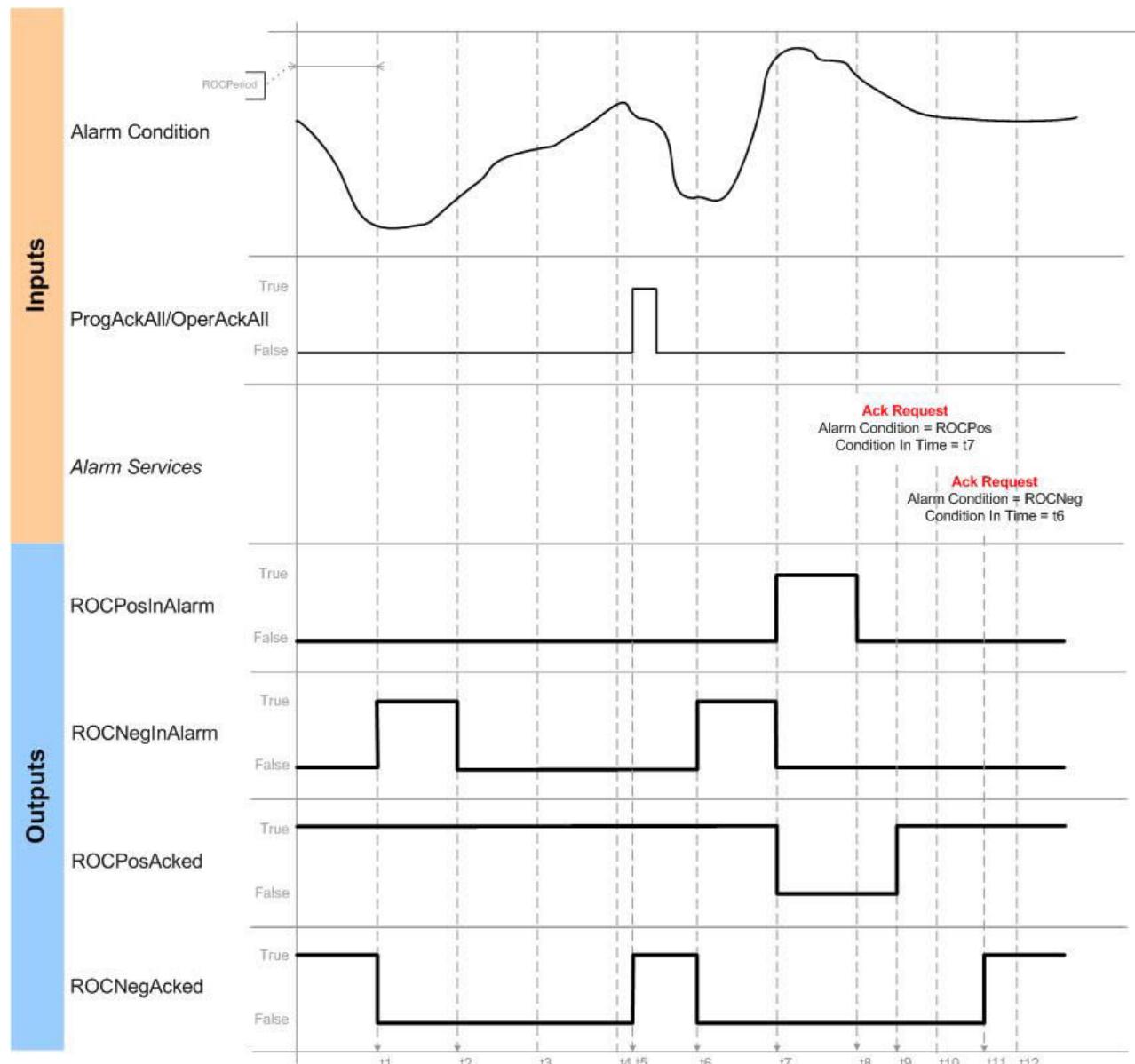
### Confirmación de comportamiento del nivel de condiciones



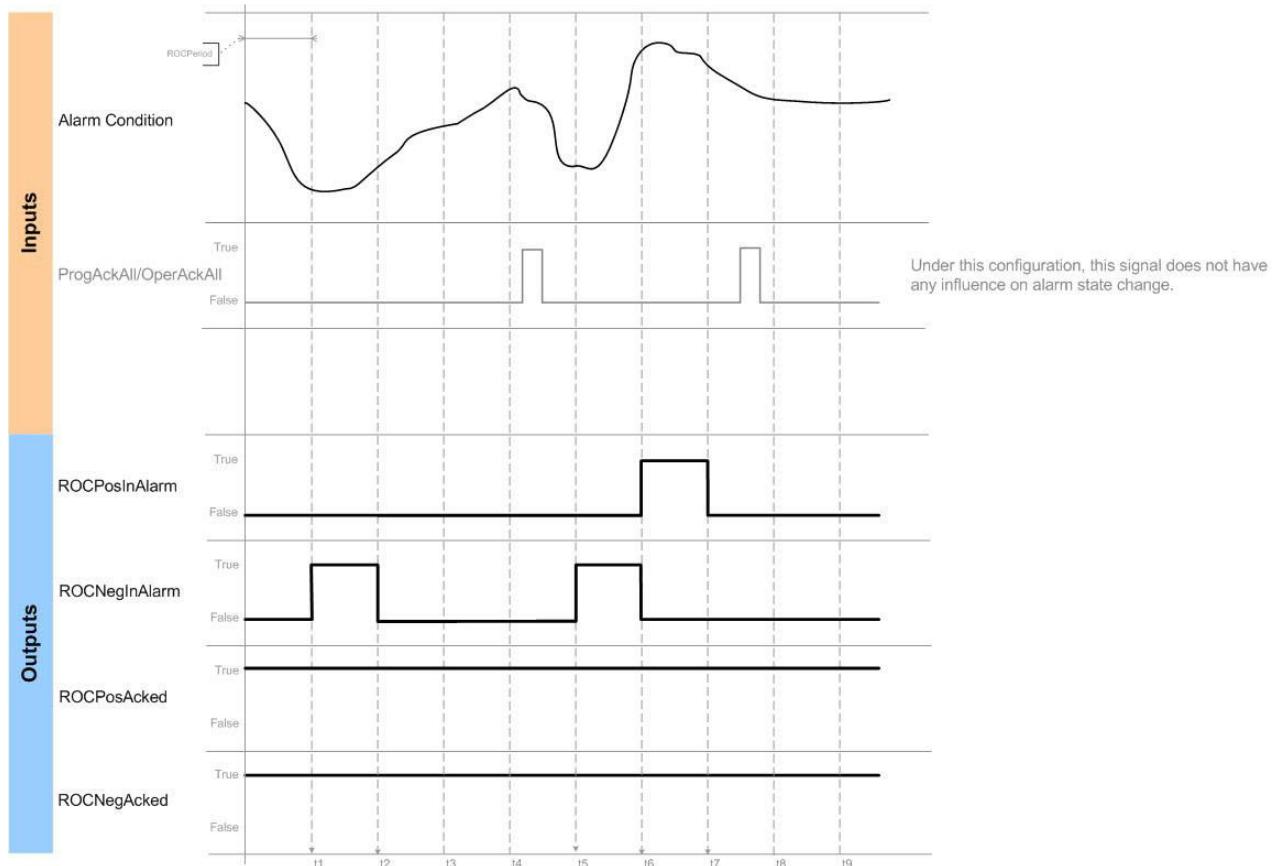
### Sin confirmación de comportamiento del nivel de condiciones



### Confirmación de comportamiento de las condiciones ROC



### Sin confirmación de comportamiento de las condiciones ROC



### Afectar a las marcas de estado matemático

No

### Condiciones de fallo

No es específico para esta instrucción. Consulte los *Atributos comunes* para fallos relacionados con el operando.

### Ejecución

En Texto estructurado, EnableIn siempre es verdadero durante un escaneado normal. Por tanto, si la instrucción se encuentra en la ruta de control activada por la lógica, se ejecutará.

Condición/estado	Acción realizada
Pre-escaneado	La condición de entrada de reglón se borra a falsa. Se han borrado todos los parámetros de la estructura ALMA. Se han confirmado todas las condiciones de alarma. Se han borrado todas las peticiones del operador. Se han borrado todas las marcas de tiempo. Se han borrado todas las marcas de entrega.
La condición de entrada de reglón es falsa	La condición de entrada de reglón se borra a falsa.
La condición de entrada de reglón es verdadera	La condición de entrada de reglón se establece en verdadera. La instrucción se ejecuta.
Post-escaneado	La condición de entrada de reglón se borra a falsa.

### Ejemplo

A continuación se muestra un ejemplo de una instrucción de ALMA en texto estructurado. En este ejemplo, se supervisa el nivel de transmisión del Tanque 32 (Tank32LT) para las condiciones de alarma. Se puede usar la etiqueta Tank32LevelAck para confirmar todas las condiciones de esta alarma.

```
ALMA(Tank32Level,Tank32LT,Tank32LevelAck,0,0);
```

### Consulte también

[Alarma analógica \(ALMA\): Lógica de escalera](#) en la página 60

[Alarma analógica \(ALMA\): Bloque de funciones](#) en la página 36

[Sintaxis de texto estructurado](#) en la página 955

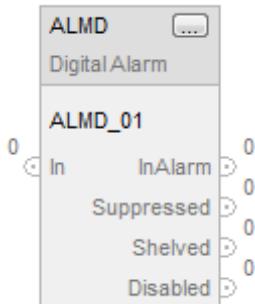
[Atributos comunes](#) en la página 923

## Alarma digital (ALMD): Bloque de funciones

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción ALMD proporciona la alarma para cualquier valor booleano discreto.

## Bloque de funciones



## Operandos

### Bloque de funciones

Estos operandos se encuentran en la instrucción.

Operando	Tipo	Formato	Descripción
ALMD tag	ALARM_DIGITAL	Estructura	Estructura de ALMD.

## Ejecución

Condición/estado	Acción realizada
Pre-escaneado	La condición de entrada de peldaño se borra a falsa La salida InAlarm se borra y se establece en falso La salida Shelved se borra y se establece en falso La salida Acked se establece en verdadero Se han confirmado todas las condiciones de alarma. Se han borrado todas las peticiones del operador. Se han borrado todas las marcas de tiempo.
La condición de entrada de reglón es falsa	La condición de entrada de reglón se borra a falsa. Diagrama de escalera: El parámetro In se borra y se establece en falso. La instrucción se ejecuta.
La condición de entrada de reglón es verdadera	La condición de entrada de reglón se establece en verdadera. Diagrama de escalera: El parámetro In se establece en verdadero. La instrucción se ejecuta.
Post-escaneado	La condición de entrada de peldaño se borra a falsa.

## Parámetros de entrada

Parámetro de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Si se borra y establece en falso, la instrucción no se ejecuta y las salidas no se actualizan. Si está establecida, la instrucción se ejecuta. El valor predeterminado es verdadero. Texto estructurado: Ningún efecto. La instrucción siempre se ejecuta.
In	BOOL	Entrada de la señal digital a la instrucción. El valor predeterminado es falso.
InFault	BOOL	Indicador de estado incorrecto para la entrada. La aplicación de usuario puede establecer InFault para indicar que hay un error en la señal de entrada. Cuando se establece, la instrucción establece InFaulted (Status.1). Cuando se borra y establece en falso, la instrucción desactiva InFaulted (Status.1). En cualquier caso, la instrucción sigue evaluando las condiciones de alarma de In. El valor predeterminado está en falso (buen estado).
Condition	BOOL	Especifica cómo se activa la alarma. Cuando Condition está establecida en verdadera, la condición de alarma se activa si In está establecido en verdadero. Cuando Condition se borra a falso, la condición de alarma se activa si In se borra a falso. El valor predeterminado es verdadero.
AckRequired	BOOL	Especifica si es necesaria la confirmación de la alarma. Cuando se establece en verdadero, se necesita confirmación. Cuando se borra y establece en falso, no se necesita confirmación ya que Acked está establecida siempre en verdadero. El valor predeterminado es verdadero.
Latched	BOOL	Especifica si la alarma está enclavada. Las alarmas enclavadas se mantienen como InAlarm si la condición de alarma pasa a falso, hasta que se recibe un comando de restablecimiento. Cuando está establecida en verdadero, la alarma está enclavada. Cuando se borra y establece en falso, la alarma está desenclavada. El valor predeterminado es falso. Sólo es posible restablecer una alarma enclavada cuando la condición de alarma es falsa.
ProgAck	BOOL	El programa del usuario lo establece en verdadero para confirmar la alarma. Solo toma efecto si no se confirma la alarma. Requiere una transición de falso a verdadero. El valor predeterminado es falso.

Parámetro de entrada	Tipo de datos	Descripción
OperAck	BOOL	La interfaz de operador lo establece en verdadero para confirmar la alarma. Solo toma efecto si no se confirma la alarma. La instrucción desactiva este parámetro. El valor predeterminado es falso.
ProgReset	BOOL	El programa del usuario lo establece en verdadero para restablecer la alarma enclavada. Solo toma efecto si la alarma enclavada es InAlarm y la condición de alarma es falso. Requiere una transición de falso a verdadero. El valor predeterminado es falso.
OperReset	BOOL	La interfaz de operador lo establece en verdadero para restablecer la alarma enclavada. Solo toma efecto si la alarma enclavada es InAlarm y la condición de alarma es falso. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso.
ProgSuppress	BOOL	El programa del usuario lo establece en verdadero para suprimir la alarma. El valor predeterminado es falso.
OperSuppress	BOOL	La interfaz de operador lo establece en verdadero para suprimir la alarma. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso.
ProgUnsuppress	BOOL	El programa del usuario lo establece en verdadero para anular la supresión de la alarma. Tiene prioridad sobre los comandos de supresión. El valor predeterminado es falso.
OperUnsuppress	BOOL	La interfaz de operador lo establece en verdadero para anular la supresión de la alarma. Tiene prioridad sobre los comandos de supresión. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso.
OperShelve	BOOL	La interfaz de operador lo establece en verdadero para archivar o rearchivar la alarma. Requiere una transición desde falso en un escaneado del programa a verdadero en el siguiente escaneado del programa. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso. Los comandos de desarchivado tienen prioridad sobre los de archivado. El archivado de una alarma pospone el procesamiento de alarma. Es como suprimir una alarma, excepto que ese archivado tiene un tiempo limitado. Si se confirma una alarma mientras se está archivando, permanece confirmada incluso si vuelve a estar activa de nuevo. Se convierte en no confirmada cuando la duración del archivado finaliza, siempre que la alarma siga activa en ese momento.

Parámetro de entrada	Tipo de datos	Descripción
ProgUnshelve	BOOL	El programa del usuario lo establece en verdadero para desarchivar la alarma. Tiene prioridad sobre los comandos de archivado. El valor predeterminado es falso. Para obtener más información sobre cómo archivar una alarma, consulte la descripción del parámetro OperShelve.
OperUnshelve	BOOL	La interfaz de operador lo establece en verdadero para desarchivar la alarma. La instrucción de alarma cambia este parámetro a falso. Tiene prioridad sobre los comandos de archivado. Está borrado de forma predeterminada. Para obtener más información sobre cómo archivar una alarma, consulte la descripción del parámetro OperShelve.
ProgDisable	BOOL	El programa del usuario lo establece en verdadero para deshabilitar la alarma. El valor predeterminado es falso.
OperDisable	BOOL	La interfaz de operador lo establece en verdadero para deshabilitar la alarma. La instrucción de alarma cambia este parámetro a verdadero. El valor predeterminado es falso.
ProgEnable	BOOL	El programa del usuario lo establece en verdadero para habilitar la alarma. Tiene prioridad sobre un comando de inhabilitación. El valor predeterminado es falso.
OperEnable	BOOL	La interfaz de operador lo establece en verdadero para habilitar la alarma. Tiene prioridad sobre el comando de inhabilitación. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso.
AlarmCountReset	BOOL	La interfaz de operador lo establece en verdadero para restablecer el conteo de alarma a cero. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso.
UseProgTime	BOOL	Especifica si se usa el reloj del controlador o el valor ProgTime para los eventos de cambio de estado de alarma de marca de tiempo. Cuando está establecido en verdadero, el valor ProgTime proporciona la marca de tiempo. Cuando se borra y establece en falso, el reloj del controlador proporciona la marca de tiempo. El valor predeterminado es falso.
ProgTime	LINT	Si UseProgTime está establecido en verdadero, este valor se usa para proporcionar el valor de marca de tiempo para todos los eventos. De este modo, se permite que la aplicación aplique las marcas de tiempo obtenidas a partir del origen de la alarma, como un módulo de entrada de secuencia de eventos.

Parámetro de entrada	Tipo de datos	Descripción
Severity	DINT	Gravedad de la alarma. Esto no afecta al procesamiento de las alarmas por el controlador pero puede usarse para clasificar y filtrar las funciones del suscriptor de alarma. Valor válido = del 1 al 1000 (1000 = la más grave; 1 = la menos grave). Valor predeterminado = 500.
MinDurationPRE	DINT	Duración mínima preseleccionada (en milisegundos) para que la condición de alarma se mantenga como verdadera antes de que la alarma se marque como InAlarm y se envíe la notificación de alarma a los clientes. El controlador recopila los datos de alarma tan pronto como se detecta la condición de alarma para que no se pierdan datos mientras se espera hasta alcanzar la duración mínima. Valor válido = de 0 al 2147483647. Valor predeterminado = 0.
ShelveDuration	DINT	Longitud de tiempo en minutos para archivar una alarma. El archivado de una alarma pospone el procesamiento de alarma. Es como suprimir una alarma, excepto que ese archivado tiene un tiempo limitado. Si se confirma una alarma mientras se está archivando, permanece confirmada incluso si vuelve a estar activa de nuevo. Se convierte en no confirmada cuando la duración del archivado finaliza (siempre que la alarma siga activa en ese momento). El tiempo mínimo es un minuto. El tiempo máximo lo define MaxShelveDuration.
MaxShelveDuration	DINT	Duración máxima del tiempo en minutos en la que se puede archivar una alarma. Para obtener más información sobre cómo archivar una alarma, consulte la descripción del parámetro ShelveDuration.

### Conectar un botón a la etiqueta OperShelve

Para evitar que la alarma se vuelva a archivar sin que lo deseé, la instrucción de alarma solo procesa la etiqueta OperShelve si hace una transición desde falso a verdadero entre un escaneado del programa y el siguiente. Si un operador presiona un botón de pulsador para archivar la alarma mientras es verdadera la etiqueta ProgUnshelve, la alarma no se archiva porque la etiqueta ProgUnshelve tiene prioridad. Sin embargo, dado que los escaneados de programa se completan en milisegundos, el operador puede seguir pulsando el botón de pulsador por lo que la etiqueta OperShelve sigue establecida sobre varios escaneados de programa incluso aunque la etiqueta ProgUnshelve se haya borrado. Esto significa que la alarma no está archivada.

Para archivar la alarma, el operador puede soltar y volver a presionar el botón de pulsador

## Parámetros de salida

Estos parámetros de salida son comunes para el bloque de funciones.

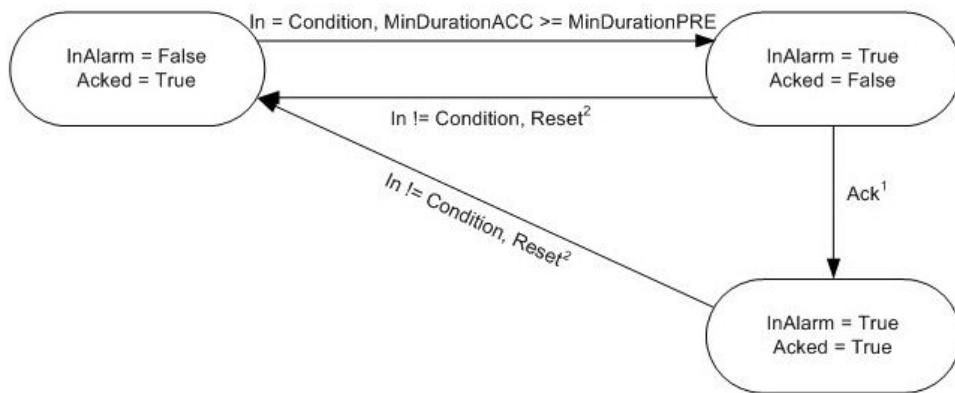
Parámetro de salida	Tipo de datos	Descripción
EnableOut	BOOL	Habilita la salida.
EnableOut	BOOL	Habilita la salida
InAlarm	BOOL	Estado activo de alarma. Establecido en verdadero cuando la alarma está activa. Borrado y establecido en falso cuando la alarma no está activa (estado normal).
Acked	BOOL	Estado de confirmación de la alarma. Establecido en verdadero cuando la alarma está confirmada. Se borra y se establece en falso cuando la alarma no está confirmada. Acked está siempre establecida en verdadero cuando AckRequired se borra y establece en falso.
InAlarmUnack	BOOL	Combinación de estado activo de alarma y estado de confirmación. Establecido en verdadero cuando la alarma está activa (InAlarm está establecido es verdadero) y sin confirmar (Acked se borra y establece en falso). Se borra y establece en falso cuando la alarma no está activa, confirmada o ambas.
Suppressed	BOOL	Estado de alarma suprimida. Establecido en verdadero cuando se haya suprimido la alarma. Se borra y establece en falso cuando no se ha suprimido la alarma.
Shelved	BOOL	Estado de archivado de la alarma. Establecido en verdadero cuando se haya archivado la alarma. Se borra y se establece en falso cuando se haya desarchivado la alarma. El archivado de una alarma pospone el procesamiento de alarma. Es como suprimir una alarma, excepto que ese archivado tiene un tiempo limitado. Si se confirma una alarma mientras se está archivando, permanece confirmada incluso si vuelve a estar activa de nuevo. Se convierte en no confirmada cuando finaliza la duración del archivado.
Disabled	BOOL	Estado de alarma deshabilitada. Establecido en verdadero cuando la alarma no se ha habilitado. Se borra y se establece en falso cuando se haya habilitado la alarma.
AlarmCount	DINT	Número de veces que se ha activado la alarma (InAlarm está establecido). Si se alcanza el valor máximo, el contador deja el valor en el valor máximo de conteo.
InAlarmTime	LINT	Marca de tiempo de detección de la alarma.
AckTime	LINT	Marca de tiempo de la confirmación de alarma. Si la alarma no requiere confirmación, esta marca de tiempo es igual a la hora de la alarma.
RetToNormalTime	LINT	Marca de tiempo de alarma que vuelve a un estado normal.
AlarmCountResetTime	LINT	Marca de tiempo que indica cuándo se ha restablecido el conteo de alarmas.

Parámetro de salida	Tipo de datos	Descripción
ShelveTime	LINT	Marca de tiempo que indica cuándo una condición de alarma se ha archivado por última vez. Este valor es establecido por el controlador cuando una alarma se archiva. Todas las alarmas se pueden archivar y desarchivar varias veces. Cada vez que la alarma se archiva, la marca de tiempo se establece en la hora actual. Para obtener más información sobre cómo archivar una alarma, consulte la descripción del parámetro Shelved.
UnshelveTime	LINT	Marca de tiempo que indica cuándo se va a desarchivar la alarma. Este valor se establece cada vez que la alarma se archiva (incluso si la alarma ya se ha archivado). La marca de tiempo viene determinada al añadir ShelveDuration a la hora actual. Si la alarma se desarchiva de forma programática o por un operador, entonces el valor se establece en la hora actual. Para obtener más información sobre cómo archivar una alarma, consulte la descripción del parámetro Shelved.
Status	DINT	Indicadores de estado combinado: Status.0 = InstructFault Status.1= InFaulted Status.2 = SeverityInv
InstructFault (Status.0)	BOOL	Existen condiciones de error de instrucción No se trata de un error mayor o menor del controlador. Se comprueba los bits de estado restantes para determinar lo que ha ocurrido.
InFaulted (Status.1)	BOOL	El programa de usuario ha establecido InFault para indicar la existencia de datos de entrada de mala calidad. La instrucción sigue evaluando In para las condiciones de alarma.
SeverityInv (Status.2)	BOOL	Configuración de gravedad de alarma. Si Gravedad < 1, la instrucción utiliza Gravedad = 1. Si la gravedad > 1000, la instrucción utiliza Severity = 1000.

### Diagramas de estado de alarmas digitales

**Acknowledgement Required, Latched**

**AckRequired = True, Latched = True**

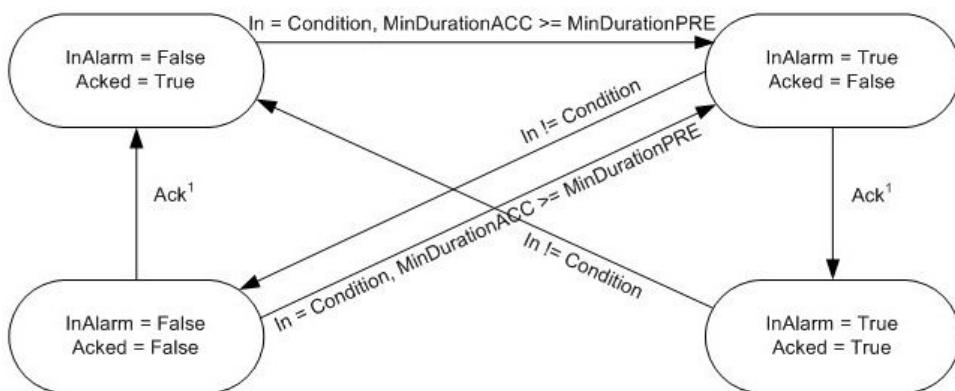


<sup>1</sup> Alarm can be acked by several different ways: ProgAck, OperAck, clients (RSLogix 5000, RSview)

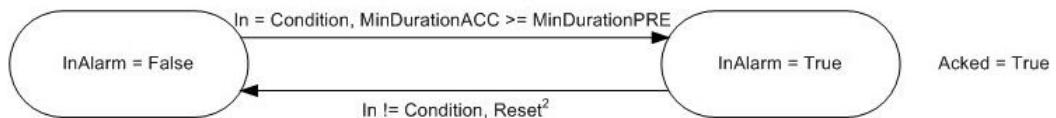
<sup>2</sup> Alarm can be reset by several different ways: ProgReset, OperReset, clients (RSLogix 5000, RSview)

**Acknowledgement Required, Not Latched**

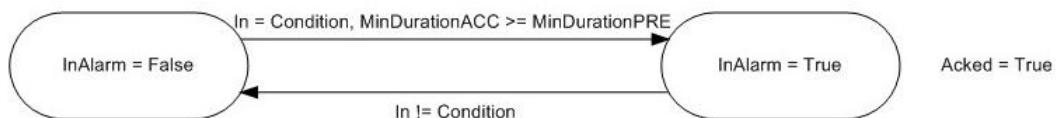
**AckRequired = True, Latched = False**



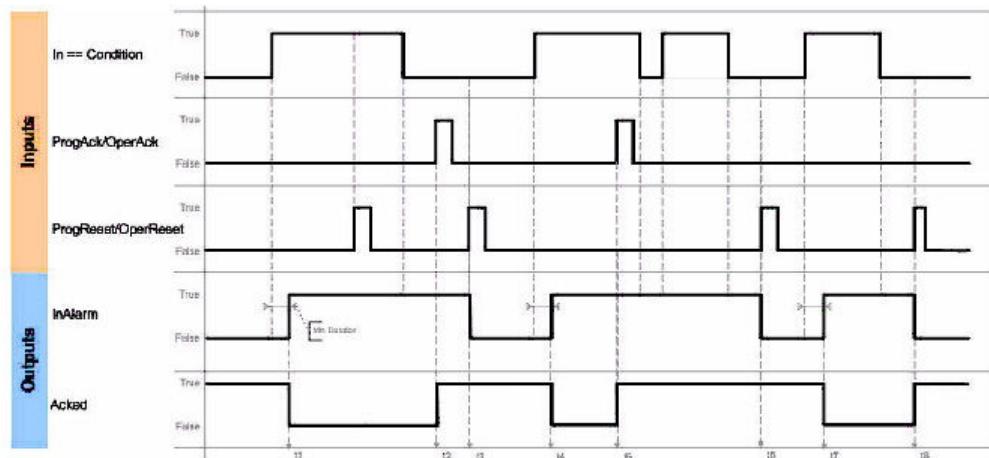
<sup>1</sup> Alarm can be acked by several different ways: ProgAck, OperAck, clients (RSLogix 5000, RSview)

**Acknowledgement Not Required, Latched****AckRequired = False, Latched = True**

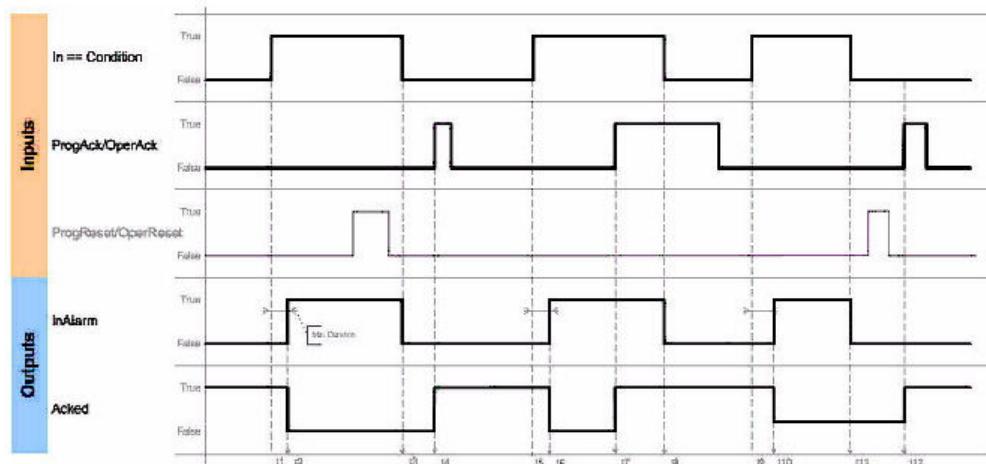
<sup>2</sup> Alarm can be reset by several different ways: ProgReset, OperReset, clients (RSLogix 5000, RSview)

**Acknowledgement Not Required, Not Latched****AckRequired = False, Latched = False****Diagramas de temporización de alarma digital**

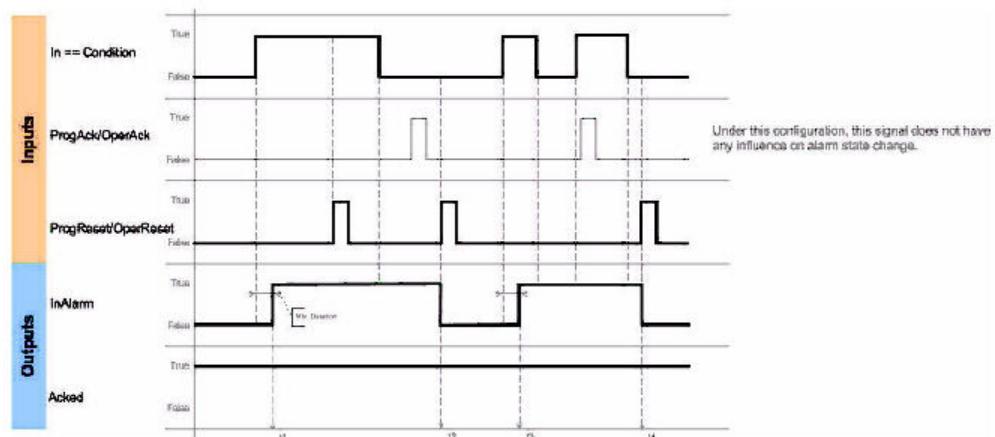
Confirmación de alarma ALMD requerida y enclavada.



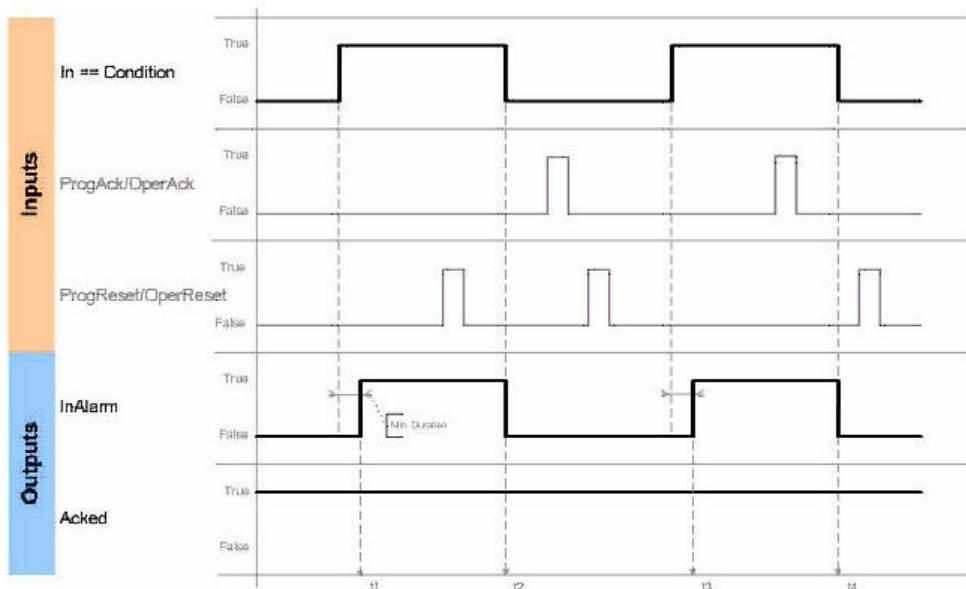
### Confirmación de alarma ALMD requerida y no enclavada.



### Confirmación de alarma ALMD no requerida y enclavada.



### Confirmación de alarma ALMD no requerida y no enclavada.



### Afectar a las marcas de estado matemático

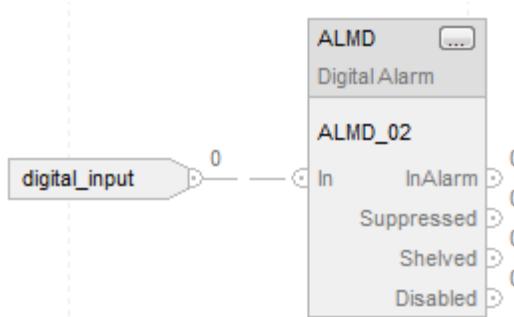
No

### Condiciones de fallo

No es específico para esta instrucción. Consulte los *Atributos comunes* para fallos relacionados con el operando.

### Ejemplo

### Bloque de funciones



### Consulte también

[Alarma digital \(ALMD\) - Lógica de escalera](#) en la página 120

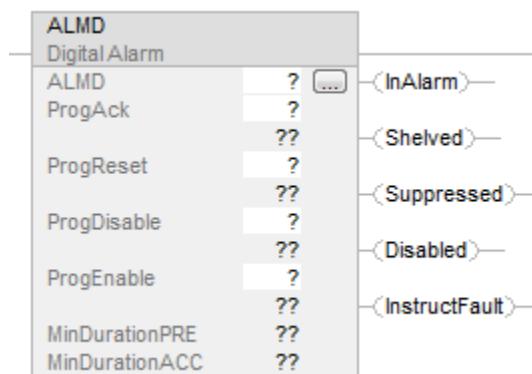
[Alarma digital \(ALMD\) - Texto estructurado](#) en la página 131

[Atributos comunes](#) en la página 923

## Alarma digital (ALMD) - Diagrama de escalera

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción ALMD proporciona la alarma para cualquier valor booleano discreto.



### Operандos

Estos operandos se encuentran en la instrucción.

Operando	Tipo	Formato	Descripción
ALMD tag	ALARM_DIGITAL	Estructura	Estructura de ALMD
ProgAck	BOOL	Etiqueta Inmediato	En la transición de falso a verdadero, confirma la alarma (si requiere confirmación).
ProgReset	BOOL	Etiqueta Inmediato	En la transición de falso a verdadero, restablece la alarma (si requiere restablecimiento).
ProgDisable	BOOL	Etiqueta Inmediato	Cuando sea Verdadero, desactiva la alarma (no anula los comandos de habilitación).
ProgEnable	BOOL	Etiqueta Inmediato	Cuando sea Verdadero, activa la alarma (tiene prioridad sobre los comandos de deshabilitación).
MinDurationPRE	DINT	Inmediato	Especifica cuánto tiempo debe cumplirse la condición de alarma antes de informar (milisegundos).
MinDurationACC	DINT	Inmediato	Indica el valor de acumulador actuar para el temporizador MinDuration de la alarma.

## Ejecución

Condición/estado	Acción realizada
Pre-escaneado	EnableOut se borra a falsa La salida InAlarm se borra y se establece en falso La salida Shelves se borra y se establece en falso La salida Acked se establece en verdadero Se han confirmado todas las condiciones de alarma. Se han borrado todas las peticiones del operador. Se han borrado todas las marcas de tiempo.
Reglón está en falso	Reglón se borra y se establece en falso. El parámetro In se borra y se establece en falso. La instrucción se ejecuta.
Reglón está en verdadero	Reglón se establece en verdadero. El parámetro In se establece en verdadero. La instrucción se ejecuta.
Post-escaneado	El bit de reglón se borra y se establece en falso.

## Parámetros de entrada

Parámetro de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Corresponde al estado de reglón. No afecta al procesamiento.
In	BOOL	Entrada de la señal digital a la instrucción. El valor predeterminado es falso.
InFault	BOOL	Indicador de estado incorrecto para la entrada. La aplicación de usuario puede establecer InFault para indicar que hay un error en la señal de entrada. Cuando se establece, la instrucción establece InFaulted (Status.1). Cuando se borra y establece en falso, la instrucción desactiva InFaulted (Status.1). En cualquier caso, la instrucción sigue evaluando las condiciones de alarma de In. El valor predeterminado está en falso (buen estado).
Condition	BOOL	Especifica cómo se activa la alarma. Cuando Condition está establecida en verdadera, la condición de alarma se activa si In está establecido en verdadero. Cuando Condition se borra a falso, la condición de alarma se activa si In se borra a falso. El valor predeterminado es verdadero.
AckRequired	BOOL	Especifica si es necesaria la confirmación de la alarma. Cuando se establece en verdadero, se necesita confirmación. Cuando se borra y establece en falso, no se necesita confirmación ya que Acked está establecida siempre en verdadero. El valor predeterminado es verdadero.
Latched	BOOL	Especifica si la alarma está enclavada. Las alarmas enclavadas se mantienen como InAlarm si la condición de alarma pasa a falso, hasta que se recibe un comando de restablecimiento. Cuando está establecida en verdadero, la alarma está enclavada. Cuando se borra y establece en falso, la alarma está desenclavada. El valor predeterminado es falso. Sólo es posible restablecer una alarma enclavada cuando la condición de alarma es falsa.

Parámetro de entrada	Tipo de datos	Descripción
ProgAck	BOOL	El programa del usuario lo establece en verdadero para confirmar la alarma. Solo toma efecto si no se confirma la alarma. Requiere una transición de falso a verdadero. El valor predeterminado es falso.
OperAck	BOOL	La interfaz de operador lo establece en verdadero para confirmar la alarma. Solo toma efecto si no se confirma la alarma. La instrucción desactiva este parámetro. El valor predeterminado es falso.
ProgReset	BOOL	El programa del usuario lo establece en verdadero para restablecer la alarma enclavada. Solo toma efecto si la alarma enclavada es InAlarm y la condición de alarma es falso. Requiere una transición de falso a verdadero. El valor predeterminado es falso.
OperReset	BOOL	La interfaz de operador lo establece en verdadero para restablecer la alarma enclavada. Solo toma efecto si la alarma enclavada es InAlarm y la condición de alarma es falso. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso.
ProgSuppress	BOOL	El programa del usuario lo establece en verdadero para suprimir la alarma. El valor predeterminado es falso.
OperSuppress	BOOL	La interfaz de operador lo establece en verdadero para suprimir la alarma. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso.
ProgUnsuppress	BOOL	El programa del usuario lo establece en verdadero para anular la supresión de la alarma. Tiene prioridad sobre los comandos de supresión. El valor predeterminado es falso.
OperUnsuppress	BOOL	La interfaz de operador lo establece en verdadero para anular la supresión de la alarma. Tiene prioridad sobre los comandos de supresión. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso.
OperShelve	BOOL	La interfaz de operador lo establece en verdadero para archivar o rarchivar la alarma. Requiere una transición desde falso en un escaneado del programa a verdadero en el siguiente escaneado del programa. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso. Los comandos de desarchivado tienen prioridad sobre los de archivado. El archivado de una alarma pospone el procesamiento de alarma. Es como suprimir una alarma, excepto que ese archivado tiene un tiempo limitado. Si se confirma una alarma mientras se está archivando, permanece confirmada incluso si vuelve a estar activa de nuevo. Se convierte en no confirmada cuando la duración del archivado finaliza, siempre que la alarma siga activa en ese momento.
ProgUnshelve	BOOL	El programa del usuario lo establece en verdadero para desarchivar la alarma. Tiene prioridad sobre los comandos de archivado. El valor predeterminado es falso. Para obtener más información sobre cómo archivar una alarma, consulte la descripción del parámetro OperShelve.

Parámetro de entrada	Tipo de datos	Descripción
OperUnshelve	BOOL	<p>La interfaz de operador lo establece en verdadero para desarchivar la alarma. La instrucción de alarma cambia este parámetro a falso. Tiene prioridad sobre los comandos de archivado.</p> <p>Está borrado de forma predeterminada.</p> <p>Para obtener más información sobre cómo archivar una alarma, consulte la descripción del parámetro OperShelve.</p>
ProgDisable	BOOL	<p>El programa del usuario lo establece en verdadero para deshabilitar la alarma.</p> <p>El valor predeterminado es falso.</p> <p>Copiado del operando de instrucción.</p>
OperDisable	BOOL	<p>La interfaz de operador lo establece en verdadero para deshabilitar la alarma. La instrucción de alarma cambia este parámetro a verdadero.</p> <p>El valor predeterminado es falso.</p>
ProgEnable	BOOL	<p>El programa del usuario lo establece en verdadero para habilitar la alarma. Tiene prioridad sobre un comando de inhabilitación.</p> <p>El valor predeterminado es falso.</p> <p>Diagrama de escalera:</p> <p>Copiado del operando de instrucción.</p> <p>Texto estructurado:</p> <p>Copiado del operando de instrucción.</p>
OperEnable	BOOL	<p>La interfaz de operador lo establece en verdadero para habilitar la alarma. Tiene prioridad sobre el comando de inhabilitación. La instrucción de alarma cambia este parámetro a falso.</p> <p>El valor predeterminado es falso.</p>
AlarmCountReset	BOOL	<p>La interfaz de operador lo establece en verdadero para restablecer el conteo de alarma a cero. La instrucción de alarma cambia este parámetro a falso.</p> <p>El valor predeterminado es falso.</p>
UseProgTime	BOOL	<p>Especifica si se usa el reloj del controlador o el valor ProgTime para los eventos de cambio de estado de alarma de marca de tiempo. Cuando está establecido en verdadero, el valor ProgTime proporciona la marca de tiempo. Cuando se borra y establece en falso, el reloj del controlador proporciona la marca de tiempo.</p> <p>El valor predeterminado es falso.</p>
ProgTime	LINT	<p>Si UseProgTime está establecido en verdadero, este valor se usa para proporcionar el valor de marca de tiempo para todos los eventos. De este modo, se permite que la aplicación aplique las marcas de tiempo obtenidas a partir del origen de la alarma, como un módulo de entrada de secuencia de eventos.</p>
Severity	DINT	<p>Gravedad de la alarma. Esto no afecta al procesamiento de las alarmas por el controlador pero puede usarse para clasificar y filtrar las funciones del suscriptor de alarma.</p> <p>Valor válido = del 1 al 1000 (1000 = la más grave; 1 = la menos grave).</p> <p>Valor predeterminado = 500.</p>

Parámetro de entrada	Tipo de datos	Descripción
MinDurationPRE	DINT	Duración mínima preseleccionada (en milisegundos) para que la condición de alarma se mantenga como verdadera antes de que la alarma se marque como InAlarm y se envíe la notificación de alarma a los clientes. El controlador recopila los datos de alarma tan pronto como se detecta la condición de alarma para que no se pierdan datos mientras se espera hasta alcanzar la duración mínima. Valor válido = 0...2147483647. Valor predeterminado = 0.
ShelveDuration	DINT	Longitud de tiempo en minutos para archivar una alarma. El archivado de una alarma pospone el procesamiento de alarma. Es como suprimir una alarma, excepto que ese archivado tiene un tiempo limitado. Si se confirma una alarma mientras se está archivando, permanece confirmada incluso si vuelve a estar activa de nuevo. Se convierte en no confirmada cuando la duración del archivado finaliza (siempre que la alarma siga activa en ese momento). El tiempo mínimo es un minuto. El tiempo máximo lo define MaxShelveDuration.
MaxShelveDuration	DINT	Duración máxima del tiempo en minutos en la que se puede archivar una alarma. Para obtener más información sobre cómo archivar una alarma, consulte la descripción del parámetro ShelveDuration.

### Conectar un botón a la etiqueta OperShelve

Para evitar que la alarma se vuelva a archivar sin que lo desee, la instrucción de alarma solo procesa la etiqueta OperShelve si hace una transición desde falso a verdadero entre un escaneado del programa y el siguiente. Si un operador presiona un botón de pulsador para archivar la alarma mientras es verdadera la etiqueta ProgUnshelve, la alarma no se archiva porque la etiqueta ProgUnshelve tiene prioridad. Sin embargo, dado que los escaneados de programa se completan en milisegundos, el operador puede seguir pulsando el botón de pulsador por lo que la etiqueta OperShelve sigue establecido sobre varios escaneados de programa incluso aunque la etiqueta ProgUnshelve se haya borrado. Esto significa que la alarma no está archivada.

Para archivar la alarma, el operador puede soltar y volver a presionar el botón.

### Parámetros de salida

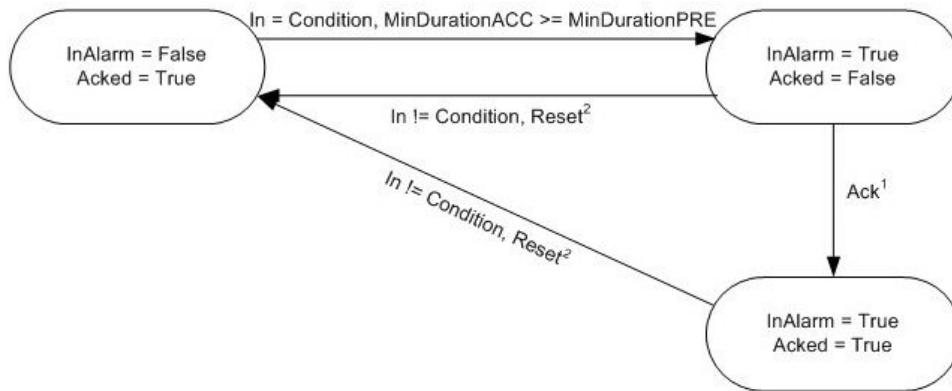
Estos parámetros de salida son comunes para la lógica de escalera.

Parámetro de salida	Tipo de datos	Descripción
EnableOut	BOOL	Habilita la salida.
InAlarm	BOOL	Estado activo de alarma. Establecido en verdadero cuando la alarma está activa. Borrado y establecido en falso cuando la alarma no está activa (estado normal).

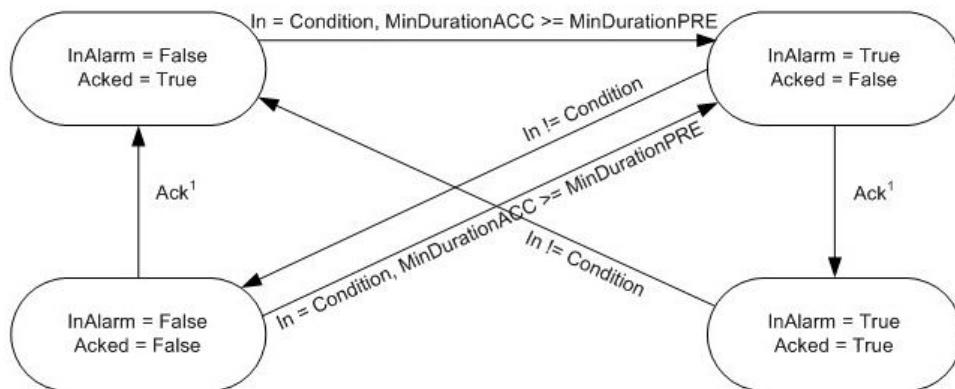
Acked	BOOL	Estado de confirmación de la alarma. Establecido en verdadero cuando la alarma está confirmada. Se borra y se establece en falso cuando la alarma no está confirmada. Acked está siempre establecida en verdadero cuando AckRequired se borra y establece en falso.
InAlarmUnack	BOOL	Combinación de estado activo de alarma y estado de confirmación. Establecido en verdadero cuando la alarma está activa (InAlarm está establecido es verdadero) y sin confirmar (Acked se borra y establece en falso). Se borra y establece en falso cuando la alarma no está activa, confirmada o ambas.
Suppressed	BOOL	Estado de alarma suprimida. Establecido en verdadero cuando se haya suprimido la alarma. Se borra y establece en falso cuando no se ha suprimido la alarma.
Shelved	BOOL	Estado de archivado de la alarma. Establecido en verdadero cuando se haya archivado la alarma. Se borra y se establece en falso cuando se haya desarchivado la alarma. El archivado de una alarma pospone el procesamiento de alarma. Es como suprimir una alarma, excepto que ese archivado tiene un tiempo limitado. Si se confirma una alarma mientras se está archivando, permanece confirmada incluso si vuelve a estar activa de nuevo. Se convierte en no confirmada cuando finaliza la duración del archivado.
Disabled	BOOL	Estado de alarma deshabilitada. Establecido en verdadero cuando la alarma no se ha habilitado. Se borra y se establece en falso cuando se haya habilitado la alarma.
Commissioned	BOOL	Estado activado de la alarma. Establecido en verdadero cuando la alarma se ha activado. Se borra a falso cuando se ha desactivado la alarma. En estos momentos siempre se establece en verdadero.
MinDurationACC	DINT	Tiempo transcurrido desde que se ha detectado la alarma. Cuando este valor alcanza el valor de MinDurationPRE, la alarma se activa (InAlarm es verdadero) y se envía una notificación a los clientes. Si la alarma se vuelve inactiva antes de que MinDurationAcc llegue a MinDurationPRE, entonces el temporizador de MinDuration se detiene pero MinDurationAcc no se restablece por lo que mantiene su último valor. MinDurationAcc se restablece cuando se detecta de forma reciente la alarma y, por lo tanto, el temporizador de MinDuration se inicia de nuevo. El temporizador de MinDuration se puede detener antes de que caduque si EnableIn se ha borrado a falso (se aplica solo al bloque de funciones) o la alarma está deshabilitada. En estos casos MinDurationAcc mantiene su último valor y no se restablece hasta que el temporizador de MinDuration se inicia de nuevo.
AlarmCount	DINT	Número de veces que se ha activado la alarma (InAlarm está establecido). Si se alcanza el valor máximo, el contador deja el valor en el valor máximo de conteo.
InAlarmTime	LINT	Marca de tiempo de detección de la alarma.
AckTime	LINT	Marca de tiempo de la confirmación de alarma. Si la alarma no requiere confirmación, esta marca de tiempo es igual a la hora de la alarma.
RetToNormalTime	LINT	Marca de tiempo de alarma que vuelve a un estado normal.
AlarmCountResetTime	LINT	Marca de tiempo que indica cuándo se ha restablecido el conteo de alarmas.

ShelveTime	LINT	Marca de tiempo que indica cuándo una condición de alarma se ha archivado por última vez. Este valor es establecido por el controlador cuando una alarma se archiva. Todas las alarmas se pueden archivar y desarchivar varias veces. Cada vez que la alarma se archiva, la marca de tiempo se establece en la hora actual. Para obtener más información sobre cómo archivar una alarma, consulte la descripción del parámetro Shelves.
UnshelveTime	LIN	Marca de tiempo que indica cuándo se va a desarchivar la alarma. Este valor se establece cada vez que la alarma se archiva (incluso si la alarma ya se ha archivado). La marca de tiempo viene determinada al añadir ShelveDuration a la hora actual. Si la alarma se desarchiva de forma programática o por un operador, entonces el valor se establece en la hora actual. Para obtener más información sobre cómo archivar una alarma, consulte la descripción del parámetro Shelves.
Status	DINT	Indicadores de estado combinado: Status.0 = InstructFault Status.1= InFaulted Status.2 = SeverityInv
InstructFault (Status.0)	BOOL	Existen condiciones de error de instrucción No se trata de un error mayor o menor del controlador. Se comprueba los bits de estado restantes para determinar lo que ha ocurrido.
InFaulted (Status.1)	BOOL	El programa de usuario ha establecido InFault para indicar la existencia de datos de entrada de mala calidad. La instrucción sigue evaluando In para las condiciones de alarma.
SeverityInv (Status.2)	BOOL	Configuración de gravedad de alarma. Si Gravedad < 1, la instrucción utiliza Gravedad = 1. Si la gravedad > 1000, la instrucción utiliza Severity = 1000.

### Diagramas de estado de alarmas digitales

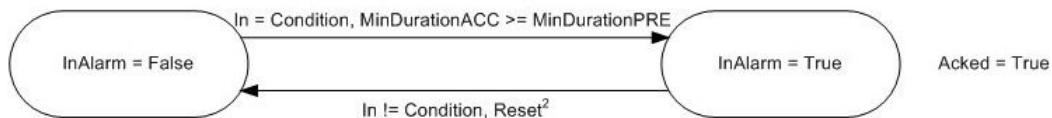
**Acknowledgement Required, Latched**
**AckRequired = True, Latched = True**

<sup>1</sup> Alarm can be acked by several different ways: ProgAck, OperAck, clients (RSLogix 5000, RSview)

<sup>2</sup> Alarm can be reset by several different ways: ProgReset, OperReset, clients (RSLogix 5000, RSview)

**Acknowledgement Required, Not Latched**
**AckRequired = True, Latched = False**

<sup>1</sup> Alarm can be acked by several different ways: ProgAck, OperAck, clients (RSLogix 5000, RSview)

**Acknowledgement Not Required, Latched**

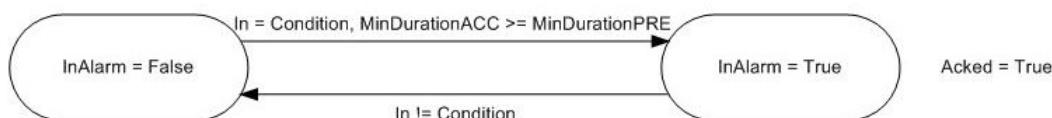
**AckRequired = False, Latched = True**



<sup>2</sup> Alarm can be reset by several different ways: ProgReset, OperReset, clients (RSLogix 5000, RSview)

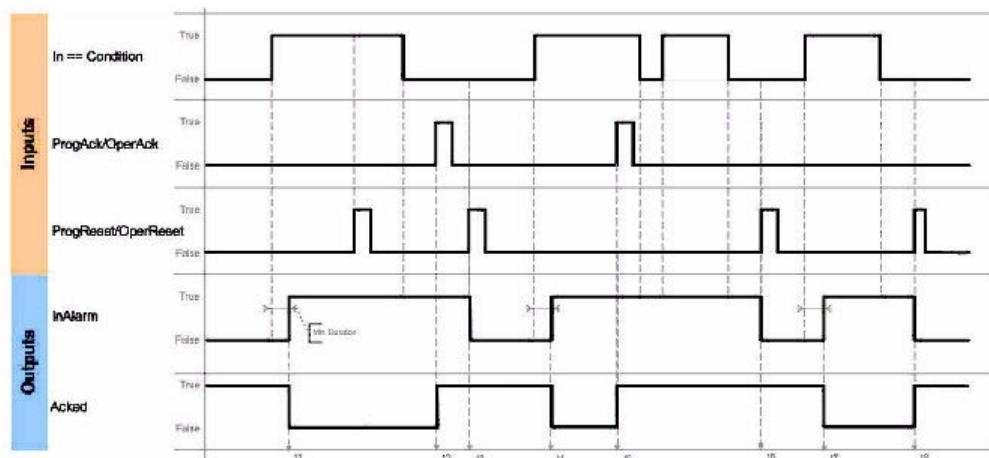
**Acknowledgement Not Required, Not Latched**

**AckRequired = False, Latched = False**

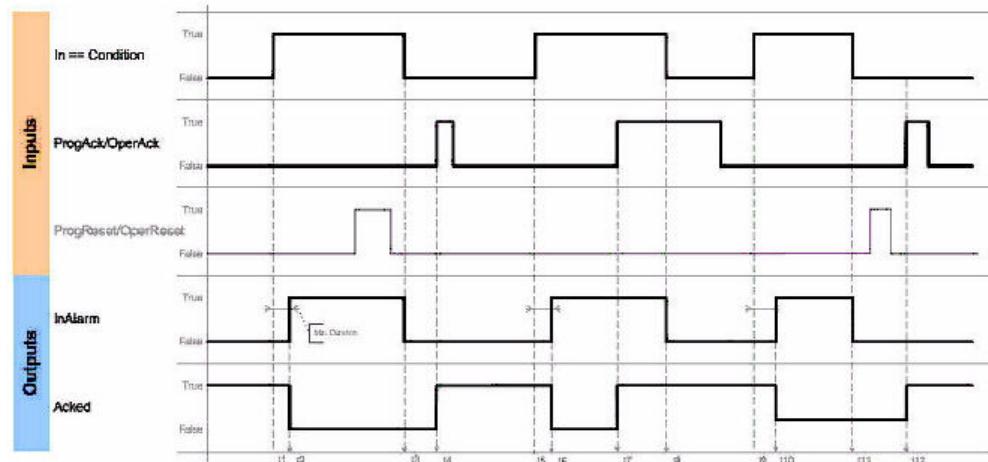


### Diagramas de temporización de alarma digital

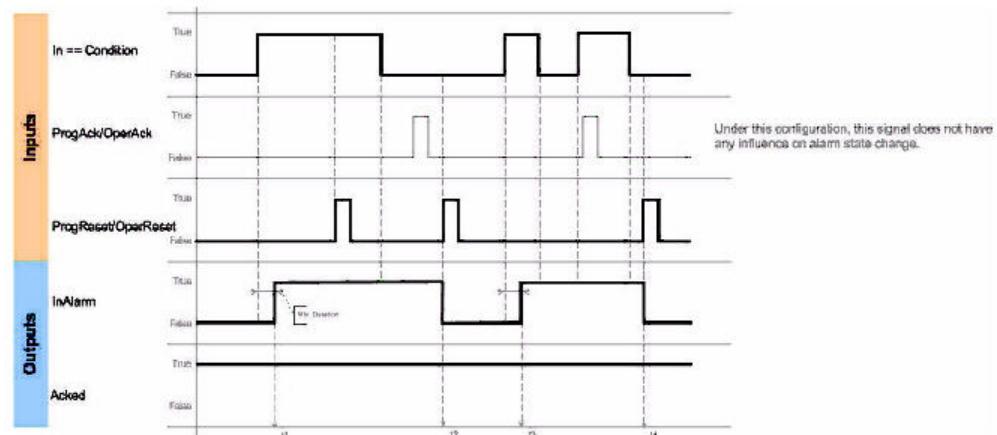
Confirmación de alarma ALMD requerida y enclavada.



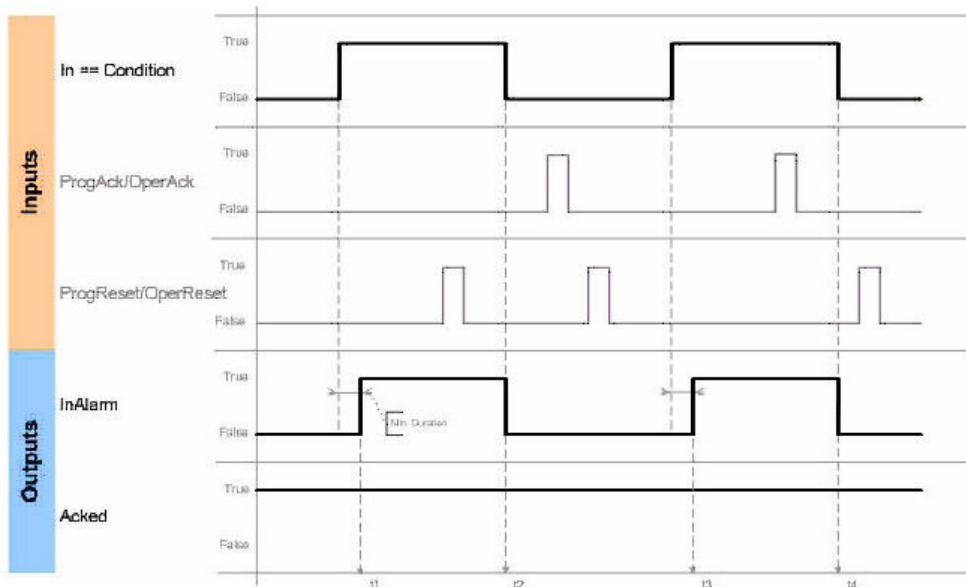
### Confirmación de alarma ALMD requerida y no enclavada.



### Confirmación de alarma ALMD no requerida y enclavada.



### Confirmación de alarma ALMD no requerida y no enclavada.



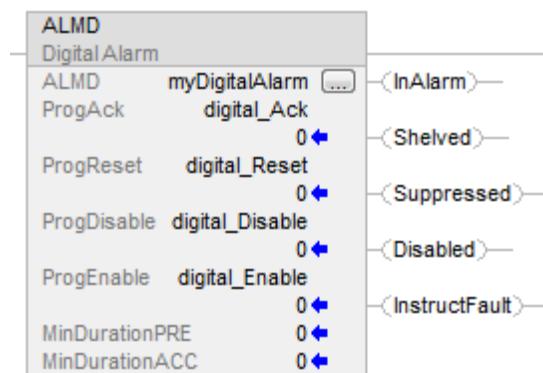
### Afectar a las marcas de estado matemático

No

### Condiciones de fallo

No es específico para esta instrucción. Consulte los *Atributos comunes* para fallos relacionados con el operando.

### Ejemplo



### Consulte también

[Alarma digital \(ALMD\): Bloque de funciones](#) en la página 108

[Alarma digital \(ALMD\) - Texto estructurado](#) en la página 131

[Atributos comunes](#) en la página 923

## Alarma digital (ALMD) - Texto estructurado

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción ALMD proporciona la alarma para cualquier valor booleano discreto.

### Texto estructurado

ALMD (ALMD,In,ProgAck,ProgReset,ProgDisable,ProgEnable)

### Operandos

#### Texto estructurado

Estos operandos se encuentran en la presentación de la instrucción.

Operando	Tipo	Formato	Descripción
ALMD	ALARM_DIGITAL	Estructura	Estructura de ALMD
In	BOOL	Etiqueta Inmediato	Se va a supervisar la condición de alarma. Cuando se ejecuta la instrucción, el valor se copia en .In.
ProgAck	BOOL	Etiqueta Inmediato	Cuando se ejecuta la instrucción, el valor se copia en ProgAck. En la transición de desactivada a activada, confirma la alarma (si requiere confirmación).
ProgReset	BOOL	Etiqueta Inmediato	Cuando se ejecuta la instrucción, el valor se copia en .ProgReset. En la transición de Falso a Verdadero, restablece la alarma (si requiere restablecimiento).
ProgDisable	BOOL	Etiqueta Inmediato	Cuando se ejecuta la instrucción, el valor se copia en .ProgDisable. Cuando sea Verdadero, desactiva la alarma (no anula los comandos de habilitación).
ProgEnable	BOOL	Etiqueta Inmediato	Cuando se ejecuta la instrucción, el valor se copia en .ProgEnable. Cuando sea Verdadero, activa la alarma (tiene prioridad sobre los comandos de deshabilitación).

Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

### Definición de la estructura de ALARM\_DIGITAL

Consulte *Definición de la estructura de las etiquetas ALARM\_DIGITAL* para ver las descripciones de los elementos de la etiqueta y la ejecución de la alarma.

## Ejecución

En Texto estructurado, EnableIn siempre es verdadero durante un escaneado normal. Por tanto, si la instrucción se encuentra en la ruta de control activada por la lógica, se ejecutará.

Condición/estado	Acción realizada
Pre-escaneado	La condición de entrada de peldaño se borra a falsa La salida InAlarm se borra y se establece en falso La salida Shelved se borra y se establece en falso La salida Acked se establece en verdadero Se han confirmado todas las condiciones de alarma. Se han borrado todas las peticiones del operador. Se han borrado todas las marcas de tiempo.
La condición de entrada de reglón es falsa	La condición de entrada de reglón se borra a falsa. Diagrama de escalera: El parámetro In se borra y se establece en falso. La instrucción se ejecuta.
La condición de entrada de reglón es verdadera	La condición de entrada de reglón se establece en verdadera. Diagrama de escalera: El parámetro In se establece en verdadero. La instrucción se ejecuta.
Post-escaneado	La condición de entrada de peldaño se borra a falsa.

## Parámetros de entrada

Parámetro de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Diagrama de escalera: Corresponde al estado de reglón. No afecta al procesamiento. Bloque de funciones: Si se borra y establece en falso, la instrucción no se ejecuta y las salidas no se actualizan. Si está establecida, la instrucción se ejecuta. El valor predeterminado es verdadero. Texto estructurado: Ningún efecto. La instrucción siempre se ejecuta.
In	BOOL	Entrada de la señal digital a la instrucción. El valor predeterminado es falso. Diagrama de escalera: Sigue la condición de peldaño. Establecido en verdadero si la condición de peldaño es verdadera. Se borra y se establece en falso si la condición de peldaño es falso. Texto estructurado: Copiado del operando de instrucción.
InFault	BOOL	Indicador de estado incorrecto para la entrada. La aplicación de usuario puede establecer InFault para indicar que hay un error en la señal de entrada. Cuando se establece, la instrucción establece InFaulted (Status.1). Cuando se borra y establece en falso, la instrucción desactiva InFaulted (Status.1). En cualquier caso, la instrucción sigue evaluando las condiciones de alarma de In. El valor predeterminado está en falso (buen estado).

Parámetro de entrada	Tipo de datos	Descripción
Condition	BOOL	Especifica cómo se activa la alarma. Cuando Condition está establecida en verdadera, la condición de alarma se activa si In está establecido en verdadero. Cuando Condition se borra a falso, la condición de alarma se activa si In se borra a falso. El valor predeterminado es verdadero.
AckRequired	BOOL	Especifica si es necesaria la confirmación de la alarma. Cuando se establece en verdadero, se necesita confirmación. Cuando se borra y establece en falso, no se necesita confirmación ya que Acked está establecida siempre en verdadero. El valor predeterminado es verdadero.
Latched	BOOL	Especifica si la alarma está enclavada. Las alarmas enclavadas se mantienen como InAlarm si la condición de alarma pasa a falso, hasta que se recibe un comando de restablecimiento. Cuando está establecida en verdadero, la alarma está enclavada. Cuando se borra y establece en falso, la alarma está desenclavada. El valor predeterminado es falso. Sólo es posible restablecer una alarma enclavada cuando la condición de alarma es falsa.
ProgAck	BOOL	El programa del usuario lo establece en verdadero para confirmar la alarma. Solo toma efecto si no se confirma la alarma. Requiere una transición de falso a verdadero. El valor predeterminado es falso. Diagrama de escalera: Copiado del operando de instrucción. Texto estructurado: Copiado del operando de instrucción.
OperAck	BOOL	La interfaz de operador lo establece en verdadero para confirmar la alarma. Solo toma efecto si no se confirma la alarma. La instrucción desactiva este parámetro. El valor predeterminado es falso.
ProgReset	BOOL	El programa del usuario lo establece en verdadero para restablecer la alarma enclavada. Solo toma efecto si la alarma enclavada es InAlarm y la condición de alarma es falso. Requiere una transición de falso a verdadero. El valor predeterminado es falso. Diagrama de escalera: Copiado del operando de instrucción. Texto estructurado: Copiado del operando de instrucción.
OperReset	BOOL	La interfaz de operador lo establece en verdadero para restablecer la alarma enclavada. Solo toma efecto si la alarma enclavada es InAlarm y la condición de alarma es falso. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso.
ProgSuppress	BOOL	El programa del usuario lo establece en verdadero para suprimir la alarma. El valor predeterminado es falso.
OperSuppress	BOOL	La interfaz de operador lo establece en verdadero para suprimir la alarma. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso.

Parámetro de entrada	Tipo de datos	Descripción
ProgUnsuppress	BOOL	<p>El programa del usuario lo establece en verdadero para anular la supresión de la alarma. Tiene prioridad sobre los comandos de supresión.</p> <p>El valor predeterminado es falso.</p>
OperUnsuppress	BOOL	<p>La interfaz de operador lo establece en verdadero para anular la supresión de la alarma. Tiene prioridad sobre los comandos de supresión. La instrucción de alarma cambia este parámetro a falso.</p> <p>El valor predeterminado es falso.</p>
OperShelve	BOOL	<p>La interfaz de operador lo establece en verdadero para archivar o rarchivar la alarma. Requiere una transición desde falso en un escaneado del programa a verdadero en el siguiente escaneado del programa. La instrucción de alarma cambia este parámetro a falso.</p> <p>El valor predeterminado es falso.</p> <p>Los comandos de desarchivado tienen prioridad sobre los de archivado.</p> <p>El archivado de una alarma pospone el procesamiento de alarma. Es como suprimir una alarma, excepto que ese archivado tiene un tiempo limitado. Si se confirma una alarma mientras se está archivando, permanece confirmada incluso si vuelve a estar activa de nuevo. Se convierte en no confirmada cuando la duración del archivado finaliza, siempre que la alarma siga activa en ese momento.</p>
ProgUnshelve	BOOL	<p>El programa del usuario lo establece en verdadero para desarchivar la alarma. Tiene prioridad sobre los comandos de archivado.</p> <p>El valor predeterminado es falso.</p> <p>Para obtener más información sobre cómo archivar una alarma, consulte la descripción del parámetro OperShelve.</p>
OperUnshelve	BOOL	<p>La interfaz de operador lo establece en verdadero para desarchivar la alarma. La instrucción de alarma cambia este parámetro a falso.</p> <p>Tiene prioridad sobre los comandos de archivado.</p> <p>Está borrado de forma predeterminada.</p> <p>Para obtener más información sobre cómo archivar una alarma, consulte la descripción del parámetro OperShelve.</p>
ProgDisable	BOOL	<p>El programa del usuario lo establece en verdadero para deshabilitar la alarma.</p> <p>El valor predeterminado es falso.</p> <p>Diagrama de escalera:</p> <p>Copiado del operando de instrucción.</p> <p>Texto estructurado:</p> <p>Copiado del operando de instrucción.</p>
OperDisable	BOOL	<p>La interfaz de operador lo establece en verdadero para deshabilitar la alarma. La instrucción de alarma cambia este parámetro a verdadero.</p> <p>El valor predeterminado es falso.</p>
ProgEnable	BOOL	<p>El programa del usuario lo establece en verdadero para habilitar la alarma. Tiene prioridad sobre un comando de inhabilitación.</p> <p>El valor predeterminado es falso.</p> <p>Diagrama de escalera:</p> <p>Copiado del operando de instrucción.</p> <p>Texto estructurado:</p> <p>Copiado del operando de instrucción.</p>

Parámetro de entrada	Tipo de datos	Descripción
OperEnable	BOOL	La interfaz de operador lo establece en verdadero para habilitar la alarma. Tiene prioridad sobre el comando de inhabilitación. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso.
AlarmCountReset	BOOL	La interfaz de operador lo establece en verdadero para restablecer el conteo de alarma a cero. La instrucción de alarma cambia este parámetro a falso. El valor predeterminado es falso.
UseProgTime	BOOL	Especifica si se usa el reloj del controlador o el valor ProgTime para los eventos de cambio de estado de alarma de marca de tiempo. Cuando está establecido en verdadero, el valor ProgTime proporciona la marca de tiempo. Cuando se borra y establece en falso, el reloj del controlador proporciona la marca de tiempo. El valor predeterminado es falso.
ProgTime	LINT	Si UseProgTime está establecido en verdadero, este valor se usa para proporcionar el valor de marca de tiempo para todos los eventos. De este modo, se permite que la aplicación aplique las marcas de tiempo obtenidas a partir del origen de la alarma, como un módulo de entrada de secuencia de eventos.
Severity	DINT	Gravedad de la alarma. Esto no afecta al procesamiento de las alarmas por el controlador pero puede usarse para clasificar y filtrar las funciones del suscriptor de alarma. Valor válido = del 1 al 1000 (1000 = la más grave; 1 = la menos grave). Valor predeterminado = 500.
MinDurationPRE	DINT	Duración mínima preseleccionada (en milisegundos) para que la condición de alarma se mantenga como verdadera antes de que la alarma se marque como InAlarm y se envíe la notificación de alarma a los clientes. El controlador recopila los datos de alarma tan pronto como se detecta la condición de alarma para que no se pierdan datos mientras se espera hasta alcanzar la duración mínima. Valor válido = 0...2147483647. Valor predeterminado = 0.
ShelveDuration	DINT	Longitud de tiempo en minutos para archivar una alarma. El archivado de una alarma pospone el procesamiento de alarma. Es como suprimir una alarma, excepto que ese archivado tiene un tiempo limitado. Si se confirma una alarma mientras se está archivando, permanece confirmada incluso si vuelve a estar activa de nuevo. Se convierte en no confirmada cuando la duración del archivado finaliza (siempre que la alarma siga activa en ese momento). El tiempo mínimo es un minuto. El tiempo máximo lo define MaxShelveDuration.
MaxShelveDuration	DINT	Duración máxima del tiempo en minutos en la que se puede archivar una alarma. Para obtener más información sobre cómo archivar una alarma, consulte la descripción del parámetro ShelveDuration.

### Conectar un botón a la etiqueta OperShelve

Para evitar que la alarma se vuelva a archivar sin que lo deseé, la instrucción de alarma solo procesa la etiqueta OperShelve si hace una transición desde falso a verdadero entre un escaneado del programa y el siguiente. Si un operador presiona

un botón de pulsador para archivar la alarma mientras es verdadera la etiqueta ProgUnshelve, la alarma no se archiva porque la etiqueta ProgUnshelve tiene prioridad. Sin embargo, dado que los escaneados de programa se completan en milisegundos, el operador puede seguir pulsando el botón de pulsador por lo que la etiqueta OperShelve sigue establecido sobre varios escaneados de programa incluso aunque la etiqueta ProgUnshelve se haya borrado. Esto significa que la alarma no está archivada.

Para archivar la alarma, el operador puede soltar y volver a presionar el botón de pulsador

### Parámetros de salida

Estos parámetros de salida son comunes para texto estructurado.

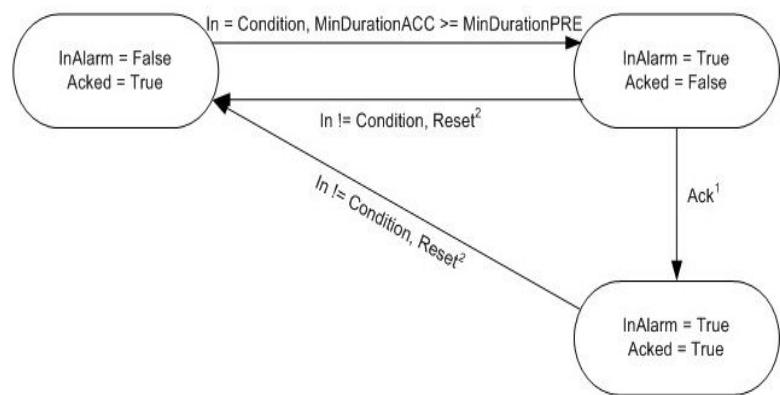
Parámetro de salida	Tipo de datos	Descripción
EnableOut	BOOL	Habilita la salida.
InAlarm	BOOL	Estado activo de alarma. Establecido en verdadero cuando la alarma está activa. Borrado y establecido en falso cuando la alarma no está activa (estado normal).
Acked	BOOL	Estado de confirmación de la alarma. Establecido en verdadero cuando la alarma está confirmada. Se borra y se establece en falso cuando la alarma no está confirmada. Acked está siempre establecida en verdadero cuando AckRequired se borra y establece en falso.
InAlarmUnack	BOOL	Combinación de estado activo de alarma y estado de confirmación. Establecido en verdadero cuando la alarma está activa (InAlarm está establecido en verdadero) y sin confirmar (Acked se borra y establece en falso). Se borra y establece en falso cuando la alarma no está activa, confirmada o ambas.
Suppressed	BOOL	Estado de alarma suprimida. Establecido en verdadero cuando se haya suprimido la alarma. Se borra y se establece en falso cuando no se ha suprimido la alarma.
Shelved	BOOL	Estado de archivado de la alarma. Establecido en verdadero cuando se haya archivado la alarma. Se borra y se establece en falso cuando se haya desarchivado la alarma. El archivado de una alarma pospone el procesamiento de alarma. Es como suprimir una alarma, excepto que ese archivado tiene un tiempo limitado. Si se confirma una alarma mientras se está archivando, permanece confirmada incluso si vuelve a estar activa de nuevo. Se convierte en no confirmada cuando finaliza la duración del archivado.
Disabled	BOOL	Estado de alarma deshabilitada. Establecido en verdadero cuando la alarma no se ha habilitado. Se borra y se establece en falso cuando se haya habilitado la alarma.
AlarmCount	DINT	Número de veces que se ha activado la alarma (InAlarm está establecido). Si se alcanza el valor máximo, el contador deja el valor en el valor máximo de conteo.
InAlarmTime	LINT	Marca de tiempo de detección de la alarma.

AckTime	LINT	Marca de tiempo de la confirmación de alarma. Si la alarma no requiere confirmación, esta marca de tiempo es igual a la hora de la alarma.
RetToNormalTime	LINT	Marca de tiempo de alarma que vuelve a un estado normal.
AlarmCountResetTime	LINT	Marca de tiempo que indica cuándo se ha restablecido el conteo de alarmas.
ShelveTime	LINT	Marca de tiempo que indica cuándo una condición de alarma se ha archivado por última vez. Este valor es establecido por el controlador cuando una alarma se archiva. Todas las alarmas se pueden archivar y desarchivar varias veces. Cada vez que la alarma se archiva, la marca de tiempo se establece en la hora actual. Para obtener más información sobre cómo archivar una alarma, consulte la descripción del parámetro Shelves.
UnshelveTime	LIN	Marca de tiempo que indica cuándo se va a desarchivar la alarma. Este valor se establece cada vez que la alarma se archiva (incluso si la alarma ya se ha archivado). La marca de tiempo viene determinada al añadir ShelveDuration a la hora actual. Si la alarma se desarchiva de forma programática o por un operador, entonces el valor se establece en la hora actual. Para obtener más información sobre cómo archivar una alarma, consulte la descripción del parámetro Shelves.
Status	DINT	Indicadores de estado combinado: Status.0 = InstructFault Status.1= InFaulted Status.2 = SeverityInv
InstructFault (Status.0)	BOOL	Existen condiciones de error de instrucción No se trata de un error mayor o menor del controlador. Se comprueba los bits de estado restantes para determinar lo que ha ocurrido.
InFaulted (Status.1)	BOOL	El programa de usuario ha establecido InFault para indicar la existencia de datos de entrada de mala calidad. La instrucción sigue evaluando In para las condiciones de alarma.
SeverityInv (Status.2)	BOOL	Configuración de gravedad de alarma. Si Gravedad < 1, la instrucción utiliza Gravedad = 1. Si la gravedad > 1000, la instrucción utiliza Severity = 1000.

### Diagramas de estado de alarmas digitales

**Acknowledgement Required, Latched**

**AckRequired = True, Latched = True**

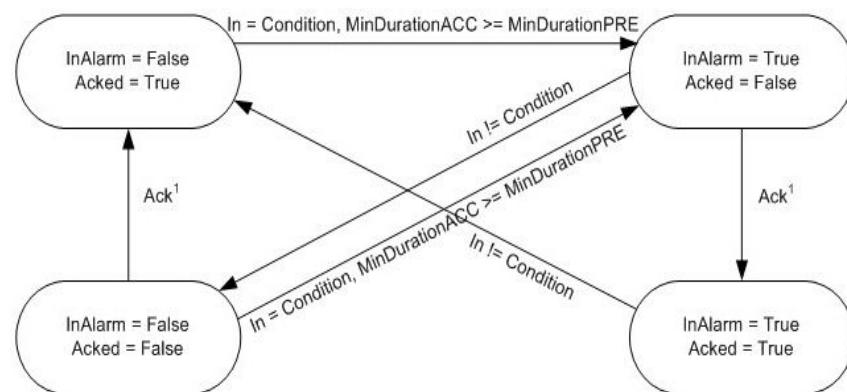


<sup>1</sup> Alarm can be acked by several different ways: ProgAck, OperAck, clients (RSLogix 5000, RSview)

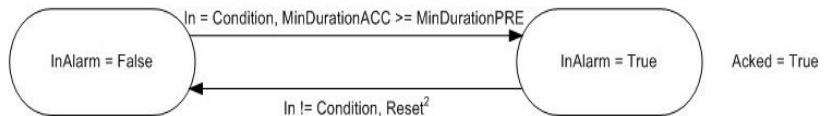
<sup>2</sup> Alarm can be reset by several different ways: ProgReset, OperReset, clients (RSLogix 5000, RSview)

**Acknowledgement Required, Not Latched**

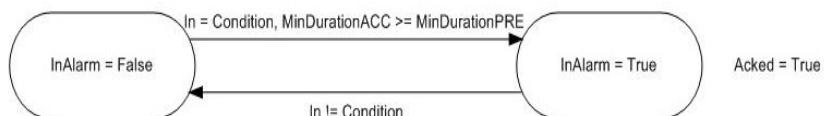
**AckRequired = True, Latched = False**



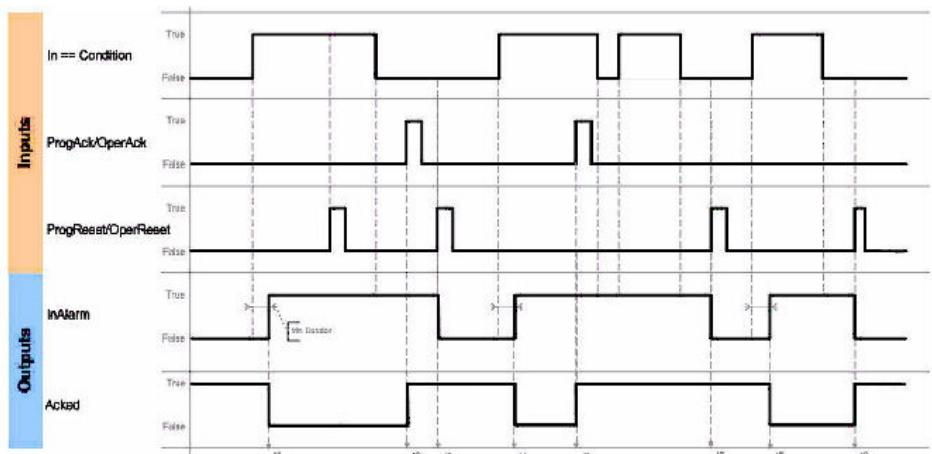
<sup>1</sup> Alarm can be acked by several different ways: ProgAck, OperAck, clients (RSLogix 5000, RSview)

**Acknowledgement Not Required, Latched****AckRequired = False, Latched = True**

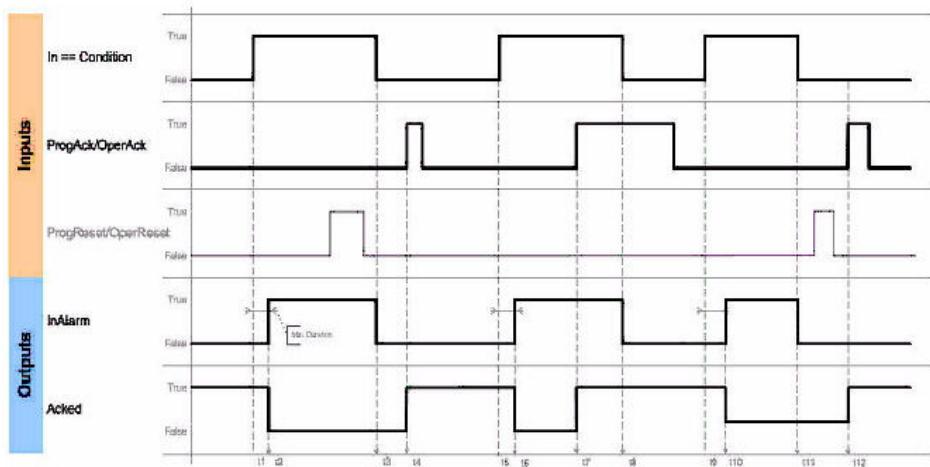
<sup>2</sup> Alarm can be reset by several different ways: ProgReset, OperReset, clients (RSLogix 5000, RSview)

**Acknowledgement Not Required, Not Latched****AckRequired = False, Latched = False****Diagramas de temporización de alarma digital**

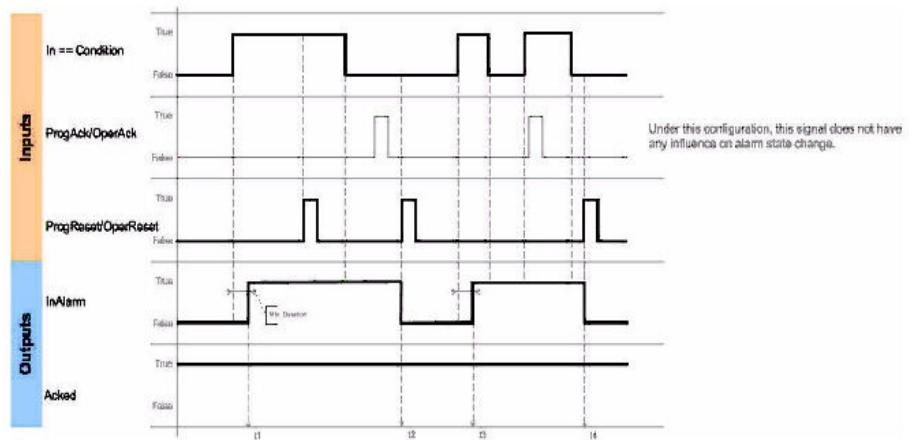
Confirmación de alarma ALMD requerida y enclavada.



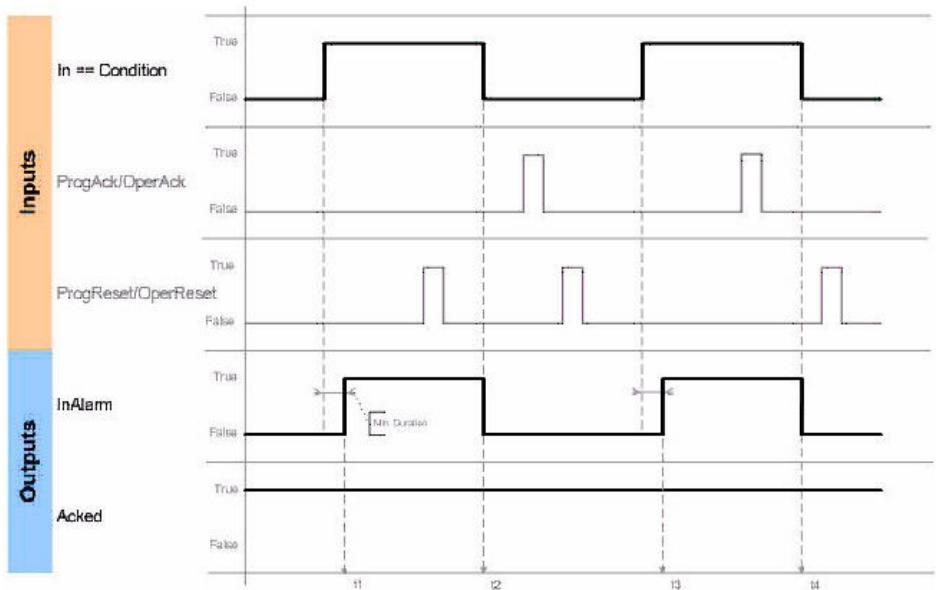
### Confirmación de alarma ALMD requerida y no enclavada.



### Confirmación de alarma ALMD no requerida y enclavada.



### Confirmación de alarma ALMD no requerida y no enclavada.



### Afectar a las marcas de estado matemático

No

### Condiciones de fallo

No es específico para esta instrucción. Consulte los *Atributos comunes* para fallos relacionados con el operando.

### Ejemplo: Alarma digital

#### Texto estructurado

A continuación se muestra un ejemplo de una instrucción de ALMD en texto estructurado. En este ejemplo, se combinan dos señales de error de motor de tal manera que si una de las dos aparece, se activa una alarma de error de motor. Se puede usar la etiqueta Motor101Ack para confirmar esta alarma.

```
Motor101FaultConditions := Motor101Overtemp OR Motor101FailToStart;
```

```
ALMD(Motor101Fault, Motor101FaultConditions, Motor101Ack, 0, 0, 0);
```

#### Consulte también

[Alarma digital \(ALMD\) - Lógica de escalera](#) en la página 120

[Alarma digital \(ALMD\): Bloque de funciones](#) en la página 108

[Atributos comunes](#) en la página 923

[Sintaxis de texto estructurado](#) en la página 955

## Instrucciones de bit

### Instrucciones de bit

Se utiliza las instrucciones de bit (de tipo relé) para monitorear y controlar los bits de estado, tales como los bit de entrada o los bits de palabras de control de temporizador.

#### Instrucciones disponibles

#### Diagrama de escalera

<a href="#">XIC</a>	<a href="#">XIO</a>	<a href="#">OTE</a>	<a href="#">OTL</a>	<a href="#">OTU</a>	<a href="#">ONS</a>	<a href="#">OSR</a>	<a href="#">OSF</a>
---------------------	---------------------	---------------------	---------------------	---------------------	---------------------	---------------------	---------------------

#### Bloque de funciones y texto estructurado

<a href="#">OSRI</a>	<a href="#">OSFI</a>
----------------------	----------------------

Si desea:	Utilice esta instrucción:
Habilitar salidas cuando el bit está establecido	XIC
Habilitar salidas cuando el bit está borrado	XIO
Establecer un bit	OTE
Establecer un bit (retentivo)	OTL
Borrar un bit (retentivo)	OTU
Habilitar salidas para un escaneado cada vez que un peldaño pase a verdadero.	ONS
Habilitar un bit para un escaneado cada vez que un reglón pase a verdadero	OSR
Habilitar un bit para un escaneado cada vez que el peldaño pase a falso.	OSF
Habilitar un bit para un escaneado cada vez que el bit de entrada se establece en el bloque de funciones	OSRI
Habilitar un bit para un escaneado cada vez que el bit de entrada se borra en el bloque de funciones	OSFI

**Consulte también**

[Comparar instrucciones](#) en la página 369

[Instrucciones de cálculo/matemáticas](#) en la página 425

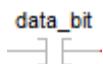
## Examinar si cerrado (XIC)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción XIC examina el bit de datos para establecer o desactivar la condición de reglón.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

#### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

#### Operандos

#### Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción
Bit de datos	BOOL	etiqueta	Se prueba el bit. Hay varios modos de direccionamiento de operandos posibles para el bit de datos, consulte <i>Direccionamiento de bit</i> para ver ejemplos.

#### Afectar a las marcas de estado matemático

No

### Fallos mayores/menores

No es específico para esta instrucción. Consulte *Índice a través de matrices* para ver si hay fallos de indexación de matrices.

### Ejecución

#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón
La condición de entrada de reglón es verdadera	Si DataBit está en verdadero, la condición de salida de reglón se establece en verdadera. Si DataBit está en falso, la condición de salida de reglón se borra y se establece en falsa.
Post-escaneado	N/A

### Ejemplo 1

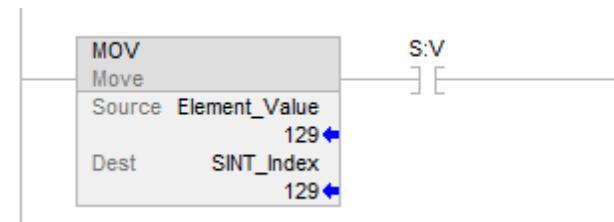
#### Diagrama de escalera



Si Limit\_Switch\_1 está en verdadero, se habilita la siguiente instrucción.

### Ejemplo 2

#### Diagrama de escalera



Si S:V (generado por MOV) está en verdadero, se habilita la siguiente instrucción.

### Ejemplo 3

#### Diagrama de escalera

```
Test_Axis_00.BusUndervoltageULFault  
<Axis_04.BusUndervoltageULFault>
```



Acceso XIC número LINT

Axis\_04 es una etiqueta AXIS\_CIP\_DRIVE.

Test\_Axis\_00 es un alias para Axis\_04.

El tipo AXIS\_CIP\_DRIVE tiene un miembro LINT llamado CIPAxisFaults.

BusUndervoltageULFault es un miembro bit de CIPAxisFaults.

Test\_Axis\_00.BusUndervoltageULFault es bit 34 para CIPAxisFaults. El valor de bit 34 es 0x400000000.

Si Test\_Axis\_00.BusUndervoltageULFault está en verdadero, se habilita la siguiente instrucción.

#### Consulte también

[Instrucciones de bit](#) en la página 143

[Direccionamiento de bit](#) en la página 936

[Índice a través de matrices](#) en la página 936

## Examinar si abierto (XIO)

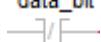
Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción XIO examina el bit de datos para establecer o desactivar la condición de reglón.

#### Idiomas disponibles

#### Diagrama de escalera

```
data_bit
```



### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

### Operандos

#### Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción
Bit de datos	BOOL	etiqueta	Se prueba el bit. Hay varios modos de direccionamiento de operandos posibles para el bit de datos, consulte <i>Direccionamiento de bit</i> para ver ejemplos.

#### Afectar a las marcas de estado matemático

No

#### Fallos mayores/menores

No es específico para esta instrucción. Consulte *Índice a través de matrices* para ver si hay fallos de indexación de matrices.

### Ejecución

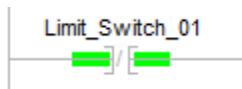
#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón
La condición de entrada de reglón es verdadera	Si Data Bit está en verdadero, la condición de salida de reglón se borra y se establece en falsa. Si Data Bit está establecido en falso, la condición de salida de reglón se establece en verdadera.
Post-escaneado	N/A

## Ejemplos

### Ejemplo 1

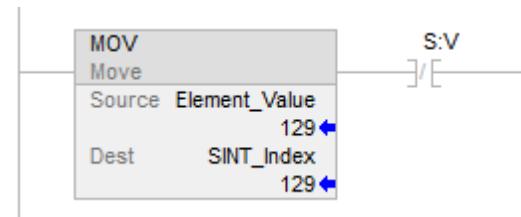
#### Diagrama de escalera



Si Limit\_Switch\_01 está establecido en falso, se habilita la siguiente instrucción.

### Ejemplo 2

#### Diagrama de escalera



Si S:V está establecido en falso, se habilita la siguiente instrucción.

#### Consulte también

[Instrucciones de bit](#) en la página 143

[Direccionamiento de bit](#) en la página 936

[Índice a través de matrices](#) en la página 936

## Un impulso (ONS)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción ONS hace un resto del reglón en verdadero cada vez que la condición de entrada de reglón cambia de falso a verdadero.

#### Idiomas disponibles

#### Diagrama de escalera



### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

### Operандos

**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
- Los miembros de un operando de estructura se sobrescriben.
- A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.

### Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción
Bit de almacenamiento	BOOL	etiqueta	Bit de almacenamiento interno. Conserva la condición de entrada de reglón de la última vez que la instrucción se ejecutó. Hay varios modos de direccionamiento de operandos posibles para el bit de datos, consulte <i>Bit de almacenamiento</i> para ver ejemplos.

### Afectar a las marcas de estado matemático

No

### Fallos mayores/menores

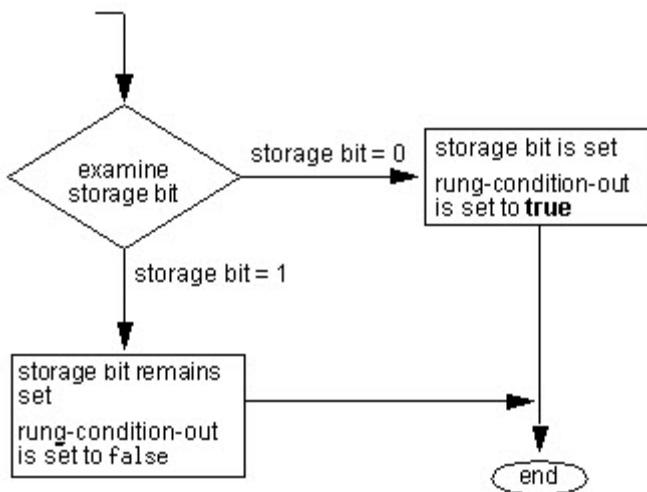
No es específico para esta instrucción. Consulte *Índice a través de matrices* para ver si hay fallos de indexación de matrices.

## Ejecución

### Diagrama de escalera

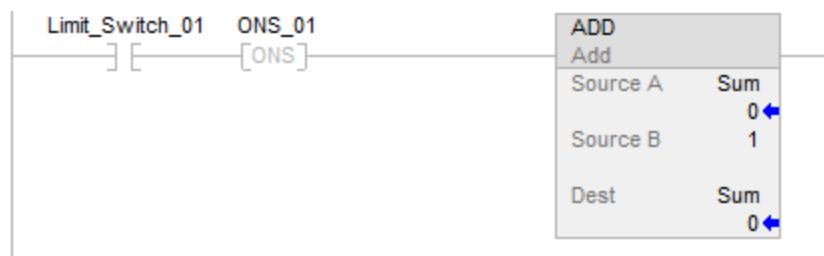
Condición/estado	Acción realizada
Pre-escaneado	El bit de almacenamiento se establece en verdadero para prevenir desencadenamientos inválidos durante el primer escaneado.
La condición de entrada de reglón es falsa	Si el bit de almacenamiento está establecido en falsa, la condición de salida de reglón se borra a falsa.
La condición de entrada de reglón es verdadera	Consultar el diagrama de flujo ONS (Verdadero)
Post-escaneado	N/A

### Diagrama de flujo ONS (Verdadero)



## Ejemplo

### Diagrama de escalera



En este ejemplo, la suma aumenta cada vez que limit\_switch\_1 cambia de falso a verdadero.

### Consulte también

[Instrucciones de bit](#) en la página 143

[Direccionamiento de bit](#) en la página 936

[Índice a través de matrices](#) en la página 936

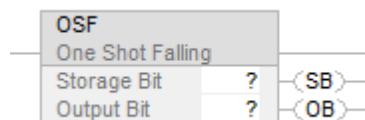
## Un impulso en flanco descendente(OSF)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción OSF establece el bit de salida para un escaneado cuando la condición de entrada de peldaño cambia de verdadero a falso.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

#### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

## Operandos

**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
- Los miembros de un operando de estructura se sobrescriben.
- A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.

## Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción
Storage Bit	BOOL	etiqueta	Almacena la condición de entrada de peldaño de cuando la instrucción se ejecutó por última vez. Hay varios modos de direccionamiento de operandos posibles para el bit de datos, consulte Bit de almacenamiento para ver ejemplos.
Output Bit	BOOL	etiqueta	Se modifica el bit.

## Afectar a las marcas de estado matemático

No

## Fallos mayores/menores

No es específico para esta instrucción. Consulte *Índice a través de matrices* para ver si hay fallos de indexación de matrices.

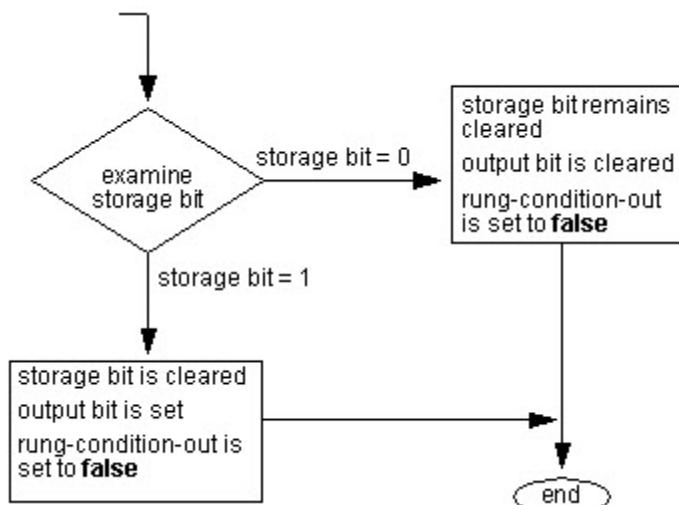
## Ejecución

### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	El bit de almacenamiento se borra y se establece en falso para prevenir desencadenamientos inválidos durante el primer escaneado del programa. El bit de salida se borra y se establece en falso.

La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón. Consultar el Diagrama de flujo OSF (Falso)
La condición de entrada de reglón es verdadera	Establecer la condición de salida de reglón a condición de entrada de reglón. El bit de almacenamiento se establece en verdadero. El bit de salida se borra y se establece en falso.
Post-escaneado	N/A

### Diagrama de flujo OSF (Falso)



### Ejemplo

#### Diagrama de escalera



Este ejemplo muestra cómo se puede usar un OSF para hacer una o más instrucciones de desencadenamiento de flanco. Cada vez que Limit\_Switch\_01 cambia de verdadero a falso, el OSF cambiará Output\_bit\_02 a verdadero. Cualquier instrucción condicionada por Output\_bit\_02 se habilitará y, puesto que Output\_bit\_02 solo es verdadero para un escaneado, se ejecutará una vez por transición.

#### Consulte también

[Instrucciones de bit](#) en la página 143

[Direccionamiento de bit](#) en la página 936

[Índice a través de matrices](#) en la página 936

## Un impulso en flanco descendente con entrada (OSFI)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

Cuando OutputBit cambia de falso a verdadero, la instrucción OSFI establece InputBit para un ciclo de ejecución.

#### Idiomas disponibles

#### Diagrama de escalera

Esta instrucción no está disponible en el diagrama de escalera.

#### Bloque de funciones



#### Texto estructurado

OSFI(OSFI\_tag)

## Operandos

### Texto estructurado

Operando	Tipo	Formato	Descripción
OSFI tag	FBD_ONESHOT	Estructura	Estructura de OSFI

Consulte *Sintaxis de texto estructurado* para fallos relacionados con los operandos.

### Bloque de funciones

Operando	Tipo	Formato	Descripción
OSFI tag	FBD_ONESHOT	Estructura	Estructura de OSFI

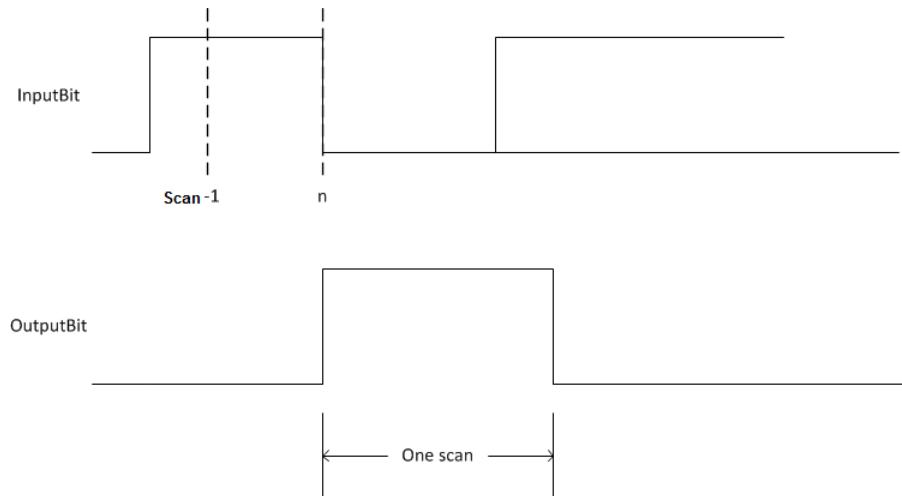
### Estructura de FBD\_ONESHOT

Parámetro de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si se borra, la instrucción no se ejecuta y las salidas no se actualizan. Está establecido de forma predeterminada.
InputBit	BOOL	Bit de entrada.

Parámetro de salida	Tipo de datos	Descripción
EnableOut	BOOL	Indica si la instrucción está habilitada.
OutputBit	BOOL	Bit de salida

## Descripción

Si InputBit es falso, y estaba establecido en verdadero la última vez que se escaneó la instrucción, se establecerá el OutputBit, si no, se borrará OutputBit.



## Afectar a las marcas de estado matemático

No

## Fallos mayores/menores

No es específico para esta instrucción. Consulte los *Atributos comunes* para fallos relacionados con el operando.

## Ejecución

### Bloque de funciones

Condición/estado	Acción realizada
Pre-escaneado	Los bits EnableIn y EnableOut se borran a falsos.
Tag.EnableIn es falso	Los bits EnableIn y EnableOut se borran a falsos.
Tag.EnableIn es verdadero	Los bits EnableIn y EnableOut se establecen en verdaderos. La instrucción se ejecuta
Primera ejecución de instrucción	N/A
Primer escaneado de instrucción	El historial anterior de InputBit se borra para requerir una transición de Verdadero a Falso del InputBit.
Post-escaneado	Los bits EnableIn y EnableOut se borran a falsos.

### Texto estructurado

Condición/estado	Acción realizada
Pre-escaneado	Consultar Pre-escaneado en la tabla Bloque de funciones.
Ejecución normal	Consultar Tag.EnableIn es verdadero en la tabla Bloque de funciones.
Post-escaneado	Consultar Post-escaneado en la tabla Bloque de funciones.

### Ejemplo

Cuando limit\_switch1 cambia de activado a desactivado, la instrucción OSFI establece OutputBit para un escaneado.

### Bloque de funciones



### Texto estructurado

```
OSFI_01.InputBit := limit_switch1;
```

```
OSFI(OSFI_01);
```

```
Output_state := OSFI_01.OutputBit;
```

### Consulte también

[Instrucciones de bit](#) en la página 143

[OSF](#) en la página 151

[Atributos comunes](#) en la página 923

[Sintaxis de texto estructurado](#) en la página 955

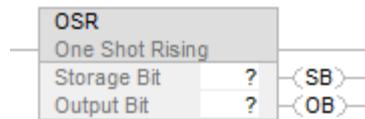
## Un impulso en flanco ascendente (OSR)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción OSR establece el bit de salida para un escaneado cuando la condición de entrada de peldaño cambia de falso a verdadero.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

#### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

#### Operандos

---

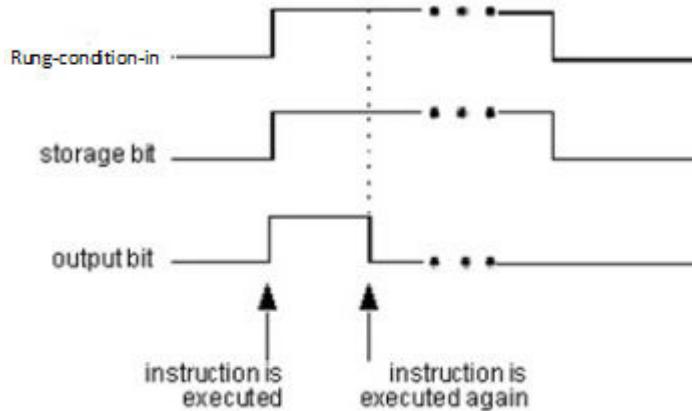
**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
  - Los miembros de un operando de estructura se sobrescriben.
  - A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.
- 

#### Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción
Storage Bit	BOOL	etiqueta	Almacena la condición de entrada de peldaño de cuando la instrucción se ejecutó por última vez. Hay varios modos de direccionamiento de operandos posibles para el bit de datos, consulte <i>Bit de almacenamiento</i> para ver ejemplos.
Output Bit	BOOL	etiqueta	Se modifica el bit.

### Descripción



### Afectar a las marcas de estado matemático

No

### Fallos mayores/menores

No es específico para esta instrucción. Consulte *Índice a través de matrices* para ver si hay fallos de indexación de matrices.

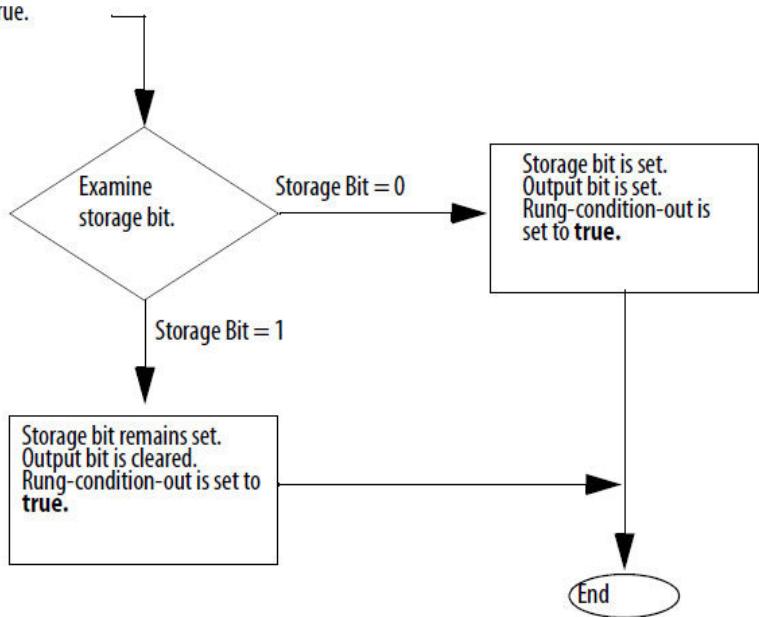
### Ejecución

#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	El bit de almacenamiento se establece en verdadero para prevenir desencadenamientos inválidos durante el primer escaneado del programa. El bit de salida se borra y se establece en falso.
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón El bit de almacenamiento se borra y se establece en falso. El bit de salida se borra y se establece en falso.
La condición de entrada de reglón es verdadera	Establecer la condición de salida de reglón a condición de entrada de reglón Consultar el Diagrama de flujo OSR (Verdadero).
Post-escaneado	N/A

### Diagrama de flujo OSR (Verdadero)

Rung-condition-in is true.



### Ejemplo

#### Diagrama de escalera



Este ejemplo muestra cómo se puede usar un OSR para hacer una o más instrucciones de desencadenamiento de flanco. Cada vez que Limit\_Switch\_01 cambia de falso a verdadero, el OSR cambiará Output\_bit\_02 a verdadero. Cualquier instrucción condicionada por Output\_bit\_02 se habilitará y, puesto que Output\_bit\_02 solo es verdadero para un escaneado, se ejecutará una vez por transición.

#### Consulte también

[Instrucciones de bit](#) en la página 143

[Direccionamiento de bit](#) en la página 936

[Índice a través de matrices](#) en la página 936

## Un impulso en flanco ascendente con entrada (OSRI)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

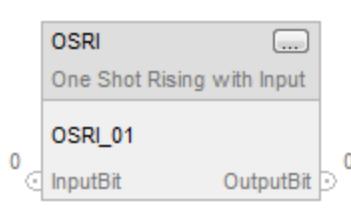
Cuando el bit de entrada cambia de desactivado a activado, la instrucción OSRI establece el bit de salida para un ciclo de ejecución.

### Idiomas disponibles

### Diagrama de escalera

Esta instrucción no está disponible en el diagrama de escalera.

### Bloque de funciones



### Texto estructurado

OSRI(OSRI\_tag);

### Operandos

### Texto estructurado

Operando	Tipo	Formato	Descripción
OSRI tag	FBD_ONESHOT	Estructura	Estructura de OSRI

### Bloque de funciones

Operando	Tipo	Formato	Descripción
OSRI tag	FBD_ONESHOT	Estructura	Estructura de OSRI

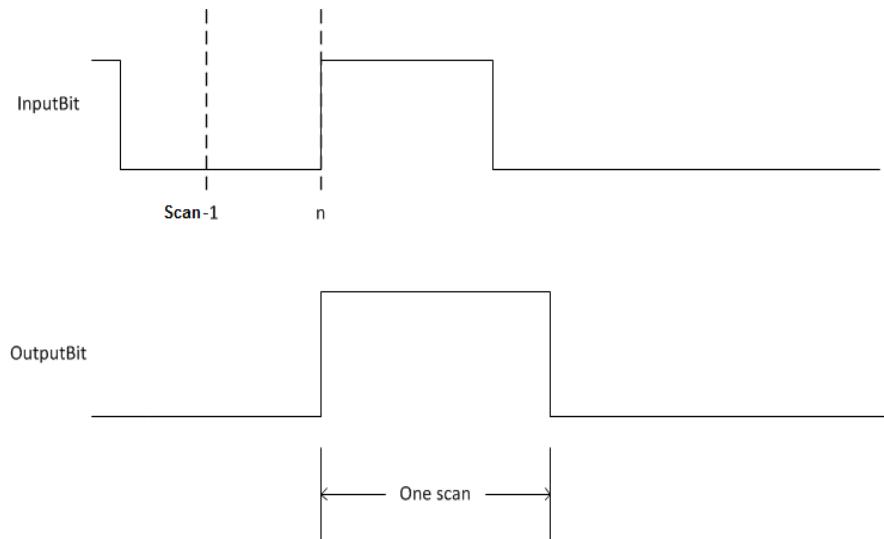
### Estructura de FBD\_ONESHOT

Parámetro de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Si no está activada, la instrucción no se ejecuta y las salidas no se actualizan. Si está activado, la instrucción se ejecuta. Está establecido de forma predeterminada.
InputBit	BOOL	Bit de entrada. Está borrado de forma predeterminada.

Parámetro de salida	Tipo de datos	Descripción
EnableOut	BOOL	Indica si la instrucción está habilitada.
OutputBit	BOOL	Bit de salida

### Descripción

Si InputBit es verdadero, y estaba establecido en falso la última vez que se escaneó la instrucción, se establecerá el OutputBit, si no, se borrará OutputBit.



### Afectar a las marcas de estado matemático

No

### Fallos mayores/menores

No es específico para esta instrucción. Consulte los *Atributos comunes* para fallos relacionados con el operando.

### Ejecución

#### Bloque de funciones

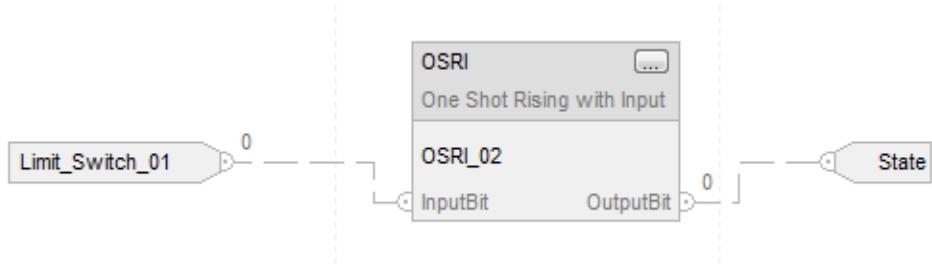
Condición/estado	Acción realizada
Pre-escaneado	Los bits EnableIn y EnableOut se borran a falsos.
Tag.Enable-in es falso	Los bits EnableIn y EnableOut se borran a falsos.
Tag.Enable-in es verdadero	Los bits EnableIn y EnableOut se establecen en verdaderos. La instrucción se ejecuta.
Primera ejecución de instrucción	N/A
Primer escaneado de instrucción	El historial anterior de InputBit se establece para requerir una transición de Falso a Verdadero del InputBit.
Post-escaneado	Los bits EnableIn y EnableOut se borran a falsos.

#### Texto estructurado

Condición/estado	Acción realizada
Pre-escaneado	Consultar Pre-escaneado en la tabla Bloque de funciones.
Ejecución normal	Consultar Tag.EnableIn es verdadero en la tabla de bloque de funciones.
Post-escaneado	Consultar Post-escaneado en la tabla de bloque de funciones.

## Ejemplos

### Bloque de funciones



Cuando limit\_switch1 cambia de activado a desactivado, la instrucción OSRI establece OutputBit para un escaneado.

### Texto estructurado

```
OSRI_01.InputBit := limit_switch1;
```

```
OSRI(OSRI_01);
```

```
State := OSRI_O1.OutputBit;
```

### Consulte también

[Instrucciones de bit](#) en la página 143

[Un impulso en flanco descendente\(OSF\)](#) en la página 151

[Un impulso \(ONS\)](#) en la página 148

[Atributos comunes](#) en la página 923

[Sintaxis de texto estructurado](#) en la página 955

## Activación de salida (OTE)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción OTE activa o desactiva el bit de datos en función de la condición de peldaño.

## Idiomas disponibles

### Diagrama de escalera

data\_bit  
—<—>—

## Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

## Texto estructurado

Esta instrucción no está disponible en texto estructurado.

## Operандos

**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
- Los miembros de un operando de estructura se sobrescriben.
- A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.

## Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción
Bit de datos	BOOL	etiqueta	Se modifica el bit. Hay varios modos de direccionamiento de operandos posibles para el bit de datos, consulte <i>Direccionamiento de bit</i> para ver ejemplos.

## Afectar a las marcas de estado matemático

No

## Fallos mayores/menores

No es específico para esta instrucción. Consulte *Índice a través de matrices* para ver si hay fallos de indexación de matrices.

## Ejecución

### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	El bit de datos se borra y se establece en falso
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón. El bit de datos se borra y se establece en falso
La condición de entrada de reglón es verdadera	Establecer la condición de salida de reglón a condición de entrada de reglón. El bit de datos se establece en verdadero.
Post-escaneado	El bit de datos se borra y se establece en falso.

### Ejemplo

### Diagrama de escalera



Cuando el conmutador es true, la instrucción OTE establece Light\_01 en verdadero. Cuando el conmutador es false, la instrucción OTE borra y establece Light\_01 en falso.

### Consulte también

[Sintaxis de texto estructurado](#) en la página 955

[Instrucciones de bit](#) en la página 143

[Direccionamiento de bit](#) en la página 936

[Índice a través de matrices](#) en la página 936

## Enclavamiento de salida (OTL)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

La instrucción OTL activa (bloquea) el bit de datos.

## Idiomas disponibles

### Diagrama de escalera

data\_bit  
—(L)—

## Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

## Texto estructurado

Esta instrucción no está disponible en texto estructurado.

## Operандos

**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
- Los miembros de un operando de estructura se sobrescriben.
- A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.

### Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción
Bit de datos	BOOL	etiqueta	Se modifica el bit. Hay varios modos de direccionamiento de operandos posibles para el bit de datos, consulte <i>Direccionamiento de bit</i> para ver ejemplos.

## Descripción

Cuando la condición de reglón es verdadera, la instrucción OTL establece el bit de datos en verdadero. El bit de datos permanece en verdadero hasta que, normalmente, lo desactiva una instrucción OTU. Cuando la condición de reglón cambia a falso, la instrucción OTL no cambia el estado del bit de datos.

## Afectar a las marcas de estado matemático

No

### Fallos mayores/menores

No es específico para esta instrucción. Consulte *Índice a través de matrices* para ver si hay fallos de indexación de matrices.

Para controllers Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580, si el operando es una referencia de matriz indirecta y el subíndice está fuera de rango, el controlador no generará un fallo cuando la instrucción OTL sea falso.

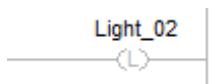
### Ejecución

#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón.
La condición de entrada de reglón es verdadera	Establecer la condición de salida de reglón a condición de entrada de reglón. El bit de datos se establece en verdadero.
Post-escaneado	N/A

### Ejemplo

#### Diagrama de escalera



Cuando esté activada, la instrucción OTL encenderá la luz.

### Consulte también

[Sintaxis de texto estructurado](#) en la página 955

[Instrucciones de bit](#) en la página 143

[Direccionamiento de bit](#) en la página 936

[Índice a través de matrices](#) en la página 936

## Desbloqueo de salida (OTU)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix

5580 y GuardLogix 5580.

La instrucción OTU desactiva (desbloquea) el bit de datos.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

#### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

#### Operандos

- Importante:** Puede ocurrir un funcionamiento inesperado si:
- Los operandos de etiqueta de salida se sobrescriben.
  - Los miembros de un operando de estructura se sobrescriben.
  - A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.

#### Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción
Bit de datos	BOOL	etiqueta	Se modifica el bit. Hay varios modos de direccionamiento de operandos posibles para el bit de datos, consulte <i>Direccionamiento de bit</i> para ver ejemplos.

#### Descripción

Cuando la condición de peldaño es verdadera, la instrucción OTU borra y establece el bit de datos en falso. Cuando la condición de peldaño cambia a falso, la instrucción OTU no cambia el estado del bit de datos.

### Afectar a las marcas de estado matemático

No

### Fallos mayores/menores

No es específico para esta instrucción. Consulte *Índice a través de matrices* para ver si hay fallos de indexación de matrices.

### Ejecución

#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón
La condición de entrada de reglón es verdadera	Establecer la condición de salida de reglón a condición de entrada de reglón El bit de datos se borra y se establece en falso.
Post-escaneado	N/A

### Ejemplo

#### Diagrama de escalera



Cuando está activada, la instrucción OTU desactiva Light\_02.

### Consulte también

[Instrucciones de bit](#) en la página 143

[Direccionamiento de bit](#) en la página 936

[Índice a través de matrices](#) en la página 936

## Instrucciones de temporizador y contador

### Instrucciones de temporizador y contador

Operaciones de control de temporizadores y contadores basadas en tiempo o en un número de eventos.

#### Instrucciones disponibles

##### Diagrama de escalera

<a href="#">TON</a>	<a href="#">TOF</a>	<a href="#">RTO</a>	<a href="#">CTU</a>	<a href="#">CTD</a>	<a href="#">RES</a>
---------------------	---------------------	---------------------	---------------------	---------------------	---------------------

##### Bloque de funciones y texto estructurado

<a href="#">TONR</a>	<a href="#">TOFR</a>	<a href="#">RTOR</a>	<a href="#">CTUD</a>
----------------------	----------------------	----------------------	----------------------

Si desea	Utilice esta instrucción
Saber cuánto tiempo se ha habilitado un temporizador.	TON
Saber cuánto tiempo se ha deshabilitado un temporizador.	TOF
Acumular tiempo.	RTO
Saber cuánto tiempo se ha habilitado un temporizador con el restablecimiento incorporado en el bloque de funciones.	TONR
Saber cuánto tiempo se ha deshabilitado un temporizador con el restablecimiento incorporado en el bloque de funciones.	TOFR
Acumular tiempo con el restablecimiento incorporado en el bloque de funciones.	RTOR
contar de manera ascendente	CTU

contar de manera descendente	CTD
contar de manera ascendente y descendente en el bloque de funciones	CTUD
Restablecer un temporizador o un contador.	RES

La base de tiempo es 1 milisecondo para todos los temporizadores. Por ejemplo, el valor .PRE de un temporizador de 2 segundos debería ser 2000.

#### Consulte también

[Instrucciones de cálculo/matemáticas](#) en la página 425

[Comparar instrucciones](#) en la página 369

[Instrucciones de bit](#) en la página 143

[Instrucciones de cadena ASCII](#) en la página 871

[Instrucciones de conversión ASCII](#) en la página 892

## Conteo descendente (CTD)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción CTD cuenta hacia abajo cada vez que la condición de entrada de reglón cambia de falso a verdadero.

#### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

## Texto estructurado

Esta instrucción no está disponible en texto estructurado.

## Operандos

**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
- Los miembros de un operando de estructura se sobrescriben.
- A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.

## Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción
Counter	COUNTER	etiqueta	Estructura de contador
Preset	DINT	inmediato	Valor del contador .PRE.
Accum	DINT	inmediato	Valor del contador .ACC.

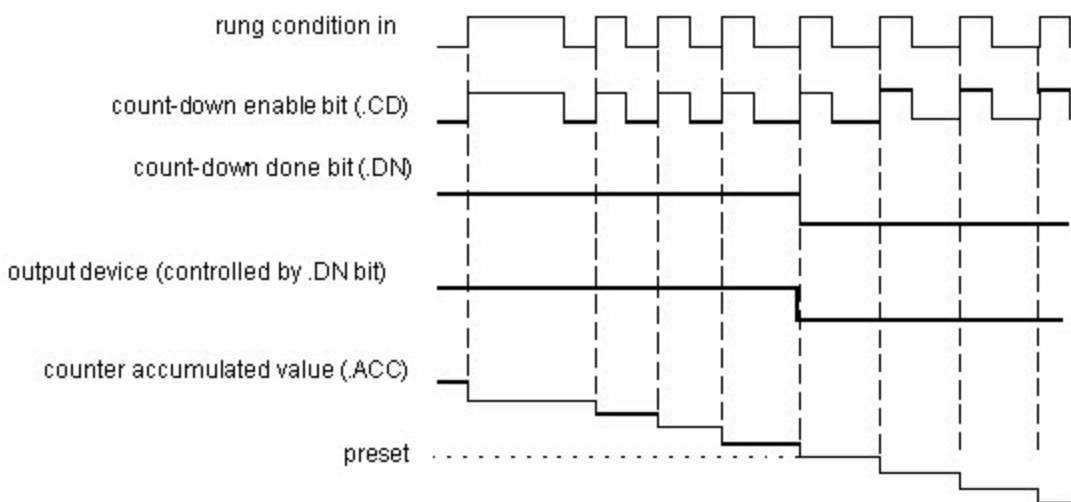
## Estructura de COUNTER

Mnemónico	Tipo de datos	Descripción
.CD	BOOL	El bit de habilitación de conteo descendente contiene una condición de entrada de reglón cuando la instrucción se ejecutó por última vez.
.DN	BOOL	Cuando el bit de efectuado está borrado, indica que la operación de conteo se ha completado.
.OV	BOOL	Cuando el bit de desbordamiento está establecido, indica que el contador ha alcanzado el límite superior de 2,147,483,647.
.UN	BOOL	Cuando el subdesbordamiento está establecido, indica que el contador se ha reducido al límite inferior de -2,147,483,648.
.PRE	DINT	El valor preestablecido indica el número al que debe llegar el valor acumulado antes de que la instrucción indique que está finalizada.
.ACC	DINT	El valor acumulado indica el número de transiciones que ha contado la instrucción.

### Descripción

La instrucción CTD normalmente se usa con una instrucción CTU que hace referencia a la misma estructura de contador.

Cuando la condición de entrada de reglón se establece en verdadera, .CD es falso y .ACC decrementará en uno. Cuando la condición de entrada de reglón se establece en falso, .CD se borrará y establecerá en falso.



### Afectar a las marcas de estado matemático

No

### Fallos mayores/menores

No es específico para esta instrucción. Consulte Índice a través de matrices para ver si hay fallos de indexación de matrices.

### Ejecución

#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	El bit .CD se establece en verdadero para prevenir decrementos inválidos durante el primer escaneado del programa.
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón Consultar el diagrama de flujo CTD (Falso)
La condición de entrada de reglón es verdadera	Establecer la condición de salida de reglón a condición de entrada de reglón Consultar el diagrama de flujo CTD (Verdadero)

Post-escaneado	N/A
----------------	-----

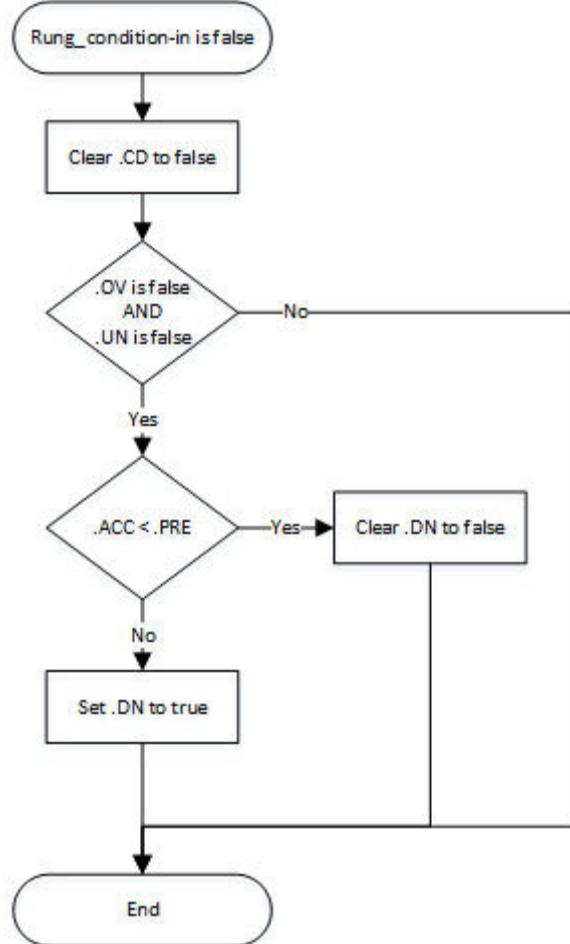
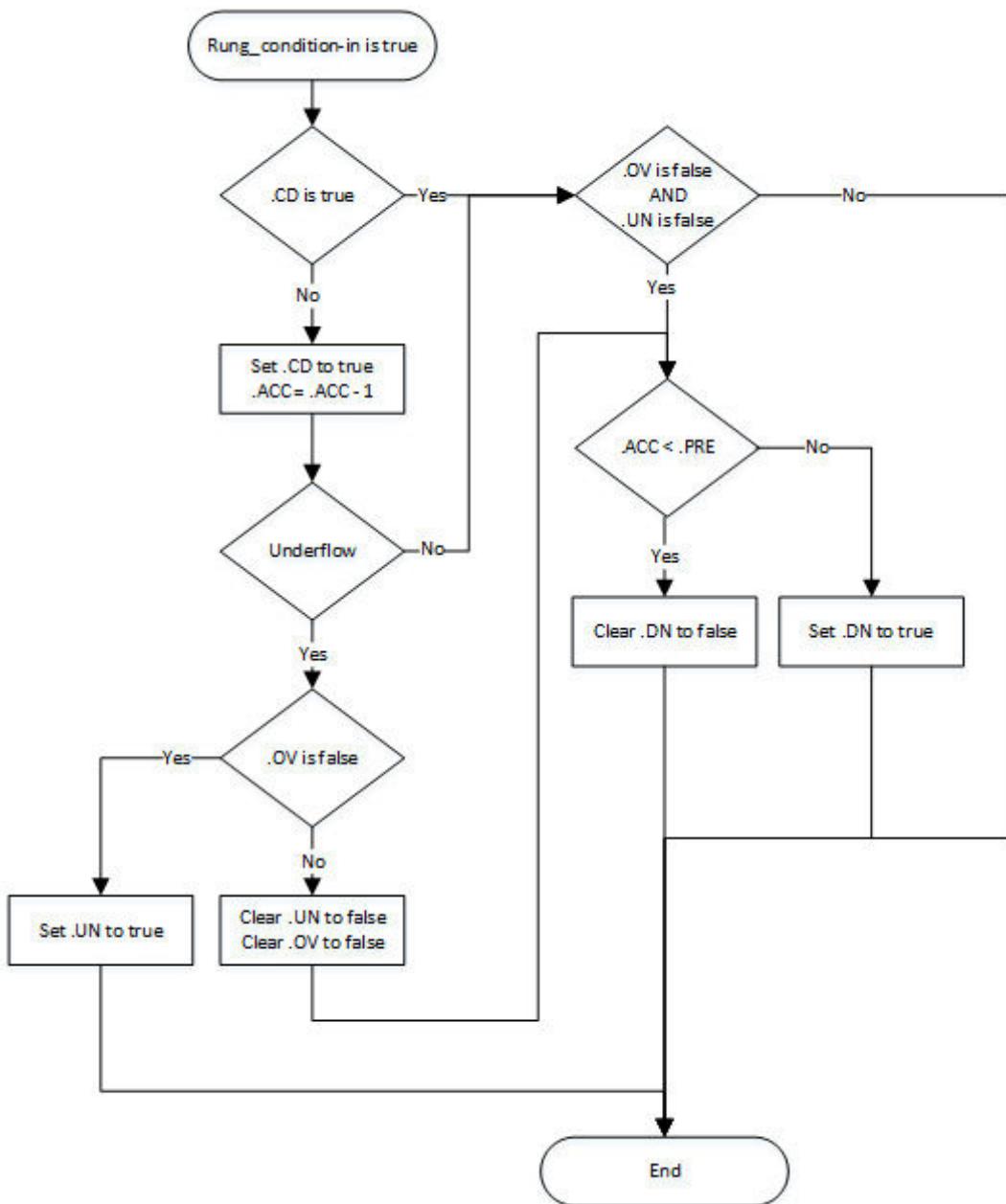
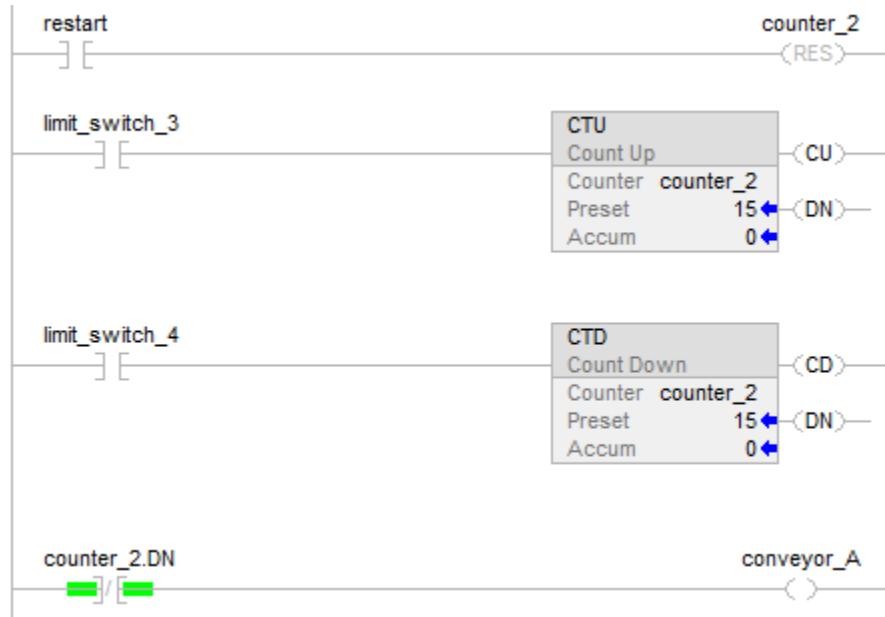
**Diagrama de flujo CTD (Falso)**

Diagrama de flujo CTD (Verdadero)



## Ejemplo

### Diagrama de escalera



Un transportador lleva partes a la zona de búfer. Cada vez que entra una parte, se habilita `limit_switch_3` y `counter_2` incrementa en 1. Cada vez que se va una parte, se habilita `limit_switch_4` y `counter_2` decremente en 1. Si hay 100 partes en la zona de búfer (`counter_2.DN` está en verdadero), se enciende `conveyor_A` y evita que el transportador entrega más partes hasta que el búfer tenga espacio para más partes.

### Consulte también

[Índice a través de matrices](#) en la página 936

[Instrucciones del contador](#) en la página 171

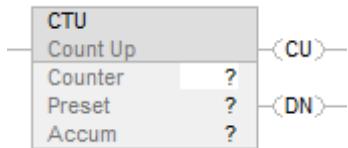
## Conteo ascendente (CTU)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción CTU cuenta hacia arriba cada vez que la condición de entrada de reglón hace una transición de falso a verdadero.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

#### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

#### Operандos

**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
- Los miembros de un operando de estructura se sobrescriben.
- A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.

#### Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción
Counter	COUNTER	etiqueta	Estructura de contador
Preset	DINT	inmediato	Valor del contador .PRE.
Accum	DINT	inmediato	Valor del contador .ACC.

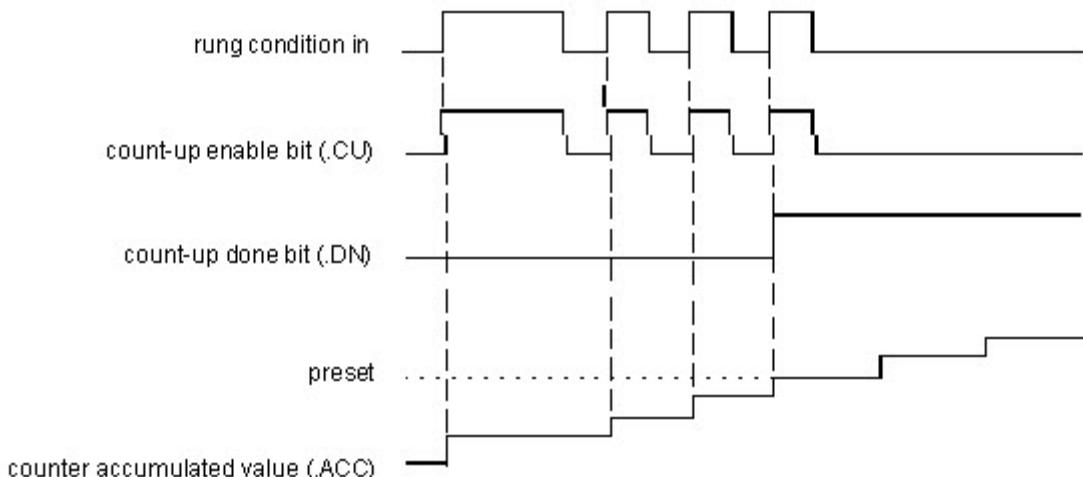
#### Estructura de COUNTER

Mnemónico	Tipo de datos	Descripción
.CU	BOOL	El bit de habilitación de conteo ascendente contiene una condición de entrada de reglón cuando la instrucción se ejecutó por última vez.
.DN	BOOL	Cuando el bit de efectuado se establece, indica que la operación de conteo se ha completado.

.OV	BOOL	Cuando el bit de desbordamiento está establecido, indica que el contador ha alcanzado el límite superior de 2,147,483,647.
.UN	BOOL	Cuando el subdesbordamiento está establecido, indica que el contador se ha reducido al límite inferior de -2,147,483,648.
.PRE	DINT	El valor preestablecido indica el número al que debe llegar el valor acumulado antes de que la instrucción indique que está finalizada.
.ACC	DINT	El valor acumulado indica el número de transiciones que ha contado la instrucción.

### Descripción

Cuando la condición de entrada de reglón se establece en verdadera, .CU es falso y .ACC aumentará en uno. Cuando la condición de entrada de reglón se establece en falso, .CU se borrará y establecerá en falso.



### Afectar a las marcas de estado matemático

No

### Fallos mayores/menores

No es específico para esta instrucción. Consulte Índice a través de matrices para ver si hay fallos de indexación de matrices.

## Ejecución

### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	El bit .CU se establece en verdadero para prevenir aumentos inválidos durante el primer escaneado del programa.
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón Consultar el diagrama de flujo CTU (Falso)
La condición de entrada de reglón es verdadera	Establecer la condición de salida de reglón a condición de entrada de reglón Consultar el diagrama de flujo CTU (Verdadero)
Post-escaneado	N/A

### Diagrama de flujo CTU (Falso)

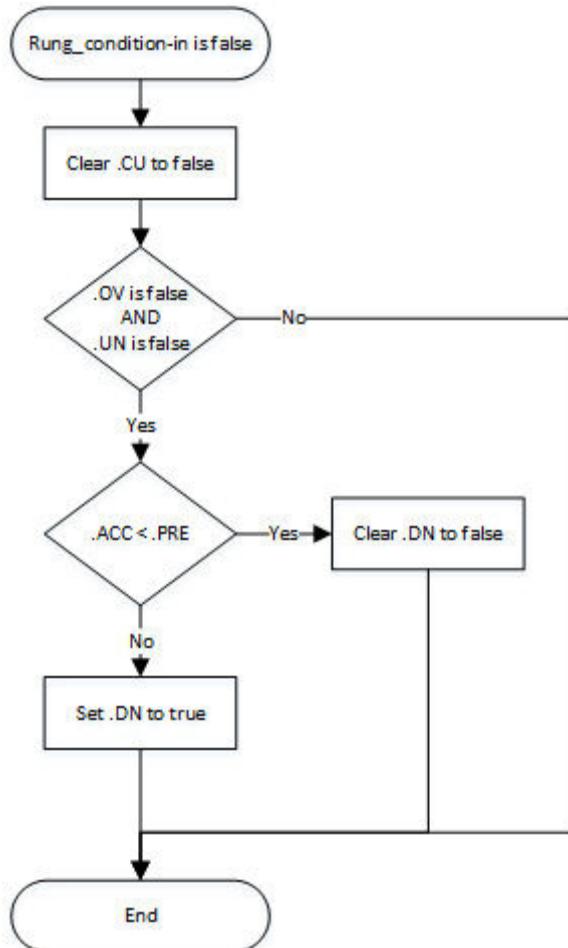
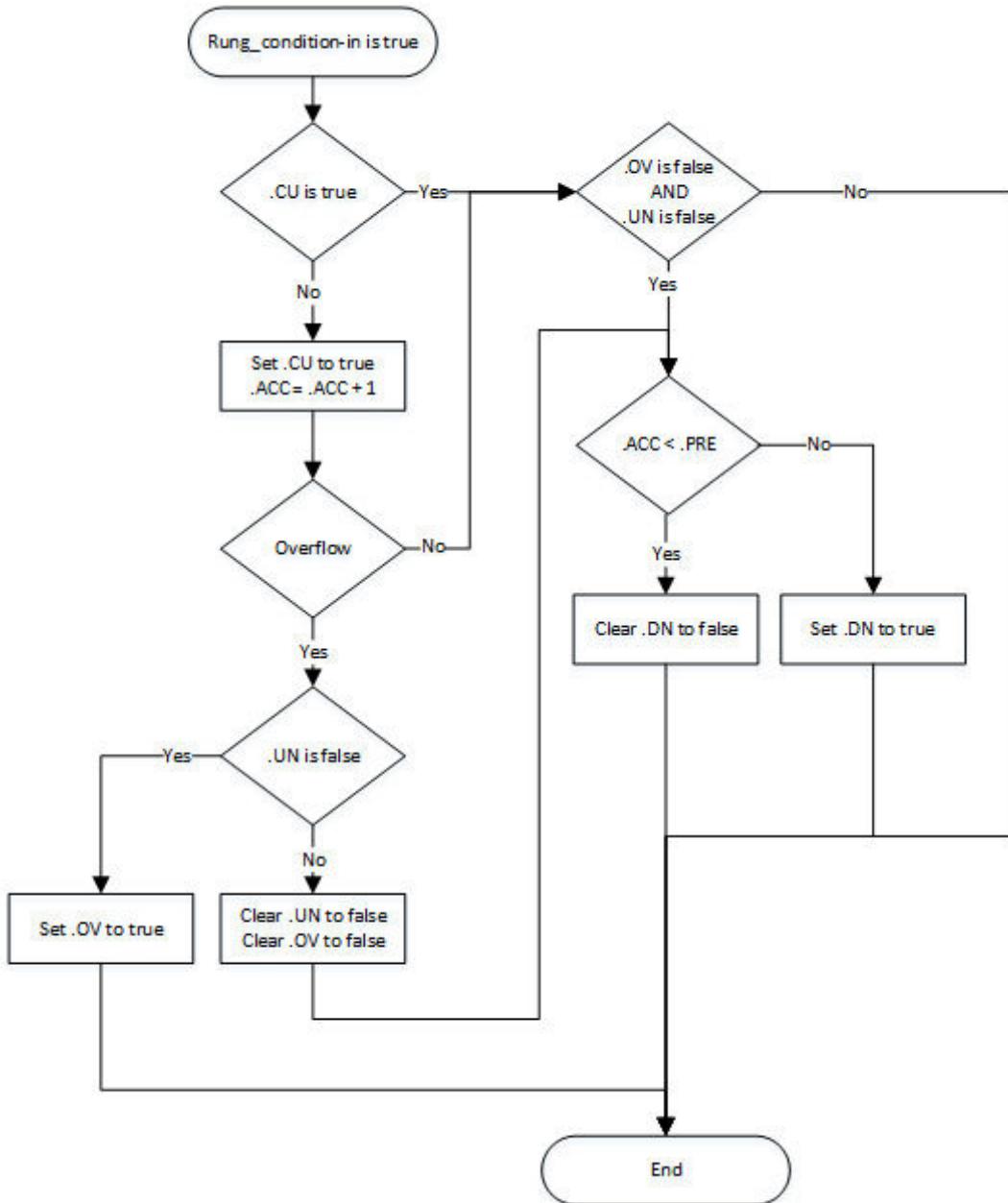
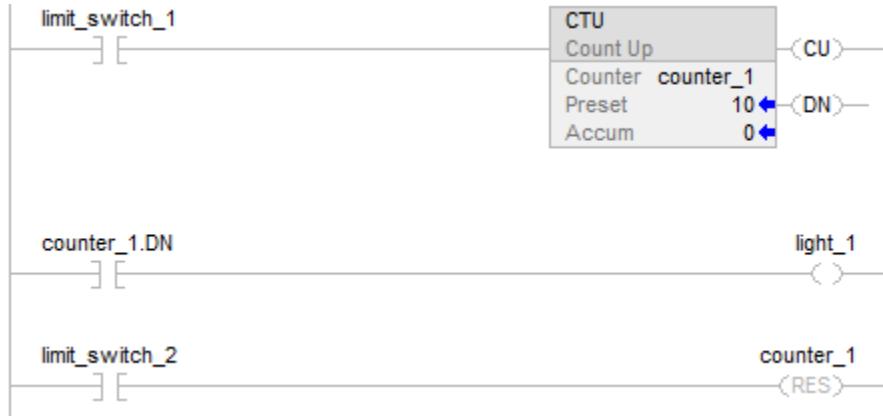


Diagrama de flujo CTU (Verdadero)



## Ejemplo

### Diagrama de escalera



Después de que limit\_switch\_1 pase de deshabilitado a habilitado 10 veces, el bit .DN se establece en verdadero y se enciende light\_1. Si limit\_switch\_1 continúa cambiando de deshabilitado a habilitado, counter\_1 seguirá incrementando su conteo y el bit .DN permanecerá establecido. Cuando se habilita limit\_switch\_2, la instrucción RES restablece counter\_1 (borra el estado de los bits y el valor .ACC) y se apaga light\_1.

### Consulte también

[Índice a través de matrices](#) en la página 936

[Instrucciones del contador](#) en la página 171

## Conteo ascendente/descendente (CTUD)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

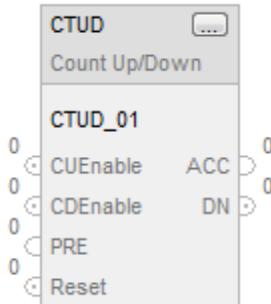
La instrucción CTUD asciende en uno cuando CUEnable pasa de tener su valor borrado a establecido. La instrucción descende en uno cuando CDEnable pasa de tener su valor borrado a establecido.

### Idiomas disponibles

### Diagrama de escalera

Esta instrucción no está disponible en el diagrama de escalera.

## Bloque de funciones



### Texto estructurado

CTUD(CTUD\_tag)

### Operandos

#### Texto estructurado

Variable	Tipo	Formato	Descripción
CTUD tag	FBD_COUNTER	Estructura	Estructura de CTUD

Consulte Sintaxis de texto estructurado para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

## Bloque de funciones

Operando	Tipo	Formato	Descripción
CTUD tag	FBD_COUNTER	Estructura	Estructura de CTUD

### Estructura de FBD\_COUNTER

Parámetro de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Si no está activada, la instrucción no se ejecuta y las salidas no se actualizan. Si está activado, la instrucción se ejecuta. Está establecido de forma predeterminada.

CUEnable	BOOL	Habilita el conteo ascendente. Cuando la entrada cambia de tener su valor borrado a establecido., el acumulador asciende en uno. Está borrado de forma predeterminada.
CDEnable	BOOL	Habilita el conteo descendiente. Cuando la entrada cambia de tener su valor borrado a establecido, el acumulador desciende en uno. Está borrado de forma predeterminada.
PRE	DINT	Valor preestablecido del contador. Este es el valor al que debe llegar el valor acumulado antes de que se establezca DN. Válido = cualquier número entero El valor predeterminado es 0.
Reset	BOOL	Petición para restablecer el temporizador. Cuando se establece, se restablece el contador. Está borrado de forma predeterminada.

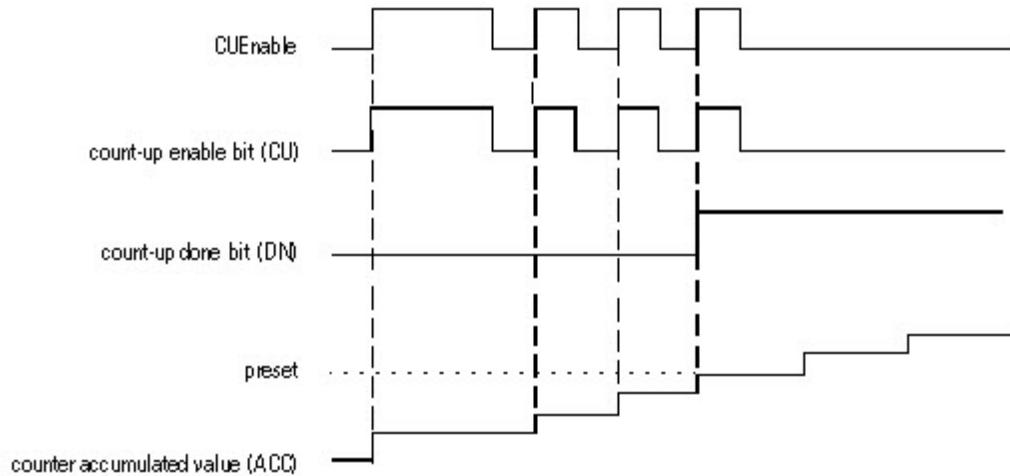
Parámetro de salida	Tipo de datos	Descripción
EnableOut	BOOL	La instrucción ha producido un resultado válido.
ACC	DINT	Valor acumulado.
CU	BOOL	Conteo ascendente habilitado.
CD	BOOL	Conteo descendente habilitado.
DN	BOOL	Conteo finalizado. Se establece cuándo el valor acumulado es mayor o igual al preestablecido.
OV	BOOL	Desbordamiento del contador. Indica que el contador ha superado el límite superior de 2 147 483 647. Después, el contador se reinvertirá hasta -2 147 483 648 y comenzará el conteo descendente de nuevo.
UN	BOOL	Subdesbordamiento del contador. Indica que el contador ha superado el límite inferior de -2 147 483 648. Después, el contador se reinvertirá hasta 2 147 483 647 y comenzará el conteo descendente de nuevo.

## Descripción

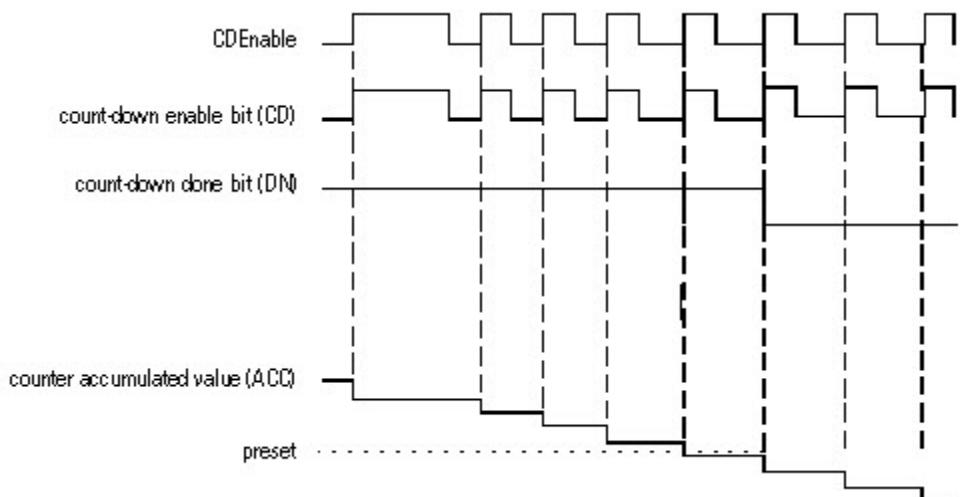
Cuando esté en verdadero y CUEnable esté establecido en verdadero, las instrucciones CTUD aumentan el contador en uno. Cuando esté en verdadero y CDEnable esté establecido en verdadero, la instrucción CTUD bajará el contador en uno.

Los dos parámetros de entrada, CUEnable y CDEnable, pueden cambiar durante el mismo escaneado. La instrucción ejecuta el conteo ascendente antes que el conteo descendente.

## Conteo ascendente



## Conteo descendente



Cuando está deshabilitada, la instrucción CTUD mantiene su valor acumulado. Establece el parámetro de entrada Reset para restablecer la instrucción.

**Afectar a las marcas de estado matemático**

No

**Fallos mayores/menores**

No es específico para esta instrucción. Consulte los *Atributos comunes* para fallos relacionados con el operando.

**Ejecución****Bloque de funciones**

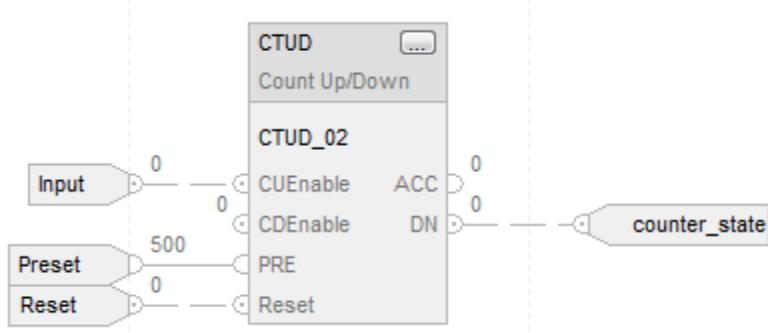
Condición/estado	Acción realizada
Pre-escaneado	Los bits EnableIn y EnableOut se borran a falsos.
Tag.EnableIn es falso	Los bits EnableIn y EnableOut se borran a falsos. Inicializar datos para solicitar una transición de cero a uno de CuEnable o CdEnable que tenga efecto en ACC.
Tag.EnableIn es verdadero	Los bits EnableIn y EnableOut se establecen en verdaderos. La instrucción se ejecuta.
Primera ejecución de instrucción	Inicializar datos para solicitar una transición de cero a uno de CuEnable o CdEnable que tenga efecto en ACC.
Primer escaneado de instrucción	Inicializar datos para solicitar una transición de cero a uno de CuEnable o CdEnable que tenga efecto en ACC.
Post-escaneado	Los bits EnableIn y EnableOut se borran a falsos.

**Texto estructurado**

Condición/estado	Acción realizada
Pre-escaneado	Consultar Pre-escaneado en la tabla Bloque de funciones.
Ejecución normal	Consultar Tag.EnableIn es verdadero en la tabla Bloque de funciones.
Post-escaneado	Consultar Post-escaneado en la tabla Bloque de funciones.

## Ejemplo

### Bloque de funciones



### Texto estructurado

```

CTUD_01.PRE := 500;
CTUD_01.Reset := Reset;
CTUD_01.CUEnable := Input;
CTUD(CTUD_01);
counter_state := CTUD_01.DN;
  
```

### Consulte también

[Atributos comunes](#) en la página 923

[Conteo ascendente \(CTU\)](#) en la página 177

[Conteo descendente \(CTD\)](#) en la página 172

[Restablecer \(RES\)](#) en la página 187

[Sintaxis de texto estructurado](#) en la página 955

## Restablecer (RES)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción RES restablece una estructura de TIMER, COUNTER o CONTROL.

### Idiomas disponibles

#### Diagrama de escalera

-<RES>-

#### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

#### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

#### Operандos

**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
- Los miembros de un operando de estructura se sobrescriben.
- A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.

#### Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción
Estructura	TIMER CONTROL COUNTER	Etiqueta	Estructura a restablecer

#### Descripción

Cuando sea verdadero, la instrucción RES borra estos elementos:

Cuando se usa una instrucción RES para:	La instrucción borra:
TIMER	El valor .ACC a 0 Borra los bits de estado a falso.
COUNTER	El valor .ACC a 0 Borra los bits de estado a falso.
CONTROL	El valor .POS a 0 Borra los bits de estado a falso.

#### Afectar a las marcas de estado matemático

No

**Fallos mayores/menores**

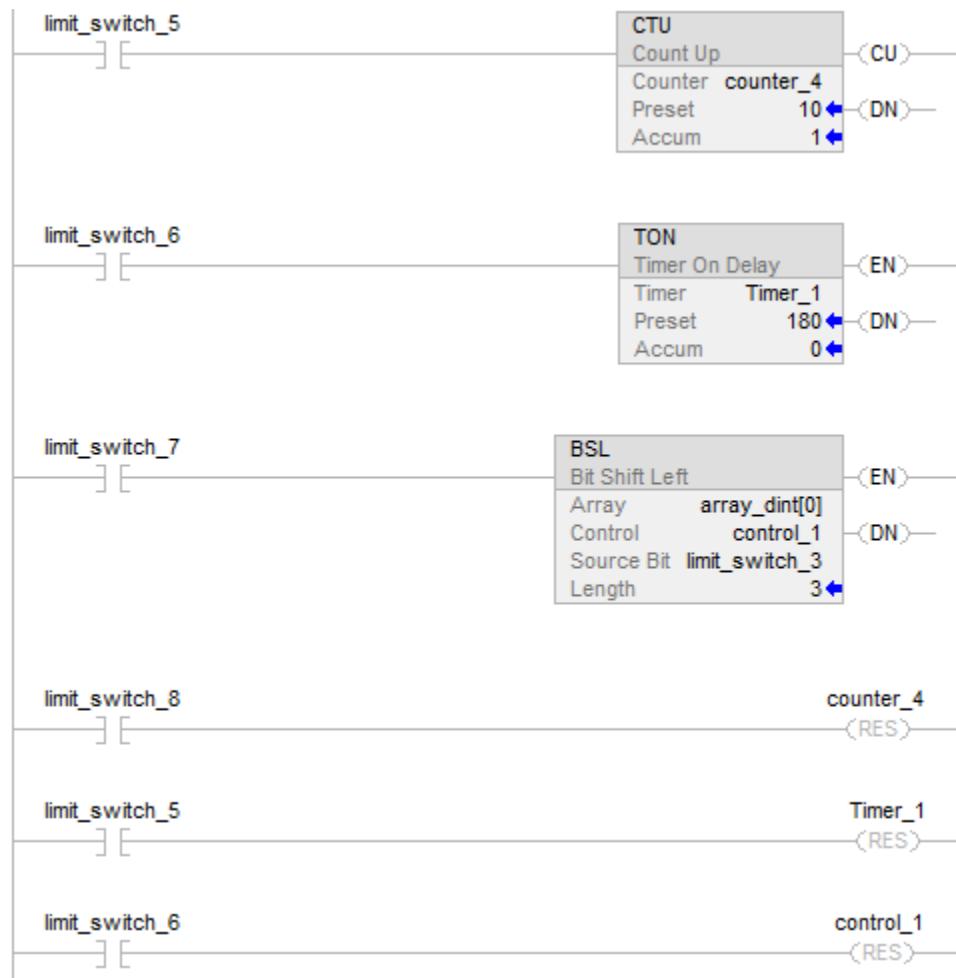
No es específico para esta instrucción. Consulte Índice a través de matrices para ver si hay fallos de indexación de matrices.

**Ejecución****Diagrama de escalera**

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón.
La condición de entrada de reglón es verdadera	Establecer la condición de salida de reglón a condición de entrada de reglón. Restablecer la estructura especificada.
Post-escaneado	N/A

## Ejemplo

### Diagrama de escalera



### Ejemplo de restablecimiento

En el siguiente ejemplo:

cuando limit\_switch\_8 está habilitado, se restablece counter\_4

cuando limit\_switch\_5 está habilitado, se restablece Timer\_1

cuando limit\_switch\_6 está habilitado, se restablece control\_1

### Consulte también

[Instrucción del contador](#) en la página 171

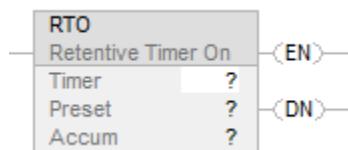
[Índice a través de matrices](#) en la página 936

**Temporizador retentivo activado (RTO)** Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción RTO es un temporizador retentivo que acumula tiempo cuando la instrucción está activada.

### Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

### Operandos

**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
- Los miembros de un operando de estructura se sobrescriben.
- A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.

### Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción
Timer	TIMER	etiqueta	Estructura de temporizador
Preset	DINT	inmediato	Valor de Timer.PRE.
Accum	DINT	inmediato	Valor de Timer.ACC.

### Estructura de TIMER

Mnemónico	Tipo de datos	Descripción
.EN	BOOL	El bit de habilitación contiene una condición de entrada de reglón de cuando la instrucción se ejecutó por última vez.
.TT	BOOL	Cuando el bit de temporización se establece, indica que la operación de conteo está en proceso.
.DN	BOOL	Cuando el bit de efectuado se establece, indica que la operación de temporización se ha completado (o pausado).
.PRE	DINT	El valor preestablecido indica el valor (en unidades de 1 milisegundo) al que debe llegar el valor acumulado antes de que la instrucción indique que está finalizada.
.ACC	DINT	El valor acumulado indica el número de milisegundos que han pasado desde que se activó la instrucción RTO.

### Descripción

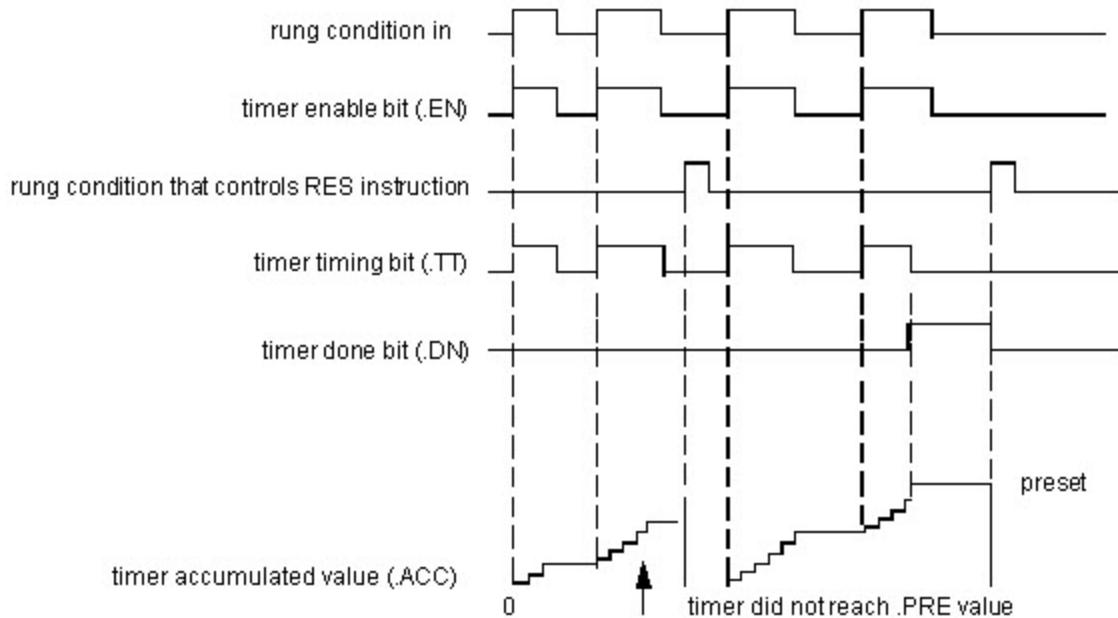
La instrucción RTO acumula tiempo hasta que:

- Se deshabilita el temporizador.
- El temporizador se completa.

La base de tiempo es siempre 1 milisegundo. Por ejemplo, para poner un temporizador de 2 segundos, hay que introducir 2000 en el valor .PRE.

El temporizador establece el bit .DN en verdadero cuando el temporizador se completa.

Cuando está activado, el temporizador se puede pausar al establecer el bit .DN en verdadero y se reanuda al borrar y establecer el bit .DN en falso.



### Cómo se ejecuta un temporizador

Un temporizador funciona restando el tiempo del último escaneado desde el tiempo actual:

$$\text{ACC} = \text{ACC} + (\text{current\_time} - \text{last\_time\_scanned})$$

Tras actualizar el ACC, el temporizador establece `last_time_scanned = current_time`. Esto hace que el temporizador se prepare para el siguiente escaneado.

### Afectar a las marcas de estado matemático

No

### Fallos mayores/menores

Se producirá un fallo mayor si:	Tipo de fallo	Código de fallo
<code>.PRE &lt; 0</code>	4	34
<code>.ACC &lt; 0</code>	4	34

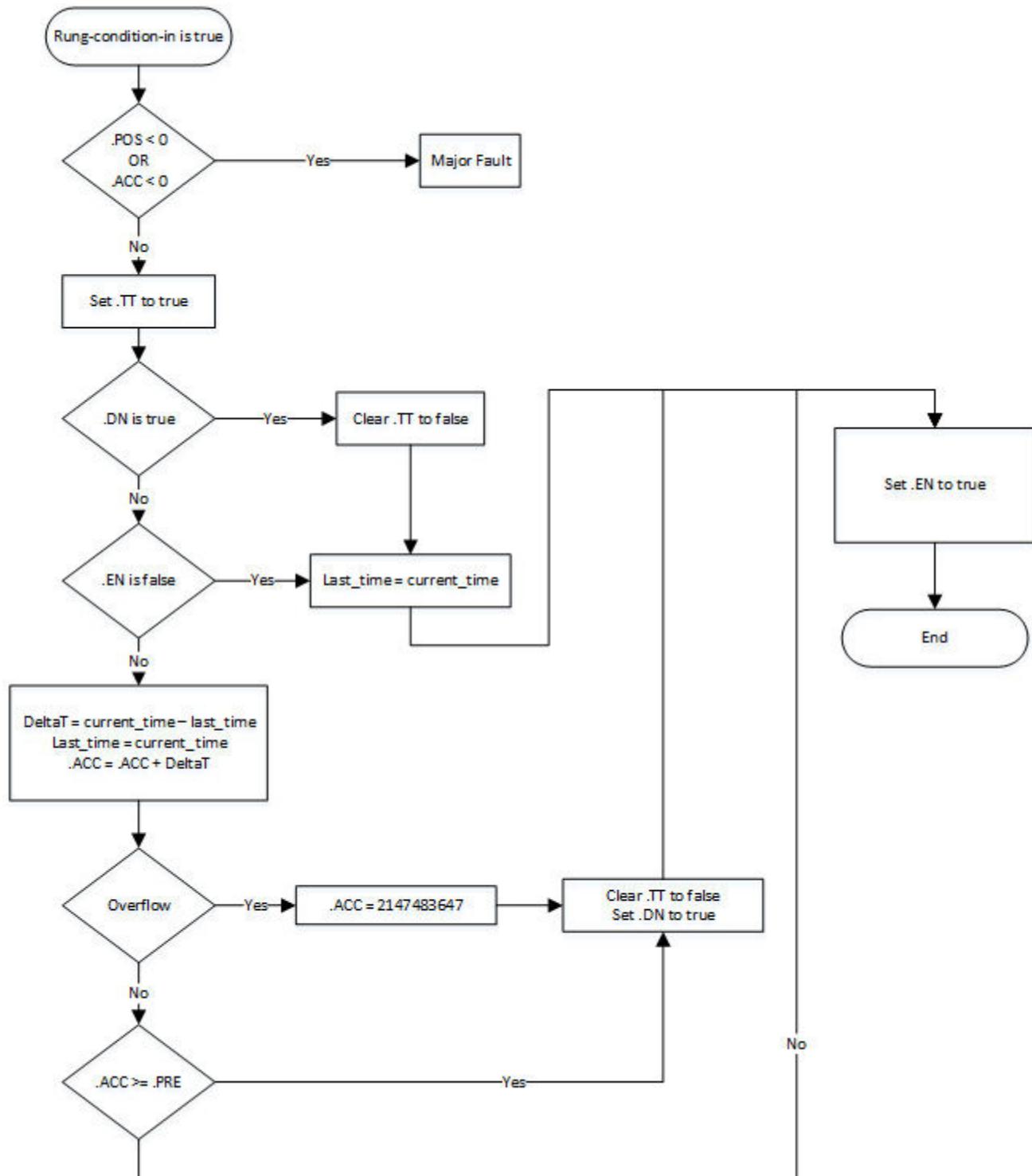
Consulte Índice a través de matrices para ver si hay fallos de indexación de matrices.

## Ejecución

### Diagrama de escalera

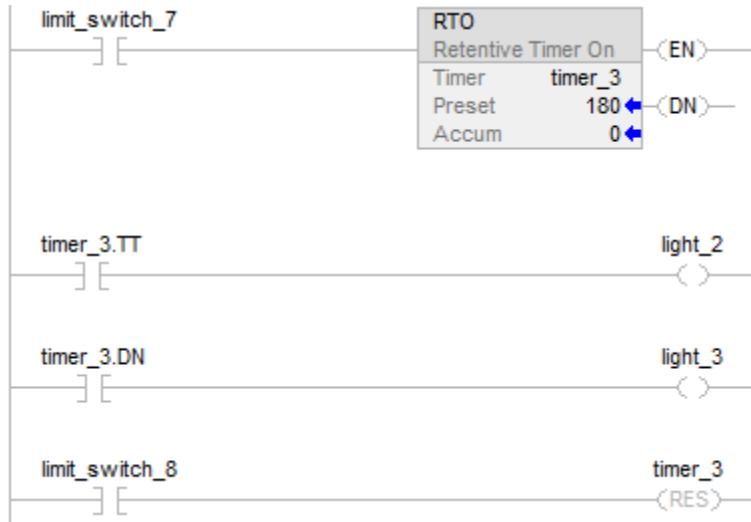
Condición/estado	Acción realizada
Pre-escaneado	El bit .EN se borra a falso. El bit .TT se borra a falso.
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón El bit .EN se borra a falso. El bit .TT se borra a falso.
La condición de entrada de reglón es verdadera	Establecer la condición de salida de reglón a condición de entrada de reglón Consultar el diagrama de flujo RTO (Verdadero)
Post-escaneado	N/A

Diagrama de flujo RTO (Verdadero)



## Ejemplo

### Diagrama de escalera



Cuando se establece limit\_switch\_7, se enciende light\_2 durante 180 milisegundos (timer\_3 cuenta el tiempo). Cuando timer\_3.acc llega a los 180, se apaga light\_2 y se enciende light\_3. Light\_3 permanece activado hasta que se restablece timer\_3. Si se borra limit\_switch\_7 mientras timer\_3 cuenta, se apaga light\_2. Cuando se establece limit\_switch\_7, la instrucción RES restablece timer\_3 (borra los bits de estado y el valor .ACC).

### Consulte también

[Índice a través de matrices](#) en la página 936

[Restablecer \(RES\)](#) en la página 187

[Temporizador de retardo a la desconexión \(TOF\)](#) en la página 201

[Temporizador de retardo a la conexión \(TON\)](#) en la página 211

[Instrucciones de temporizador y contador](#) en la página 171

## Temporizador retentivo activado con restablecimiento (RTOR)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

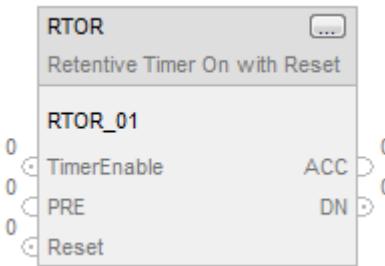
La instrucción RTOR es un temporizador retentivo que acumula tiempo cuando TimerEnable está establecido.

## Idiomas disponibles

### Diagrama de escalera

Esta instrucción no está disponible en el diagrama de escalera.

### Bloque de funciones



### Texto estructurado

RTOR(RTOR\_tag)

### Operандos

#### Texto estructurado

Variable	Tipo	Formato	Descripción
RTOR tag	FBD_TIMER	Estructura	Estructura de RTOR

Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

### Bloque de funciones

Operando	Tipo	Formato	Descripción
RTOR tag	FBD_TIMER	Estructura	Estructura de RTOR

### Estructura de FBD\_TIMER

Parámetro de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Si no está activada, la instrucción no se ejecuta y las salidas no se actualizan. Si está activado, la instrucción se ejecuta. Está establecido de forma predeterminada.

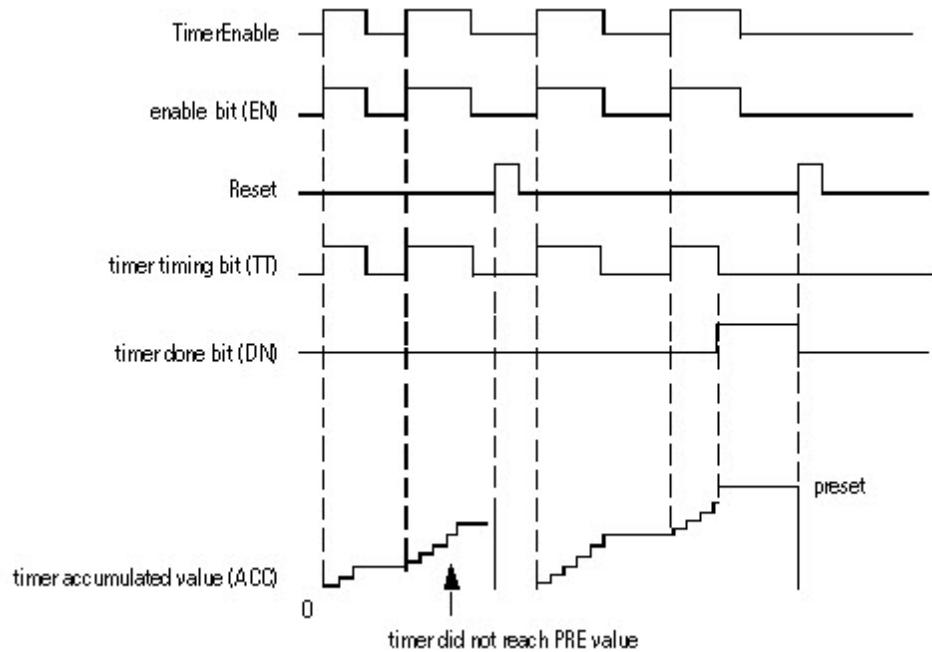
TimerEnable	BOOL	Si se establece, esto habilita el temporizador a funcionar y acumular tiempo. Está borrado de forma predeterminada.
PRE	DINT	Valor preestablecido del temporizador. Es el valor en unidades de 1 msec al que ACC debe llegar antes de que acabe el temporizador. Si no es válido, la instrucción establece el bit correspondiente en Status. y el temporizador no se ejecuta. Válido = de 0 al número entero positivo máximo
Reset	BOOL	Petición para restablecer el temporizador. Cuando se establece, se restablece el temporizador. Cuando se establece el parámetro de entrada Reset, la instrucción borra EN, TT y DN, y establece ACC = 0.

Parámetro de salida	Tipo de datos	Descripción
EnableOut	BOOL	La instrucción ha producido un resultado válido.
ACC	DINT	Tiempo acumulado en milisegundos. Este valor se conserva aunque la entrada TimerEnable se borra.
EN	BOOL	Salida del temporizador habilitada. Indica que la instrucción del temporizador está habilitada.
TT	BOOL	Salida de temporización del temporizador. Cuando se establece, una operación de temporización está en proceso.
DN	BOOL	Salida de temporización finalizada. Indica cuándo el tiempo acumulado es mayor o igual al preestablecido.
Status	DINT	Estado del bloque de funciones.
InstructFault (Status.0)	BOOL	La instrucción detectó uno de los siguientes errores de ejecución. No se trata de un error mayor o menor del controlador. Se comprueba los bits de estado restantes para determinar lo que ha ocurrido.
PresetInv (Status.1)	BOOL	El valor preestablecido no es válido.

### Descripción

La instrucción RTOR acumula tiempo hasta que sea falso. Cuando la instrucción RTOR es falso, mantiene su valor ACC. Debe borrar el valor .ACC con la entrada Reset.

La base de tiempo es siempre 1 msec. Por ejemplo, para poner un temporizador de 2 segundos, hay que introducir 2000 en el valor PRE.



Establece el parámetro de entrada Reset para restablecer la instrucción. Si TimerEnable está establecido cuando Reset está establecido, a instrucción RTOR comenzará una temporización de nuevo cuando Reset se borre.

### Cómo se ejecuta un temporizador

Un temporizador funciona restando el tiempo del último escaneado desde el tiempo actual:

- $ACC = ACC + (current\_time - last\_time\_scanned)$
- Tras actualizar el ACC, el temporizador establece  $last\_time\_scanned = current\_time$ . Esto hace que el temporizador se prepare para el siguiente escaneado.

**Importante:** Asegúrese de escanear el temporizador al menos una vez cada 69 minutos mientras esté en marcha. En caso contrario, el valor ACC no será correcto.

El valor de `last_time_scanned` alcanza hasta los 69 minutos. El cálculo del temporizador se reinvierte si no se escanea el temporizador en 69 minutos. El valor ACC no será correcto si esto ocurre.

Cuando el temporizador esté en marcha, hay que escanearlo en 69 minutos si lo ha puesto en:

- Subrutina

- Sección de código que está entre las instrucciones JMP y LBL
- Diagrama de funciones secuenciales (SFC)
- Evento o tarea periódica
- Estado de rutina de una fase

#### Afectar a las marcas de estado matemático

No

#### Fallos mayores/menores

No es específico para esta instrucción. Consulte los *Atributos comunes* para fallos relacionados con el operando.

#### Ejecución

##### Bloque de funciones

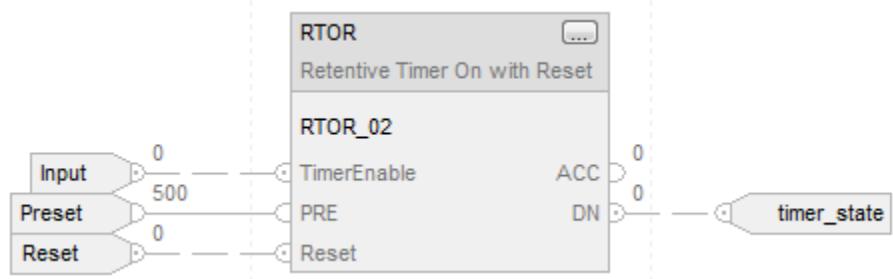
Condición/estado	Acción realizada
Pre-escaneado	Los bits EnableIn y EnableOut se borran a falsos.
Tag.EnableIn es falso	Los bits EnableIn y EnableOut se borran a falsos.
Tag.EnableIn es verdadero	Los bits EnableIn y EnableOut se establecen en verdaderos. La instrucción se ejecuta. Cuando se establece el parámetro de entrada Reset, la instrucción borra EN, TT y DN, y establece ACC = 0.
Primera ejecución de instrucción	EN, TT y DN se borran a falso. La instrucción se ejecuta.
Primer escaneado de instrucción	N/A
Post-escaneado	EnableIn y EnableOut se borran a falsos.

##### Texto estructurado

Condición/estado	Acción realizada
Pre-escaneado	Consultar Pre-escaneado en la tabla Bloque de funciones.
Ejecución normal	Consultar Tag.EnableIn es verdadero en la tabla Bloque de funciones.
Post-escaneado	Consultar Post-escaneado en la tabla Bloque de funciones.

## Ejemplo

### Bloque de funciones



### Texto estructurado

```
RTOR_01.PRE := 500;  
RTOR_01.Reset := Reset;  
RTOR_01.TimerEnable := Input;  
RTOR(RTOR_01);  
timer_state := RTOR_01.DN;
```

### Consulte también

[Atributos comunes](#) en la página 923

[Temporizador retentivo activado \(RTO\)](#) en la página 191

[Restablecer \(RES\)](#) en la página 187

[Sintaxis de texto estructurado](#) en la página 955

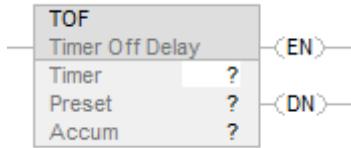
## Temporizador de retardo a la desconexión (TOF)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción TOF es un temporizador no retentivo que acumula tiempo cuando la instrucción está activada (la condición de entrada de reglón es falso).

### Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

### Operандos

**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
- Los miembros de un operando de estructura se sobrescriben.
- A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.

### Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción
Timer	TIMER	etiqueta	Estructura de temporizador
Preset	DINT	inmediato	Valor de Timer.PRE.
Accum	DINT	inmediato	Valor de Timer.ACC.

### Estructura de TIMER

Mnemónico	Tipo de datos	Descripción
.EN	BOOL	El bit de habilitación contiene una condición de entrada de reglón de cuando la instrucción se ejecutó por última vez.
.TT	BOOL	Cuando el bit de temporización se establece, indica que la operación de conteo está en proceso.
.DN	BOOL	Cuando el bit de efectuado está borrado, indica que la operación de temporización se ha completado (o pausado).

.PRE	DINT	El valor preestablecido indica el valor (en unidades de 1 milisegundo) al que debe llegar el valor acumulado antes de que la instrucción indique que está finalizada.
.ACC	DINT	El valor acumulado indica el número de milisegundos que han pasado desde que se activó la instrucción TOF.

### Descripción

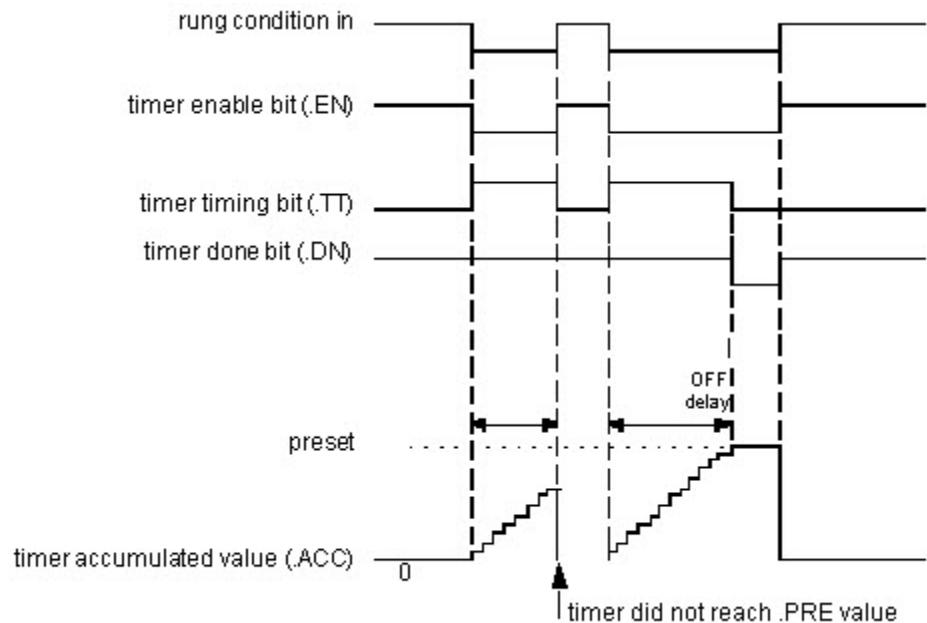
La instrucción TOF acumula tiempo hasta que:

- Se deshabilita el temporizador
- El temporizador se completa

La base de tiempo es siempre 1 milisegundo. Por ejemplo, para poner un temporizador de 2 segundos, hay que introducir 2000 en el valor .PRE.

El temporizador borrará el bit .DN a falso cuando el temporizador se completa.

Cuando está activado, el temporizador se puede pausar al borrar el bit .DN a falso y se reanuda al establecer el bit .DN en verdadero.



### Cómo se ejecuta un temporizador

Un temporizador funciona restando el tiempo del último escaneado desde el tiempo actual:

$$\text{ACC} = \text{ACC} + (\text{current\_time} - \text{last\_time\_scanned})$$

Tras actualizar el ACC, el temporizador establece last\_time\_scanned = current\_time. Esto hace que el temporizador se prepare para el siguiente escaneado.

#### Afectar a las marcas de estado matemático

No

#### Fallos mayores/menores

Se producirá un fallo mayor si:	Tipo de fallo	Código de fallo
.PRE < 0	4	34
.ACC < 0	4	34

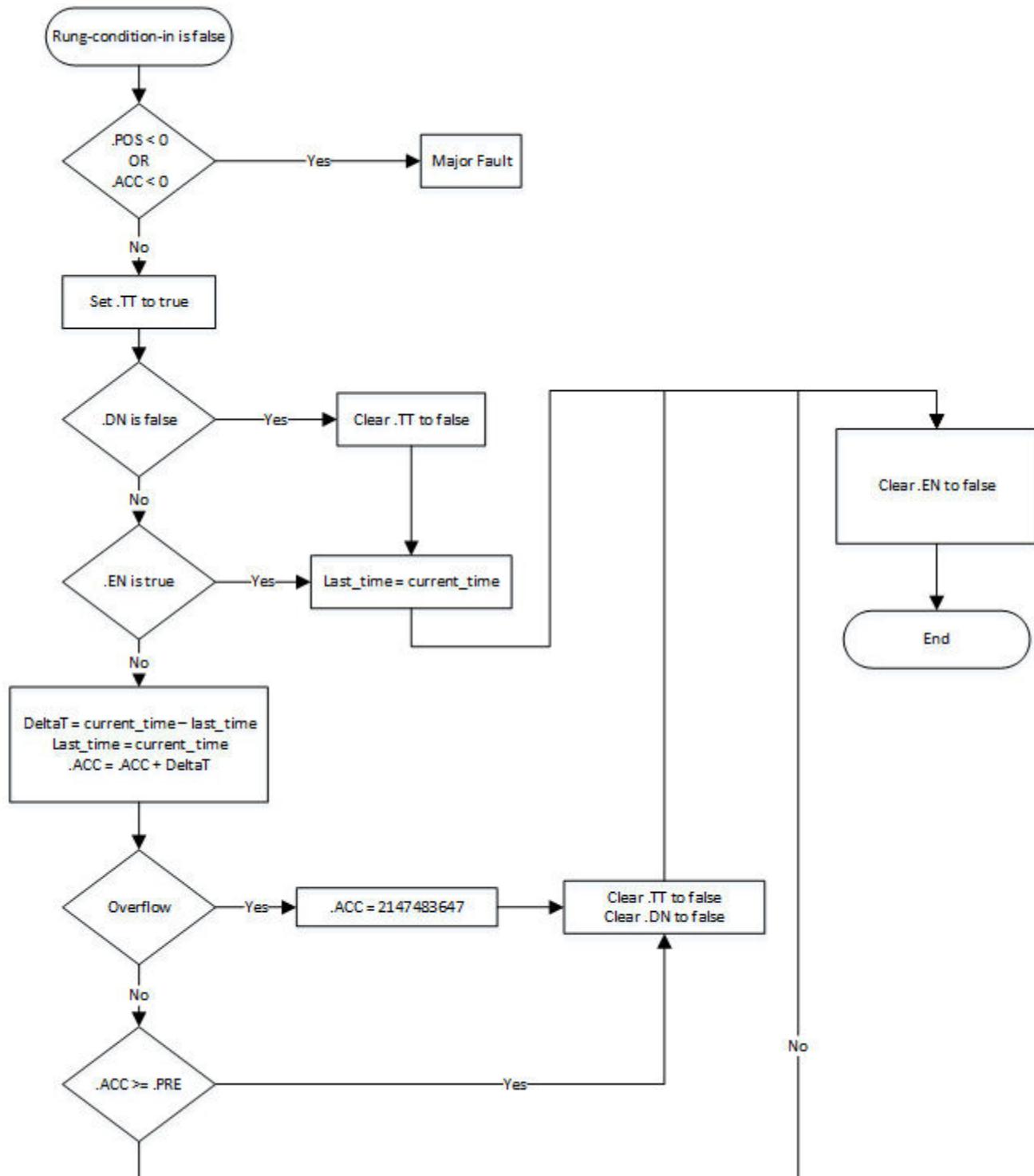
Consulte *Índice a través de matrices* para ver si hay fallos de indexación de matrices.

#### Ejecución

##### Diagrama de escalera

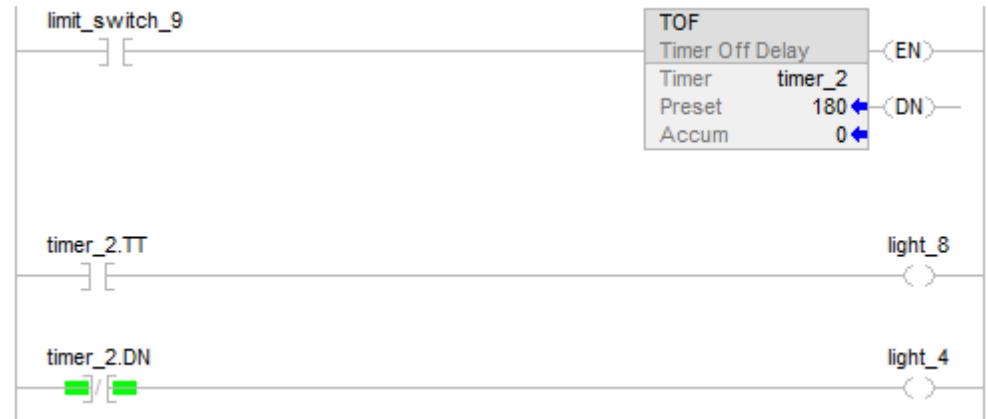
Condición/estado	Acción realizada
Pre-escaneado	El bit .EN se borra a falso. El bit .TT se borra a falso. El bit .DN se borra a falso. El valor .ACC se establece para ser igual el valor .PRE.
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón Consulte el diagrama de flujo TOF (Falso).
La condición de entrada de reglón es verdadera	Establecer la condición de salida de reglón a condición de entrada de reglón El bit .EN se establece en verdadero. El bit .TT se borra a falso. El bit .DN se establece en verdadero. El valor .ACC se borra a 0.
Post-escaneado	El bit .EN se borra a falso. El bit .TT se borra a falso. El bit .DN se borra a falso. El valor .ACC se establece para ser igual el valor .PRE.

Diagrama de flujo TOF (Falso)



## Ejemplo

### Diagrama de escalera



Cuando se borra limit\_switch\_9, se enciende light\_8 durante 180 milisegundos (timer\_2 cuenta el tiempo). Cuando timer\_2.acc llega a los 180, se apaga light\_8 y se enciende light\_4. Light\_4 permanece encendida hasta que se habilita la instrucción TOF. Si se establece en verdadero limit\_switch\_9 mientras timer\_2 cuenta, se apaga light\_8.

### Consulte también

[Instrucciones de temporizador y contador](#) en la página 171

[Índice a través de matrices](#) en la página 936

## Temporizador de retardo a la desconexión con restablecimiento (TOFR)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

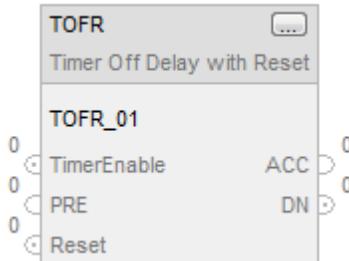
La instrucción TOFR es un temporizador no retentivo que acumula tiempo cuando TimerEnable está borrado.

### Idiomas disponibles

### Diagrama de escalera

Esta instrucción no está disponible en el diagrama de escalera.

## Bloque de funciones



### Texto estructurado

TOFR(TOFR\_tag)

### Operando

### Texto estructurado

Variable	Tipo	Formato	Descripción
TOFR tag	FBD_TIMER	Estructura	Estructura de TOFR

Consulte Sintaxis de texto estructurado para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

## Bloque de funciones

Operando	Tipo	Formato	Descripción
TOFR tag	FBD_TIMER	Estructura	Estructura de TOFR

### Estructura de FBD\_TIMER

Parámetros de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Si no está activada, la instrucción no se ejecuta y las salidas no se actualizan. Si está activado, la instrucción se ejecuta. Está establecido de forma predeterminada.
TimerEnable	BOOL	Si se borra, esto habilita el temporizador a funcionar y acumular tiempo. Está borrado de forma predeterminada.

PRE	DINT	Valor preestablecido del temporizador. Es el valor en unidades de 1 msec al que ACC debe llegar antes de que acabe el temporizador. Si no es válido, la instrucción establece el bit correspondiente en Status. y el temporizador no se ejecuta. Válido = de 0 al número entero positivo máximo
Reset	BOOL	Petición para restablecer el temporizador. Cuando se establece, restablece el temporizador. Está borrado de forma predeterminada. Cuando se activa el parámetro de entrada Reset, la instrucción borra EN, TT y DN, y establece ACC = PRE. Nota: esto es diferente a usar una instrucción RES en una instrucción TOF.

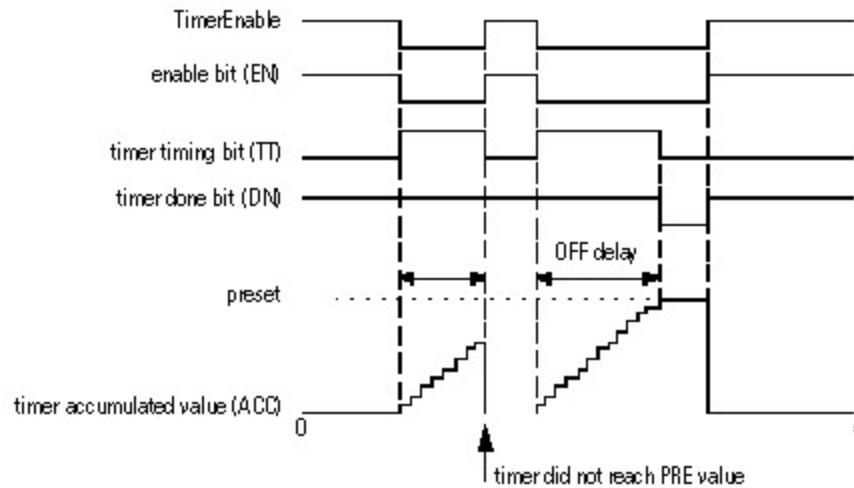
Parámetros de salida	Tipo de datos	Descripción
EnableOut	BOOL	La instrucción ha producido un resultado válido.
ACC	BOOL	Tiempo acumulado en milisegundos.
EN	BOOL	Salida del temporizador habilitada Indica que la instrucción del temporizador está habilitada.
TT	BOOL	Salida de temporización del temporizador. Cuando se establece, una operación de temporización está en proceso.
DN	BOOL	Salida de temporización finalizada. Indica cuándo el tiempo acumulado es mayor o igual al preestablecido.
Status	DINT	Estado del bloque de funciones.
InstructFault (Status.0)	BOOL	La instrucción detectó uno de los siguientes errores de ejecución. No se trata de un error mayor o menor del controlador. Se comprueba los bits de estado restantes para determinar lo que ha ocurrido.
PresetInv (Status.1)	BOOL	El valor preestablecido no es válido.

### Descripción

Cuando sea verdadero, la instrucción TOFR acumula tiempo hasta que:

- La instrucción TOFR está deshabilitada.
- ACC  $\geq$  PR E

La base de tiempo es siempre 1 msec. Por ejemplo, para poner un temporizador de 2 segundos, hay que introducir 2000 en el valor PRE.



Establece el parámetro de entrada Reset para restablecer la instrucción. Si TimerEnable es falso cuando Reset es verdadero, la instrucción TOFR no comenzará una temporización de nuevo cuando Reset se establezca en falso.

### Cómo se ejecuta un temporizador

Un temporizador funciona restando el tiempo del último escaneado desde el tiempo actual:

$$\text{ACC} = \text{ACC} + (\text{current\_time} - \text{last\_time\_scanned})$$

Tras actualizar el ACC, el temporizador establece last\_time\_scanned = current\_time. Esto hace que el temporizador se prepare para el siguiente escaneado.

---

**Importante:** Asegúrese de escanear el temporizador al menos una vez cada 69 minutos mientras esté en marcha. En caso contrario, el valor ACC no será correcto.

---

El valor de last\_time\_scanned alcanza hasta los 69 minutos. El cálculo del temporizador se reinvierte si no se escanea el temporizador en 69 minutos. El valor ACC no será correcto si esto ocurre.

Cuando el temporizador esté en marcha, hay que escanearlo en 69 minutos si lo ha puesto en:

- Subrutina

- Sección de código que está entre las instrucciones JMP y LBL
- Diagrama de funciones secuenciales (SFC)
- Evento o tarea periódica
- Estado de rutina de una fase

#### Afectar a las marcas de estado matemático

No

#### Fallos mayores/menores

No es específico para esta instrucción. Consulte los *Atributos comunes* para fallos relacionados con el operando.

#### Ejecución

##### Bloque de funciones

Condición/estado	Acción realizada
Pre-escaneado	Los bits EnableIn y EnableOut se borran a falsos.
Tag. EnableIn es falso	Los bits EnableIn y EnableOut se borran a falsos.
Tag. EnableIn es verdadero	Los bits EnableIn y EnableOut se establecen en verdaderos. El algoritmo principal de la instrucción se ejecutará y las salidas se actualizarán.
Primera ejecución de instrucción	N/A
Primer escaneado de instrucción	Se borran EN, TT y DN. No se modifica el valor ACC.
Post-escaneado	Los bits EnableIn y EnableOut se borran a falsos.

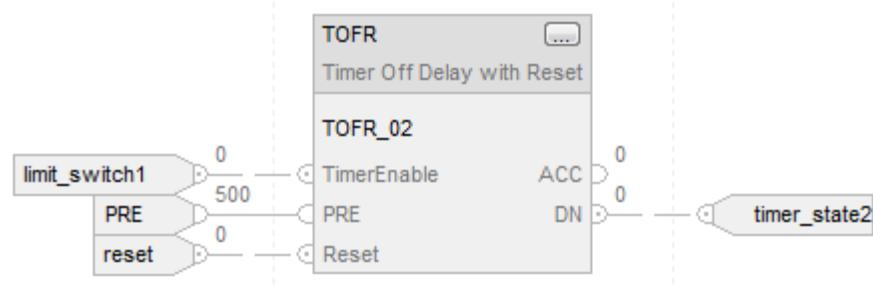
##### Texto estructurado

Condición/estado	Acción realizada
Pre-escaneado	Consultar Pre-escaneado en la tabla Bloque de funciones.
Ejecución normal	Consultar Tag.EnableIn es verdadero en la tabla Bloque de funciones.
Post-escaneado	Consultar Post-escaneado en la tabla Bloque de funciones.

## Ejemplo

En cada escaneado que se borra limit\_switch1, la instrucción TOFR aumenta el valor ACC según el tiempo que haya pasado hasta que el valor ACC alcance el valor PRE. Cuando ACC  $\geq$  PRE, se borra el parámetro DN y se establece timer\_state2.

## Bloque de funciones



## Texto estructurado

```

TOFR_01.PRE := 500;
TOFR_01.Reset := Reset;
TOFR_01.TimerEnable := Input;
TOFR(TOFR_01);
timer_state := TOFR_01.DN;

```

## Consulte también

[Atributos comunes](#) en la página 923

[Sintaxis de texto estructurado](#) en la página 955

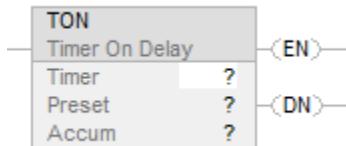
## Temporizador de retardo a la conexión (TON)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción TON es un temporizador no retentivo que acumula tiempo cuando la instrucción está activada.

## Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

### Operandos

**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
- Los miembros de un operando de estructura se sobrescriben.
- A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.

### Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción
Timer	TIMER	etiqueta	Estructura de temporizador
Preset	DINT	inmediato	Valor de Timer.PRE.
Accum	DINT	inmediato	Valor de Timer.ACC.

### Estructura de TIMER

Mnemónico	Tipo de datos	Descripción
.EN	BOOL	El bit de habilitación contiene una condición de entrada de reglón de cuando la instrucción se ejecutó por última vez.
.TT	BOOL	Cuando el bit de temporización se establece, indica que la operación de conteo está en proceso.
.DN	BOOL	Cuando el bit de efectuado se establece, indica que la operación de temporización se ha completado (o pausado).

.PRE	DINT	El valor preestablecido indica el valor (en unidades de 1 milisecondo) al que debe llegar el valor acumulado antes de que la instrucción indique que está finalizada.
.ACC	DINT	El valor acumulado indica el número de milisegundos que han pasado desde que se activó la instrucción TON.

### Descripción

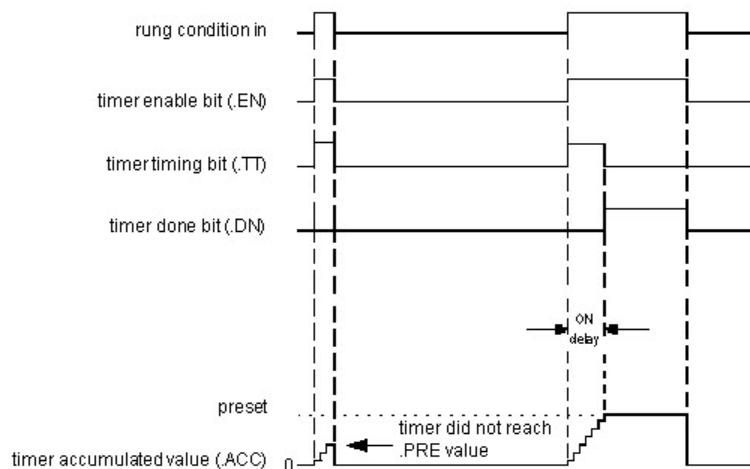
La instrucción TON acumula tiempo desde el momento en el que está habilitada hasta:

- Se deshabilita el temporizador
- El temporizador se completa

La base de tiempo es siempre 1 milisecondo. Por ejemplo, para poner un temporizador de 2 segundos, hay que introducir 2000 en el valor .PRE.

El temporizador establece el bit .DN en verdadero cuando el temporizador se completa.

Cuando está activado, el temporizador se puede pausar al establecer el bit .DN en verdadero y se reanuda al borrar y establecer el bit .DN en falso.



### Cómo se ejecuta un temporizador

Un temporizador funciona restando el tiempo del último escaneado desde el tiempo actual:

$$\text{ACC} = \text{ACC} + (\text{current\_time} - \text{last\_time\_scanned})$$

Tras actualizar el ACC, el temporizador establece last\_time\_scanned = current\_time. Esto hace que el temporizador se prepare para el siguiente escaneado.

#### Afectar a las marcas de estado matemático

No

#### Fallos mayores/menores

Se producirá un fallo mayor si:	Tipo de fallo	Código de fallo
.PRE < 0	4	34
.ACC < 0	4	34

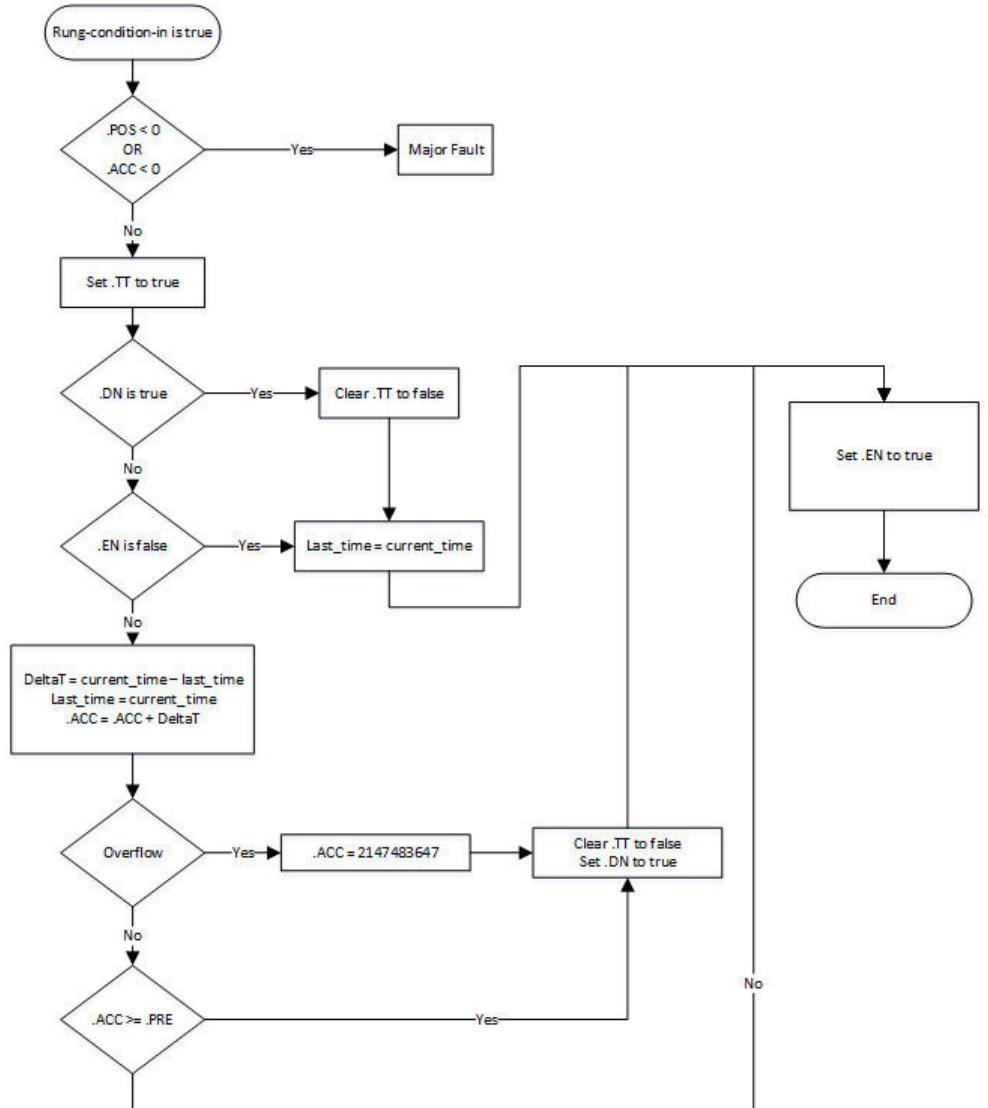
Consulte Índice a través de matrices para ver si hay fallos de indexación de matrices.

#### Ejecución

#### Diagrama de escalera

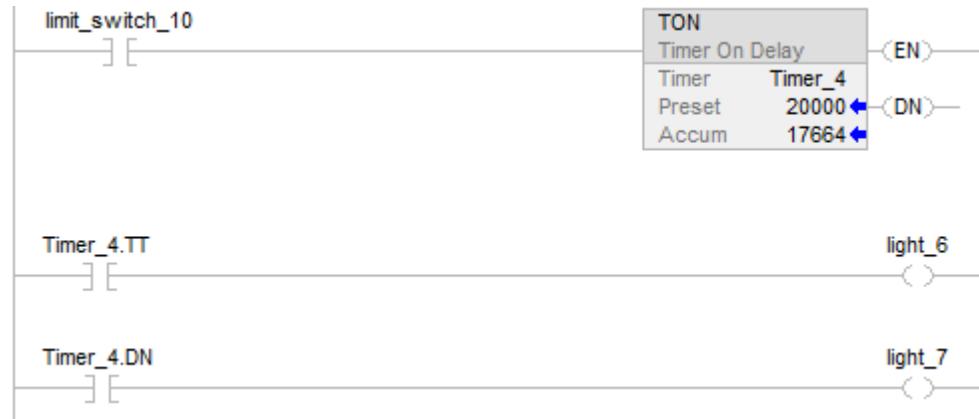
Condición/estado	Acción realizada
Pre-escaneado	El bit .EN se borra a falso. El bit .TT se borra a falso. El bit .DN se borra a falso. El valor .ACC se borra a 0.
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón El bit .EN se borra a falso. El bit .TT se borra a falso. El bit .DN se borra a falso. El valor .ACC se borra a 0.
La condición de entrada de reglón es verdadera	Establecer la condición de salida de reglón a condición de entrada de reglón Consultar el Diagrama de flujo TON (Verdadero)
Post-escaneado	El bit .EN se borra a falso. El bit .TT se borra a falso. El bit .DN se borra a falso. El valor .ACC se borra a 0.

## Diagrama de flujo TON (Verdadero)



## Ejemplo

### Diagrama de escalera



Cuando limit\_switch\_10 se establece en verdadero, se enciende light\_6 durante 20.000 milisegundos (Timer\_4 cuenta el tiempo). Cuando Timer\_4.acc llega a los 20.000, se apaga light\_6 y se enciende light\_7. Si se borra limit\_switch\_10 a falso mientras Timer\_4 cuenta, se apaga light\_6. Cuando se borra limit\_switch\_10 a falso, se restablecen los bits de estado de Timer\_4 y el valor de .ACC.

### Consulte también

[Instrucciones del contador](#) en la página 171

[Índice a través de matrices](#) en la página 936

## Temporizador de retardo a la conexión con restablecimiento (TONR)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

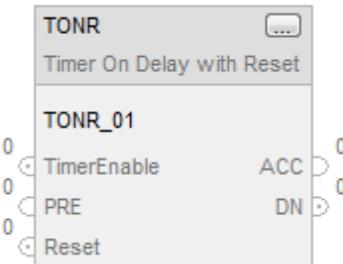
La instrucción TONR es un temporizador no retentivo que acumula tiempo cuando TimerEnable está establecido.

### Idiomas disponibles

### Diagrama de escalera

Esta instrucción no está disponible en el diagrama de escalera.

## Bloque de funciones



### Texto estructurado

```
TONR(TONR_tag);
```

### Operандos

#### Texto estructurado

Operando	Tipo	Formato	Descripción
TONR tag	FBD_TIMER	Estructura	Estructura de TONR

Consulte Sintaxis de texto estructurado para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

## Bloque de funciones

Operando	Tipo	Formato	Descripción
TONR tag	FBD_TIMER	Estructura	Estructura de TONR

### Estructura de FBD\_TIMER

Parámetro de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Si no está activada, la instrucción no se ejecuta y las salidas no se actualizan. Si está activado, la instrucción se ejecuta. Está establecido de forma predeterminada.
TimerEnable	BOOL	Si se establece, esto habilita el temporizador a funcionar y acumular tiempo. Está borrado de forma predeterminada.

PRE	DINT	Valor preestablecido del temporizador. Es el valor en unidades de 1 msec al que ACC debe llegar antes de que acabe el temporizador. Si no es válido, la instrucción establece el bit correspondiente en Status. y el temporizador no se ejecuta. Válido = de 0 al número entero positivo máximo
Reset	BOOL	Petición para restablecer el temporizador. Cuando se establece, restablece el temporizador. Está borrado de forma predeterminada. Cuando se establece el parámetro de entrada Reset, la instrucción borra EN, TT y DN, y establece ACC = 0.

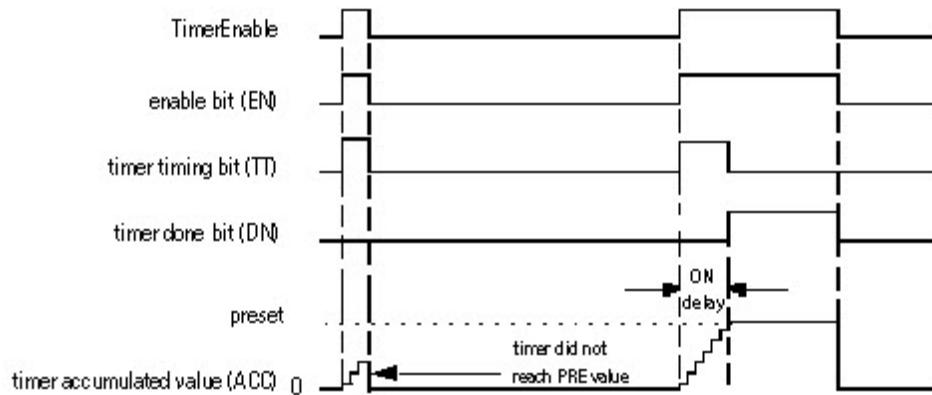
Parámetro de salida	Tipo de datos	Descripción
EnableOut	BOOL	La instrucción ha producido un resultado válido.
ACC	BOOL	Tiempo acumulado en milisegundos.
ENF	BOOL	Salida del temporizador habilitada Indica que la instrucción del temporizador está habilitada.
TT	BOOL	Salida de temporización del temporizador. Cuando se establece, una operación de temporización está en proceso.
DN	BOOL	Salida de temporización finalizada. Indica cuándo el tiempo acumulado es mayor o igual al preestablecido.
Status	DINT	Estado del bloque de funciones.
InstructFault (Status.0)	BOOL	La instrucción detectó uno de los siguientes errores de ejecución. No se trata de un error mayor o menor del controlador. Se comprueba los bits de estado restantes para determinar lo que ha ocurrido.
PresetInv (Status.1)	BOOL	El valor preestablecido no es válido.

### Descripción

Cuando sea verdadero, la instrucción TONR acumula tiempo hasta que:

- La instrucción TONR está deshabilitada.
- ACC  $\geq$  PR E

La base de tiempo es siempre 1 msec. Por ejemplo, para poner un temporizador de 2 segundos, hay que introducir 2000 en el valor PRE.



Establece el parámetro de entrada Reset para restablecer la instrucción. Si TimerEnable está establecido cuando Reset es verdadero, la instrucción TONR no comenzará una temporización de nuevo cuando Reset es falso.

### Cómo se ejecuta un temporizador

Un temporizador funciona restando el tiempo del último escaneado desde el tiempo actual:

- $ACC = ACC + (current\_time - last\_time\_scanned)$

Tras actualizar el ACC, el temporizador establece  $last\_time\_scanned = current\_time$ . Esto hace que el temporizador se prepare para el siguiente escaneado.

---

**Importante:** Asegúrese de escanear el temporizador al menos una vez cada 69 minutos mientras esté en marcha. En caso contrario, el valor ACC no será correcto.

---

El valor de  $last\_time\_scanned$  alcanza hasta los 69 minutos. El cálculo del temporizador se reinvierte si no se escanea el temporizador en 69 minutos. El valor ACC no será correcto si esto ocurre.

Cuando el temporizador esté en marcha, hay que escanearlo en 69 minutos si lo ha puesto en:

- Subrutina
- Sección de código que está entre las instrucciones JMP y LBL
- Diagrama de funciones secuenciales (SFC)
- Evento o tarea periódica
- Estado de rutina de una fase

### Afectar a las marcas de estado matemático

No

### Fallos mayores/menores

No es específico para esta instrucción. Consulte los *Atributos comunes* para fallos relacionados con el operando.

### Ejecución

#### Bloque de funciones

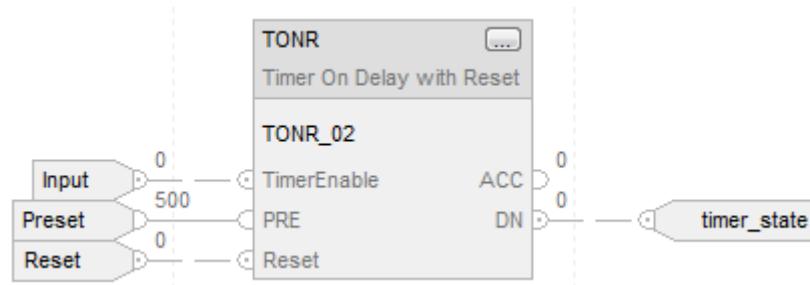
Condición/estado	Acción realizada
Pre-escaneado	Los bits EnableIn y EnableOut se borran a falsos.
Tag.EnableIn es falso	Los bits EnableIn y EnableOut se borran a falsos.
Tag.EnableIn es verdadero	Los bits EnableIn y EnableOut se establecen en verdaderos. El algoritmo principal de la instrucción se ejecuta y las salidas se actualizan.
Primera ejecución de instrucción	N/A
Primer escaneado de instrucción	Se borran EN, TT y DN. El valor ACC se establece en 0.
Post-escaneado	Los bits EnableIn y EnableOut se borran a falsos.

#### Texto estructurado

Condición/estado	Acción realizada
Pre-escaneado	Consultar Pre-escaneado en la tabla Bloque de funciones.
Ejecución normal	Consultar Tag.EnableIn es verdadero en la tabla Bloque de funciones.
Post-escaneado	Consultar Post-escaneado en la tabla Bloque de funciones.

### Ejemplo

#### Bloque de funciones



**Texto estructurado**

```
TONR_01.PRE := 500;  
TONR_01.Reset := Reset;  
TONR_01.TimerEnable := Input;  
TONR(TONR_01);  
timer_state := TONR_01.DN;
```

**Consulte también**

[Atributos comunes](#) en la página 923

[Temporizador de retardo a la conexión \(TON\)](#) en la página 211

[Restablecer \(RES\)](#) en la página 187

[Sintaxis de texto estructurado](#) en la página 955



# **Entrada/salida:**

## **Instrucciones de entrada/salida**

Las instrucción de entrada/salida leen o escriben datos para o desde un controlador o un bloque de datos para o desde otro módulo en otra red.

### **Instrucciones disponibles**

#### **Diagrama de escalera y Texto estructurado**

<a href="#">MSG</a>	<a href="#">GSV</a>	<a href="#">SSV</a>	<a href="#">IOT</a>
---------------------	---------------------	---------------------	---------------------

### **Bloque de funciones**

No disponible

<b>Si desea:</b>	<b>Utilice esta instrucción:</b>
Enviar datos para o desde otro módulo	MSG
Obtener la información de estado del controlador	GSV
Activar la información de estado del controlador	SSV
Mandar valores de salida a un módulo E/S o un controlador de consumo en un punto concreto de su lógica Desencadenar una tarea de evento en otro controlador	IOT

### **Consulte también**

[Especificación de detalles de comunicación](#) en la página 253

[Especificar mensajes CIP](#) en la página 360

[Seleccionar del tipo de mensaje](#) en la página 342

[Ejemplos de configuración de MSG](#) en la página 234

[Determinar la información de la memoria del controlador](#) en la página 272

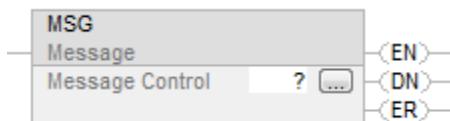
## Mensaje (MSG)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción MSG lee de manera asíncrona o escribe un bloque de datos en otro módulo de red.

### Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

### Texto estructurado

```
MSG(MessageControl);
```

### Operandos

### Diagrama de escalera

Operando	Tipo	Formato	Descripción
Message	MSG	etiqueta	Estructura de mensaje

### Texto estructurado

Operando	Tipo	Formato	Descripción
Message	MSG	etiqueta	Estructura de mensaje

Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

## Estructura de MESSAGE

**Importante:** Si comprueba los bits de estado más de una vez, use una copia de los bits si los comprueba en más de un sitio dentro de su lógica. En caso contrario, los bits podrían cambiar durante el escaneado y la lógica no funcionará como se espera.

Una manera de hacer una copia es mediante la palabra FLAGS. Copie la palabra FLAGS a otra etiqueta y compruebe los bits en esta copia.

**Importante:** No cambie los siguientes bits de una instrucción MSG:

- DN
- EN
- ER
- EW
- ST

No cambie estos bits por separado ni como parte de la palabra FLAGS. Si lo hace, puede que el controlador produzca un fallo irrecuperable. Cuando hay un fallo irrecuperable en el controlador, el controlador borra el proyecto desde su memoria.

Mnemónico	Tipo de datos	Descripción	
.FLAGS	INT	El miembro .FLAGS da acceso a los miembros de estado (bits) en una palabra de 16 bits.	
		Este bit 2	es este miembro .EW
		4	.ER
		5	.DN
		6	.ST
		7	.EN
		8	.TO
		9	.EN_CC
		<b>Importante:</b> No cambie los bits EW, ER, DN o ST del miembro FLAGS. Por ejemplo, no borre por completo la palabra FLAGS. El controlador ignora el cambio y usa los valores de los bits almacenados de manera interna.	

.ERR	INT	Si se establece el bit .ER, la palabra del código de error identifica el código de error para la instrucción MSG.
.EXERR	INT	La palabra de código de error extendido proporciona información adicional del código de error en algunos códigos de error.
.REQ_LEN	INT	La longitud requerida muestra cuántas palabras intentará transferir la instrucción de mensaje.
.DN_LEN	INT	La longitud realizada identifica cuántas palabras ya se han transferido.
.EW	BOOL	El bit de habilitación a la espera se establece cuando el controlador detecta que una petición de mensaje ha entrado en cola. El controlador restablece el bit .EW cuando se activa el bit .ST. <b>Importante:</b> No cambie el bit .EW. El controlador ignora el cambio y usa los valores del los bit almacenados de manera interna.
.ER	BOOL	El bit de error se establece cuando el controlador detecta que ha fallado una transferencia. El bit .ER se restablecerá la próxima vez que EnableIn cambie de falso a verdadero. <b>Importante:</b> No cambie el bit .ER. El controlador ignora el cambio y usa los valores del los bit almacenados de manera interna.
.DN	BOOL	El bit de efectuado se establece cuando el último paquete de mensaje se ha transferido con éxito. El bit .DN se restablecerá la próxima vez que EnableIn cambie de falso a verdadero. <b>Importante:</b> No cambie el bit .DN. El controlador ignora el cambio y usa los valores del los bit almacenados de manera interna.
.ST	BOOL	El bit de comienzo se establece cuando el controlador comienza a ejecutar la instrucción MSG. El bit .ST se restablece cuando se activa el bit .DN o el .ER. <b>Importante:</b> No cambie el bit .ST. El controlador ignora el cambio y usa los valores del los bit almacenados de manera interna.

.EN	BOOL	<p>El bit de habilitación se establece cuando EnableIn se establece en verdadero y permanece establecido hasta que se establezca el bit .DN o el .ER y EnableIn sea falso. Si EnableIn se borra y se establece en falso, pero los bits .DN y .ER se borran, el bit .EN permanece activado.</p> <p><b>Importante:</b> No cambie el bit .EN. El controlador ignora el cambio y usa los valores del los bit almacenados de manera interna.</p>
.TO	BOOL	<p>Si activa manualmente el bit .TO, el controlador dejará de procesar el mensaje y activará el bit .ER.</p>
.EN_CC	BOOL	<p>El bit de habilitación de caché determina cómo manejar la conexión MSG. Si quiere que el controlador mantenga la conexión (como cuando repite la misma instrucción MSG muchas veces), establezca el bit .EN_CC. Si ejecuta pocas veces la instrucción MSG y necesita la conexión del controlador para otras cosas, borre el bit .EN_CC.</p> <p>Las conexiones de las instrucciones MSG que se salgan del puerto serial no se almacenan en la caché aunque el bit .EN_CC esté establecido.</p>
.ERR_SRC	SINT	Muestra la ruta del error en el cuadro de diálogo Configuración de mensaje.
.Destination Link	INT	Para cambiar el vínculo de Destination de DH+ o CIP con el mensaje de ID de Source, establezca este miembro en el valor necesario.
.Destination Node	INT	Para cambiar el nodo de Destination de DH+ o CIP con el mensaje de ID de Source, establezca este miembro en el valor necesario.
.SourceLink	INT	Para cambiar el vínculo de Source de DH+ o CIP con el mensaje de ID de Source, establezca este miembro en el valor necesario.
.Class	INT	Para cambiar la clase del parámetro de un mensaje de CIP genérico, establezca este miembro en el valor necesario.
.Attribute	INT	Para cambiar el atributo del parámetro de un mensaje de CIP genérico, establezca este miembro en el valor necesario.
.Instance	DINT	Para cambiar el parámetro Instance de un mensaje de CIP genérico, se establece este miembro en el valor necesario.
.LocalIndex	DINT	Si usa un asterisco (*) para designar un número de elemento en la matriz local, LocalIndex le dará el número de elemento. Para cambiar el número de elemento, establezca este miembro en el valor necesario.

		<b>Si el mensaje:</b>	<b>La matriz local es:</b>
		Lee datos	Elemento de Destination
		Escribe datos	Elemento de Source
.Channel	SINT	Para mandar el mensaje a un canal diferente del módul 1756-DHRI0, se establece este miembro en el valor necesario. Se utiliza indistintamente el carácter ASCII A o B.	
.Rack	SINT	Para cambiar el número de bastidor de un mensaje de transferencia en bloques, se establece este miembro en el número de rack necesario (octal).	
.Group	SINT	Para cambiar el número de grupo de un mensaje de transferencia en bloques, se establece este miembro en el número de grupo necesario (octal).	
.Slot	SINT	Para cambiar el número de ranura de un mensaje de transferencia en bloques, active este miembro en el número de ranura necesario (octal).	<b>Si el mensaje muestra algo de esta red:</b>
		E/S remotas universales	Se especifica el número de ranura en: octal
		ControlNet	decimal (0-15)
.Path	STRING	Para mandar el mensaje a un controlador diferente, se establece este miembro en la nueva ruta.  Se introduce la ruta como un valor hexadecimal.  Omita los puntos (.).  Por ejemplo, para una ruta que sea 1, 0, 2, 42, 1, 3, escriba \$01\$00\$02\$2A\$01\$03.  Para examinar un dispositivo y crear automáticamente una porción o un todo de la nueva cadena, haga clic en la etiqueta de cadena y elija Ir al editor de ruta de mensaje (Go to Message Path Editor).	
.Remotelnde x	DINT	Si usa un asterisco (*) para designar un número de elemento en la matriz remota, RemotelIndex le dará el número de elemento. Para cambiar el número de elemento, establezca este miembro en el valor necesario.	<b>Si el mensaje</b> <b>Entonces la matriz remota es</b>

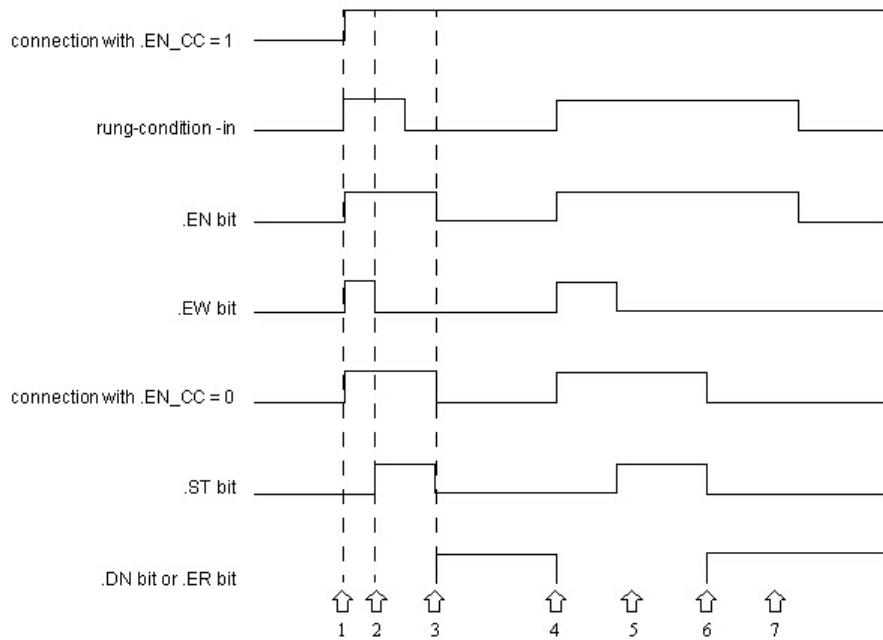
		<table border="1"> <tr><td>Lee datos</td><td>Elemento de Source</td></tr> <tr><td>Escribe datos</td><td>Elemento de Destination</td></tr> </table>	Lee datos	Elemento de Source	Escribe datos	Elemento de Destination
Lee datos	Elemento de Source					
Escribe datos	Elemento de Destination					
.RemoteElement	STRING	Para especificar una etiqueta o dirección diferente en el controlador al que se envía el mensaje, se establece este miembro en el valor necesario. Se introduce la etiqueta o la dirección como caracteres ASCII.				
		<b>Si el mensaje</b> <b>Entonces la matriz remota es</b>				
		Lee datos      Elemento de Source				
		Escribe datos      Elemento de Destination				
.Unconnecte dTimeout	DINT	Tiempo de espera para un mensaje desconectado o para hacer una conexión. El valor predeterminado es 30 segundos. Si el mensaje está desconectado, el bit ER se enciende si el controlador no obtiene una respuesta dentro del tiempo UnconnectedTimeout. Si el mensaje está conectado, el bit ER se enciende si el controlador no obtiene una respuesta para hacer la conexión dentro del tiempo UnconnectedTimeout.				
.Connection Rate	DINT	Tiempo de espera para un mensaje conectado una vez tenga conexión. Este tiempo de espera es para la respuesta del otro dispositivo.				
.TimeoutMultipli er	SINT	Se aplica solo después de que se haya realizado la conexión. Tiempo de espera = ConnectionRate x TimeoutMultiplier El valor predeterminado de ConnectionRate es 7,5 segundos. El valor predeterminado de TimeoutMultiplier es 0 (que igual a un factor de multiplicación de 4). El tiempo de espera predeterminado para mensajes conectados es de 30 segundos (7.5 segundos x 4 = 30 segundos). Para cambiar el tiempo de espera, cambie ConnectionRate y deje el valor predeterminado de TimeoutMultiplier.				

### Descripción

La instrucción MSG transfiere elementos de datos. Se trata de una instrucción de transición:

- En un diagrama de escalera, EnableIn cambia de borrado a establecido cada vez que se ejecuta la instrucción.

- El tamaño de cada elemento depende del tipo de datos que especifica y el tipo de comando de mensaje que se utiliza.



Dónde	Descripción
1	EnableIn es verdadero .EN se establece .EW se establece Conexión abierta.
2	Mensaje enviado. .ST se establece .EW está borrado.
3	Mensaje finalizado o erróneo, EnableIn es falso. .DN o .ER se establece. .ST está borrado. Conexión cerrada (si .EN_CC = 0). .EN está borrado (porque EnableIn es falso).
4	EnableIn es verdadero y .DN o .ER se estableció previamente. .EN se establece .EW se establece Conexión abierta. .DN o .ER está borrado.
5	Mensaje enviado. .ST se establece .EW está borrado.

6	Mensaje finalizado o erróneo, EnableIn todavía es verdadero. .DN o .ER se establece. .ST está borrado. Conexión cerrada (si .EN_CC = 0).
7	EnableIn cambia a falso y .DN o .ER se establece. .EN está borrado.

### Afectar a las marcas de estado matemático

No

### Fallos mayores/menores

No es específico para esta instrucción. Consulte los Atributos comunes para fallos relacionados con el operando.

### Ejecución

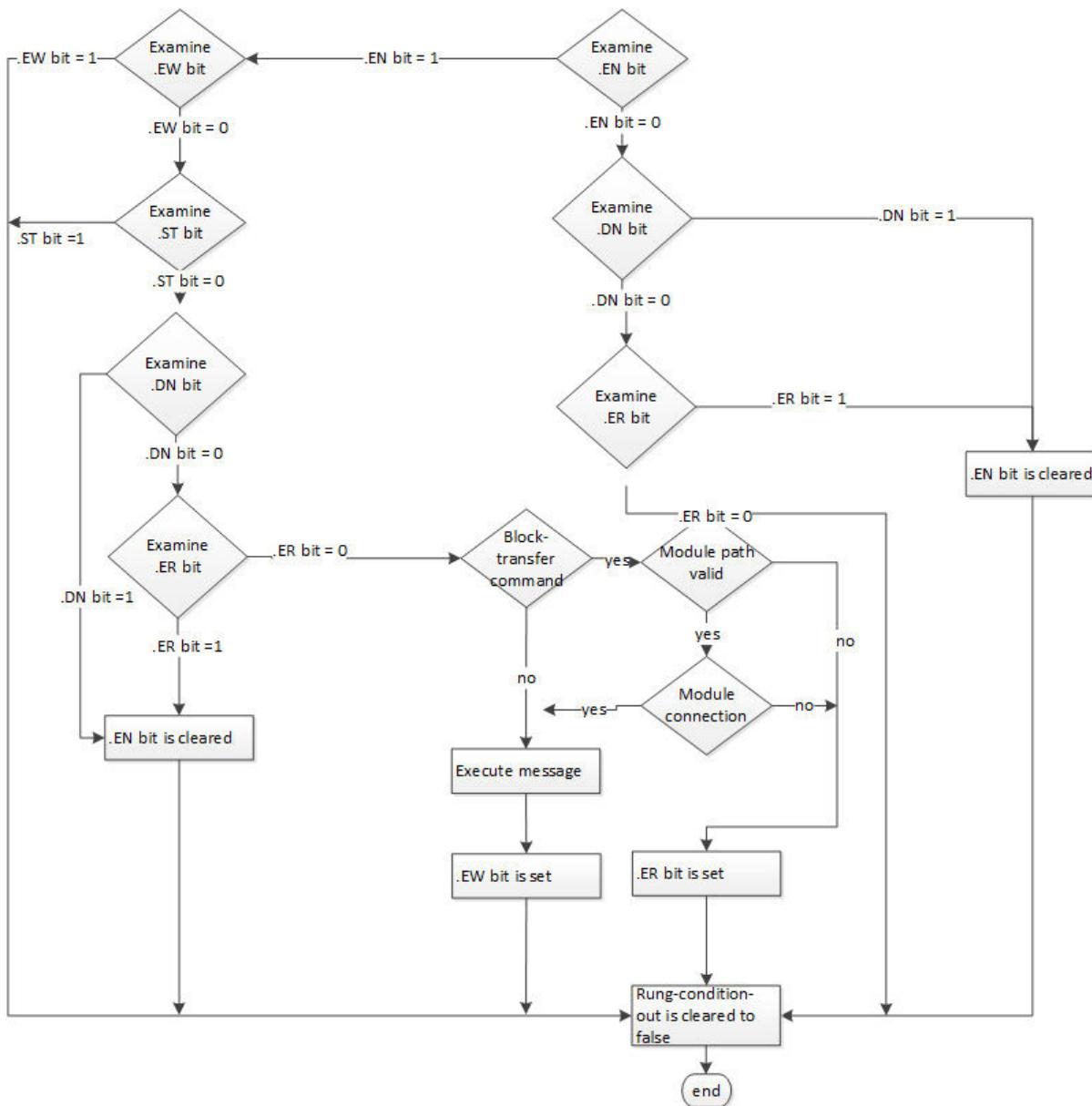
#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	Los bits .EWS, .ST, .DN y .ER se han desactivado.
La condición de entrada de reglón es falsa	Consultar el diagrama de flujo MSG (Falso)
La condición de entrada de reglón es verdadera	Consultar el diagrama de flujo MSG (Verdadero)
Post-escaneado	N/A

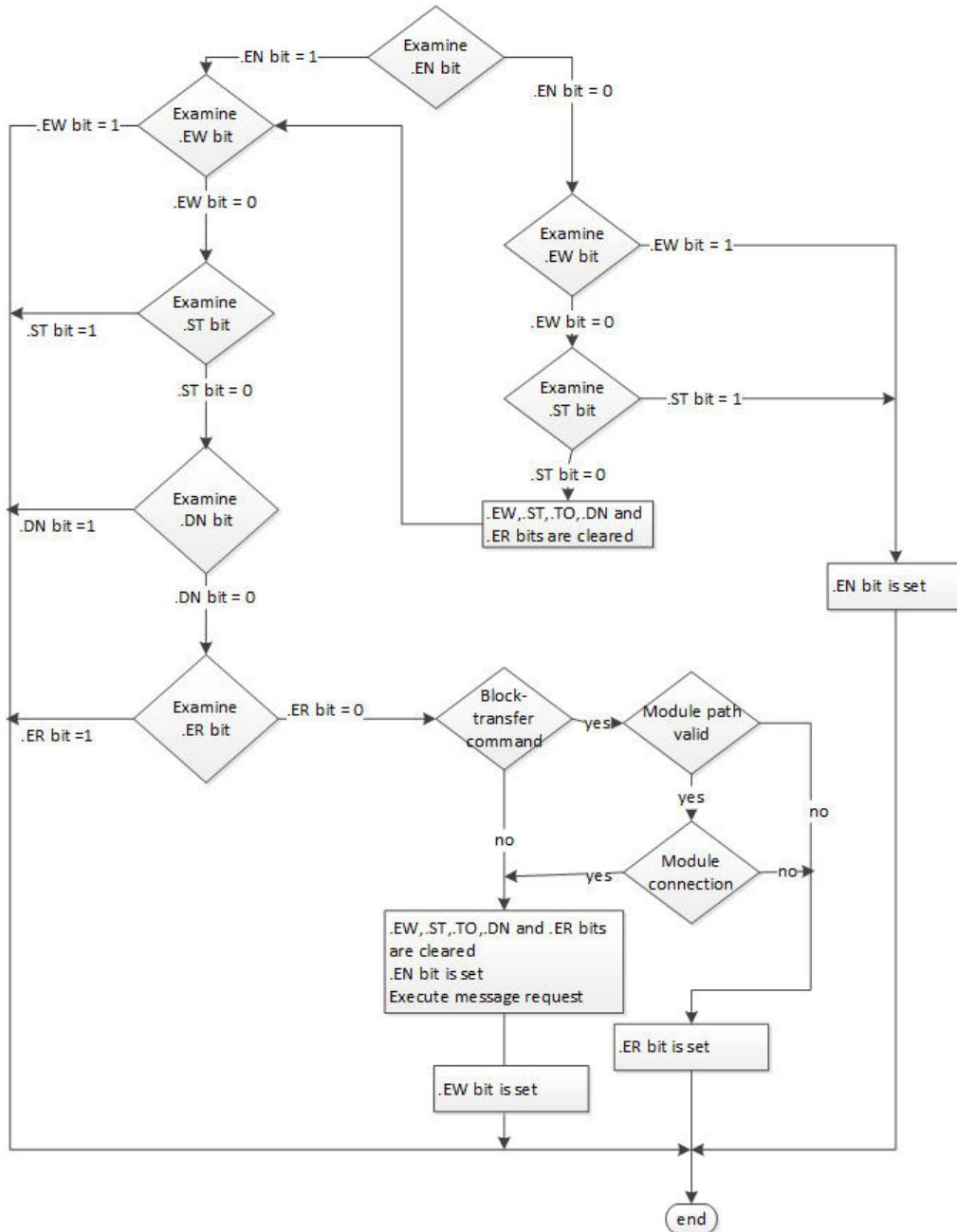
#### Texto estructurado

Condición/estado	Acción realizada
Pre-escaneado	Consultar Pre-escaneado en la tabla de diagrama de escalera
Ejecución normal	Consultar el diagrama de flujo MSG (Verdadero)
Post-escaneado	Consultar Post-escaneado en la tabla de Diagramas de escalera

Diagrama de flujo MSG (Falso)



### Diagrama de flujo MSG (Verdadero)



### Ejemplo

#### Diagrama de escalera



**Texto estructurado**

MSG (MessageControl);

**Consulte también**

[Sintaxis de texto estructurado](#) en la página 955

[Códigos de error de mensaje](#) en la página 244

[Seleccionar del tipo de mensaje](#) en la página 342

[Especificación de detalles de comunicación](#) en la página 253

[Ejemplos de configuración de MSG](#) en la página 234

[Atributos comunes](#) en la página 923

## Ejemplos de configuración de MSG

Los siguientes ejemplos muestran las etiquetas de fuente y destino, y los elementos para diferentes combinaciones de controladores.

La tabla explica la ruta para las instrucciones MSG que tienen origen en un controlador LOGIX 5000 y se han escrito para otro controlador.

Ruta del mensaje	Ejemplo de Source y Destination	
LOGIX 5000 -> LOGIX 5000	Etiqueta de Source	array_1[0]
	Etiqueta de Destination	array_2[0]
	Puede usar una etiqueta de alias en la etiqueta de origen del controlador de origen LOGIX 5000.	
	No puede usar una de alias para la etiqueta de destino. La de destino tiene que ser una etiqueta de base.	
LOGIX 5000 -> PLC-5 LOGIX 5000 -> SLC	Etiqueta de Source	array_1[0]
	Elemento de Destination	N7:10
	Puede usar una etiqueta de alias para la etiqueta en la etiqueta de origen del controlador de origen LOGIX 5000.	
LOGIX 5000 -> PLC-2	Etiqueta de Source	array_1[0]
	Elemento de Destination	010

La tabla explica la ruta para las instrucciones MSG que tienen origen en un controlador LOGIX 5000 y se han leído en otro controlador.

Ruta del mensaje	Ejemplo de Source y Destination	
LOGIX 5000 -> LOGIX 5000	Etiqueta de Source	array_1[0]
	Etiqueta de Destination	array_2[0]

	No puede usar una etiqueta de alias para la etiqueta de origen. La de origen tiene que ser una etiqueta de base. Puede usar una etiqueta de alias para la etiqueta en la etiqueta de destino del controlador de origen LOGIX 5000.	
LOGIX 5000 -> PLC-5 LOGIX 5000 -> SLC	Elemento de Source	N7:10
	Etiqueta de Destination	array_1[0]
Puede usar una etiqueta de alias para la etiqueta en la etiqueta de destino del controlador de origen LOGIX 5000.		
LOGIX 5000 -> PLC-2	Elemento de Source	010
	Etiqueta de Destination	array_1[0]

### Consulte también

[Mensaje \(MSG\)](#) en la página 224

## Códigos y tipos de fallos mayores

Estos son los tipos y códigos de fallos mayores.

Lista de major fault:

Tipo	Código	Causa	Método de recuperación
1	1	El controller está encendido en modo Marcha	Ejecute el administrador de encendido.
1	16	Se ha detectado un fallo en la configuración de la comunicación de I/O. (Solo controllers CompactLogix 1768-L4x).	Vuelva a configurar el número de modules de comunicación en el lado del bus 1768 del controller: <ul style="list-style-type: none"><li>• 1768-L43 tiene un máximo de dos módulos.</li><li>• 1768-L45 tiene un máximo de cuatro módulos.<ul style="list-style-type: none"><li>• Hasta cuatro modules Sercos</li><li>• Hasta dos modules de comunicación NetLinx</li></ul></li></ul>
1	40	Si el controlador usa una batería, esta no contiene suficiente energía para guardar el programa de usuario al apagar.  Si el controlador usa un ESM (módulo de almacenamiento de energía), este no contiene suficiente energía para guardar el programa de usuario al apagar.	<ul style="list-style-type: none"><li>• Para controladores que usan batería, sustituya la batería.</li><li>• Para controladores que usan un ESM (módulo de almacenamiento de energía):<ul style="list-style-type: none"><li>• Permita que el ESM se cargue completamente antes de apagar el controlador.</li><li>• Sustituya el ESM si se puede extraer, o sustituya el controlador si el ESM no se puede extraer.</li></ul></li><li>• Si el problema persiste, póngase en contacto con el soporte de Rockwell Automation.</li></ul>
1	60	Para un controller sin tarjeta de memoria instalada, el controller: <ul style="list-style-type: none"><li>• Detectó un fallo no recuperable.</li><li>• Borró el proyecto de la memoria.</li></ul>	<ol style="list-style-type: none"><li>1. Borre el fallo.</li><li>2. Descargue el proyecto.</li><li>3. Cambie a modo marcha/marcha remota. Siga los siguientes pasos si el fallo persiste.<ol style="list-style-type: none"><li>1. Antes de desconectar y volver a conectar el controller, registre el estado de los indicadores de estado OK y RS232.</li><li>2. Póngase en contacto con el soporte técnico de Rockwell Automation.</li></ol></li></ol>

1	61	Para un controller con tarjeta de memoria instalada, el controller: <ul style="list-style-type: none"><li>• Detectó un fallo no recuperable.</li><li>• Escribió información de diagnóstico en la tarjeta de memoria.</li><li>• Borró el proyecto de la memoria.</li></ul>	1. Borre el fallo. 2. Descargue el proyecto. 3. Cambie a modo marcha/marcha remota. Si el fallo persiste, póngase en contacto con el soporte de Rockwell Automation.
1	62	Para un controller con tarjeta Secure Digital (SD) instalada, el controller: <ul style="list-style-type: none"><li>• Detectó un falló no recuperable.</li><li>• Escribió información de diagnóstico en la tarjeta de memoria.</li></ul> Cuando se encuentre en este estado, el controller no abrirá ninguna conexión ni permitirá cambiar al modo marcha.	1. Borre el fallo. 2. Descargue el proyecto. 3. Cambie a modo marcha/marcha remota. Si el fallo persiste, póngase en contacto con el soporte de Rockwell Automation.
3	16	Ha fallado una conexión de module de I/O requerida	Compruebe: <ul style="list-style-type: none"><li>• Que el module de I/O está en el chasis.</li><li>• Los requisitos de codificación electrónica</li><li>• La ficha Major faults de Propiedades de controller y la ficha Conexión de Propiedades del module si desea más información sobre el fallo.</li></ul>
3	20 21	Possible problema con el chasis.	No recuperable: cambie el chasis.
3	23	Al menos una conexión requerida no se estableció antes de pasar a modo Marcha	Espere a que la luz del I/O de controller se ponga verde antes de cambiar al modo Marcha.
4	16	Instrucción desconocida detectada	Elimine la instrucción desconocida. Probablemente la ha provocado un proceso de conversión de program.
4	20	Subíndice de registro demasiado grande; la estructura de control POS o LEN no es válida.	Ajuste el valor de modo que entre dentro del rango definido. No supere el tamaño de registro ni sobrepase las dimensiones definidas.
4	21	LEN o POS de la estructura de CONTROL es menor que 0	Ajuste el valor para que sea mayor que 0.
4	31	Los parámetros de la instrucción JSR no coinciden con el SBR asociado o con la instrucción RET	Pase el número apropiado de parámetros. Si se pasan demasiados parámetros, los sobrantes se ignoran sin provocar ningún error.
4	34	Una instrucción de temporizador tiene una preselección negativa o un valor acumulado	Arregle el program para que no cargue un valor negativo en el temporizador preseleccionado o acumulado.
4	42	JMP se dirige a una etiqueta que no existe o que ha sido borrada	Corrija el destino de JMP o agregue la etiqueta que falta.
4	82	Un diagrama de funciones secuenciales (SFC) ha invocado una subrutina y esta ha intentado saltar hacia atrás al SFC que invoca. Ocurre cuando el SFC utiliza una instrucción JSR o FOR para invocar a la subrutina.	Retire el salto hacia atrás al SFC que invoca.
4	83	Los datos probados no estaban dentro de los límites requeridos Esto ocurre con subregímenes de tabla que se usan con tablas booleanas y direccionamiento a nivel de bit.	Ajuste el valor para que esté dentro del rango válido. No supere el tamaño de la tabla ni sobrepase las dimensiones definidas.
4	84	Overflow de pila	Reduzca los niveles de anidamiento de subrutinas o el número de parámetros.
4	89	En una instrucción SFR, la routine de destino no contiene el paso de destino.	Corrija el destino de SFR o agregue el paso que falta.

4	90	Uso de una instrucción de seguridad fuera de una task de seguridad.	Coloque la instrucción de seguridad dentro de la task de seguridad.
4	91	La instrucción de equipment phase se invoca desde fuera de un program de equipment phase.	Use la instrucción únicamente en un program de equipment phase.
4	94	Se han superado los límites de anidamiento.	Restructure el proyecto para reducir los niveles de anidamiento de las subroutines.
4	990 - 999	Major fault definido por el usuario.	
6	1	Se agotó el tiempo del temporizador de control Watchdog para la task  La task de usuario no se ha completado dentro del periodo especificado. Un error de program ha provocado un lazo infinito, o el program es demasiado complejo para ejecutarse tan rápidamente como se ha especificado, o bien hay una task de priority superior que impide que se termine esta task (se intenta realizar un trabajo excesivo con un solo controller).	Incremente el temporizador de task, acorte el tiempo de ejecución, eleve la prioridad de esta task, simplifique las task de prioridad más alta o pase algunos códigos a otro controller.
7	40	Falló el almacenamiento en memoria no volátil.	<ul style="list-style-type: none"> <li>• Intente almacenar otra vez el proyecto en memoria no volátil.</li> <li>• Si no se puede almacenar el proyecto en memoria no volátil, reemplace la tarjeta de memoria.</li> <li>• Si está utilizando un controller 1756-L7x, compruebe que la tarjeta SD esté desbloqueado.</li> </ul>
7	41	La carga desde la memoria no volátil falló debido a una desigualdad en el tipo de controller.	Cambie a un controller del tipo correcto o descargue el proyecto y guárdelo en la tarjeta de memoria.
7	42	Falló la carga desde memoria no volátil debido a que la revisión de firmware del proyecto en memoria no volátil no coincide con la revisión de firmware del controller.	Actualice el firmware del controller al mismo nivel de revisión que el proyecto que está en memoria no volátil.
7	43	La carga desde la memoria no volátil falló debido a una suma de comprobación incorrecta.	Póngase en contacto con el soporte técnico de Rockwell Automation.
7	44	Se produjo un fallo al restaurar la memoria del procesador.	Póngase en contacto con el soporte técnico de Rockwell Automation.

7	50	<p>El certificado del archivo de registro no se puede verificar. Cuando se inicia el controller, este intenta verificar la combinación de clave y certificado del archivo de registro. En función de la verificación, el controller realizará una de las siguientes acciones:</p> <ul style="list-style-type: none"> <li>• Si el controller verifica el certificado del archivo de registro existente, el controller continuará con el directorio de registro existente.</li> <li>• Si no se puede verificar el certificado existente, el controller registrará un fallo mayor e intentará crear un nuevo certificado.           <ul style="list-style-type: none"> <li>• Si el controller crea un nuevo certificado correctamente, creará un subdirectorio de registro de copia de seguridad, moverá los archivos existentes a dicho directorio y continuará registrando y firmando con la nueva clave de verificación y el nuevo certificado de archivo de registro.</li> <li>• Si el controller no puede crear un nuevo certificado, el controller escribirá las entradas del registro en el directorio de registro existente, pero no actualizará los archivos de firma de dicho directorio.</li> </ul> </li> </ul>	Borre el fallo y desconecte y vuelva a conectar el controller. Si el problema persiste, póngase en contacto con el soporte de Rockwell Automation.
8	1	Se ha intentado situar el controller en modo Marcha con el interruptor de llave durante la descarga	Espere a que se complete la descarga y borre el error.
11	1	La posición actual del axis excedió el límite de fin de carrera positivo	Desplace el axis en dirección negativa hasta que la posición se encuentre dentro del límite de fin de carrera y ejecute Restablecer fallo de motion axis.
11	2	La posición actual del axis excedió el límite de fin de carrera negativo	Mueva el axis en dirección positiva hasta que la posición se encuentre dentro del límite de fin de carrera, y ejecute Restablecer fallo de motion axis.
11	3	La posición real excedió la tolerancia de error de posición	Mueva la posición dentro de la tolerancia y ejecute Restablecer fallo de motion axis.
11	4	La conexión de los canales A, B o Z del encoder está interrumpida.	Vuelva a conectar el canal de encoder y ejecute Restablecer fallo de motion axis.
11	5	Se detectó ruido en el encoder, o bien las señales no están en la cuadratura	Arregle el cableado del encoder y ejecute Restablecer fallo de motion axis.
11	6	Se ha activado la entrada de fallo de variador	Borre el fallo de variador y ejecute Restablecer fallo de motion axis.
11	7	Falló la conexión síncrona	Primero, ejecute Restablecer fallo de motion axis. Si eso no funciona, extraiga el servo module y vuelva a ponerlo. Si tampoco funciona, reemplace el servo module.
11	8	El servo module ha detectado un fallo grave de hardware	Reemplace el module
11	9	Falló la conexión asíncrona	Primero, ejecute Restablecer fallo de motion axis. Si eso no funciona, extraiga el servo module y vuelva a ponerlo. Si tampoco funciona, reemplace el servo module.
11	10	Ha ocurrido un fallo en el motor.	Consulte el tag de axis DriveFaults para obtener más información.
11	11	Ha ocurrido un fallo térmico del motor.	Consulte el tag de axis DriveFaults para obtener más información.
11	12	Ha ocurrido un fallo térmico del motor.	Consulte el tag de axis DriveFaults para obtener más información.

11	13	Ha ocurrido un fallo del anillo SERCOS.	Compruebe la integridad de la red de fibra óptica del anillo SERCOS y los dispositivos que hay en ella.
11	14	Ha ocurrido un fallo en la entrada de habilitación del variador.	Vuelva a habilitar la entrada de habilitación del variador y borre el fallo.
11	15	Ha ocurrido un fallo de pérdida de phase del variador.	Restablezca la conexión de energía en el variador y borre el fallo.
11	16	Ha ocurrido un fallo de protección del variador.	Consulte el tag de axis GuardFaults para obtener más información.
11	32	La task de movimiento ha sufrido una superposición	El régimen aproximado es demasiado alto para mantener una operación correcta. Borre el tag del fallo de grupo, eleve el régimen de actualización del grupo y borre el major fault.
12	32	Tras desconectar y volver a conectar la energía en un controller secundario descalificado, no se ha encontrado ningún controller ni chasis homólogo.	Compruebe que: <ul style="list-style-type: none"> <li>• Se haya conectado un chasis homólogo.</li> <li>• Se haya aplicado energía en ambos chasis redundantes.</li> <li>• Los controllers compatibles tengan el mismo: <ul style="list-style-type: none"> <li>• número de catálogo.</li> <li>• número de ranura.</li> <li>• revisión de firmware.</li> </ul> </li> </ul>
12	33	Se ha identificado un controller no compatible en el nuevo chasis primario tras una conmutación.	Existen dos posibilidades: <ul style="list-style-type: none"> <li>• Retire el controller no compatible e identifique la causa de la conmutación.</li> <li>• Añada un controller homólogo al chasis secundario</li> <li>• Identifique la causa de la conmutación y sincronice el sistema.</li> </ul>
12	34	Justo después de que ocurra una conmutación, las posiciones del interruptor de llave de los controllers primario y secundario no coinciden. El controller primario anterior está en modo program y el nuevo controller primario está en modo marcha.	Existen dos posibilidades: <ul style="list-style-type: none"> <li>• Para borrar el fallo, cambie los interruptores de llave de la posición de modo en marcha a la posición de modo program, y de nuevo a modo marcha.</li> <li>• Use la aplicación Logix Designer para conectar los controladores. A continuación, borre los fallos y cambie el modo de ambos controllers al modo marcha.</li> </ul>
14	1	Se agotó el tiempo del temporizador de control Watchdog para la task de seguridad.  La task de usuario no se ha completado dentro del periodo especificado. Un error de program ha causado un lazo infinito, el program es demasiado complejo para ejecutarse tan rápidamente como se ha especificado, una task con mayor priority está impidiendo que esta task finalice o se ha retirado el homólogo de seguridad.	Borre el fallo.  Si existe una firma de task de seguridad, la memoria de seguridad vuelve a iniciarse y la task de seguridad empieza a ejecutarse.  Si no existe una firma de task de seguridad, debe volver a descargar el program para poder ejecutar la task de seguridad.  Si lo ha retirado, vuelva a insertar el homólogo de seguridad.
14	2	Existe un error en una routine de la task de seguridad.	Corrija el error en la routine en la lógica del programa de usuario.
14	3	Falta el homólogo de seguridad.	Instale un homólogo de seguridad compatible.
14	4	El homólogo de seguridad no está disponible.	Instale un homólogo de seguridad compatible.
14	5	El hardware del homólogo de seguridad es incompatible.	Instale un homólogo de seguridad compatible.
14	6	El firmware del homólogo de seguridad es incompatible.	Instale un homólogo de seguridad compatible.

14	7	<p>Task de seguridad inoperativa.</p> <p>Este fallo ocurre cuando la lógica de seguridad no es válida, por ejemplo: si existe una desigualdad de lógica entre el controller primario y el homólogo de seguridad, si se supera el tiempo de espera de watchdog o si la memoria está dañada.</p>	<p>Borre el fallo.</p> <p>Si existe una firma de task de seguridad, la memoria de seguridad vuelve a iniciarse usando la firma de task de seguridad y la task de seguridad empieza a ejecutarse.</p> <p>Si no existe una firma de task de seguridad, debe volver a descargar el program para poder ejecutar la task de seguridad.</p>
14	8	No se encontró el maestro de la hora coordinada del sistema (CST).	Borre el fallo. Configure un dispositivo para que sea el CST maestro.
14	9	Fallo de controller no recuperable del homólogo de seguridad.	Borre el fallo y descargue el program. Si el fallo persiste, sustituya el homólogo de seguridad.
17	34	La temperatura interna del controller ha excedido el límite operativo.	Se tomarán medidas para reducir la temperatura ambiente del module. Siga los límites recomendados para la temperatura ambiente (entrada) y mantenga el espacio libre necesario alrededor el chasis.
18	1	El variador de movimiento CIP no se ha iniciado correctamente.	Para determinar la acción correctiva, consulte los atributos de Fallo de Inicialización, donde obtendrá más información sobre el tipo de fallo producido.
18	2	<p>El variador de movimiento CIP no se ha iniciado correctamente.</p> <p>Este fallo se muestra cuando se produce un fallo de inicialización específico del fabricante.</p>	Para determinar la acción correctiva, consulte los atributos de Fallo de inicialización CIP - Mfg, donde obtendrá más información sobre el tipo de fallo producido.
18	3	Se establece el bit de fallo de axis físico, el cual indica un fallo en el axis físico.	Para determinar la acción correctiva, consulte los atributos de Fallo de Axis CIP, donde obtendrá más información sobre el tipo de fallo producido.
18	4	<p>Se establece el bit de fallo de axis físico, el cual indica un fallo en el axis físico.</p> <p>Esta fallo se indica cuando se produce un fallo de axis específico del fabricante.</p>	Para determinar la acción correctiva, consulte los atributos de Fallo de inicialización - Mfg, donde obtendrá más información sobre el tipo de fallo producido.
18	5	Ha ocurrido un fallo de movimiento.	Para determinar la acción correctiva, consulte el atributo de Fallo de movimiento y Bits de fallo de movimiento, donde obtendrá más información sobre el tipo de fallo producido.
18	6	Ha ocurrido un fallo de variador de movimiento CIP. Normalmente el fallo afecta a todos los axis asociados con el module y todos los axis asociados se apagan.	Para solucionar el fallo, vuelva a configurar el module de movimiento que falló.
18	7	Ha ocurrido un fallo de grupo de movimiento. Normalmente el fallo afecta a todos los axis asociados con un grupo de movimiento.	Para solucionar el fallo, vuelva a configurar todo el subsistema de movimiento.
18	8	<p>Ha ocurrido un fallo durante la configuración de un variador de movimiento CIP.</p> <p>Normalmente, este fallo ocurre después de intentar actualizar sin éxito un atributo de la configuración del axis de un variador de movimiento CIP.</p>	Para determinar la acción correctiva, consulte los atributos Fallo de configuración en el código de error de atributo e ID de error de atributo asociados con el movimiento o el module 1756-ENxT.
18	9	Ha ocurrido un fallo de recuperación de posición absoluta (APR) y la posición absoluta del axis no se puede recuperar.	Para determinar la acción correctiva, consulte el Fallo APR para identificar la causa del fallo.
18	10	<p>Ha ocurrido un fallo de recuperación de posición absoluta (APR) y la posición absoluta del axis no se puede recuperar.</p> <p>Esta fallo se indica cuando se produce un fallo de APR específico del fabricante.</p>	Para determinar la acción correctiva, consulte los atributos de Fallo APR - Mfg para identificar la causa del fallo.

18	128	Ha ocurrido un fallo específico de la función de seguridad Guard Motion. Este fallo sólo sucede cuando se usa un variador con la función Guard Safety.	Para determinar la acción correctiva, consulte los atributos de Guard Motion y bits de estado de protección para identificar la causa del fallo.
20	1	La licencia necesaria no se encuentra o bien ha caducado durante la transition al modo de marcha o prueba.	Inserte una CmCard con todas las licencias que necesita el proyecto en el controller.

## Códigos y tipos de fallos menores

Los siguientes son los tipos y códigos de fallos menores.

Lista de fallos menores:

Tipo	Código	Causa	Método de recuperación
1	15	<ul style="list-style-type: none"> <li>Hay una fuente de alimentación 1769 conectada directamente al CompactBus 1768 del controlador, con una configuración no válida.</li> <li>La fuente de alimentación 1768 que abastece al controlador ha fallado.</li> </ul>	<ul style="list-style-type: none"> <li>Quite la fuente de alimentación del CompactBus 1768, desconecte el sistema y vuelva a conectarlo.</li> <li>Sustituya la fuente de alimentación.</li> </ul>
3	1	Estado de bus desactivado. Las conexiones entre el controlador y los módulos de E/S están interrumpidas.	<p>Siga estos pasos para identificar el origen del fallo de BUS OFF:</p> <ol style="list-style-type: none"> <li>El número de módulos de expansión locales del proyecto coincide con el número de módulos instalados físicamente en el sistema.</li> <li>Todas las bases de montaje están bloqueadas y los módulos de E/S están instalados de forma segura en esas bases.</li> <li>Todos los módulos de E/S 1734 POINT están configurados para utilizar la velocidad en baudios automática.</li> </ol> <p>Si estos pasos no resuelven la condición de fallo, contacte con el soporte técnico de Rockwell Automation.</p>
3	94	La actualización de RPI actual de un módulo de E/S se superpone con la actualización de RPI anterior.	<p>Defina el índice de RPI de los módulos de E/S en un valor numérico más alto.</p> <p>Rockwell Automation recomienda que los sistemas de control CompactLogix 5370 L2 y CompactLogix 5370 L3 no se ejecuten con fallos de superposición de RPI de los módulos.</p>
4	4	Se ha producido un desbordamiento aritmético en una instrucción.	Arregle el programa examinando las operaciones aritméticas (orden) o ajustando los valores.
4	5	No se encontró la instancia especificada en una instrucción GSV/SSV.	Compruebe el nombre de instancia.
4	6	En una instrucción GSV/SSV, existen dos posibilidades: <ul style="list-style-type: none"> <li>El nombre de clase especificado no se admite</li> <li>El nombre de atributo especificado no es válido</li> </ul>	Compruebe el nombre de clase y el nombre de atributo.
4	7	La etiqueta de destino GSV/SSV es demasiado pequeña para todos los datos.	Arregle el destino o el origen para que haya suficiente espacio.
4	30	Se han transmitido parámetros incorrectos a través del puerto ASCII.	Verifique los parámetros de configuración de ASCII.

4	35	Tiempo de delta de PID $\leq 0$ .	Ajuste el tiempo de delta de PID para que sea mayor que 0.
4	36	Punto de ajuste de PID fuera de rango.	Modifique el punto de ajuste para que esté dentro del rango.
4	51	El valor LEN de la etiqueta de cadena es mayor que el tamaño de DATA de la etiqueta de cadena.	<ul style="list-style-type: none"> <li>Compruebe que ninguna instrucción esté escribiendo en el miembro LEN de la etiqueta de cadena.</li> <li>En el valor LEN, introduzca el número de caracteres que contiene la cadena.</li> </ul>
4	52	La cadena de salida es mayor que el destino.	Cree un nuevo tipo de datos de cadena que sea lo bastante grande para la cadena de salida. Use el nuevo tipo de datos de cadena como el tipo de datos para el destino.
4	53	El número de salida sobrepasa los límites del tipo de datos de destino.	Existen posibilidades: <ul style="list-style-type: none"> <li>Reduzca el tamaño del valor ASCII.</li> <li>Use un tipo de datos mayor para el destino.</li> </ul>
4	56	El valor de Start o Quantity no es válido.	<ul style="list-style-type: none"> <li>Comprueba que el valor de Start se encuentre entre 1 y el tamaño de DATA del Source.</li> <li>Compruebe que el valor de Start más el valor de Quantity sea menor o igual al tamaño DATA del Source.</li> </ul>
4	57	La instrucción AHL no pudo ejecutarse porque el puerto en serie está establecido en sin comunicación.	Existen posibilidades: <ul style="list-style-type: none"> <li>Cambie el parámetro Línea de control del puerto en serie.</li> <li>Elimine la instrucción AHL.</li> </ul>
6	2	Superposición de tarea periódica. La tarea periódica no se ha completado antes de que llegue el momento de ejecutarla de nuevo	Realice cambios como simplificación de programas, ampliación del período o aumento de la prioridad relativa.
6	3	Superposición de tarea de evento. La tarea de evento no se ha completado antes de que llegue el momento de ejecutarla de nuevo.	Realice cambios como simplificación de programas, ampliación del período, aumento de la prioridad relativa o retraso de la activación del evento.
7	49	Cuando el controlador carga un proyecto desde la memoria no volátil, registra su fallo menor y establece el objeto FaultLog, atributo MinorFaultBits, bit 7.	Borre el fallo.
9	0	Error desconocido durante el servicio del puerto en serie	Póngase en contacto con el Soporte técnico de Rockwell Automation si el problema persiste.
9	1	La línea CTS no es correcta para la configuración actual.	Desconecte y vuelva a conectar el cable del puerto en serie al controlador. Compruebe que el cableado es correcto.
9	2	Error de lista de encuesta. Se ha detectado un fallo con la lista de encuestas de la estación maestra de DF1, como la especificación de más estaciones que el tamaño del archivo, la especificación de más de 255 estaciones, el intento de indexar pasando el final de la lista o encuestar la dirección de emisión (STN #255).	Compruebe que no existen los siguientes errores: <ul style="list-style-type: none"> <li>El número total de estaciones es mayor que el espacio de la etiqueta de lista de encuestas.</li> <li>El número total de estaciones es mayor que 255.</li> <li>El puntero de la estación actual es mayor que el final de la etiqueta de la lista de encuestas.</li> <li>Se ha encontrado un número de estaciones mayor que 254.</li> </ul>
9	3	No está especificada la etiqueta de la estación activa maestra RS-232 DF1.	Especifique una etiqueta para utilizarla como la etiqueta de la estación activa en la ficha Protocolo de puerto en serie, bajo las Propiedades del controlador.
9	5	Se ha superado el tiempo límite de encuesta de estación esclava de DF1. El tiempo de espera de vigilancia de encuesta para la estación esclava ha caducado. La estación maestra no ha encuestado este controlador en el tiempo especificado.	Determine y corrija el retraso para la encuesta.

9	9	Se perdió contacto para el módem. Las líneas de control DCD o DSR no se reciben en el orden y/o el estado adecuados.	Conexión del módem con el controlador correcta.
9	10	Los datos se han perdido o caído del puerto en serie.	Ralentice el régimen en que el iniciador está enviando los datos.
10	10	No hay batería, o bien se debe reemplazar.	Instale una batería nueva.
10	11	La batería del homólogo de seguridad no se detecta o se debe cambiar.	Instale una batería nueva.
10	12	El módulo de almacenamiento de energía (ESM) no está instalado. Si el controlador está desconectado, el programa y atributo WallClockTime no se conservan.	Instale un ESM en el controlador.
10	13	El ESM instalado no es compatible con el controlador.	Cambie el ESM instalado por uno que sea compatible con el controlador.
10	14	El ESM debe ser sustituido debido a un fallo de hardware. No se puede conservar el programa del controlador ni el atributo WallClockTime cuando está apagado.	Sustituya el ESM.
10	15	El ESM no puede almacenar suficiente energía en el ESM para conservar el atributo WallClockTime o el programa del controlador cuando está apagado.	Sustituya el ESM.
10	16	El suministro de alimentación ininterrumpida (UPS) no se encuentra o no está listo.	Existen posibilidades: <ul style="list-style-type: none"> <li>• Instale el UPS.</li> <li>• Compruebe el UPS para asegurarse de que está correctamente cargado para suministrar alimentación de respaldo en caso de pérdida de alimentación.</li> </ul>
10	17	La batería del UPS presenta fallos y es necesario sustituirla.	Sustituya la batería del UPS.
13	21	El tiempo de reloj se encuentra fuera de rango.	Asegúrese de que la fecha y el tiempo de reloj sean correctas.
14	12	El proyecto de seguridad está configurado como SIL2/PLd y hay presente un homólogo de seguridad.	Asegúrese de que no hay instalado ningún homólogo de seguridad a la derecha del controlador primario.
17	1...n	Ha fallado un diagnóstico de controlador interno.	Póngase en contacto con el Soporte técnico de Rockwell Automation y facilíteleles el tipo y código del fallo.
17	35	La temperatura interna del controlador se acerca al límite operativo.	Se tomarán medidas para reducir la temperatura ambiente del módulo. Siga los límites recomendados para la temperatura ambiente (entrada) y mantenga el espacio libre necesario alrededor el chasis.
17	36	No hay ningún ventilador o no mantiene la velocidad deseada.	Sustituya el ventilador.
20	1	La licencia necesaria no se encuentra, o bien ha caducado mientras el controlador se encuentra en modo de ejecución o prueba.	Inserte una CmCard con todas las licencias que necesita el proyecto en el controlador.

Palabras clave: código de fallo:2, códigos de fallo:2, fallos:2Palabras clave: faults:5

## Códigos de error de mensaje

Los códigos de error dependen del tipo de instrucción MSG.

**Consulte también**

[Códigos de error](#) en la página 245

[Códigos de error extendidos](#) en la página 246

[Códigos de error PLC y SLC \(.ERR\)](#) en la página 249

[Códigos de error en la transferencia en bloque](#) en la página 252

## Códigos de error

La aplicación Logix Designer no muestra siempre la descripción completa.

Código de error (Hex)	Descripción	Visualización en el software
0001	Fallo de conexión (códigos de error extendidos)	Igual que en la descripción
0002	Recurso insuficiente	
0003	Valor no válido	
0004	Error de sintaxis IOI (ver códigos de error extendidos)	
0005	Destino desconocido, clase no compatible, instancia o elemento estructural no definido (ver códigos de error extendidos)	
0006	Espacio de paquete insuficiente	
0007	La conexión se ha perdido	
0008	Servicio no compatible	
0009	Error en segmento de datos o valor de atributo no válido	
000A	Error de lista de atributos	
000B	El estado ya existe	
000C	Conflicto de modelo de objeto	
000D	El objeto ya existe	
000E	El atributo no se puede establecer	
000F	Permiso denegado	
0010	Conflicto en el estado del dispositivo	
0011	No hay espacio para la respuesta	
0012	Fragmento primitivo	
0013	Datos de comando insuficientes	
0014	Atributo no compatible	
0015	Demasiados datos	
001A	Solicitud de puente demasiado larga	
001B	Respuesta de puente demasiado larga	
001C	Escasez de listas de atributos	
001D	Lista de atributos no válida	Igual que en la descripción
001E	Error de servicios incrustados	
001F	Error de conexión (ver errores de código extendidos)	
0022	La respuesta no válida recibida	
0025	Error en segmento clave	

0026	Error de IOI no válido	
0027	Atributo no esperado en la lista	
0028	Error DeviceNet: ID de miembro no válido	
0029	Error DeviceNet: no se puede establecer el miembro	
00D1	El módulo no se encuentra en estado de marcha	Error desconocido
00FB	Puerto del mensaje no compatible	
00FC	Tipo de datos del mensaje no compatible	
00FD	Mensaje no iniciado	
00FE	Tiempo de espera del mensaje	
00FF	Error general (ver errores de código extendidos)	

## Códigos de error extendidos

La aplicación Logix Designer no muestra ningún texto para los códigos de error extendidos.

A continuación se muestran los códigos de error extendidos para el código de error 0001.

Código de error extendidos (hex)	Descripción
0100	Conexión en uso
0103	Transporte no compatible
0106	Conflicto de propiedad
0107	Conexión no encontrada
0108	Tipo de conexión no válido
0109	Tamaño de conexión no válido
0110	Módulo no configurado
0111	EPR no compatible
0113	Error de escritura MSG
0114	Módulo equivocado
0115	Tipo de dispositivo equivocado
0116	Revisión equivocada
0118	Formato de configuración no válido
011A	Aplicación fuera de conexiones
0203	Tiempo de espera de la conexión
0204	Tiempo de espera del mensaje no conectado
0205	Error de parámetro de envío no conectado
0206	Mensaje demasiado largo

0301	No hay memoria de búfer
0302	Bandwidth no disponible
0303	No hay pantallas disponibles
0305	Discrepancia de firmas
0311	Puerto no disponible
0312	Dirección de vínculo no disponible
0315	Tipo de segmento no válido
0317	Conexión no programada

A continuación se muestran los códigos de error extendidos para el código de error 001F.

Código de error extendidos (hex)	Descripción
0203	Tiempo de espera de la conexión

A continuación se muestran los códigos de error extendidos para los códigos de error 0004 y 0005.

Código de error extendidos (hex)	Descripción
0000	estado extendido fuera de memoria
0001	estado extendido fuera de instancias

A continuación se muestran los códigos de error extendidos para el código de error 00FF.

Código de error extendidos (hex)	Descripción
2001	IOI excesivo
2002	Valor de parámetro erróneo
2018	Rechazo de semáforo
201B	Tamaño demasiado pequeño
201C	Tamaño no válido
2100	Fallo de privilegio
2101	Posición del interruptor de llave no válida
2102	Contraseña no válida
2103	No se ha emitido ninguna contraseña
2104	Dirección fuera de rango
2105	Dirección y cuántos fuera de rango
2106	Datos en uso
2107	El tipo no es valido o no es compatible
2108	Controlador en modo carga o descarga

2109	Intento de cambiar el número de las dimensiones de la matriz
210A	Nombre de símbolo no válido
210B	El símbolo no existe
210E	Búsqueda con fracaso
210F	La tarea no se puede iniciar
2110	No se puede escribir
2111	No se puede leer
2112	Rutina compartida no editable
2113	Controlador en modo de fallo
2114	Modo Marcha inhibido

## Códigos de error PLC y SLC (.ERR)

La revisión 10.x (y las posteriores) del firmware de Logix proporciona nuevos códigos de error para los errores que están asociados con los tipos de mensaje PLC y SLC™ (mensajes PCCC).

Esto permite que el software RSLogix 5000 muestre una mejor descripción de la mayoría de los errores. Antes de este cambio, el software no daba ninguna descripción para los errores asociados con el código de error 00F0.

Otra ventaja es que los códigos de error son ahora más coherentes con los errores enviados por otros controladores, como los PLC-5®.

En la siguiente tabla se puede observar el cambio en los códigos de error, desde la versión R9.x (y anteriores) hasta la versión R10.x (y posteriores). A raíz de este cambio, el miembro .ERR devuelve un valor único para cada error PCCC. Ya no es necesario .EXERR para estos errores.

### Códigos de error PLC y SLC (hex)

R9.x y anteriores		R10.x y posteriores		Descripción
.ERR	.EXERR	.ERR	.EXERR	
0010		1000		Comando ilegal o formato desde el procesador local
0020		2000		El módulo de comunicación no funciona
0030		3000		El nodo remoto está perdido, desconectado o apagado
0040		4000		El procesador está conectado pero con fallo (hardware)
0050		5000		Número de estación equivocado
0060		6000		La función solicitada no está disponible
0070		7000		El procesador está en modo Programa
0080		8000		El archivo de compatibilidad de procesador no existe
0090		9000		El nodo remoto no puede poner el comando en el búfer
00B0		B000		No se puede acceder al procesador porque está descargando
00F0	0001	F001		El procesador ha convertido la dirección de manera incorrecta
00F0	0002	F002		Dirección incompleta

00F0	0003	F003		Dirección incorrecta
00F0	0004	F004		Formato de dirección ilegal: símbolo no encontrado
00F0	0005	F005		Formato de dirección ilegal: el símbolo tiene un número de caracteres igual a 0 o superior al número máximo permitido por el dispositivo
00F0	0006	F006		El archivo de la dirección no existe en el procesador de destino
00F0	0007	F007		El archivo de destino es demasiado pequeño para el número de palabras solicitado
00F0	0008	F008		No se puede realizar la solicitud La situación ha cambiado durante la operación con múltiples paquetes
00F0	0009	F009		Archivo o datos demasiado grandes Memoria no disponible
00F0	000A	F00A		El procesador de destino no puede poner la información solicitada en paquetes
00F0	000B	F00B		Error de privilegio; acceso denegado
00F0	000C	F00C		La función solicitada no está disponible
00F0	000D	F00D		Solicitud es redundante
00F0	000E	F00E		No se puede ejecutar el comando
00F0	000F	F00F		Desbordamiento; desbordamiento de histograma
00F0	0010	F010		Sin acceso
00F0	0011	F011		El tipo de datos solicitado no coincide con los datos disponibles
00F0	0012	F012		Parámetros de comando incorrectos
00F0	0013	F013		La referencia de dirección se encuentra en el área eliminada
00F0	0014	F014		Error de ejecución de comando por razón desconocida Desbordamiento de histograma PLC-3®
00F0	0015	F015		Error de conversión de datos
00F0	0016	F016		El escáner no está disponible para comunicarse con un adaptador de rack 1771
00F0	0017	F017		El adaptador no está disponible para comunicarse con el módulo
00F0	0018	F018		La respuesta del módulo 1771 no era válida
00F0	0019	F019		Etiqueta duplicada
00F0	001A	F01A		Propietario activo del archivo: el archivo está siendo utilizado
00F0	001B	F01B		Propietario activo del programa: alguien lo está descargando o editando en línea
00F0	001C	F01C		El archivo del disco está protegido contra escritura o no accesible por alguna otra razón (solo fuera de línea)
00F0	001D	F01D		El archivo del disco está siendo utilizado por otra aplicación No se ha realizado la actualización (solo fuera de línea)



## Códigos de error en la transferencia en bloque

Estos son los códigos de error específicos para la transferencia en bloque en LOGIX 5000.

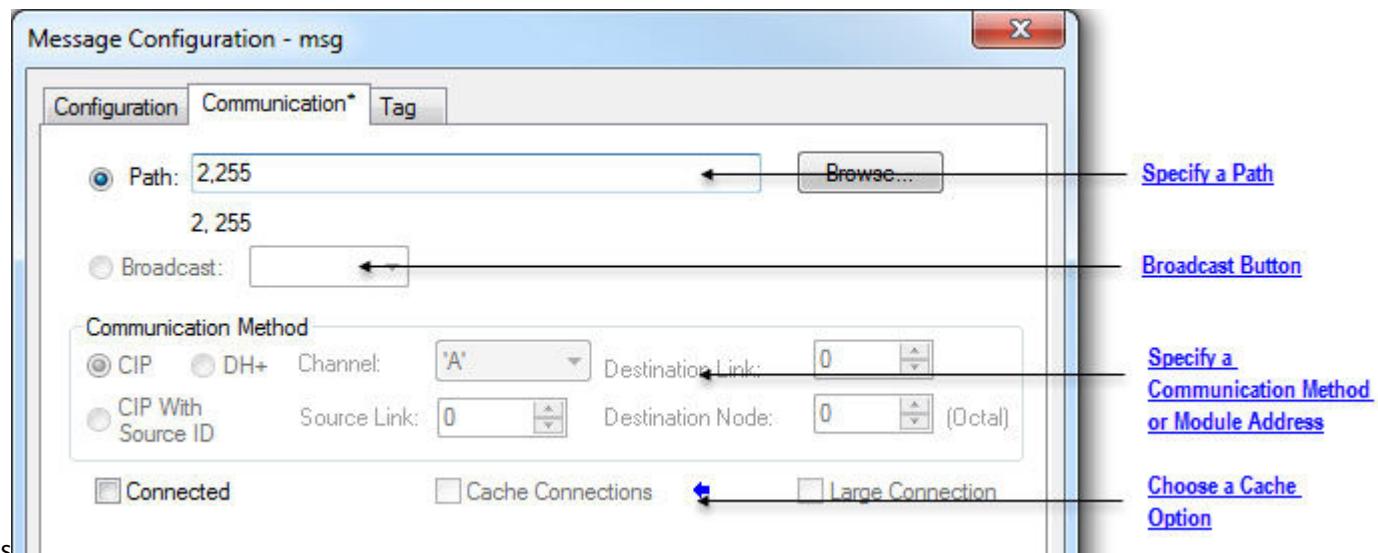
Código de error (Hex)	Descripción	Visualización en el software
00D0	El escáner no ha recibido una respuesta de transferencia en bloque por parte del módulo de transferencia en bloque en los 3,5 segundos posteriores a la solicitud.	Error desconocido
00D1	La suma de comprobación de la respuesta leída no coincidía con la suma de comprobación del flujo de datos.	
00D2	El escáner ha solicitado una lectura o escritura, pero el módulo de transferencia en bloque ha respondido lo contrario.	
00D3	El escáner ha solicitado una longitud y el módulo de transferencia en bloque ha respondido con una longitud diferente.	
00D6	El escáner ha recibido una respuesta del módulo de transferencia en bloque indicando el fallo de la solicitud de escritura.	
00EA	El escáner no se ha configurado para una comunicación con el rack que pudiese contener este módulo de transferencia en bloque.	
00EB	La ranura lógica especificada no está disponible para este tamaño de rack.	
00EC	Actualmente hay una solicitud de transferencia en bloque en progreso y es necesaria una respuesta antes de poder iniciar otra solicitud.	
00ED	El tamaño de la solicitud de transferencia en bloque no es consecuente con las solicitudes de tamaños válidos de transferencia en bloque.	
00EE	El tipo de solicitud de transferencia en bloque no es consecuente con el BT_READ o BT_WRITE esperado.	
00EF	El escáner no ha podido encontrar una ranura disponible en la tabla de transferencia en bloque que pueda albergar la solicitud de transferencia en bloque.	
00F0	El escáner ha recibido una solicitud para restablecer los canales E/S remotos mientras había transferencias en bloque pendientes.	
00F3	Las colas para las transferencia en bloque remotas están llenas.	
00F5	No hay ningún canal de comunicación configurado para el rack o ranura solicitado.	
00F6	No hay ningún canal de comunicación configurado para E/S remotas.	
00F7	El tiempo de espera de la transferencia en bloque, fijado en la instrucción, terminó antes de finalizar.	
00F8	Error en el protocolo de transferencia en bloque: transferencia en bloque no solicitada.	

00F9	Se han perdido los datos de la transferencia en bloque debido a un canal de comunicación erróneo.	
00FA	El módulo de transferencia en bloque ha solicitado una longitud diferente a la de la instrucción de transferencia en bloque asociada.	
00FB	La suma de comprobación de los datos de lectura de transferencia en bloque era errónea.	
00FC	Había una transferencia inválida de datos de escritura entre el adaptador y el módulo de transferencia en bloque.	
00FD	La suma del tamaño de la transferencia en bloque y del índice en la tabla de datos de transferencia en bloque era mayor que el tamaño del archivo de la tabla de datos de transferencia en bloque.	

## Especificación de detalles de comunicación

Se configura una difusión en programas de texto estructurado o lógica de escalera. En la lógica de escalera, añada un reglón y haga clic en las propiedades del **MSG** para acceder al cuadro de diálogo **Configuración de mensaje** (Message Configuration) y configurar un nuevo mensaje. En el texto estructurado, escriba **MSG** (aMsg) y haga clic en el botón derecho en el aMsg para abrir el cuadro de diálogo **Configuración de mensaje** (Message Configuration) y configurar el mensaje.

Para configurar una instrucción MSG, especifique lo siguiente en la pestaña **Comunicación** (Communication):



### Especificar una ruta

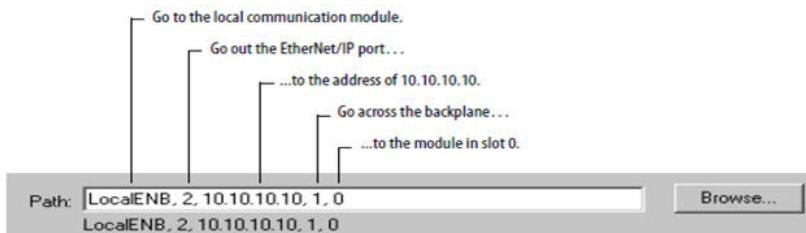
Esta ruta muestra el camino que toma el mensaje para llegar al destino. Utiliza los nombres de la configuración E/S del controlador, los números que ha introducido o ambos. Puede asignar la ruta por defecto usando el botón de difusión, que debe estar activado junto con el protocolo del sistema y el tipo de mensaje.

Si	Entonces
La configuración E/S del controlador tiene el módulo que recibe el mensaje.	Se navega para seleccionar el módulo.
La configuración E/S del controlador solo tiene el módulo de comunicación local.	Se navega para seleccionar el módulo de comunicación local y escriba el resto de la ruta.
La configuración E/S del controlador no tiene ninguno de los módulos necesarios para el mensaje.	Se escribe la ruta.

**Cons** También es compatible con THIS, que indica una ruta a  
**ejo:** sí mismo. THIS se utiliza para enviar un mensaje desconectado al controlador.

### Ejemplos

La configuración E/S del controlador solo tiene el módulo de comunicación local:



Para escribir una ruta, se usa el formato:

port, next\_address, port, next\_address,

Dónde	Es	
	Para esta red	Tipo
Port	Backplane	1
	DF1 (serie, canal serie 0)	2
	ControlNet	
	EtherNet/IP	
	DH+ canal A	
	DH+ canal B	3

	Canal 1 DF1 (canal serie 1)	
Next_address	Backplane	Número de ranura del módulo
	DF1 (serie)	Dirección de la estación (0-254)
	ControlNet	Número de nodo (1-99 decimal)
	DH+	8# seguido por el número de nodo (1-77 octal) Por ejemplo, para especificar la dirección de nodo octal de 37, escriba 8#37.
	EtherNet/IP	Especifique un módulo en una red EtherNet/IP usando alguno de estos formatos: <ul style="list-style-type: none"><li>• Dirección IP. Por ejemplo, 10.10.10.10</li><li>• Dirección IP:Puerto. Por ejemplo, 10.10.10.10:24</li><li>• Nombre DNS. Por ejemplo, tanks</li><li>• Nombre DNS: Puerto Por ejemplo, tanks:24</li></ul>

### Botón de Difusión (Broadcast)

El botón **Difusión** (Broadcast) se usa con el puerto serial.

- La funcionalidad para el software RSLogix 5000, desde la versión 18, mejora la capacidad de definir la ruta y el tipo de mensaje necesarios a fin de enviar un mensaje a su destino.

Cuando está habilitado, el botón **Difusión** (Broadcast) permite definir una ruta por defecto seleccionando un/unos canal/es disponible/s en un cuadro combinado. El número de canales que aparecerá en el cuadro combinado depende del controlador actual.

Por defecto, el botón **Ruta** (Path) de la pestaña **Comunicación** (Communication) está activo.

Se realizan estos pasos para habilitar el botón **Difusión** (Broadcast) y se selecciona un canal para asignar una ruta por defecto para el mensaje.

1. En el **Organizador de controlador** (Controller Organizer), haga clic con el botón derecho en **Controlador** (Controller) y seleccione **Propiedades** (Properties). Aparecerá el cuadro de diálogo **Propiedades del controlador** (Controller Properties).
2. Haga clic en la pestaña **Protocolo del sistema** (System Protocol).
3. Seleccione **DF1 Maestro** (DF1 Master) en el cuadro **Protocolo** (Protocol). El modo de encuesta establece por defecto 'Basado en mensaje' (el esclavo puede iniciar mensajes).
4. Haga clic en **Aceptar** (OK).

5. En la lógica de escalera, haga clic en el cuadro que está dentro de la etiqueta MSG. Aparecerá el cuadro de diálogo **Configuración de mensaje** (Message Configuration) con la pestaña **Configuración** (Configuration) abierta.
6. En el cuadro **Tipo de mensaje** (Message Type), seleccione **Escritura de tabla de datos CIP** (CIP Data Table Write).
7. Haga clic en **Aceptar** (OK). Acaba de habilitar el botón **Difusión** (Broadcast) en la pestaña **Comunicación** (Communication).
8. Haga clic en la pestaña **Comunicación** (Communication).
9. Junto al botón **Difusión** (Broadcast), seleccione un canal en el cuadro combinado. El número de canales que aparecerá en el cuadro combinado depende del controlador.  
Al seleccionar el canal 0 o 1, la ruta del mensaje correspondiente en el cuadro de diálogo **Configuración de mensaje** (Message Configuration) se establecerá por defecto en 2.255 (canal 0) o 3.255 (canal 1). Ruta (Path) se pone en gris para evitar que introduzca un valor de ruta de manera manual.
10. Haga clic en **Aceptar** (OK).

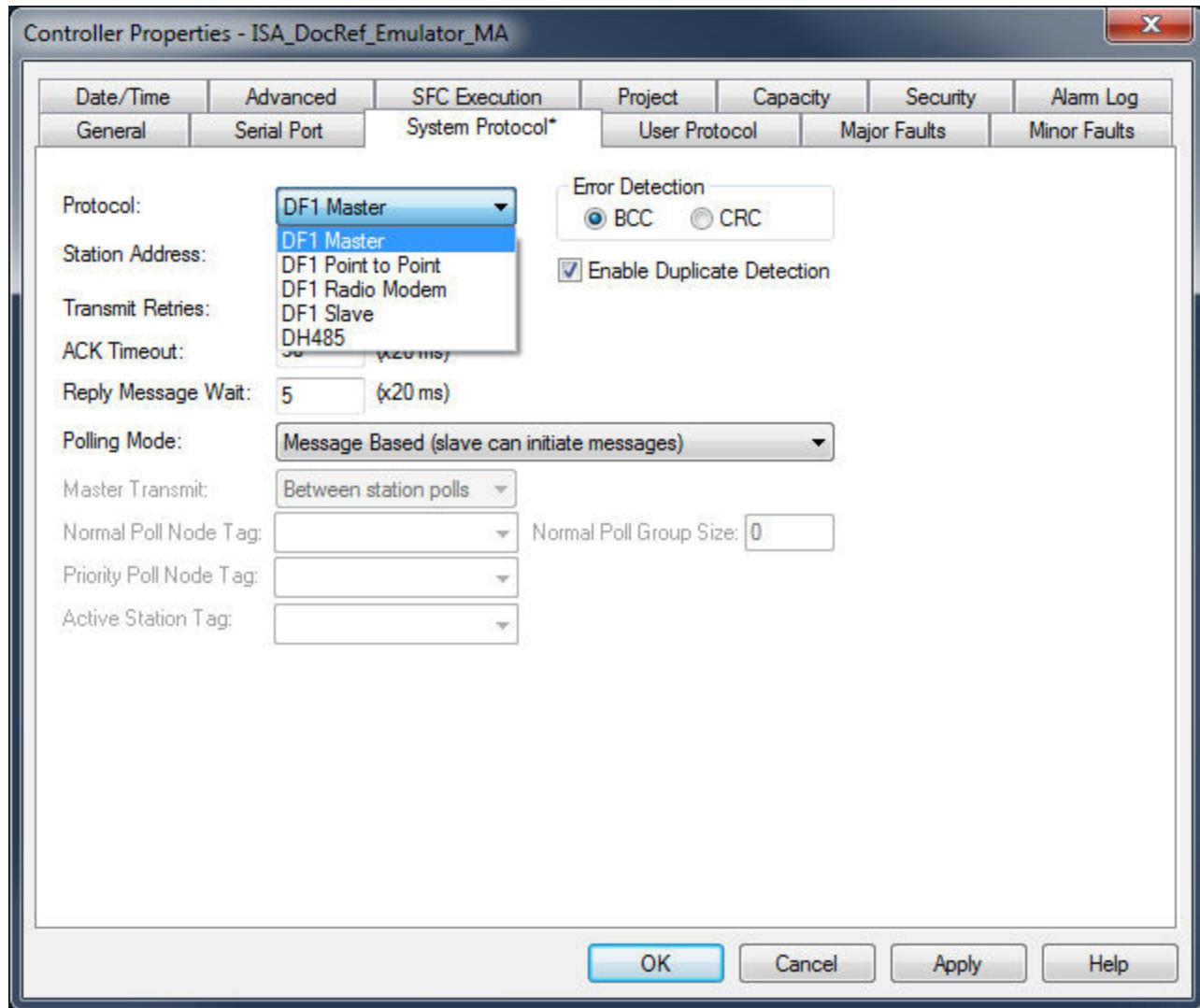
#### Configuración de la pestaña Protocolo del sistema (System Protocol)

Para ejecutar la Difusión (Broadcast) en los controladores ControlLogix en la aplicación Logix Designer, deberá configurar la pestaña **Protocolo del sistema** (System Protocol) en el cuadro de diálogo **Propiedades del controlador** (Controller Properties). El protocolo debe ser compatible con el tipo de mensaje de ‘escritura’ del cuadro de diálogo **Configuración de mensaje** (Message Configuration).

Siga estos pasos para configurar el Protocolo del sistema (System Protocol) de manera que sea compatible con la función Difusión (Broadcast).

1. Cree o abra un controlador ya existente en la aplicación.
2. En el **Organizador de controlador** (Controller Organizer), haga clic con el botón derecho en el nombre del controlador y seleccione **Propiedades** (Properties). Aparecerá el cuadro de diálogo **Propiedades del controlador** (Controller Properties).

3. Si el controlador tiene un puerto serial, haga clic en la pestaña **Protocolo del sistema** (System Protocol).



4. En el cuadro Protocolo (Protocol), seleccione un protocolo.

**IMPORTANTE** El cuadro **Tipo de mensaje** (Message Type) en el cuadro de diálogo **Pestaña de configuración de mensaje** (Message Configuration Tab) debe ser de escritura en pantalla para que sea compatible con el protocolo del sistema. De lo contrario, el botón **Difusión** (Broadcast) se deshabilitará.

5. Introduzca la información en la pestaña **Protocolo del sistema** (System Protocol) para cada protocolo expuesto en las siguientes tablas.

Tema	Descripción
Protocolo (Protocol)	Maestro DF-1 (DF-1 Master)

Dirección de estación (Station Address)	Escriba el número de dirección de estación del controlador
Reintentos de transmisión (Transmit Retries)	3
Tiempo de espera de ACK (ACK Timeout)	50
Espera de mensaje de respuesta (Reply Message Wait)	5
Modo de encuesta (Polling Mode)	<p>Seleccione uno de los siguientes modos:</p> <ul style="list-style-type: none"> <li>• <b>Basado en mensaje</b> Encuestar el esclavo usando una instrucción de mensaje</li> <li>• <b>Esclavo puede iniciar mensaje</b> (Slave can initiate message) para que el esclavo realice la difusión</li> <li>• <b>Estándar</b> (Standard). para tener programada la encuesta del esclavo</li> </ul>
Detección de error (Error Detection)	BCC
Detección de duplicados (Duplicate Detection)	Habilitada (marcada)
Tema	Descripción
Protocolo (Protocol)	Esclavo DF-1
Dirección de estación (Station Address)	Escriba el número de dirección de estación del controlador
Reintentos de transmisión (Transmit Retries)	3
Tiempo de espera de encuesta de esclavos (Slave Poll Timeout)	3000
Supresión de EOT (EOT Suppression)	Inhabilitada (desmarcada)
Detección de error (Error Detection)	BCC
Detección de duplicados (Duplicate Detection)	Habilitada (marcada)
Tema	Descripción
Protocolo (Protocol)	Esclavo DF-1
Dirección de estación (Station Address)	Escriba el número de dirección de estación del controlador
Habilitar almacenamiento y enviar (Enable Store and Forward)	Habilitar cuadro (marca de verificación) para usar la etiqueta de usar almacenamiento y enviar
Detección de error (Error Detection)	BCC

6. Haga clic en **Aceptar** (OK).

### Para las transferencias en bloque

Para los mensajes de transferencia en bloque, se añaden los siguientes módulos a la configuración E/S del controlador:

Para las transferencias en bloque a través de esta red:	Se añaden estos módulos a la configuración E/S:
ControlNet	Módulo de comunicación local (por ejemplo, el módulo 1756-CNB) Módulo adaptador remoto (por ejemplo, el módulo 1771-ACN)
E/S remotas universales	Módulo de comunicación local (por ejemplo, el módulo 1756-DHRIo) Un módulo adaptador remoto (por ejemplo, el módulo 1771-ASB) para cada bastidor, o sección de rack, en el chasis. Módulo de transferencia en bloque (opcional)

### Especificar un método de comunicación o dirección de módulo

Se usa la siguiente tabla para seleccionar un método de comunicación o dirección de módulo para el mensaje:

Si el dispositivo de destino es	Seleccione	Y especifique	
Controlador LOGIX 5000	CIP	Channel	No son necesarias más especificaciones.
Controlador PLC-5 por red EtherNet/IP			
Controlador PLC-5 por red ControlNet			
Controlador SLC 5/05			
Controlador PLC-5 por red DH+	DH+	Vínculo de origen	ID del vínculo asignado al backplane del controlador en la tabla de encaminamiento del módulo 1756-DHRIo. El nodo de origen en la tabla de encaminamiento será automáticamente el número de ranura del controlador.
Controlador SLC por red DH+		Vínculo de destino	ID del vínculo DH+ remoto en donde reside el dispositivo de destino.
Procesador PLC-3		Nodo de destino	Dirección de estación del dispositivo de destino, en octal.
Procesador PLC-2			

			Si hay solo un vínculo DH+ y no se ha usado el software RSLinx Classic para configurar el módulo DH/RIO para vínculos remotos, se especifica 0 para el Vínculo de origen y el Vínculo de destino.
Aplicación en una estación de trabajo que recibe un mensaje no solicitado encaminado a través de una red EtherNet/IP o ControlNet red mediante el software RSLinx Classic o FactoryTalk Linx.	CIP con ID de origen  Esto permite que la aplicación reciba datos de un controlador.	Vínculo de origen	ID remoto del tema del software RSLinx Classic o del acceso directo en FactoryTalk Linx.
		Vínculo de destino	ID de vínculo virtual configurado en el software RSLinx Classic o FactoryTalk Linx (0...65535).
		Nodo de destino	ID de destino (0...77 octal) proporcionado por la aplicación a RSLinx Classic o FactoryTalk Linx. Para un tema DDE en RSLinx Classic, utilice 77.
			El número de ranura del controlador ControlLogix se usa como el Nodo de Source.
Módulo de transferencia en bloque a través de una red de E/S remotas universales.	RIO	Channel	Channel A o Channel B del módulo 1756-DHRIO conectado a la red RIO.
		Rack	Número de rack (octal) del módulo.
		Grupo	Número de grupo del módulo.
		Ranura	Número de ranura del módulo.
Módulo de transferencia en bloque a través de una red ControlNet.	ControlNet	Ranura	Número de ranura del módulo.

### Seleccione una opción de caché

Puede usar una conexión para enviar o recibir datos en función de la configuración de una instrucción MSG.

Tipo de mensaje:	Método de comunicación:	Usa una connexión:
Escritura o lectura de tabla de datos CIP		Su opción (1)
PLC-2, PLC-3, PLC-5 o SLC (todos los tipos)	CIP CIP con ID de origen	
	DH+	X
CIP genérico		Su opción (2)
Lectura o escritura de transferencia en bloque		X

- Los mensajes de lectura o escritura de tabla de datos CIP se pueden conectar o desconectar. Para la mayoría de aplicaciones, Rockwell Automation le

recomienda dejar CIP los mensajes de lectura o escritura de tabla de datos conectados.

2. Los mensajes genéricos CIP se pueden conectar o desconectar. Sin embargo, para la mayoría de aplicaciones, le recomendamos que deje los mensajes genéricos CIP desconectados.

Si una instrucción MSG utiliza una conexión, tiene la opción de dejar la conexión abierta (caché) o cerrarla cuando el mensaje se haya transmitido.

<b>Si usted:</b>	<b>Entonces:</b>
La conexión en caché	La conexión permanece abierta después de que la instrucción MSG haya finalizado. Esto optimiza el tiempo de ejecución. Si se abre una conexión cada vez que se ejecuta un mensaje, se aumenta el tiempo de ejecución.
No almacena la conexión en caché	La conexión se cierra después de que la instrucción MSG haya finalizado. Esto libera la conexión para otros usos.

El controlador tiene los siguientes límites en el número de conexiones que puede almacenar en caché.

<b>Si tiene este controlador:</b>	<b>Entonces, puede almacenar en caché:</b>
CompactLogix 5370 o ControlLogix 5570	Hasta 32 conexiones.
ControlLogix 5580	Hasta 256 conexiones.

Si varios mensajes van al mismo dispositivo, pueden compartir una conexión.

<b>Si las instrucciones MSG van:</b>	<b>Y están:</b>	<b>Entonces:</b>
A diferentes dispositivos		Cada instrucción MSG usa 1 conexión.
Al mismo dispositivo	Habilitadas al mismo tiempo	Cada instrucción MSG usa 1 conexión.
	NO habilitadas al mismo tiempo	La instrucción MSG usa 1 conexión y 1 búfer de caché. Comparten la conexión y el búfer

**Consejo:** A la hora de compartir una conexión, si el controlador cambia entre enviar un mensaje de lectura y escritura de transferencia en bloque, ambos mensajes cuentan como una conexión. Al almacenar en caché ambos mensajes, quedarán registrados como uno solo en la lista de caché.

### Pautas

Cuando planear y programar las instrucciones MSG, siga estas pautas:

Pauta	Detalles
1. Cree una etiqueta de control para cada instrucción MSG.	<p>Cada instrucción MSG necesita su propia etiqueta de control.</p> <p>Tipo de dato = MESSAGE</p> <p>Alcance = controlador</p> <p>La etiqueta no puede ser parte de un tipo de dato definido por el usuario o de una matriz.</p>
1. Mantenga los datos de origen y/o destino al alcance del controlador.	<p>Una instrucción MSG solo puede acceder a etiquetas que estén en la carpeta Etiquetas del controlador (al alcance del controlador)</p>
1. Si la MSG se dirige a un dispositivo que emplea enteros de 16 bits, use un búfer de INT en la MSG y DINT durante todo el proyecto.	<p>Si el mensaje se dirige a un dispositivo que emplea enteros de 16 bits, como un controlador PLC-5 o SLC 500, y transfiere enteros (no REAL), utilice un búfer de INT en el mensaje y DINT durante todo el proyecto.</p> <p>Esto aumentará la eficacia del proyecto, ya que los controladores Logix realizan la ejecución de manera más eficiente y emplean menos memoria al trabajar con enteros de 32 bits (DINT).</p> <p>Para realizar la conversión entre INT y DINT, consulte el <a href="#">Logix 5000 Controllers Common Procedures Programming Manual</a>, publicación <a href="#">1756-PM001</a>.</p>
1. La conexión en caché las MSG que se ejecuten con más frecuencia.	<p>La conexión en caché para las instrucciones MSG que se ejecutan con más frecuencia, hasta el número máximo permisible para la revisión del controlador.</p> <p>Esto optimizará el tiempo de ejecución porque el controlador no tendrá que abrir una conexión cada vez que se ejecuta el mensaje.</p>
5. En el caso de los controladores CompactLogix 5370 o ControlLogix 5570, si desea habilitar más de 16 MSG al mismo tiempo, deberá utilizar algún tipo de estrategia de gestión. En el caso de los controladores ControlLogix 5580, si desea habilitar más de 256 MSGs al mismo tiempo, deberá utilizar algún tipo de estrategia de gestión.	<p>En el caso de los controladores CompactLogix 5370 o ControlLogix 5570, si habilita más de 16 MSGs al mismo tiempo, es posible que algunas instrucciones MSG experimenten retrasos al entrar en la cola.</p> <p>En el caso de los controladores ControlLogix 5580, si habilita más de 256 MSGs al mismo tiempo, es posible que algunas instrucciones MSG experimenten retrasos al entrar en la cola.</p> <p>Para garantizar que todos los mensajes se ejecuten, utilice una de esta opciones:</p> <ul style="list-style-type: none"> <li>Habilitar cada mensaje en secuencia.</li> <li>Habilitar los mensajes en grupos.</li> </ul> <p>Programar un mensaje para que se comunique con diferentes dispositivos. Para más información, consulte el <a href="#">LOGIX 5000 Controllers Common Procedures Programming Manual</a>, publicación <a href="#">1756-PM001</a>.</p> <p>Programar la lógica para coordinar la ejecución de los mensajes. Para más información, consulte el <a href="#">LOGIX 5000 Controllers Common Procedures Programming Manual</a>, publicación <a href="#">1756-PM001</a>.</p>

6. (Solo para los controladores CompactLogix 5370 o ControlLogix 5570) Mantenga de que el número de MSG no conectadas ni almacenadas en caché es inferior al número de búferes no conectados.	El controlador puede tener 10-40 búferes no conectados. El número predeterminado es 10 para los controladores CompactLogix 5370 o ControlLogix 5570.
	Si todos los búferes no conectados están en uso cuando una instrucción abandona la cola de mensajes, la instrucción sufrirá un error y no se enviarán los datos.
	Puede aumentar el número de búferes no conectados (hasta un máximo de 40), pero deberá seguir observando el pautas 5.
	Para aumentar el número de búferes no conectados, consulte el <a href="#">LOGIX 5000 Controllers Common Procedures Programming Manual</a> , publicación <a href="#">1756-PM001</a> .

## Especificar mensajes SLC

Se utiliza los tipos de mensajes SLC para comunicarse con los controladores SLC y MicroLogix. La siguiente tabla especifica los tipos de datos a los que la instrucción le permite tener acceso. La tabla también muestra los tipos de datos LOGIX 5000 correspondientes.

Para este tipo de datos SLC o MicroLogix:	Usa este tipo de datos LOGIX 5000:
F	REAL
L (controladores MicroLogix 1200 y 1500)	DINT
N	INT

## Especificar mensajes de transferencia en bloques

Los tipos de mensajes de transferencia en bloques se utilizan para comunicar los módulos de transferencia en bloque con la red E/S remotas universales.

Para:	Seleccione este comando:
Leer datos de un módulo de transferencia en bloque. Este tipo de mensaje reemplaza la instrucción BTR.	Lectura de transferencia en bloque (Block-Transfer Read)
Escribir datos para un módulo de transferencia en bloque. Este tipo de mensaje reemplaza la instrucción BTW.	Escritura de transferencia en bloque (Block-Transfer Write)

Para configurar un mensaje de transferencia en bloques, siga estas directrices:

- Las etiquetas de origen (para BTW) y de destino (para BTR) tienen que ser lo suficientemente grandes para aceptar los datos necesarios, a excepción de las estructuras MESSAGE, AXIS y MODULE.
- Especifica cuántos enteros de 16 bits (INT) va a mandar o recibir. Puede elegir desde 0 a 64 enteros.

**Cons ejo:** Para hacer que el módulo de transferencia en bloque determina cuántos enteros de 16 bits va a mandar (BTR), o para que el controlador manda 64 enteros (BTW), escriba **0** en el número de elementos.

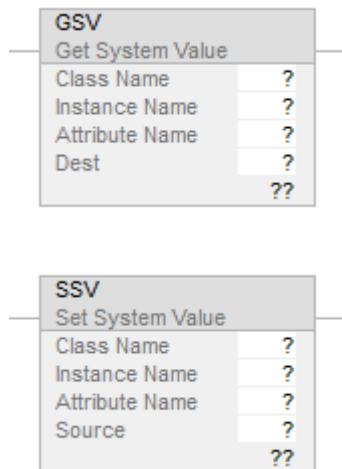
## Obtener valor del sistema (GSV) y Establecer valor del sistema (SSV)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

Las instrucciones GSV/SSV obtienen y establecen los datos del sistema del controlador que están almacenados en objetos.

### Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones

Estas instrucciones no están disponibles en el bloque de funciones.

### Texto estructurado

GSV(ClassName,InstanceName,AttributeName,Dest)

SSV(ClassName,InstanceName,AttributeName,Source)

### Operandos

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción. Consulte *Conversión de datos*.

### Diagrama de escalera y Texto estructurado

Operando	Tipo	Formato	Descripción
Class name		nombre	El nombre de la clase del objeto
Instance name		nombre	El nombre del objeto específico, si el objeto requiere un nombre.
Attribute name		nombre	El atributo del objeto El tipo de dato depende del atributo seleccionado.
Destination (GSV)	SINT INT DINT REAL estructura	etiqueta	El destino de los datos de atributo.
Source (SSV)	SINT INT DINT REAL estructura	etiqueta	La etiqueta que contiene los datos a copiar al atributo.

### Descripción

Las instrucciones GSV/SSV obtienen y establecen los datos de estado del controlador que están almacenados en objetos. El controlador almacena los datos de estado en objetos. No hay un archivo de estado, como en el procesador PLC-5.

Si es verdadero, la instrucción GSV recupera la información especificada y la coloca en el destino. Si es verdadero, la instrucción SSV establece el atributo especificado con los datos del origen.

Cuando introduce una instrucción GSV o SSV, el software de programación muestra las clases de objeto, los nombres de objeto y los nombres de atributo válidos para cada instrucción. Para la instrucción GSV, se pueden obtener los valores para todos los atributos. Para la instrucción SSV, el software solo muestra los atributos que se pueden establecer (SSV).

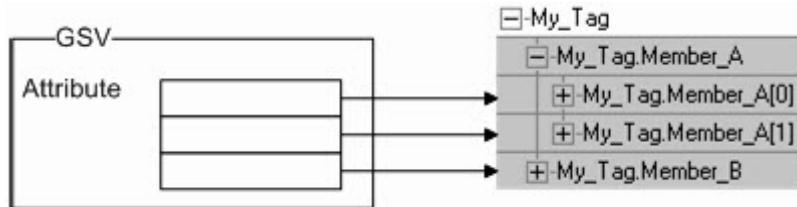


Use las instrucciones SSV con cuidado. Al efectuar cambios en objetos, el controlador puede funcionar de manera inesperada o provocar lesiones en el personal.

Debe asegurarse y confirmar que las instrucciones no cambian ningún dato que no desee cambiar.

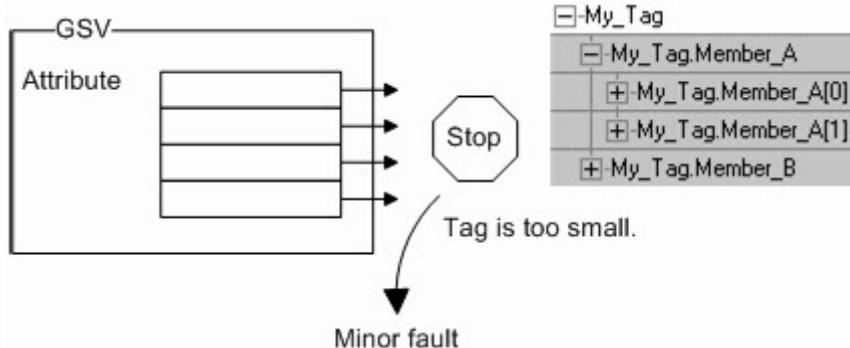
Las instrucciones SSV y las instrucciones GSV escriben y leen respectivamente un miembro hacia otros miembros de la etiqueta. Si la etiqueta es demasiado pequeña, las instrucciones no podrán escribir o leer los datos. En su lugar, se registrará un fallo menor.

#### Ejemplo 1



Member \_A es demasiado pequeño para el atributo. Por lo tanto, la instrucción GSV escribe el último valor en Member \_B.

#### Ejemplo 2



My \_Tag es demasiado pequeño para el atributo. Por lo tanto, la instrucción GSV se detiene y registra un fallo menor. La etiqueta de Destination permanece inalterada.

Los objetos GSV/SSV definen todos los atributos del objeto y sus tipos de datos asociados. Por ejemplo, el atributo MajorFaultRecord del objeto del programa requiere un tipo de dato DINT[11].

#### Afectar a las marcas de estado matemático

No.

#### Fallos mayores/menores

Ocurrirá un fallo menor si:	Tipo de fallo	Código de fallo
Hay una dirección de objeto no válida	4	5
El objeto especificado que no es compatible con GSV/SSV	4	6
Hay un atributo no válido	4	6
No había suficiente información para una instrucción SSV.	4	6

El destino GSV no era suficientemente largo como para almacenar los datos solicitados.	4	7
--	---	---

Consulte los *Atributos comunes* para fallos relacionados con el operando.

## Ejecución

### Diagrama de escalera

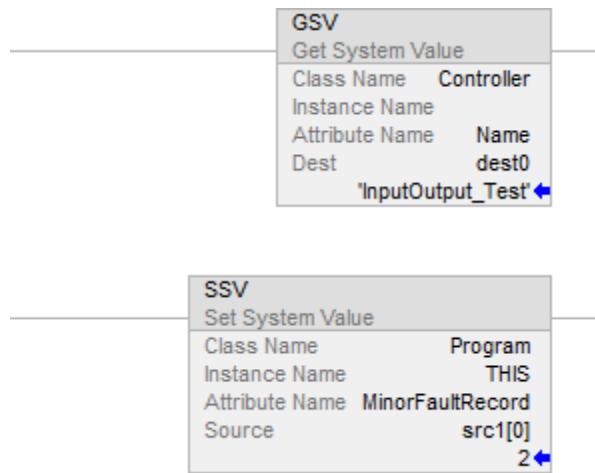
Condición	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	La instrucción se ejecuta.
Post-escaneado	N/A

### Texto estructurado

Condición	Acción realizada
Pre-escaneado	Consultar Pre-escaneado en la tabla de Diagramas de escalera.
Ejecución normal	Consultar La condición de entrada de reglón es verdadera en la tabla de Diagramas de escalera
Post-escaneado	Consultar Post-escaneado en la tabla de Diagramas de escalera.

## Ejemplo

### Diagramas de escalera



### Texto estructurado

GSV (Program,THIS,LASTSCANTIME,dest1);

SSV (Program, THIS, MinorFaultRecord, src[0]);

### Consulte también

[Conversiones de datos](#) en la página 927

[Atributos comunes](#) en la página 923

[Objetos GSV/SSV](#) en la página 284

[Objetos de seguridad GSV/SSV](#) en la página 336

[Ejemplo de programación GSV/SSV](#) en la página 280

## Salida inmediata (IOT)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción IOT actualiza de forma inmediata los datos de salida especificados (etiqueta de salida de un módulo E/S o etiqueta producida). La conexión al módulo debe ser abierta para habilitar la ejecución de la instrucción IOT.

### Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

### Texto estructurado

IOT (output\_tag)

## Operandos

Diagrama de escalera

Operando	TIPO	FORMATO	DESCRIPCIÓN
Update Tag		Etiqueta	Etiqueta que contiene datos a copiar en la etiqueta de atributos que desee actualizar. Ya sea: La etiqueta de salida de un módulo E/S o etiqueta producida

## Texto estructurado

Los operandos son los mismos que para la instrucción IOT del diagrama de escalera.

Consulte Sintaxis de texto estructurado para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

## Descripción

La instrucción IOT anula el intervalo de paquete solicitado (RPI) de una conexión de salida y envía datos nuevos a través de una conexión.

Una conexión de salida es una conexión que está asociada a la etiqueta de salida de un módulo E/S o a una etiqueta producida. Si la conexión es para una etiqueta producida, la instrucción IOT también enviará el desencadenador de evento al controlador de consumo. Esto permitirá que la instrucción IOT desencadene una tarea de evento en el controlador de consumo.

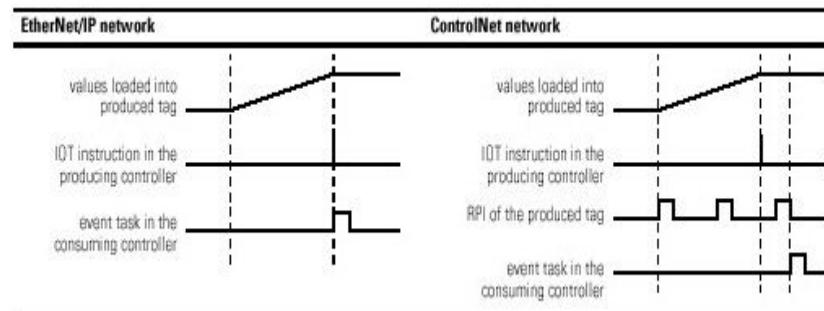
Para usar una instrucción IOT y una etiqueta producida con el fin de desencadenar una tarea de evento en un controlador de consumo, marque la casilla de verificación Enviar Desencadenador de evento a consumidor por programa (instrucción IOT) (Programmatically (IOT Instruction) Send Event Trigger to Consumer) en el cuadro de diálogo **Propiedades de etiqueta** (Tag Properties).

**Consejo:** Para los Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580. (al controlar E/S 5069 por una red remota) se utiliza una optimización para agrupar las conexiones de módulos configuradas con el mismo índice RPI en un solo paquete de envío por la red. Si se utiliza la IOT en una de estas etiquetas, se podría provocar una actualización de algunas etiquetas de datos para otros módulos que están configurados con el mismo RPI, en el mismo backplane y agrupados junto con esa etiqueta. Si prefiere que no ocurra, puede evitarlo fijando un RPI que no sea exactamente igual al de otras conexiones de módulos.

El tipo de red entre los controladores determina el momento en el que el controlador de consumo recibe los nuevos datos y el desencadenador de evento a través de la instrucción IOT.

A través de esta red	El dispositivo de consumo recibe los datos y el desencadenador de evento.
Backplane	Inmediatamente
EtherNet/IP	Inmediatamente
ControlNet	Dentro del intervalo real de paquete (API) de la etiqueta consumida (conexión)

Los siguientes diagramas comparan la recepción de datos a través de una instrucción IOT por redes EtherNet/IP y ControlNet.



### Afectar a las marcas de estado matemático

No

### Condiciones de fallo

No es específico para esta instrucción. Consulte los Atributos comunes para fallos relacionados con el operando.

## Ejecución

### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	La instrucción actualiza la conexión de la etiqueta especificada y restablece el temporizador RPI de la conexión.
Post-escaneado	N/A

### Texto estructurado

Condición/estado	Acción realizada
Pre-escaneado	N/A
Ejecución normal	Consultar La condición de entrada de reglón es verdadera en el Diagrama de escalera
Post-escaneado	N/A

## Ejemplo

Cuando se ejecuta la instrucción IOT, envía inmediatamente los valores de la etiqueta Local:5:0 al módulo de salida.

### Diagrama de escalera



### Texto estructurado

IOT (Local:5:0);

### Consulte también

[Atributos comunes](#) en la página 923

[Sintaxis de texto estructurado](#) en la página 955

## Acceso a los valores del sistema

Este procedimiento le ayudará obtener o usar información sobre el controlador LOGIX 5000.

Si desea:	Consulte este tema de ayuda:
usar palabras clave específicas en su lógica para monitorizar eventos específicos	<a href="#">Marcas de estado de monitor</a> en la página 342

obtener o establecer los valores del sistema	<a href="#">Obtener y establecer valores del sistema</a> en la página 278
obtener información sobre la memoria del controlador	<a href="#">Determinar la información de la memoria del controlador</a> en la página 272

## Determinar la información de la memoria del controlador

La memoria del controlador se divide en memoria de E/S y memoria de expansión. La siguiente tabla muestra cómo el controlador utiliza cada tipo de memoria:

Esto	Utiliza la memoria de
etiquetas de E/S	memoria de E/S
etiquetas producidas	
etiquetas consumidas	
comunicación a través de instrucciones MSG	
comunicación con estaciones de trabajo	
etiquetas que no sean de E/S, producidas o consumidas	memoria de expansión
rutinas de lógica	
comunicación con etiquetas de encuesta (OPC/DDE) que utilizan RSLinx Classic.	memoria de expansión y memoria de E/S

Tenga en cuenta que el controlador devuelve valores en número de palabras de 32 bits. Para ver un valor en bytes, simplemente multiplique por 4.

Se utiliza este procedimiento para obtener la siguiente información sobre la memoria del controlador:

- E/S disponible (libre) y memoria de expansión;
- E/S total y memoria de expansión;
- bloque contiguo mayor de E/S y memoria de expansión

### Obtener información de la memoria desde el controlador

Para obtener información de la memoria desde el controlador, ejecute una instrucción Mensaje (MSG) configurada de la siguiente manera:

Desde el cuadro de diálogo Propiedades del mensaje (Message Properties), pestaña Configuración (Configuration):

For this item	Type or select	Which means:																																							
Message Type	CIP Generic	Execute a Control and Information Protocol command.																																							
Service Type	Custom	Create a CIP Generic message that is not available in the drop-down list.																																							
Service Code	3	Use the GetAttributeList service. This lets you read specific information about the controller.																																							
Class	72	Get information from the user memory object.																																							
Instance	1	This object contains only 1 instance.																																							
Attribute	0	Null value																																							
Source Element	<i>source_array</i> of type SINT[12]	<table border="1"> <thead> <tr> <th>In this element</th> <th>Enter</th> <th>Which means:</th> </tr> </thead> <tbody> <tr> <td><i>source_array[0]</i></td> <td>5</td> <td>Get 5 attributes</td> </tr> <tr> <td><i>source_array[1]</i></td> <td>0</td> <td>Null value</td> </tr> <tr> <td><i>source_array[2]</i></td> <td>1</td> <td>Get free memory</td> </tr> <tr> <td><i>source_array[3]</i></td> <td>0</td> <td>Null value</td> </tr> <tr> <td><i>source_array[4]</i></td> <td>2</td> <td>Get total memory</td> </tr> <tr> <td><i>source_array[5]</i></td> <td>0</td> <td>Null value</td> </tr> <tr> <td><i>source_array[6]</i></td> <td>5</td> <td>Get largest contiguous block of additional free expansion memory</td> </tr> <tr> <td><i>source_array[7]</i></td> <td>0</td> <td>Null value</td> </tr> <tr> <td><i>source_array[8]</i></td> <td>6</td> <td>Get largest contiguous block of free I/O memory</td> </tr> <tr> <td><i>source_array[9]</i></td> <td>0</td> <td>Null value</td> </tr> <tr> <td><i>source_array[10]</i></td> <td>7</td> <td>Get largest contiguous block of free expansion memory</td> </tr> <tr> <td><i>source_array[11]</i></td> <td>0</td> <td>Null value</td> </tr> </tbody> </table>	In this element	Enter	Which means:	<i>source_array[0]</i>	5	Get 5 attributes	<i>source_array[1]</i>	0	Null value	<i>source_array[2]</i>	1	Get free memory	<i>source_array[3]</i>	0	Null value	<i>source_array[4]</i>	2	Get total memory	<i>source_array[5]</i>	0	Null value	<i>source_array[6]</i>	5	Get largest contiguous block of additional free expansion memory	<i>source_array[7]</i>	0	Null value	<i>source_array[8]</i>	6	Get largest contiguous block of free I/O memory	<i>source_array[9]</i>	0	Null value	<i>source_array[10]</i>	7	Get largest contiguous block of free expansion memory	<i>source_array[11]</i>	0	Null value
In this element	Enter	Which means:																																							
<i>source_array[0]</i>	5	Get 5 attributes																																							
<i>source_array[1]</i>	0	Null value																																							
<i>source_array[2]</i>	1	Get free memory																																							
<i>source_array[3]</i>	0	Null value																																							
<i>source_array[4]</i>	2	Get total memory																																							
<i>source_array[5]</i>	0	Null value																																							
<i>source_array[6]</i>	5	Get largest contiguous block of additional free expansion memory																																							
<i>source_array[7]</i>	0	Null value																																							
<i>source_array[8]</i>	6	Get largest contiguous block of free I/O memory																																							
<i>source_array[9]</i>	0	Null value																																							
<i>source_array[10]</i>	7	Get largest contiguous block of free expansion memory																																							
<i>source_array[11]</i>	0	Null value																																							
Source Length	12	Write 12 bytes (12 SINTs).																																							
Destination	<i>INT_array</i> of type INT[29]																																								

Desde el cuadro de diálogo Propiedades del mensaje (Message Properties), pestaña Comunicación (Configuration):

For this item	Type:
Path	1, <i>slot_number_of_controller</i>

#### Elegir la información de memoria que se desee

La instrucción MSG devuelve la siguiente información a INT\_array (la etiqueta de destino de la instrucción MSG).

**Importante:** Para un controlador 1756-L55M16, la instrucción MSG devuelve dos valores para cada categoría de memoria de expansión. Para determinar la memoria de expansión libre o total de un controlador 1756-L55M16, se agrega ambos valores para la categoría.

If you want the:	Then copy these array elements:	Description:
amount of free I/O memory (32-bit words)	<i>INT_array[3]</i>	lower 16 bits of the 32 bit value
	<i>INT_array[4]</i>	upper 16 bits of the 32 bit value
amount of free expansion memory (32-bit words)	<i>INT_array[5]</i>	lower 16 bits of the 32 bit value
	<i>INT_array[6]</i>	upper 16 bits of the 32 bit value
1756-L55M16 controllers only—amount of additional free expansion memory (32-bit words)	<i>INT_array[7]</i>	lower 16 bits of the 32 bit value
	<i>INT_array[8]</i>	upper 16 bits of the 32 bit value
total size of I/O memory (32-bit words)	<i>INT_array[11]</i>	lower 16 bits of the 32 bit value
	<i>INT_array[12]</i>	upper 16 bits of the 32 bit value
total size of expansion memory (32-bit words)	<i>INT_array[13]</i>	lower 16 bits of the 32 bit value
	<i>INT_array[14]</i>	upper 16 bits of the 32 bit value
1756-L55M16 controllers only—additional expansion memory (32-bit words)	<i>INT_array[15]</i>	lower 16 bits of the 32 bit value
	<i>INT_array[16]</i>	upper 16 bits of the 32 bit value
1756-L55M16 controllers only—largest contiguous block of additional free expansion memory (32-bit words)	<i>INT_array[19]</i>	lower 16 bits of the 32 bit value
	<i>INT_array[20]</i>	upper 16 bits of the 32 bit value
largest contiguous block of free I/O memory (32-bit words)	<i>INT_array[23]</i>	lower 16 bits of the 32 bit value
	<i>INT_array[24]</i>	upper 16 bits of the 32 bit value
largest contiguous block of free expansion memory (32-bit words)	<i>INT_array[27]</i>	lower 16 bits of the 32 bit value
	<i>INT_array[28]</i>	upper 16 bits of the 32 bit value

### Convertir INT a DINT

La instrucción MSG devuelve cada valor de memoria como dos INT separados.

- El primer INT representa los 16 bits inferiores del valor.
- El segundo INT representa los 16 bits superiores del valor.

Para convertir los INT separados en un valor utilizable, se utiliza una instrucción Copiar (COP), donde:

En este operando:	Especifique:	Que significa:
Source	el primer INT del par de 2 elementos (16 bits inferiores)	Comenzar con los 16 bits inferiores
Destination	la etiqueta DINT en la que almacenar el valor de 32 bits	Copiar el valor a la etiqueta DINT
Length	1	Copiar una vez el número de bytes en el tipo de datos Destination. En este caso, la instrucción copia 4 bytes (32 bits), que combina los 16 bits inferiores y superiores en un valor de 32 bits.

## Códigos de estado de DeviceNet

Los siguientes valores son los códigos de estado de DeviceNet.

Código de estado	Descripción del estado	Acción recomendada
0-63	Dirección de nodo DeviceNet del escáner o dispositivo esclavo.	Ninguno.
65	La opción AutoScan está activa y el escáner está en modo inactivo.	Ninguno.
67	El escáner es escáner secundario.	Ninguno.
68	El escáner primario no ha detectado ningún escáner secundario.	Configurar otro escáner como escáner secundario.
69	Las configuraciones primaria y secundaria no coinciden.	Comprobar la configuración del escáner secundario.
70	La dirección del escáner ya está en uso por otro dispositivo de la red.	Cambiar la dirección del escáner a una dirección no utilizada.
71	Datos no válidos en la lista de escaneado.	Utilizar el software RSNetWorx para reconfigurar la lista de escaneado.
72	El dispositivo esclavo dejó de comunicarse. Si la comunicación no se restablece con el dispositivo esclavo durante el siguiente intento, el código de estado cambiará a 78.	<ul style="list-style-type: none"> <li>• Verificar la alimentación del dispositivo esclavo y las conexiones de red.</li> <li>• Si se encuesta el dispositivo esclavo, verifique que el tiempo de retardo entre escaneados sea adecuado para que el dispositivo devuelva datos.</li> <li>• Verificar que el dispositivo esclavo esté funcionando correctamente.</li> </ul>
73	La información de identidad del dispositivo esclavo no coincide con la clave electrónica en el escáner.	<ul style="list-style-type: none"> <li>• Asegurarse de que el dispositivo esclavo correcto esté conectado en esta dirección.</li> <li>• Asegurarse de que el dispositivo esclavo coincida con la clave electrónica especificada (proveedor, código de producto, tipo de producto).</li> <li>• Verificar que el dispositivo esclavo esté funcionando correctamente.</li> </ul>
74	El escáner detectó el rebasamiento de datos en el puerto de comunicación DeviceNet.	<ul style="list-style-type: none"> <li>• Comprobar el tráfico de comunicación de red.</li> <li>• Verificar que el dispositivo esclavo esté funcionando correctamente.</li> </ul>
75	Una o ambas de las siguientes condiciones están presentes. <ul style="list-style-type: none"> <li>• El escáner no tiene una lista de escaneado.</li> <li>• El escáner no ha recibido comunicación de ningún otro dispositivo.</li> </ul>	Verificar que el escáner tenga lo siguiente. <ul style="list-style-type: none"> <li>• Una lista de escaneado configurada.</li> <li>• Una conexión correctamente cableada a la red.</li> </ul>
76	No hay tráfico de red directo para el escáner. El escáner escucha otra comunicación de red pero no escucha ninguna dirigida a éste.	Ninguno.

77	Durante la inicialización, el tamaño de datos esperado por el dispositivo esclavo no coincide con el tamaño en la entrada de la lista de escaneado correspondiente.	<ul style="list-style-type: none"> <li>Utilizar el software RSNetWorx para comprobar el dispositivo esclavo y la lista de escaneado para conocer los tamaños de entrada y salida correctos para el dispositivo esclavo.</li> <li>Verificar que el dispositivo esclavo esté funcionando correctamente.</li> </ul>
78	El dispositivo esclavo está configurado en la lista de escaneado, pero no se está comunicando.	<ul style="list-style-type: none"> <li>Verificar la alimentación del dispositivo esclavo y las conexiones de red.</li> <li>Si se encuesta el dispositivo esclavo, asegúrese de que el retardo entre escaneados sea lo suficientemente largo para que el dispositivo esclavo devuelva sus datos.</li> <li>Si es necesario, se utiliza el software RSNetWorx para realizar lo siguiente. <ul style="list-style-type: none"> <li>Añadir el dispositivo esclavo a la red DeviceNet.</li> <li>Eliminar el dispositivo esclavo de la lista de escaneado del escáner.</li> <li>Inhibir el dispositivo esclavo en la lista de escaneado del escáner.</li> </ul> </li> <li>Verificar que el dispositivo esclavo esté funcionando correctamente.</li> </ul>
79	El escáner no ha podido transmitir un mensaje.	<ul style="list-style-type: none"> <li>Asegurarse de que el escáner esté conectado a una red válida.</li> <li>Comprobar si hay cables desconectados.</li> <li>Verificar la velocidad en baudios de la red.</li> </ul>
80	El escáner está en modo inactivo.	<p>Si lo desea, ponga el escáner en modo Marcha haciendo lo siguiente.</p> <ul style="list-style-type: none"> <li>Poniendo el controlador en modo Marcha/Marcha remota mediante el interruptor de llave en el controlador o mediante la aplicación AND Logix Designer</li> <li>Activando el bit O.CommandRegister.Run para el escáner.</li> </ul>
81	El controlador ha establecido el escáner en modo de fallo.	El bit O.CommandRegister.Fault para el escáner está activado. Corregir la condición que provocó que el controlador estableciera este bit y luego desactivarlo.

82	Error detectado en la secuencia de mensajes de E/S fragmentados desde el dispositivo esclavo.	<ul style="list-style-type: none"> <li>Utilizar el software RSNetWorx para realizar lo siguiente.           <ul style="list-style-type: none"> <li>Comprobar la entrada de la lista de escaneado para el dispositivo esclavo para asegurarse de que los tamaños de datos de entrada y salida sean correctos.</li> <li>Comprobar la configuración del dispositivo esclavo.</li> </ul> </li> <li>Verificar que el dispositivo esclavo esté funcionando correctamente.</li> </ul>
83	El dispositivo esclavo devuelve respuestas de error cuando el escáner intenta comunicarse con éste.	<ul style="list-style-type: none"> <li>Utilizar el software RSNetWorx para realizar lo siguiente.           <ul style="list-style-type: none"> <li>Comprobar la precisión de la lista de escaneado.</li> <li>Comprobar la configuración del dispositivo esclavo. El dispositivo esclavo puede estar en la lista de escaneado de otro escáner.</li> </ul> </li> <li>Ciclo de apagado y encendido del dispositivo esclavo.</li> <li>Verificar que el dispositivo esclavo esté funcionando correctamente.</li> </ul>
84	El escáner está inicializando la red DeviceNet.	Ninguno. Este código se borra una vez que el escáner intenta inicializar todos los dispositivos esclavos de la red.
85	Durante el tiempo de ejecución, el tamaño de datos enviado por el dispositivo esclavo no coincide con el tamaño en la entrada de la lista de escaneado correspondiente.	Dado que los datos de encuesta de longitud variable no son compatibles, verifique que el dispositivo esclavo esté funcionando correctamente.
86	El dispositivo esclavo está en modo inactivo o no produce datos mientras el escáner está en modo Marcha.	<ul style="list-style-type: none"> <li>Comprobar la configuración y el estado del dispositivo esclavo.</li> <li>Si establece una relación maestro/esclavo entre 2 escáneres, asegúrese de que ambos escáneres estén en modo Marcha.</li> </ul>
87	El escáner no puede escuchar las entradas compartidas del dispositivo esclavo porque el escáner propietario no ha establecido comunicación con ese dispositivo esclavo.	<ul style="list-style-type: none"> <li>Verificar la conexión y configuración del escáner propietario.</li> <li>Puede ser que el dispositivo esclavo no esté produciendo datos.</li> </ul>
88	El escáner no puede escuchar las entradas compartidas del dispositivo esclavo porque los parámetros de E/S (por ejemplo, encuesta o estroboscopio, clave electrónica, tamaño de datos) para ese dispositivo esclavo están configurados de manera diferente entre este escáner y el escáner propietario.	En este escáner, reconfigurar los parámetros de E/S para la entrada de la lista de escaneado de entradas compartidas de modo que coincidan con esos mismos parámetros en el escáner propietario.
89	El escáner no pudo configurar un dispositivo esclavo utilizando los parámetros de Recuperación automática del dispositivo (ADR).	Asegurarse de haber instalado un dispositivo esclavo compatible.

90	El controlador ha establecido el escáner en modo deshabilitado.	Si lo desea, habilite el escáner desactivando el bit O.CommandRegister.DisableNetwork para el escáner.
91	Condición de bus desactivado probablemente debido a errores de cable o señal.	<ul style="list-style-type: none"> <li>• Ciclo de apagado y encendido en el escáner, dispositivos esclavos o red.</li> <li>• Verificar que todos los dispositivos estén establecidos a la misma velocidad en baudios.</li> <li>• Comprobar el cableado de DeviceNet para asegurarse de que no existan cortocircuitos entre los cables CAN (azul y blanco) y los cables de alimentación o blindaje (negro, rojo y blindado).</li> <li>• Comprobar el sistema de medios para las siguientes fuentes de ruido. <ul style="list-style-type: none"> <li>• Dispositivo ubicado cerca del cable de alimentación de alto voltaje.</li> <li>• Resistencia de terminación incorrecta o no utilizada.</li> <li>• Conexión a tierra incorrecta.</li> </ul> </li> <li>• Dispositivo en la red que produce ruido o datos incorrectos en la red.</li> </ul>
92	El cable de DeviceNet no suministra energía al puerto de comunicación del escáner.	<ul style="list-style-type: none"> <li>• Verificar que la fuente de alimentación de 24V CC de la red funcione correctamente.</li> <li>• Verificar el buen estado de los cables.</li> <li>• Comprobar las conexiones de los cables al escáner.</li> </ul>
95	El firmware del escáner se está actualizando o se está descargando una configuración.	Ninguno. No desconecte el escáner mientras la actualización está en proceso; de lo contrario, se perderán los datos existentes en la memoria del escáner.
97	El controlador ha colocado el escáner en el modo de detención.	El bit O.CommandRegister.HaltScanner para el escáner está activado. Desactivar este bit y luego realizar un ciclo de apagado y encendido en el escáner.
98	Error de firmware general.	Sustituir el dispositivo.
99	Fallo de sistema.	Sustituir el dispositivo.

## Obtener y establecer valores del sistema

El controlador almacena datos del sistema en objetos. No hay un archivo de estado, como en el controlador PLC-5. Utilizar las instrucciones GSV/SSV para obtener o establecer los datos del sistema del controlador que se almacenan en los objetos:

- La instrucción GSV recupera la información especificada y la coloca en el destino.

- La instrucción SSV establece el atributo especificado con datos del origen.

Atención: Se utiliza la instrucción SSV con cuidado. Al efectuar cambios en objetos, el controlador puede funcionar de manera inesperada o provocar lesiones en el personal.

Para obtener o establecer un valor del sistema:

1. Abra el proyecto de la aplicación Logix Designer.
2. En el menú **Ayuda** (Help), haga clic en **Contenido** (Contents).
3. Haga clic en **Índice** (Index).
4. Escriba **objetos gsv/ssv** (gsv/ssv objects) y haga clic en **Mostrar** (Display).
5. Haga clic en el objeto requerido.

Para obtener o establecer	Haga clic en
eje de un módulo servo	AXIS
fracción de tiempo de costo del sistema	CONTROLLER
hardware físico de un controlador	CONTROLLERDEVICE
hora coordinada del sistema para los dispositivos en un chasis	CST
Driver de comunicación DF1 para el puerto serial (solo para controladores con puertos seriales)	DF1
historial de fallos de un controlador	FAULTLOG
atributos de una instrucción de mensaje	MESSAGE
estado, fallos, ruta de comunicación y modo de un módulo	MODULE
grupo de ejes	MOTIONGROUP
información de fallos o tiempo de escaneado para un programa	PROGRAM
número de instancia de una rutina	ROUTINE
configuración del puerto serial (solo para controladores con puertos seriales)	SERIALPORT
propiedades o tiempo transcurrido de una tarea	TASK
tiempo de reloj de un controlador	WALLCLOCKTIME
estado de sincronización de tiempo de un controlador	TIMESYNCHRONIZE

6. En la lista de atributos para el objeto, identifique el atributo al que deseé tener acceso.
7. Cree una etiqueta para el valor del atributo.

Si el tipo de datos del atributo es	Entonces
un elemento (por ejemplo, DINT)	Cree una etiqueta para el atributo.
más de un elemento (por ejemplo, DINT[7])	Cree un tipo de datos definidos por el usuario que coincida con la organización de los datos utilizados por el atributo. A continuación, cree una etiqueta para el atributo y use el tipo de datos que creó.

8. En su rutina de lógica de escalera, introduzca la instrucción apropiada.

Para	Introduzca esta instrucción
obtener el valor de un atributo	GSV
establecer el valor de un atributo	SSV

9. Asigne los operandos necesarios a la instrucción.

Consulte la instrucción GSV/SSV para obtener información sobre estos operandos.

#### Consulte también

[Obtener valor del sistema \(GSV\) y Establecer valor del sistema](#) en la página 264

## Ejemplo de programación GSV/SSV

Los siguientes ejemplos usan instrucciones GSV para obtener información de fallos.

#### Ejemplo 1: Obtención de información de fallos E/S

Este ejemplo obtiene la información de fallos del módulo E/S disc\_in\_2 y se colocan los datos en una disc\_in\_2\_info de estructura definida por el usuario.

### Diagrama de escalera



### Texto estructurado

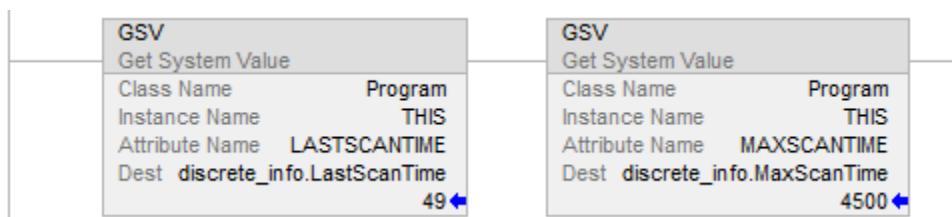
```

GSV(MODULE,disc_in_2,FaultCode,disc_in_2_info.FaultCode);
GSV(MODULE,disc_in_2,FaultInfo,disc_in_2_info.FaultInfo);
GSV(MODULE,disc_in_2,Mode,disc_in_2_info.Mode);
  
```

### Ejemplo 2: Obtención de información sobre el estado del programa

Este ejemplo obtiene información sobre datos discretos del programa y se colocan los datos en una discrete\_info de estructura definida por el usuario.

### Diagrama de escalera



### Texto estructurado

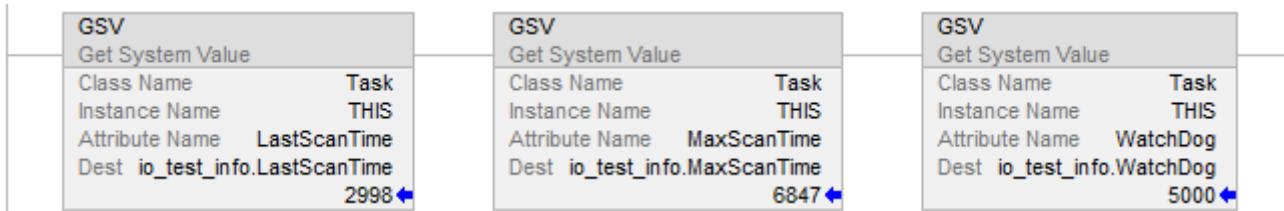
```

GSV(PROGRAM,DISCRETE,LASTSCANTIME,discrete_info.LastScanTime);
GSV(PROGRAM,DISCRETE,MAXSCANTIME,discrete_info.MaxScanTime);
  
```

### Ejemplo 3: Obtención de información sobre el estado de las tareas

Este ejemplo obtiene información sobre la tarea IO\_test y se colocan los datos en una io\_test\_info de estructura definida por el usuario.

#### Diagrama de escalera



#### Texto estructurado

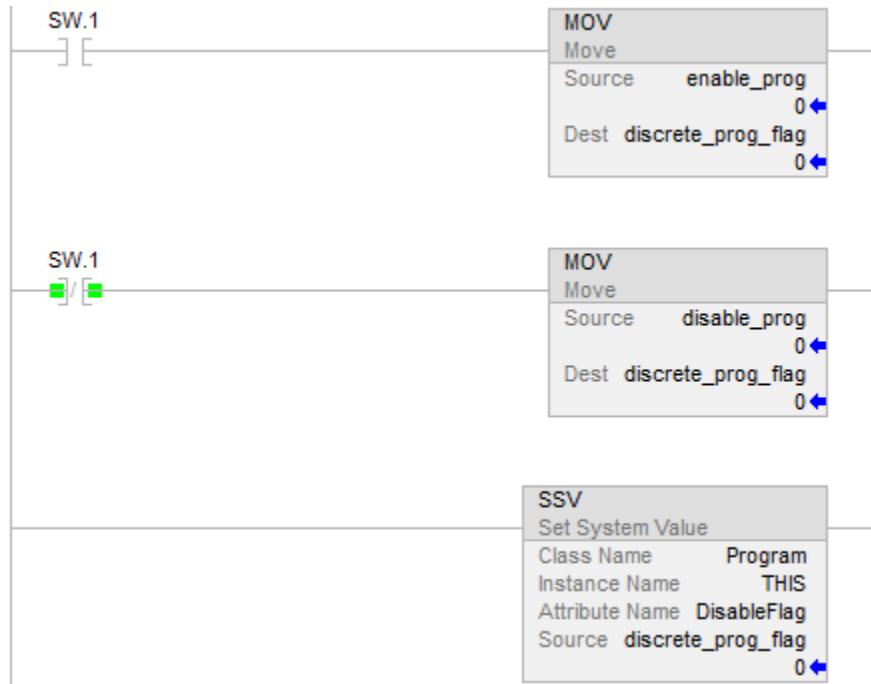
```
GSV(TASK, IO_TEST, LASTSCANTIME, io_test_info.LastScanTime);
GSV(TASK, IO_TEST, MAXSCANTIME, io_test_info.MaxScanTime);
GSV(TASK, IO_TEST, WATCHDOG, io_test_info.Watchdog);
```

#### Establecimiento de marcas de Habilitar y Deshabilitar.

El siguiente ejemplo usa la instrucción SSV para habilitar o deshabilitar un programa. También se puede usar este método para habilitar o deshabilitar un módulo E/S, que es una solución de software similar al uso de bits de inhibición con un procesador PLC-5.

En función del estado de SW.1, coloque el valor adecuado en el atributo de las marcas de deshabilitación de los datos discretos del programa.

### Diagrama de escalera



### Texto estructurado

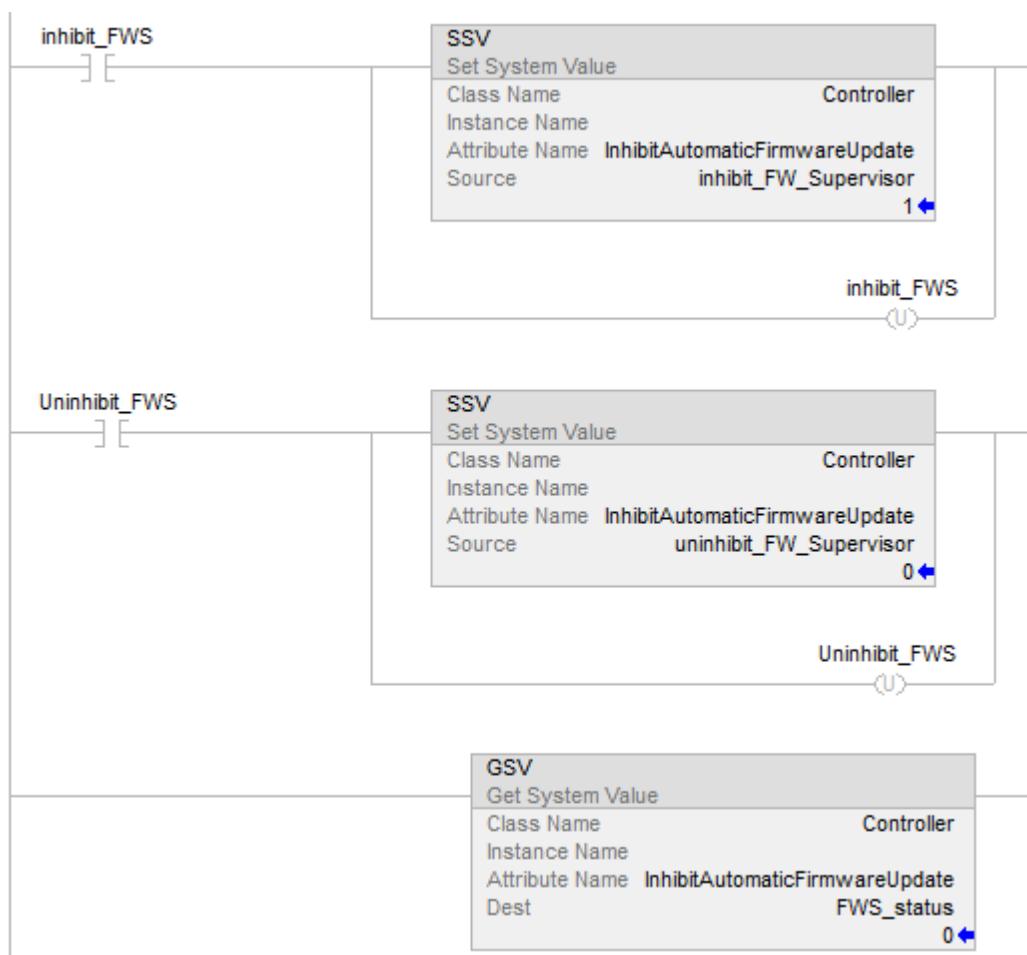
```

IF SW.1 THEN
    discrete_prog_flag := enable_prog;
ELSE
    discrete_prog_flag := disable_prog;
END_IF;
SSV(PROGRAM,DISCRETE,DISABLEFLAG,discrete_prog_flag);
    
```

### Inhibición y desinhibición de la Actualización automática de FirmwareSupervisor

El siguiente ejemplo usa la instrucción GSV/SSV para inhibir o desinhibir el atributo Actualización automática de firmware del controlador. Si escribe el valor 1, la función quedará inhibida. Si escribe el valor 0, la función quedará desinhibida. El estado del atributo también se puede leer con un GSV.

### Diagrama de escalera



## Objetos GSV/SSV

Al introducir una instrucción GSV/SSV, se especifica el objeto y su atributo de acceso. En algunos casos, habrá más de una instancia del mismo tipo de objeto. Asegúrese de especificar el nombre de objeto. Por ejemplo, cada tarea tiene su propio objeto TASK que requiere especificar el nombre de la tarea para obtener acceso.

---

**Importante:** Para la instrucción GSV, solo se copia el tamaño de datos especificado en el destino. Por ejemplo, si el atributo se especifica como SINT y el destino es DINT, solo se actualizan los 8 bits inferiores del destino DINT, sin modificar los 24 bits restantes.

---

---

**Importante:** El búfer de alarma se eliminó de las funciones de suscripción para alarma en firmware de v21 y ya no está disponible. Las instrucciones GSV que anteriormente hicieran referencia al atributo de búfer de alarma se invalidan cuando se verifica el proyecto. Es responsabilidad del programador corregir cualquier código de las aplicaciones que aún use este atributo.

---

Estos son los objetos GSV/SSV. Los objetos disponibles para el acceso dependen del controlador.

- [AddOnInstructionDefinition](#) en la página 286
- [Axis](#) en la página 289
- [Controller](#) en la página 300
- [ControllerDevice](#) en la página 302
- [CoordinateSystem](#) en la página 304
- [CST](#) en la página 308
- [DF1](#) en la página 312
- [FaultLog](#) en la página 315
- [HardwareStatus](#) en la página 316
- [Message](#) en la página 307
- [Module](#) en la página 318
- [MotionGroup](#) en la página 306
- [Program](#) en la página 325
- [Redundancy](#) en la página 321
- [Routine](#) en la página 321
- [Safety](#) en la página 326
- [SerialPort](#) en la página 328
- [Task](#) en la página 329
- [TimeSynchronize](#) en la página 331
- [WallClockTime](#) en la página 335

#### Consulte también

[Obtener valor del sistema \(GSV\) y Establecer valor del sistema \(SSV\)](#) en la página 264

[Instrucciones de entrada/salida](#) en la página 223

## Acceso al objeto AddOnInstructionDefinition

El objeto **AddOnInstructionDefinition** le permite personalizar instrucciones para conjuntos de lógica que se usan con frecuencia, proporciona una interfaz común para esta lógica y suministra documentos para la instrucción.

Para obtener más información, consulte LOGIX 5000 Controllers Add-On Instructions Programming Manual, publicación 1756-PM010.

Atributo	Tipo de datos	Instrucción dentro de la Tarea estándar	Instrucción dentro de la Tarea de seguridad	Descripción
LastEditDate	LINT	GSV	Ninguno	Sello de hora y fecha de la última edición a una definición de Instrucción Add-On.
MajorRevision	DINT	GSV	Ninguno	Número de revisión mayor de la Instrucción Add-On.
MinorRevision	DINT	GSV	Ninguno	Número de revisión menor de la Instrucción Add-On.
Name	Cadena	GSV	GSV	Nombre de la Instrucción Add-On.
RevisionExtendedText	Cadena	GSV	Ninguno	Texto descriptivo de la revisión de la Instrucción Add-On.
SafetySignature ID	DINT	GSV	Ninguno	En un proyecto de seguridad, el número de ID, la fecha y la marca de tiempo de una definición de Instrucción Add-On.
SignatureID	DINT	GSV	Ninguno	Número de identificación de 32 bits de una definición de Instrucción Add-On.
Vendor	Cadena	GSV	Ninguno	Proveedor que ha creado la Instrucción Add-On

### Consulte también

[Códigos y tipos de fallos mayores](#) en la página 235

[Códigos y tipos de fallos menores](#) en la página 241

## Acceso al objeto ALARMBUFFER

El objeto ALARMBUFFER es parte de la infraestructura Publicador/Suscriptor. La infraestructura Publicador/Suscriptor es parte del subsistema de comunicaciones del controlador Logix. El subsistema de comunicaciones del controlador Logix implementa los patrones de mensajes Publicador/Suscriptor para CIP, lo que permite que otros dispositivos reciban mensajes enviados por el subsistema del controlador. Actualmente, los subsistemas Alarmas digitales, Alarmas analógicas y Fase de equipos en lote usan la infraestructura Publicador/Suscriptor para enviar mensajes a través de CIP a aplicaciones suscritas.

Use el objeto ALARMBUFFER para ayudarle a determinar la existencia de conexiones al subsistema Publicador/Suscriptor y su estado. Existe una instancia del objeto AlarmBuffer para cada aplicación suscrita. Esto significa que puede existir un objeto AlarmBuffer en algún momento, pero no existir en otro momento. Es por eso que la instrucción Obtener valor del sistema (GSV) devuelve

un estado como parte de la etiqueta de destino (INT[0].0). Cuando el bit de estado está a cero, lo más probable es que el objeto AlarmBuffer ya no exista.

Atributo	Tipo de datos	Instrucción	Descripción														
AlarmBufferInstance	DINT[n]	GSV	<p>Devuelve los ID del objeto AlarmBuffer.</p> <table border="1"> <tr> <td>DINT[0]</td><td>Número de objetos AlarmBuffer.</td></tr> <tr> <td>DINT[1...n-1]</td><td>Los ID del objeto AlarmBuffer.</td></tr> </table> <p>Si el número de objetos AlarmBuffer es mayor que n-1, solo se devolverán los ID de los primeros objetos (n-1).</p> <p>No es necesario especificar un ID de instancia AlarmBuffer para este atributo.</p>	DINT[0]	Número de objetos AlarmBuffer.	DINT[1...n-1]	Los ID del objeto AlarmBuffer.										
DINT[0]	Número de objetos AlarmBuffer.																
DINT[1...n-1]	Los ID del objeto AlarmBuffer.																
AlarmBufferStatus	INT[2]	GSV	<p>Devuelve el estado del objeto AlarmBuffer especificado. Debe especificar el ID de la instancia AlarmBuffer para obtener el estado de la instancia individual.</p> <table border="1"> <tr> <td>INT[0].0</td><td>1-El atributo AlarmBufferStatus es válido. 0-El atributo AlarmBufferStatus no es válido.</td></tr> <tr> <td>INT[1]</td><td>Valor del atributo AlarmBuffer Status.</td></tr> </table> <p>El atributo Status contiene lo siguiente:</p> <table border="1"> <tr> <td>INT[1].0</td><td>1-Paquete de multimensaje habilitados. 0-Paquete de multimensaje deshabilitados.</td></tr> <tr> <td>INT[1].1</td><td>1-Búfer habilitado. 0-Búfer deshabilitado.</td></tr> <tr> <td>INT[1].2</td><td>1-Datos almacenados en el búfer. 0-El búfer está vacío.</td></tr> <tr> <td>INT[1].3</td><td>1-El búfer está lleno. 0-El búfer no está lleno.</td></tr> <tr> <td>INT[1].4</td><td>1-Los mensajes de Estado de inicialización NO se enviarán (en el momento de la suscripción y en la conmutación de redundancia). 0-Los mensajes de Estado de inicialización SÍ se enviarán.</td></tr> </table> <p>Todos los demás bits están reservados y quedan fijados a 0.</p>	INT[0].0	1-El atributo AlarmBufferStatus es válido. 0-El atributo AlarmBufferStatus no es válido.	INT[1]	Valor del atributo AlarmBuffer Status.	INT[1].0	1-Paquete de multimensaje habilitados. 0-Paquete de multimensaje deshabilitados.	INT[1].1	1-Búfer habilitado. 0-Búfer deshabilitado.	INT[1].2	1-Datos almacenados en el búfer. 0-El búfer está vacío.	INT[1].3	1-El búfer está lleno. 0-El búfer no está lleno.	INT[1].4	1-Los mensajes de Estado de inicialización NO se enviarán (en el momento de la suscripción y en la conmutación de redundancia). 0-Los mensajes de Estado de inicialización SÍ se enviarán.
INT[0].0	1-El atributo AlarmBufferStatus es válido. 0-El atributo AlarmBufferStatus no es válido.																
INT[1]	Valor del atributo AlarmBuffer Status.																
INT[1].0	1-Paquete de multimensaje habilitados. 0-Paquete de multimensaje deshabilitados.																
INT[1].1	1-Búfer habilitado. 0-Búfer deshabilitado.																
INT[1].2	1-Datos almacenados en el búfer. 0-El búfer está vacío.																
INT[1].3	1-El búfer está lleno. 0-El búfer no está lleno.																
INT[1].4	1-Los mensajes de Estado de inicialización NO se enviarán (en el momento de la suscripción y en la conmutación de redundancia). 0-Los mensajes de Estado de inicialización SÍ se enviarán.																
BufferSize	INT[2]	GSV	<p>Devuelve el tamaño del búfer (en kB) del objeto AlarmBuffer especificado. Debe especificar el ID de la instancia AlarmBuffer para obtener el tamaño del búfer de la instancia individual.</p> <table border="1"> <tr> <td>INT[0].0</td><td>1-El atributo BufferSize es válido. 0-El atributo BufferSize no es válido.</td></tr> <tr> <td>INT[1]</td><td>Valor del atributo Buffer Size.</td></tr> </table>	INT[0].0	1-El atributo BufferSize es válido. 0-El atributo BufferSize no es válido.	INT[1]	Valor del atributo Buffer Size.										
INT[0].0	1-El atributo BufferSize es válido. 0-El atributo BufferSize no es válido.																
INT[1]	Valor del atributo Buffer Size.																
BufferUsage	INT[2]	GSV	<p>Devuelve el porcentaje de espacio en el búfer usado por el objeto AlarmBuffer especificado. Debe especificar el ID de la instancia AlarmBuffer para obtener el valor del uso del búfer de la instancia individual.</p> <table border="1"> <tr> <td>INT[0].1</td><td>1-El atributo BufferUsage es válido. 0-El atributo BufferUsage no es válido.</td></tr> <tr> <td>INT[1]</td><td>Valor del atributo BufferUsage</td></tr> </table>	INT[0].1	1-El atributo BufferUsage es válido. 0-El atributo BufferUsage no es válido.	INT[1]	Valor del atributo BufferUsage										
INT[0].1	1-El atributo BufferUsage es válido. 0-El atributo BufferUsage no es válido.																
INT[1]	Valor del atributo BufferUsage																

SubscriberName	STRING	GSV	<p>Devuelve el nombre del suscriptor del objeto AlarmBuffer especificado. Debe especificar la ID de la instancia AlarmBuffer para obtener el nombre del suscriptor de la instancia individual.</p> <p>Cualquier tipo de cadena se puede usar de referencia como una etiqueta de destino.</p> <p>Si el Nombre del suscriptor no cabe en la cadena de la etiqueta de destino proporcionada, la instrucción solo proporcionará la parte del nombre que quepa en la etiqueta de destino.</p> <p>Si la instancia del objeto AlarmBuffer especificada por el ID de la instancia no existe en el momento en que se llama a la instrucción, la longitud de la cadena (miembro .LEN) se fija a cero.</p> <p>Tenga en cuenta que si no se proporciona ningún nombre de suscriptor cuando el suscriptor crea un objeto AlarmBuffer, el atributo de nombre del suscriptor se establecerá en un número de serie de un dispositivo asociado con una conexión que sirva para llamar al servicio Crear en el objeto AlarmBuffer.</p>
----------------	--------	-----	--

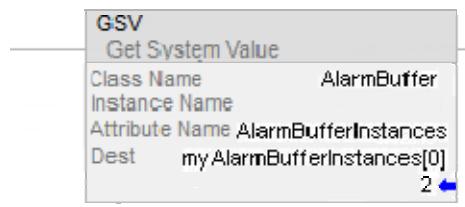
### Ejemplo de instrucción GSV

El programa puede contener una instrucción GSV para obtener la lista actual de AlarmBufferInstances en el controlador. La instrucción devolverá el recuento total de objetos búfer de alarma que haya actualmente en el controlador (DINT[0]) junto con el ID de la instancia de objeto AlarmBuffer asociada (DINT[1] – DINT[n-1]) para cada objeto AlarmBuffer que esté presente en el controlador. La instrucción GSV muestra el valor del número de objetos AlarmBuffer (DINT[0]) bajo el nombre de etiqueta Dest (Destino)

El programa puede usar el ID de la instancia de objeto AlarmBuffer para obtener información relacionada con un instancia específica del objeto AlarmBuffer que esté presente en el controlador. Se devuelve una palabra de estado (INT[0]), indicando si son datos válidos o no, en la etiqueta de destino para los atributos AlarmBufferStatus, BufferSize y BufferUsage, ya que los objetos de búfer de alarma se pueden crear y eliminar en cualquier momento. El valor devuelto se almacena en (INT[1]) cuando Attribute Name es igual a AlarmBufferStatue, BufferSize o BufferUsage. El valor devuelto es el nombre del suscriptor cuando Attribute Name es SubscriberName. No se devuelve ningún estado para el atributo SubscriberName.

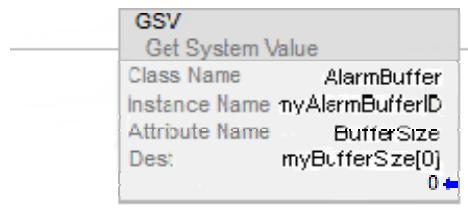
### Diagrama de escalera

A continuación se muestra un ejemplo en el que la instrucción GSV recupera los ID del objeto AlarmBuffer.



Aunque la instrucción GSV de AlarmBufferInstances devuelva los valores en una matriz, no se puede usar la dirección de la matriz para obtener los valores de los atributos de esa instancia. Debe copiar o mover el valor en myAlarmBufferInstances[x] (donde x = 1, 2, 3,...) a una etiqueta directa (no indexada), como se muestra en el ejemplo de myAlarmBufferID en la siguiente figura.

A continuación se muestra un ejemplo en el que la instrucción GSV recupera el tamaño del búfer del objeto AlarmBuffer.



El número mostrado bajo el nombre de etiqueta Dest (Destino) corresponde al valor del bit válido o no válido cuando el nombre de attribute es AlarmBufferStatus, BufferSize o BufferUsage.

### Texto estructurado

A continuación se muestra un ejemplo en el que la instrucción GSV recupera los ID del objeto AlarmBuffer.

- GSV(AlarmBuffer, AlarmBufferInstances, myAlarmBufferInstances[0]);

A continuación se muestra un ejemplo en el que la instrucción GSV recupera el objeto AlarmBuffer.

- GSV(AlarmBuffer, myAlarmBufferID, BufferSize, myBufferSize[0]);

## Acceso al objeto Axis

El objeto AXIS proporciona información de estado sobre un eje. Especifica el nombre de etiqueta del eje para determinar el objeto AXIS que desea.

Para obtener más información sobre el objeto AXIS, consulte el *SERCOS and Analog Motion Configuration and Startup User Manual*, publicación MOTION-UM001.

Cuando un atributo está marcado con un asterisco (\*), significa que el atributo está ubicado en el controlador ControlLogix y en el módulo de movimiento. Al usar una instrucción SSV para escribir uno de estos valores, el controlador actualizará automáticamente la copia en el módulo. Sin embargo, este proceso no es inmediato. Se suministra la etiqueta de estado del eje, ConfigUpdateInProcess, para indicar el momento en el que se completa el proceso.

Por ejemplo, si ejecuta una SSV a PositionLockTolerance, ConfigUpdateInProcess en la etiqueta del eje está establecido hasta que haya una actualización correcta en el módulo. Por lo tanto, la lógica posterior al SSV podría esperar a que este bit se restableciera antes de continuar en el programa.

Atributo	Tipo de datos	Instrucción	Descripción	
* AccelerationFeedForwardGain	REAL	GSV SSV	El porcentaje (%) de salida de comandos de par necesario para generar la aceleración solicitada en el comando.	
ACStopMode	SINT	GSV SSV	El tipo de detención a realizar en el eje.	
			Valor	Significado
			0	detención rápida
			1	desactivación rápida
			2	desactivación por hardware
ActualPosition	REAL	GSV	La posición real del eje en unidades de posición.	
ActualVelocity	REAL	GSV	La velocidad real del eje en unidades de posición/segundo.	
AnalogInput1	REAL	GSV SSV	Este atributo solo aplica a un eje asociado Entrada analógica 2, un variador Kinetix7000. Este atributo tiene un rango de números enteros de +/-16384, y representa el valor analógico de un dispositivo analógico conectado a las entradas analógicas del variador Kinetix7000. Estas entradas son útiles para aplicaciones de conversión web con celda de carga (en la medición de fuerza web en un rodillo) o tensor (en la medición directa de fuerza/posición web) y se pueden conectar directamente al variador que controla la web.	
AverageVelocity	REAL	GSV	La velocidad promedio del eje en unidades de posición/segundo.	
AverageVelocityTimebase	REAL	GSV SSV	La base de tiempo en segundos de la velocidad promedio del eje.	
AxisConfigurationState	SINT	GSV	El estado de la configuración del eje.	
			Valor	Significado
			0 – 126	aún no configurado
			127	datos del eje consumidos no válidos (debido a revisiones de incompatibilidad entre productor y consumidor)
			128	configurado
			3	esperando respuesta
			4	configurado
AxisEventBits	DINT	GSV	Los bits de evento del servo para el servolazo. (En la estructura de AXIS, son el miembro AxisEvent).	
			Bit	Nombre del bit
			0	WatchEventArmed Status
			1	WatchEventStatus

			2	RegEvent1ArmedStatus	registro del armado del evento
			3	RegEvent1Status	evento del registro
			4	HomeEventArmedStatus	evento armado de posición inicial
			5	HomeEventStatus	evento de posición inicial
AxisState	SINT	GSV	El estado de funcionamiento del eje.		
			Valor	Significado	
			0	eje preparado	
			1	control de variador directo	
			2	control de servo	
			3	fallo de eje	
			4	Desactivación de eje	
Bandwidth	REAL	GSV SSV	El ancho de banda de ganancia de unidad (Hz) que el controlador utiliza para calcular la ganancia en una instrucción Aplicar ajustes a eje de movimiento (Motion Apply Axis Tuning, MAAT).		
C2CConnectionInstance	DINT	GSV	La instancia de conexión del controlador que produce los datos del eje.		
C2CMapTableInstance	DINT	GSV	La instancia de asignación del controlador que produce los datos del eje.		
CommandPosition	REAL	GSV	La posición de comando del eje en unidades de posición.		
CommandVelocity	REAL	GSV	La velocidad de comando del eje en unidades de posición.		
ConversionConstant	REAL	GSV SSV	El factor de conversión empleado para convertir de sus unidades a conteos de retroalimentación en unidades de posición/conteo.		
DampingFactor	REAL	GSV SSV	El valor empleado en el cálculo del ancho de banda de servo de posición máximo durante la ejecución de la instrucción Ajuste de eje de movimiento de marcha (MRAT)		
*DriveFaultAction	SINT	GSV SSV	La operación realizada cuando hay un fallo de variador.		
			Valor	Significado	
			0	desconectar el eje	
			1	deshabilitar el variador	
			2	detener el movimiento del comando	
			3	cambiar solo el bit de estado	
DynamicsConfigurationBits	DINT	GSV SSV	La revisión 16 mejoró la gestión del controlador sobre los cambios en un perfil de curva en S. ¿Desea volver al comportamiento de la revisión 15 o anteriores para las curvas en S? NO - Dejar este bit en ON (valor predeterminado). Sí - Poner en OFF uno o más de estos bits:		
			<b>Para desactivar este cambio</b>		<b>Desacactivar este bit</b>

			<b>Retardo de detención en la curva en S reducida</b> Este cambio aplica a la instrucción Detención de movimiento del eje (Motion Axis Stop, MAS). Le permite usar una variación de desaceleración mayor para detener el eje de <b>aceleración</b> más rápidamente. El controlador usa la variación de desaceleración de la instrucción de detención si es mayor que la variación de aceleración actual.	0
			<b>Retrocesos de velocidad en la curva en S reducida</b> Antes de la revisión 16, se podía conseguir que un eje variaría momentáneamente su dirección bajando la variación de desaceleración mientras el eje desaceleraba. Normalmente, esto ocurría al intentar reiniciar un impulso o moverse con un índice de desaceleración menor mientras el eje se detenía. Este cambio evita que el eje retroceda en estas situaciones.	1
			<b>Sobreimpulsos de velocidad en la curva en S reducida</b> Se puede provocar que un eje experimente un sobreimpulso con respecto a su velocidad programada si la variación de aceleración disminuye mientras el eje acelera. Este cambio mantiene los sobreimpulsos por debajo del 50 % de la velocidad programada.	2
FaultConfigurationBits	DINT		Tipo de eje	Configuración de fallos
*FeedbackFaultAction	SINT	GSV SSV	La operación realizada cuando hay un fallo de pérdida de codificador.	
			Valor	Significado
			0	desconectar el eje
			1	deshabilitar el variador
			2	detener el movimiento del comando
			3	cambiar solo el bit de estado
*FeedbackNoiseFaultAction	SINT	GSV SSV	La operación realizada cuando hay un fallo de ruido de codificador.	
			Valor	Significado
			0	desconectar el eje
			1	deshabilitar el variador
			2	detener el movimiento del comando
			3	cambiar solo el bit de estado

*FrictionCompensation	REAL	GSV SSV	El nivel de salida fijado en voltios que se utiliza para compensar la fricción estática.
GroupInstance	DINT	GSV	El número de instancia del grupo de movimiento que contiene el eje.
HardOvertravelFaultAction	SINT	GSV SSV	<b>Valor</b> <b>Significado</b>
			0      desactivación
			1      deshabilitar el variador
			2      detener el movimiento
			3      sólo estado
HomeConfigurationBits	DINT	GSV SSV	Los bits de configuración de movimiento para el eje.
			Bit      Significado
			0      dirección de la posición inicial
			1      interruptor de posición inicial normalmente cerrado
			2      Flanco negativo del marcador de posición inicial
HomeMode	SINT	GSV SSV	El modo posición inicial para el eje.
			Valor      Significado
			0      posición inicial pasiva
			1      posición inicial activa (valor predeterminado)
			2      absoluto
HomePosition	REAL	GSV SSV	La posición inicial del eje en unidades de posición.
HomeReturnSpeed	REAL	GSV SSV	La velocidad de retorno a la posición inicial del eje en unidades de posición/segundo.
HomeSequence	SINT	GSV SSV	La secuencia de posición inicial para el eje.
			Valor      Significado
			0      posición inicial inmediata
			1      posición inicial de interruptor
			2      posición inicial de marcador
			3      posición inicial marcador-interruptor (valor predeterminado)
HomeSpeed	REAL	GSV SSV	La velocidad de posición inicial del eje en unidades de posición/segundo.
Instance	DINT	GSV	El número de instancia del eje.
InterpolatedActualPosition	REAL	GSV	Para las capturas de posición basadas en tiempo, este atributo proporciona la posición real del eje interpolada.  La posición se especifica en unidades de posición y se basa en el valor del atributo InterpolationTime.  Para interpolar una posición de eje real, use una instrucción SSV para establecer el atributo InterpolationTime.

InterpolatedCommandPosition	REAL	GSV	<p>Para las capturas de posición basadas en tiempo, este atributo proporciona la posición interpolada del eje en el comando.</p> <p>La posición se especifica en unidades de posición y se basa en el valor del atributo InterpolationTime.</p> <p>Para interpolar una posición de eje en el comando, use una instrucción SSV para establecer el atributo InterpolationTime.</p>																											
InterpolationTime	DINT	GSV SSV	<p>Use este atributo para proporcionar una referencia para las capturas de la posición basadas en tiempo.</p> <p>Para interpolar una posición, use una instrucción SSV para establecer el atributo InterpolationTime. A continuación, el controlador actualiza los siguientes atributos:</p> <ul style="list-style-type: none"> <li>• InterpolatedActualPosition</li> <li>• InterpolatedCommandPosition</li> </ul> <p>Para proporcionar un valor a InterpolationTime, puede usar cualquier evento que produzca una marca de tiempo CST, como:</p> <ul style="list-style-type: none"> <li>• El atributo RegistrationTime</li> <li>• marca de tiempo de una salida digital</li> </ul> <p>El atributo InterpolationTime solo usa los 32 bits menores de una marca de tiempo CST.</p>																											
MapTableInstance	DINT	GSV	La instancia de asignación E/S del módulo servo.																											
MasterOffset	REAL	GSV	Desplazamiento de posición que se aplica actualmente al maestro de una leva de posición. Especificado en unidades de posición del eje maestro.																											
MaximumAcceleration	REAL	GSV SSV	La aceleración máxima del eje en unidades de posición/segundo <sup>2</sup> .																											
MaximumDeceleration	REAL	GSV SSV	La desaceleración máxima del eje en unidades de posición/segundo <sup>2</sup> .																											
*MaximumNegativeTravel	REAL	GSV SSV	El límite de recorrido negativo máximo en unidades de posición.																											
*MaximumPositiveTravel	REAL	GSV SSV	El límite de recorrido positivo máximo en unidades de posición.																											
MaximumSpeed	REAL	GSV SSV	La velocidad máxima del eje en unidades de posición/segundo.																											
ModuleChannel	SINT	GSV	El canal del módulo servo.																											
MotionStatusBits	DINT	GSV	<p>Los bits de estado de movimiento para el eje. (En la estructura de AXIS, son el miembro MotionStatus).</p> <table border="1"> <thead> <tr> <th>Bit</th> <th>Nombre del bit</th> <th>Significado</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>AccelStatus</td> <td>aceleración</td> </tr> <tr> <td>1</td> <td>DecelStatus</td> <td>desaceleración</td> </tr> <tr> <td>2</td> <td>MoveStatus</td> <td>mover</td> </tr> <tr> <td>3</td> <td>JogStatus</td> <td>impulsar</td> </tr> <tr> <td>4</td> <td>GearingStatus</td> <td>engranaje</td> </tr> <tr> <td>5</td> <td>HomingStatus</td> <td>posición inicial</td> </tr> <tr> <td>6</td> <td>StoppingStatus</td> <td>detener</td> </tr> <tr> <td>7</td> <td>AxisHomedStatus</td> <td>estado de posición de inicio</td> </tr> </tbody> </table>	Bit	Nombre del bit	Significado	0	AccelStatus	aceleración	1	DecelStatus	desaceleración	2	MoveStatus	mover	3	JogStatus	impulsar	4	GearingStatus	engranaje	5	HomingStatus	posición inicial	6	StoppingStatus	detener	7	AxisHomedStatus	estado de posición de inicio
Bit	Nombre del bit	Significado																												
0	AccelStatus	aceleración																												
1	DecelStatus	desaceleración																												
2	MoveStatus	mover																												
3	JogStatus	impulsar																												
4	GearingStatus	engranaje																												
5	HomingStatus	posición inicial																												
6	StoppingStatus	detener																												
7	AxisHomedStatus	estado de posición de inicio																												

			8	PositionCamStatus	leva de posición		
			9	TimeCamStatus	leva de tiempo		
			10	PositionCamPendingStatus	leva de posición pendiente		
			11	TimeCamPendingStatus	leva de tiempo pendiente		
			12	GearingLockStatus	bloqueo de engranajes		
			13	PositionCamLockStatus	bloqueo de leva de posición		
			14	MasterOffsetMoveStatus	movimiento de desplazamiento maestro		
			15	CoordinatedMotionStatus	coordinar movimiento		
			16	TransformStateStatus	transformar estado		
			17	ControlledByTransformStatus	control por transformación		
*OutputLPFilterBandwidth	REAL	GSV SSV	El ancho de banda (Hz) del filtro de salida digital de paso bajo del servo.				
*OutputLimit	REAL	GSV SSV	El valor en voltios de la tensión máxima de salida del servo en el eje.				
*OutputOffset	REAL	GSV SSV	El valor en voltios que se usa para desplazar los efectos de los desplazamientos acumulados en la salida del módulo servo DAC y la entrada del servovariador.				
PositionError	REAL	GSV	La diferencia entre la posición de comando y la posición real del eje.				
*PositionErrorFaultAction	SINT	GSV SSV	La operación que se realiza cuando hay un fallo de posición.				
			Valor	Significado			
			0	desconectar el eje			
			1	deshabilitar el variador			
			2	detener el movimiento del comando			
			3	cambiar solo el bit de estado			
*PositionErrorTolerance	REAL	GSV SSV	La cantidad de error de posición (medido en unidades de posición) que el servo tolera antes de emitir un fallo del error de posición.				
PositionIntegratorError	REAL	GSV	La suma del error de posición en un eje, en unidades de posición.				
*PositionIntegralGain	REAL	GSV SSV	El valor ( $1/\text{msec}^2$ ) usado para lograr una posición exacta del eje a pesar de perturbaciones como la fricción estática y la gravedad.				
PositionLockTolerance	REAL	GSV SSV	La cantidad de error de posición en unidades de posición que el módulo servo tolera al dar una indicación de estado bloqueado de posición verdadera.				
*PositionProportionalGain	REAL	GSV SSV	El valor ( $1/\text{msec}$ ) que el controlador multiplica por el error de posición para corregir el error de posición.				

PositionServoBandwidth	REAL	GSV SSV	El ancho de banda de ganancia de unidad que el controlador utiliza para calcular la ganancia en una instrucción Aplicar ajustes a eje de movimiento (Motion Apply Axis Tuning, MAAT).														
*PositionUnwind	DINT	GSV SSV	El valor usado para realizar el desbobinado automático del eje rotatorio en conteos/revolución.														
ProcessStatus	INT	GSV	<p>El estado de la última instrucción Diagnóstico de conexión en marcha movimiento (Motion Run Hookup Diagnostic, MRHD).</p> <table border="1"> <thead> <tr> <th>Valor</th><th>Significado</th></tr> </thead> <tbody> <tr> <td>0</td><td>proceso de prueba correcto</td></tr> <tr> <td>1</td><td>prueba en curso</td></tr> <tr> <td>2</td><td>proceso de prueba abortado por el usuario</td></tr> <tr> <td>3</td><td>la prueba ha superado el tiempo de espera de 2 segundos</td></tr> <tr> <td>4</td><td>el proceso de prueba ha fallado debido a un fallo del servo</td></tr> <tr> <td>5</td><td>incremento de prueba insuficiente</td></tr> </tbody> </table>	Valor	Significado	0	proceso de prueba correcto	1	prueba en curso	2	proceso de prueba abortado por el usuario	3	la prueba ha superado el tiempo de espera de 2 segundos	4	el proceso de prueba ha fallado debido a un fallo del servo	5	incremento de prueba insuficiente
Valor	Significado																
0	proceso de prueba correcto																
1	prueba en curso																
2	proceso de prueba abortado por el usuario																
3	la prueba ha superado el tiempo de espera de 2 segundos																
4	el proceso de prueba ha fallado debido a un fallo del servo																
5	incremento de prueba insuficiente																
ProgrammedStopMode	SINT	GSV SSV	<p>El tipo de detención a realizar en el eje.</p> <table border="1"> <thead> <tr> <th>Valor</th><th>Significado</th></tr> </thead> <tbody> <tr> <td>0</td><td>detención rápida</td></tr> <tr> <td>1</td><td>desactivación rápida</td></tr> <tr> <td>2</td><td>desactivación por hardware</td></tr> </tbody> </table>	Valor	Significado	0	detención rápida	1	desactivación rápida	2	desactivación por hardware						
Valor	Significado																
0	detención rápida																
1	desactivación rápida																
2	desactivación por hardware																
Registration1Position	REAL	GSV	La posición de registro del eje en unidades de posición.														
RegistrationTime	DINT	GSV	<p>Use este atributo para proporcionar una marca de tiempo para capturas de posición basadas en tiempo:</p> <ul style="list-style-type: none"> <li>el atributo RegistrationTime contiene los 32 bits menores de la marca de tiempo CST de un evento de registro de eje</li> <li>La marca de tiempo CST se mide en microsegundos</li> <li>Para interpolar una posición basada en un evento de registro de eje: <ul style="list-style-type: none"> <li>Use una instrucción GSV para obtener el valor del atributo RegistrationTime.</li> <li>Use una instrucción SSV para establecer el atributo InterpolationTime al valor del atributo RegistrationTime.</li> </ul> </li> </ul>														
RotaryAxis	SINT	GSV Etiqueta	<p>0 = Lineal 1 = Rotatorio</p> <p>Cuando el atributo Rotary Axis se establece como verdadero (1), permite el desbobinado del eje. Esta función proporciona un rango de posición infinito al desbobinar la posición del eje a través de una revolución física completa. El número de conteos de codificador por revolución física del eje se especifica mediante el atributo Position Unwind. Para las operaciones lineales, los conteos no dan la vuelta. Están limitados a +/- 2.000 millones.</p>														

ServoFaultBits	DINT	GSV	Los bits de fallo del servo para el servolazo. (En la estructura de AXIS, son el miembro AxisEvent).
			Bit
			Nombre del bit
			Significado
			0 PosSoftOvertravelFault fallo de fin de carrera positivo
			1 NegSoftOvertravel Fault fallo de fin de carrera negativo
			2 PositionErrorFault fallo de error de posición
			3 FeedbackFault fallo de pérdida en canal A de codificador
			4 FeedbackFault fallo de pérdida en canal B de codificador
			5 FeedbackFault fallo de pérdida en canal Z de codificador
ServoOutputLevel	REAL	GSV	El nivel de tensión de salida en voltios para el servolazo del eje.
ServoStatusBits	DINT	GSV	Los bits de estado para el servolazo. (En la estructura de AXIS, son el miembro ServoStatus).
			Bit
			Nombre del bit
			Significado
			0 ServoActionStatus acción del servo
			1 DriveEnableStatus variador habilitado
			2 OutputLimitStatus límite de salida
			3 PositionLockStatus bloqueo de posición
			13 TuneStatus proceso de ajuste
			14 ProcessStatus diagnóstico de prueba
*SoftOvertravelFaultAction	SINT	GSV SSV	La operación realizada cuando hay un fallo de fin de carrera controlado por software.
			Valor
			Significado
			0 desconectar el eje
			1 deshabilitar el variador
			2 detener el movimiento del comando
			3 cambiar solo el bit de estado

StartActualPosition	REAL	GSV	La posición real en unidades de posición en el eje cuando empieza un nuevo movimiento de comando para el eje.						
StartCommandPosition	REAL	GSV	La posición de comando en unidades de posición en el eje cuando empieza un nuevo movimiento de comando para el eje.						
StartMasterOffset	REAL	GSV	El desplazamiento maestro cuando la última instrucción Desplazamiento eje de movimiento (Motion Axis Move, MAM) ejecutó uno de estos movimientos: <ul style="list-style-type: none"><li>• AbsoluteMasterOffset</li><li>• IncrementalMasterOffset</li></ul> Especificado en unidades de posición del eje maestro.						
StrobeActualPosition	REAL	GSV	La posición real en unidades de posición de un eje cuando se ejecuta la instrucción Posición de estroboscopio de grupo de movimiento (MGSP).						
StrobeCommandPosition	REAL	GSV	La posición de comando en unidades de posición de un eje cuando se ejecuta la instrucción Posición de estroboscopio de grupo de movimiento (MGSP).						
StrobeMasterOffset	REAL	GSV	El desplazamiento maestro cuando se ejecuta la instrucción Posición de estroboscopio de grupo de movimiento (MGSP). Especificado en unidades de posición del eje maestro.						
TestDirectionForward	SINT	GSV	<p>La dirección del recorrido del eje durante la instrucción Diagnóstico de conexión en marcha movimiento (MRHD) según se ve en el módulo servo.</p> <table border="1"> <tr> <td>Valor</td><td>Significado</td></tr> <tr> <td>0</td><td>dirección negativa (retroceso)</td></tr> <tr> <td>1</td><td>dirección positiva (avance)</td></tr> </table>	Valor	Significado	0	dirección negativa (retroceso)	1	dirección positiva (avance)
Valor	Significado								
0	dirección negativa (retroceso)								
1	dirección positiva (avance)								
TestIncrement	REAL	GSV SSV	La cantidad de movimiento necesaria para iniciar una prueba de Diagnóstico de conexión en marcha movimiento (MRHD).						
*TorqueScaling	REAL	GSV SSV	El valor usado para convertir la salida del servolazo en la tensión equivalente en el variador.						
TuneAcceleration	REAL	GSV	El valor de aceleración, en unidades de posición/segundo <sup>2</sup> , medido durante la última instrucción Ajuste de eje de movimiento de marcha (MRAT).						
TuneAccelerationTime	REAL	GSV	El tiempo de aceleración, en segundos, medido durante la última instrucción Ajuste de eje de movimiento de marcha (MRAT).						
TuneDeceleration	REAL	GSV	El valor de desaceleración, en unidades de posición/segundo, medido durante la última instrucción Ajuste de eje de movimiento de marcha (MRAT).						
TuneDecelerationTime	REAL	GSV	El tiempo de desaceleración, en segundos, medido durante la última instrucción Ajuste de eje de movimiento de marcha (MRAT).						
Tunelnertia	REAL	GSV	El valor de inercia en mV/Kconteos/segundo para el eje según se calcula a partir de las medidas que el controlador realiza durante la última instrucción Ajuste de eje de movimiento de marcha (MRAT).						

TuneRiseTime	REAL	GSV	El tiempo de elevación del eje, en segundos, medido durante la última instrucción Ajuste de eje de movimiento de marcha (MRAT).																		
TuneSpeedScaling	REAL	GSV	El factor de escalado en el variador del eje, en mV/Kconteos/segundo, medido durante la última instrucción Ajuste de eje de movimiento de marcha (MRAT).																		
TuneStatus	INT	GSV	<p>El estado de la última instrucción Ajuste de eje de movimiento de marcha (MRAT).</p> <table border="1"> <tr> <td>Valor</td><td>Significado</td></tr> <tr> <td>0</td><td>proceso de ajuste exitoso</td></tr> <tr> <td>1</td><td>ajuste en proceso</td></tr> <tr> <td>2</td><td>proceso de ajuste abortado por el usuario</td></tr> <tr> <td>3</td><td>el ajuste ha superado el tiempo de espera de 2 segundos</td></tr> <tr> <td>4</td><td>el proceso de ajuste ha fallado debido a un fallo del servo</td></tr> <tr> <td>5</td><td>el eje ha llegado al límite de recorrido de ajuste</td></tr> <tr> <td>6</td><td>polaridad del eje establecida de manera incorrecta</td></tr> <tr> <td>7</td><td>velocidad de ajuste demasiado pequeña para realizar medidas</td></tr> </table>	Valor	Significado	0	proceso de ajuste exitoso	1	ajuste en proceso	2	proceso de ajuste abortado por el usuario	3	el ajuste ha superado el tiempo de espera de 2 segundos	4	el proceso de ajuste ha fallado debido a un fallo del servo	5	el eje ha llegado al límite de recorrido de ajuste	6	polaridad del eje establecida de manera incorrecta	7	velocidad de ajuste demasiado pequeña para realizar medidas
Valor	Significado																				
0	proceso de ajuste exitoso																				
1	ajuste en proceso																				
2	proceso de ajuste abortado por el usuario																				
3	el ajuste ha superado el tiempo de espera de 2 segundos																				
4	el proceso de ajuste ha fallado debido a un fallo del servo																				
5	el eje ha llegado al límite de recorrido de ajuste																				
6	polaridad del eje establecida de manera incorrecta																				
7	velocidad de ajuste demasiado pequeña para realizar medidas																				
TuningConfigurationBits	DINT	GSV SSV	<p>Los bits de configuración de ajuste para el eje.</p> <table border="1"> <tr> <td>Bit</td><td>Significado</td></tr> <tr> <td>0</td><td>dirección de ajuste (0=avance, 1=retroceso)</td></tr> <tr> <td>1</td><td>integrador de error de posición de ajuste</td></tr> <tr> <td>2</td><td>integrador de error de velocidad de ajuste</td></tr> <tr> <td>3</td><td>bit de prealimentación de velocidad de ajuste</td></tr> <tr> <td>4</td><td>prealimentación de aceleración</td></tr> <tr> <td>5</td><td>filtro de paso bajo de velocidad de ajuste</td></tr> </table>	Bit	Significado	0	dirección de ajuste (0=avance, 1=retroceso)	1	integrador de error de posición de ajuste	2	integrador de error de velocidad de ajuste	3	bit de prealimentación de velocidad de ajuste	4	prealimentación de aceleración	5	filtro de paso bajo de velocidad de ajuste				
Bit	Significado																				
0	dirección de ajuste (0=avance, 1=retroceso)																				
1	integrador de error de posición de ajuste																				
2	integrador de error de velocidad de ajuste																				
3	bit de prealimentación de velocidad de ajuste																				
4	prealimentación de aceleración																				
5	filtro de paso bajo de velocidad de ajuste																				
TuningSpeed	REAL	GSV SSV	La velocidad máxima, en unidades de posición/segundo, iniciada por la instrucción Ajuste de eje de movimiento de marcha (MRAT).																		
TuningTravelLimit	REAL	GSV SSV	El límite de recorrido usado por la instrucción Ajuste de eje de movimiento de marcha (MRAT) para limitar la acción durante el ajuste.																		
VelocityCommand	REAL	GSV	La referencia de velocidad actual para un eje, en unidades de posición/segundo, al servolazo de velocidad.																		
VelocityError	REAL	GSV	La diferencia, en unidades de posición/segundo, entre la velocidad de comando y la real de un servoeje.																		
VelocityFeedback	REAL	GSV	La velocidad real, en unidades de posición/segundo, del eje según la estimación del módulo servo.																		

*VelocityFeedforwardGain	REAL	GSV SSV	El porcentaje (%) de salida de comando de velocidad necesario para generar la velocidad solicitada en el comando.
*VelocityIntegralGain	REAL	GSV SSV	El valor (1/msec) que el controlador multiplica por el valor VelocityError para corregir el error de velocidad.
VelocityIntegratorError	REAL	GSV	La suma del error de velocidad para un eje especificado.
*VelocityProportionalGain	REAL	GSV SSV	El valor (1/msec) que el controlador multiplica por VelocityError para corregir el error de velocidad.
*VelocityScaling	REAL	GSV SSV	El valor usado para convertir la salida del servolazo en la tensión equivalente en el variador.
VelocityServoBandwidth	REAL	GSV SSV	El ancho de banda (Hz) del variador según se calcula a partir de las medidas realizadas durante la última instrucción Ajuste de eje de movimiento de marcha (MRAT).
WatchPosition	REAL	GSV	La posición de control del eje, en unidades de posición.

### Consulte también

[Códigos y tipos de fallos mayores](#) en la página 235

[Códigos y tipos de fallos menores](#) en la página 241

## Acceso al objeto Controller

El objeto **Controller** proporciona información de estado sobre la ejecución del controlador.

Atributo	Tipo de datos	Instrucción	Descripción
Audit Value	DINT[2], LINT	GSV	<p>El valor de auditoría es un valor único que se genera cuando se descarga un proyecto al controlador o cuando se carga desde un almacenamiento extraíble. El valor se actualiza cuando se detecta un cambio.</p> <p>Para especificar los cambios monitorizados, se usa el atributo ChangesToDetect.</p> <p><b>Consejo:</b> Le recomendamos usar el tipo de datos DINT[2] para evitar limitaciones al trabajar con el tipo de datos LINT en los controladores Rockwell Automation.</p>
ChangesToDetect	DINT[2], LINT	GSV, SSV	<p>Usado para especificar los cambios que se están monitorizando. Cuando hay un cambio monitorizado, Audit Value se actualiza.</p> <p><b>Consejo:</b> Le recomendamos usar el tipo de datos DINT[2] para evitar limitaciones al trabajar con el tipo de datos LINT en los controladores Rockwell Automation.</p>

CanUseRPIFromProducer	DINT	GSV	<p>Identifica si se debe usar el RPI especificado por el productor.</p> <table border="0"> <tr> <td>Valor</td><td>Significado</td></tr> <tr> <td>0</td><td>No se usa el RPI especificado por el productor</td></tr> <tr> <td>1</td><td>Se usa el RPI especificado por el productor</td></tr> </table>	Valor	Significado	0	No se usa el RPI especificado por el productor	1	Se usa el RPI especificado por el productor
Valor	Significado								
0	No se usa el RPI especificado por el productor								
1	Se usa el RPI especificado por el productor								
ControllerLog Execution Modification Count	DINT	GSV SSV	Número de entradas en el registro del controlador que se originan en un cambio de propiedades de programa/tarea, una edición en línea o un cambio de fracción de tiempo del controlador. También se puede configurar para incluir entradas de registro que se originan en forzados. El número se restablece si la RAM entra en un estado incorrecto. El número no está limitado en el mayor DINT y puede haber un reinicio.						
ControllerLog TotalEntryCount	DINT	GSV SSV	Número de entradas en el registro del controlador desde la última actualización de firmware. El número se restablece si la RAM entra en un estado incorrecto. El número está limitado en el mayor DINT.						
DataTablePad Percentage	INT	GSV	Porcentaje (0-100) de memoria libre reservada en la tabla de datos.						
IgnoreArrayFaultsDuringPostScan	SINT	GSV SSV	<p>Se usa para configurar la supresión de los fallos seleccionados que surgen al realizar el post-escaneado de una acción SFC. Solo es válido cuando las SFC se configuran para un restablecimiento automático.</p> <ul style="list-style-type: none"> <li>• 0. Este valor no suprime los fallos que ocurren durante la ejecución del post-escaneado. Este es el valor predeterminado y el comportamiento recomendado.</li> <li>• 1. Este valor suprime de manera automática los fallos mayores 4/20 (subíndice de matriz demasiado grande) y 4/83 (valor fuera de rango) durante las acciones SFC de post-escaneado.</li> </ul> <p>Cuando se suprime un fallo, el controlador usa un administrador de fallos interno para borrar el fallo automáticamente. Esto provoca que se omita la instrucción con fallos y la ejecución se reanuda en la siguiente instrucción</p> <p>Puesto que el administrador de fallos es interno, no es necesario configurar un administrador de fallos para obtener este comportamiento. De hecho, aunque se configurara un administrador de fallos, un fallo suprimido no lo desencadenaría.</p>						
InhibitAutomaticFirmwareUpdate	BOOL	GSV SSV	<p>Identifica si se debe habilitar el supervisor de firmware.</p> <ul style="list-style-type: none"> <li>• 0. Este valor ejecuta el supervisor de firmware.</li> <li>• 1. Este valor no ejecuta el supervisor de firmware.</li> </ul>						

KeepTestEditsOnSwitch over	SINT	GSV	Identifica si se deben mantener ediciones de pruebas en la conmutación del controlador. <ul style="list-style-type: none"> <li>• 0. Este valor deja de comprobar automáticamente las ediciones en la conmutación.</li> <li>• 1. Este valor continúa comprobando las ediciones en la conmutación.</li> </ul>
Name	Cadena	GSV	Nombre del controlador.
Redundancy Enabled	SINT	GSV	Identifica si el controlador se configura para la redundancia. <ul style="list-style-type: none"> <li>• 0. Este valor indica que el controlador no está configurado para la redundancia.</li> <li>• 1. Este valor indica que el controlador está configurado para la redundancia.</li> </ul>
ShareUnused TimeSlice	INT	GSV SSV	Identifica el método que han usado la tarea continua y las tareas en fondo para compartir las fracciones de tiempo no utilizadas. <ul style="list-style-type: none"> <li>• 0. Este valor indica que el sistema operativo no le da el control a la tarea continua aunque las tareas en fondo hayan terminado.</li> <li>• 1. Este valor indica que la tarea continua se ejecuta incluso cuando las tareas en fondo se han completado. Este es el valor predeterminado.</li> <li>• 2. Este valor (o un valor mayor) registra un fallo menor y deja el establecimiento sin cambios.</li> </ul>
TimeSlice	INT	GSV SSV	Porcentaje de CPU disponible (10-90) que ha sido asignado a comunicaciones. Este valor no puede cambiar cuando el interruptor de llave está en la posición Marcha.

### Consulte también

[Códigos y tipos de fallos mayores](#) en la página 235

[Códigos y tipos de fallos menores](#) en la página 241

## Acceso al objeto ControllerDevice

El objeto **ControllerDevice** identifica el hardware físico del controlador.

Atributo	Tipo de datos	Instrucción	Descripción
DeviceName	SINT[33]	GSV	Cadena ASCII que identifica el número de catálogo del controlador y la placa de memoria. El primer byte contiene un conteo del número de caracteres ASCII devueltos en la cadena de matriz.

ProductCode	INT	GSV	Cada valor identifica el tipo de controlador: 15 SoftLogix5800 49 PowerFlex® con DriveLogix5725 52 PowerFlex con DriveLogix5730 53 Emulador 54 1756-L61 ControlLogix 55 1756-L62 ControlLogix 56 1756-L63 ControlLogix 57 1756-L64 ControlLogix 64 1769-L31 CompactLogix 65 1769-L35E CompactLogix 67 1756-L61S GuardLogix 68 1756-L62S GuardLogix 69 1756-LSP GuardLogix 72 1768-L43 CompactLogix 74 1768-L45 CompactLogix 76 1769-L32C CompactLogix 77 1769-L32E CompactLogix 80 1769-L35CR CompactLogix 85 1756-L65 ControlLogix 86 1756-L63S GuardLogix 87 1769-L23E-QB1 CompactLogix 88 1769-L23-QBFC1 CompactLogix 89 1769-L23E-QBFC1 CompactLogix 92 1756-L71 93 1756-L72 94 1756-L73 95 1756-L74 96 1756-L75 106 1769-L30ER 107 1769-L33ER 108 1769-L36ERM 109 1769-L30ER-NSE 110 1769-L33ERM 146 1756-L7SP 147 1756-L72S 148 1756-L73S 149 1769-L24ER-QB1B 150 1769-L24ER-QBFC1B 151 1769-L27ERM-QBFC1B 153 1769-L16ER-BB1B 154 1769-L18ER-BB1B 155 1769-L18ERM-BB1B 156 1769-L30ERM 158 1756-L71S
ProductRev	INT	GSV	Identifica la revisión actual del producto. La visualización debe ser hexadecimal. El byte inferior contiene la revisión mayor; el byte superior contiene la revisión menor.
SerialNumber	DINT	GSV	Número serial del dispositivo. El número serial se asigna cuando se fabrica el dispositivo.

Status	INT	GSV	Bits de estado del dispositivo 7...4 Significado 0000 Reservado 0001 Actualización de Flash en curso 0010 Reservado 0011 Reservado 0100 Flash incorrecto 0101 Modos de fallo 0110 Marcha 0111 Programa  Bits de estado de fallo 11...8 Significado 0001 Fallo menor recuperable 0010 Fallo menor no recuperable 0100 Fallo mayor recuperable 1000 Fallo mayor no recuperable  Bits de estado del controlador 13...12 Significado 01 Interruptor de llave en funcionamiento 10 Interruptor de llave en el programa 11 Interruptor de llave en control remoto 15...14 Significado 01 El controlador está cambiando de modo 10 Modo de depuración si el controlador está en modo Marcha
Type	INT	GSV	Identifica el dispositivo como un controlador. Controlador = 14.
Vendor	INT	GSV	Identifica al proveedor del dispositivo. Allen-Bradley = 0001.

### Consulte también

[Códigos y tipos de fallos mayores](#) en la página 235

[Códigos y tipos de fallos menores](#) en la página 241

## Acceso al objeto CoordinateSystem

El objeto COORDINATESYSTEM proporciona la información de estado de la ejecución del sistema de coordenadas de movimiento.

Atributo	Tipo de datos	Instrucción	Significado
CoordinateMotionStatus	DINT	GSV SSV	Se establece cuando se requiere un bloqueo de eje para una instrucción MCLM o MCCM, y el bloqueo ha pasado la Posición de bloqueo. Se borra cuando se inicia una instrucción MCLM o MCCM.
AccelStatus	BOOL	GSV SSV	Se establece cuando el vector acelera. Se borra cuando se está realizando una combinación o cuando el movimiento del vector va rápido o está frenando.

DecelStatus	BOOL	GSV SSV	Se establece cuando el vector desacelera. Se borra cuando se está realizando una combinación o cuando el movimiento del vector está acelerando o se completa el movimiento.
ActualPosToleranceStatus	BOOL	GSV SSV	Se establece solo con el tipo de terminación de Tolerancia real. Cuando las siguientes dos condiciones se cumplen, se establece el bit. 1) Se ha completado la interpolación. 2) La distancia real al extremo programado es menor que el valor configurado de Tolerancia real del sistema de coordenadas. Una vez finalizada la instrucción, permanece activado. Se restablece cuando empieza una instrucción nueva.
CommandPosToleranceStatus	BOOL	GSV SSV	Se activa con todos los tipos de terminaciones en las que la distancia al extremo programado sea menor al valor configurado de Tolerancia de comando del sistema de coordenadas. Una vez finalizada la instrucción, permanece activado. Se restablece cuando empieza una instrucción nueva.
StoppingStatus	BOOL	GSV SSV	El bit Stopping Status se borra cuando se ejecuta la instrucción MCCM.
MoveStatus	BOOL	GSV SSV	Se establece cuando la instrucción MCCM comienza un movimiento de eje. Se borra en el bit .PC de la última instrucción de movimiento o cuando se ejecuta una instrucción de movimiento que causa una parada.
MoveTransitionStatus	BOOL	GSV SSV	Se establece cuando el tipo terminación de No Decel o Command Tolerance se cumple. Cuando una combinación colineal se mueve, no se establece el bit porque la máquina está siempre en ruta. Se borra cuando se completa una combinación, comienza el movimiento de una instrucción pendiente o se ejecuta una instrucción de movimiento que causa una parada. Indica que no está en ruta.
MovePendingStatus	BOOL	GSV SSV	<p>El bit de movimiento pendiente se establece cuando se pone en cola una instrucción de movimiento coordinado. Cuando la instrucción haya comenzado a ejecutarse, el bit se borrará, siempre y cuando durante ese tiempo no se hayan puesto en cola instrucciones de movimiento coordinado. Si solo hay una instrucción de movimiento coordinado, el usuario podría no detectar el bit de estado en la aplicación Logix Designer, ya que la transición desde en cola a ejecución se realiza más rápidamente que el periodo de actualización.</p> <p>El valor real del bit aparece cuando hay varias instrucciones. Siempre que haya una instrucción en cola, se activará el bit pendiente. Esto le da al programador de la aplicación Logix Designer una manera de reestructurar la ejecución de varias instrucciones de movimiento coordinado. Se puede hacer que la lógica de escalera que contenga instrucciones de movimiento coordinado se ejecute de manera más rápida cuando el programador permita que se pongan en cola otras instrucciones mientras se ejecuta la instrucción anterior. Cuando se borra el bit MovePendingStatus, la siguiente instrucción de movimiento coordinado se puede ejecutar (lo que quiere decir, organización en la cola).</p>

MovePendingQueueFullStat us	BOOL	GSV SSV	Se establece cuando la cola de instrucción está llena. Se borra cuando hay espacio en la cola para que entre una nueva instrucción de movimiento coordinado.
TransformSourceStatus	BOOL	GSV SSV	El sistema de coordenadas es el origen de una transformación activa.
TransformTargetStatus	BOOL	GSV SSV	El sistema de coordenadas es el destino de una transformación activa.
CoorMotionLockStatus	BOOL	GSV SSV	<p>Se establece cuando se requiere un bloqueo de eje para una instrucción MCLM o MCCM, y el bloqueo ha pasado la Posición de bloqueo. Se borra cuando se inicia una instrucción MCLM o MCCM.</p> <p>Para las enumeraciones Inmediato solo hacia delante (Immediate Forward Only) e Inmediato solo inverso (Immediate Reverse Only), el bit se activa inmediatamente cuando se inicia una instrucción MCLM o MCCM.</p> <p>Cuando la enumeración es Posición solo hacia delante (Position Forward Only) o Posición solo inversa (Position Reverse Only), el bit se activa cuando el eje maestro cruza la Posición de bloqueo en la dirección concreta. El bit nunca se establece si la enumeración es NONE.</p> <p>El bit CoordMotionLockStatus se borra cuando el eje maestro cambia la dirección y el eje esclavo deja de seguir al eje maestro. El bit CoordMotionLockStatus se establece de nuevo cuando el sistema de coordenadas esclavo vuelve a seguir al eje maestro. El bit CoordMotionLockStatus también se borra cuando se inicia una instrucción MCS.</p>

## Acceso al objeto MotionGroup

El objeto MOTIONGROUP proporciona información de estado sobre un grupo de ejes para el módulo servo. Especifica el nombre de etiqueta del grupo de movimiento para determinar qué objeto MOTIONGROUP desea.

Atributo	Tipo de datos	Instrucción	Descripción
Alternate1UpdateMultiplier	USINT	GSV	El período de actualización de los ejes que están asociados con el programa de actualización alternativo 1.
Alternate1UpdatePeriod	UDINT	GSV	El período de actualización de los ejes que están asociados con el programa de actualización alternativo 1. El valor es el producto del período de actualización alternativo 1 y el período de actualización aproximado.
Alternate2UpdateMultiplier	USINT	GSV	El período de actualización de los ejes que están asociados con el programa de actualización alternativo 2.
Alternate2UpdatePeriod	UDINT	GSV	El período de actualización de los ejes que están asociados con el programa de actualización alternativo 2. El valor es el producto del período de actualización alternativo 1 y el período de actualización aproximado.

AutoTagUpdate	USINT	GSV SSV	Controla la conversión automática y la actualización de los atributos de estado de movimiento.
CoarseUpdatePeriod	UDINT	GSV	El período de actualización aproximado generalmente se conoce como período de actualización base.
Cycle Start Time	LTIME	GSV	Este valor de 64 bits (ms) corresponde al evento de temporizador que inicia el ciclo de actualización.
INSTANCE	DINT	GSV	El número de instancia de este objeto MOTION_GROUP
MaximumInterval	LTIME	GSV SSV	El intervalo máximo entre ejecuciones sucesivas de esta tarea.
MinimumInterval	LTIME	GSV	El intervalo mínimo entre ejecuciones sucesivas de esta tarea.
StartTime	LTIME	GSV	El valor de tiempo de reloj cuando se inició la última ejecución de la tarea
TaskAverageIOTime	UDINT	GSV SSV	La entrada de la tarea de movimiento promedio hasta el tiempo de salida, es decir, el tiempo transcurrido desde el inicio de la tarea de movimiento hasta enviar los datos de conexión. (Constante de tiempo = 250 CUP)
TaskAverageScanTime	UDINT	GSV SSV	El tiempo promedio de escaneado de la tarea de movimiento. (Constante de tiempo = 250 CUP)
TaskLastIOTime	UDINT	GSV	La entrada de la última tarea de movimiento promedio hasta el tiempo de salida, es decir, el tiempo transcurrido desde el inicio de la tarea de movimiento hasta enviar los datos de conexión.
TaskLastScanTime	UDINT	GSV	El tiempo de escaneado de la última tarea de movimiento. (tiempo transcurrido)
TaskMaximumIOTime	UDINT	GSV SSV	La entrada de la tarea de movimiento máximo hasta el tiempo de salida, es decir, el tiempo transcurrido desde el inicio de la tarea de movimiento hasta enviar los datos de conexión.
TaskMaximumScanTime	UDINT	GSV SSV	El tiempo máximo de escaneado de la tarea de movimiento. (tiempo transcurrido)
Time Offset	LTIME	GSV	El valor de desplazamiento de tiempo entre el tiempo de reloj y el valor del temporizador local para el controlador asociado con el valor actual de Cycle Start Time.

### Consulte también

[Códigos y tipos de fallos mayores](#) en la página 235

[Códigos y tipos de fallos menores](#) en la página 241

## Acceso al objeto Message

Se accede al objeto Message a través de instrucciones GSV/SSV. Se especifica el nombre de etiqueta del mensaje para determinar el objeto Message que desea. El objeto Message proporciona una interfaz para configurar y desencadenar las comunicaciones entre dispositivos similares. Este objeto sustituye el tipo de datos MG del procesador PLC-5.

Atributo	Tipo de datos	Instrucción	Descripción
ConnectionPath	SINT[130]	GSV SSV	Datos para configurar la ruta de conexión. Los primeros dos bytes (bajo y alto) tienen la longitud en bytes de la ruta de la conexión.

ConnectionRate	DINT	GSV SSV	Régimen del paquete solicitado de la conexión.
MessageType	SINT	GSV SSV	Especifica el tipo de mensaje. El valor tiene un significado específico: • 0. No inicializado
Port	SINT	GSV SSV	Indica el puerto por el que se debería enviar el mensaje. Cada valor tiene un significado concreto: • 1. Backplane. • 2. Puerto serial.
Timeout Multiplier	SINT	GSV SSV	Determina el tiempo en el que la conexión debería considerarse finalizada y cerrada. Cada valor tiene un significado concreto: • 0. La conexión sobrepasará el tiempo de espera en 4 veces del régimen de actualización. Este es el valor predeterminado. • 1. La conexión sobrepasará el tiempo de espera en 8 veces del régimen de actualización • 2. La conexión sobrepasará el tiempo de espera en 16 veces del régimen de actualización
Unconnected Timeout	DINT	GSV SSV	Tiempo de espera, en microsegundos, para todos los mensajes desconectados. El valor predeterminado es 30.000.000 microsegundos (30 s).

### Consulte también

[Códigos y tipos de fallos mayores](#) en la página 235

[Códigos y tipos de fallos menores](#) en la página 241

### Acceso al objeto CST

El objeto hora coordinada del sistema (CST) proporciona la hora coordinada del sistema para los dispositivos en un chasis.

Atributo	Tipo de datos	Instrucción	Descripción
CurrentStatus	INT	GSV	<p>Estado actual de la hora coordinada del sistema. Cada bit tiene un significado específico:</p> <ul style="list-style-type: none"> <li>• 0. Hardware del temporizador con fallos. El hardware del temporizador interno del dispositivo se encuentra en estado de fallo.</li> <li>• 1. Rampa habilitada. El valor actual de los 16+ bits inferiores del temporizador suben siguiendo una rampa hasta el valor solicitado, en lugar de romperse hasta el valor más bajo.</li> <li>• 2. Maestro de tiempo del sistema. El objeto CST es un origen de tiempo de maestro en el sistema ControlLogix.</li> <li>• 3. Sincronizado. El objeto CST maestro sincroniza CurrentValue de 64 bits del objeto CST a través de una actualización del tiempo del sistema.</li> <li>• 4. Maestro de red local. El objeto CST es el origen de tiempo de maestro en la red local.</li> <li>• 5. Modo relé. El objeto CST actúa en un modo de relé de tiempo.</li> <li>• 6. Maestro duplicado detectado. Se ha detectado un maestro de tiempo en la red local duplicado. Este bit siempre es 0 para los nodos que dependen del tiempo.</li> <li>• 7. No utilizado.</li> <li>• 8-9. 00. Nodo dependiente del tiempo.</li> <li>• 01. Nodo maestro del tiempo.</li> <li>• 10. Nodo relé de tiempo.</li> <li>• 11. No utilizado.</li> <li>• 10-15. No utilizado.</li> </ul>
CurrentValue	DINT[2]	GSV	Valor actual del temporizador. DINT[0] contiene los 32 bits inferiores; DINT[1] contiene los 32 bits superiores. El origen de tiempo se ajusta para coincidir con el valor suministrado en los servicios de actualización y desde la sincronización de la red de comunicación local. El ajuste es una rampa hasta el valor solicitado o un ajuste inmediato hasta ese mismo valor, tal y como informa el atributo CurrentStatus.

#### Consulte también

[Códigos y tipos de fallos mayores](#) en la página 235

[Códigos y tipos de fallos menores](#) en la página 241

## Acceso al objeto Datalog

El objeto DATALOG proporciona información de estado sobre un registro de datos específico. Especifica el nombre del registro de datos para determinar qué objeto DATALOG desea.

Atributo	Tipo de datos	Instrucción dentro de la Tarea estándar	Instrucción dentro de la Tarea de seguridad	Descripción
CaptureFull	BOOL	GSV	Ninguno	Un estado que indica que: <ul style="list-style-type: none"> <li>• La captura de datos más reciente se detiene para recopilar muestras, o</li> <li>• Las muestras más antiguas en la captura de datos más reciente se sobrescriben debido al tamaño de la captura excedida.</li> </ul>
CollectionCapacity	DINT	GSV	Ninguno	Muestra la frecuencia proporcionada por el controlador sobre cuántos bytes se pueden recopilar por segundo para cada tipo de controlador. El porcentaje de CPU que se utiliza para el registro de datos se puede calcular en función de esta frecuencia y el número de bytes que el controlador debe recopilar para todos los registros de datos configurados.

CollectionState	INT	GSV	Ninguno	<p>Muestra el estado actual de recopilación de datos del registro de datos. Puede ser:</p> <ul style="list-style-type: none"> <li><b>Fuera de línea:</b> no conectado con el controlador.</li> <li><b>Deshabilitado</b> – el registro de datos no se realizará hasta que esté habilitado.</li> <li><b>Esperando desencadenamiento</b> – esperando desencadenamiento de inicio o activación de instantánea. El estado Esperando desencadenamiento de detención se combina con Recopilando muestras. Este estado puede coexistir con Captura llena.</li> <li><b>Recopilando muestras</b> – recopilando muestras activamente, no premuestreos o postmuestreos. El estado Recopilando premuestreos se combina con el estado Esperando desencadenamiento. Este estado puede coexistir con Captura llena.</li> <li><b>Recopilando postmuestreos</b> – se ha producido un desencadenamiento de detención y se están recopilando postmuestreos. Este estado puede coexistir con Captura llena.</li> <li><b>Captura llena:</b> o la captura de datos más reciente ha dejado de recopilar muestras o las muestras más antiguas de la captura de datos más reciente se han sobrescrito debido a que superan el tamaño de la captura. Este estado puede coexistir con Esperando desencadenamiento, Recopilando muestras, Recopilando posmuestreos o Registro de datos lleno.</li> <li><b>Registro de datos lleno</b> – el registro de datos se ha detenido debido a que supera las capturas de datos. Este estado puede coexistir con Captura llena. A la recopilación de datos se le puede cambiar el nombre emitiendo un comando de restablecimiento o un comando de borrado seguido de un comando de habilitar.</li> <li><b>Fallado:</b> ha ocurrido un fallo y se ha detenido la recopilación de datos. No se recopilarán más datos hasta que el fallo se haya borrado y se haya emitido un servicio de habilitar o restablecer. Este estado puede coexistir con Captura llena.</li> </ul>
CurrentCaptureNumber	INT	GSV	Ninguno	Indica el número de la captura actual. Por ejemplo, si la configuración dice que las capturas de datos para guardar son 10, el número de captura actual puede ser de 1 a 10.
DataCapturesToKeep	SINT	GSV	Ninguno	Indica el número configurado de capturas de datos para mantener el registro de datos especificado.
Enabled	SINT	GSV	Ninguno	Indica si el registro de datos especificado está habilitado o no.
FaultReason	INT	GSV	Ninguno	Indica el motivo del fallo actual.

PreviousCaptureUsedStorage	DINT	GSV	Ninguno	Indica la cantidad de almacenamiento utilizado por la captura de datos anterior.
ReservedStorage	DINT	GSV	Ninguno	Indica el porcentaje de almacenamiento total que está reservado para el registro de datos actual.
UsedStorage	DINT	GSV	Ninguno	Indica el porcentaje del almacenamiento total que se llena actualmente con las muestras de datos recopiladas para el registro de datos actual.

**Consulte también**[Códigos y tipos de fallos mayores](#) en la página 235[Códigos y tipos de fallos menores](#) en la página 241**Acceso al objeto DF1**

El objeto DF1 proporciona una interfaz al driver de comunicación DF1 que se puede configurar para el puerto serial.

Atributo	Tipo de datos	Instrucción	Descripción
ACKTimeout	DINT	GSV	La cantidad de tiempo de espera para confirmar la transmisión de un mensaje (solo punto a punto y maestro). Los valores válidos son 0-32.767. Retardo en conteos de períodos de 20 ms. El valor por defecto es 50 (1 segundo).
Diagnostic Counters	INT[19]	GSV	Matriz de contadores de diagnóstico para el driver de comunicación DF1.

Desplazamiento de palabras		DF1 punto a punto	DF1 slaveMaster
0	Firma (0x0043)	Firma (0x0042)	Firma (0x0044)
1	Bits de módem	Bits de módem	Bits de módem
2	Paquetes enviados	Paquetes enviados	Paquetes enviados
3	Paquetes recibidos	Paquetes recibidos	Paquetes recibidos
4	Paquetes no entregados	Paquetes no entregados	Paquetes no entregados
5	No utilizado	Mensajes reintentados	Mensajes reintentados
6	NAK recibidos	NAK recibidos	No utilizado
7	ENQ recibidos	Paquetes de encuesta recibidos	No utilizado
8	Paquetes deficientes con NAK	Paquetes deficientes sin ACK	Paquetes deficientes sin ACK
9	Sin memoria enviado NAK	Sin memoria sin ACK	No utilizado
10	Paquetes duplicados recibidos	Paquetes duplicados recibidos	Paquetes duplicados recibidos
11	Caracteres deficientes recibidos	No utilizado	No utilizado
12	Conteo de recuperaciones DCD	Conteo de recuperaciones DCD	Conteo de recuperaciones DCD

13	Conteo de módem perdido	Conteo de módem perdido	Conteo de módem perdido
14	No utilizado	No utilizado	Máximo tiempo de escaneado prioritario
15	No utilizado	No utilizado	Último tiempo de escaneado prioritario
16	No utilizado	No utilizado	Máximo tiempo de escaneado normal
17	No utilizado	No utilizado	Último tiempo de escaneado normal
18	ENQ enviados	No utilizado	No utilizado
Duplicate Detection	SINT	GSV	Habilita la detección de mensajes duplicados. Cada valor tiene un significado concreto: <ul style="list-style-type: none"> <li>• 0. Detección de mensajes duplicados deshabilitada.</li> <li>• No cero. Detección de mensajes duplicados habilitada.</li> </ul>
Embedded ResponseEnable	SINT	GSV	Habilita la función de respuesta incrustada (solo punto a punto). Cada valor tiene un significado concreto: <ul style="list-style-type: none"> <li>• 0. Se inicia solo después de recibir uno. Este es el valor predeterminado.</li> <li>• 1. Habilitado sin condiciones.</li> </ul>
EnableStoreFwd	SINT	GSV	Habilita el comportamiento de almacenamiento y envío al recibir un mensaje. Cada valor tiene un significado concreto: <ul style="list-style-type: none"> <li>• 0. No enviar mensaje</li> <li>• No cero. Ver la tabla de almacenamiento y envío al recibir un mensaje. Este es el valor predeterminado.</li> </ul>
ENQTransmit Limit	SINT	GSV	El número de consultas (ENQ) que se enviará después de un tiempo de espera de ACK (solo punto a punto). Los valores válidos son 0-127. El valor predeterminado es 3.
EOTSuppression	SINT	GSV	Habilita la supresión de transmisiones EOT en respuesta a los paquetes de encuesta (solo esclavo). Cada valor tiene un significado concreto: <ul style="list-style-type: none"> <li>• 0. Supresión de EOT deshabilitada (deshabilitada).</li> <li>• No cero. Supresión de EOT habilitada.</li> </ul>
ErrorDetection	SINT	GSV	Especifica el plan de detección de errores. Cada valor tiene un significado concreto: <ul style="list-style-type: none"> <li>• 0. BCC. Este es el valor predeterminado.</li> <li>• 1. CRC.</li> </ul>
MasterMessageTransmi t	SINT	GSV	El valor actual de la transmisión de mensajes maestra (solo maestro). Cada valor tiene un significado concreto: <ul style="list-style-type: none"> <li>• 0. Entre encuestas de estaciones. Este es el valor predeterminado.</li> <li>• 1. En la secuencia de la encuesta. Esto sustituye al número de la estación del maestro.</li> </ul>

MaxStationAddress	SINT	GSV	Valor actual (0-31) de la dirección máxima de nodo en una red DH-485. El valor predeterminado es 31.
NAKReceiveLimit	SINT	GSV	El número de NAK recibidos como respuesta a un mensaje antes de detener la transmisión (solo comunicación punto a punto). Los valores válidos son de 0 a 127. El valor predeterminado es 3.
NormalPollGroupSize	INT	GSV	Número de estaciones a encuestar en la matriz de nodo de encuesta normal tras encuestar todas las estaciones en la matriz de nodo de encuesta de prioridad (solo maestro). Los valores válidos son de 0 a 255. El valor predeterminado es 0.
PollingMode	SINT	GSV	Modo de encuesta actual (solo maestro). El valor predeterminado es 1. Cada valor tiene un significado concreto: <ul style="list-style-type: none"> <li>• 0. Basado en mensaje, pero no permite que los esclavos inicien mensajes.</li> <li>• 1. Basado en mensaje, pero permite que los esclavos inicien mensajes. Este es el valor predeterminado.</li> <li>• 2. Estándar, transferencia de un solo mensaje por escaneado de nodo.</li> <li>• 3. Estándar, transferencias de varias mensajes por escaneado de nodo.</li> </ul>
ReplyMessageWait	DINT	GSV	El tiempo (actuando como maestro) de espera después de recibir un ACK antes de encuestar al esclavo para una respuesta (solo maestro). Los valores válidos son de 0 a 65.535. Retardo en conteos de períodos de 20 ms. El valor predeterminado es 5 períodos (100 ms).
SlavePollTimeout	DINT	GSV	La cantidad de tiempo, en milisegundos, que el esclavo espera la encuesta del maestro antes de que el esclavo diga que no puede transmitir porque el maestro está inactivo (solo esclavo). Los valores válidos son de 0 a 32.767. Retardo en conteos de períodos de 20 ms. El valor predeterminado es 3000 períodos (1 minuto).
StationAddress	INT	GSV	Dirección de estación actual en el puerto serial. Los valores válidos son de 0 a 254. El valor predeterminado es 0.
TokenHoldFactor	SINT	GSV	Valor actual (1-4) del número máximo de mensajes enviado por este nodo antes de pasar el token en una red DH-485. El valor predeterminado es 1.
TransmitRetries	SINT	GSV	Número de veces que se puede reenviar un mensaje sin obtener confirmación (solo maestro y esclavo). Los valores válidos son de 0 a 127. El valor predeterminado es 3.
PendingACK Timeout	DINT	SSV	Valor pendiente para el atributo ACKTimeout.
Pending Duplicate Detection	SINT	SSV	Valor pendiente para el atributo DuplicateDetection.
Pending Embedded ResponseEnable	SINT	SSV	Valor pendiente para el atributo EmbeddedResponse.

PendingEnableStoreFwd	SINT	SSV	Valor pendiente para el atributo EnableStoreFwd.
PendingENQTransmitLimit	SINT	SSV	Valor pendiente para el atributo ENQTransmitLimit.
PendingEOTSuppression	SINT	SSV	Valor pendiente para el atributo EOTSuppression.
PendingErrorDetection	SINT	SSV	Valor pendiente para el atributo ErrorDetection.
PendingMasterMessageTransmit	SINT	SSV	Valor pendiente para el atributo MasterMessageTransmit.
PendingMaxStationAddress	SINT	SSV	Valor pendiente para el atributo MaxStationAddress.
PendingNAKReceiveLimit	SINT	SSV	Valor pendiente para el atributo NAKReceiveLimit.
PendingNormalPollGroupSize	INT	SSV	Valor pendiente para el atributo NormalPollGroupSize.
PendingPollingMode	SINT	SSV	Valor pendiente para el atributo PollingMode.
PendingReplyMessageWait	DINT	SSV	Valor pendiente para el atributo ReplyMessageWait.
PendingSlavePollTimeoutut	DINT	SSV	Valor pendiente para el atributo SlavePollTimeout.
PendingStationAddress	INT	SSV	Valor pendiente para el atributo StationAddress.
PendingTokenHoldFactory	SINT	SSV	Valor pendiente para el atributo TokenHoldFactor.
PendingTransmitRetries	SINT	SSV	Valor pendiente para el atributo TransmitRetries.

### Consulte también

[Códigos y tipos de fallos mayores](#) en la página 235

[Códigos y tipos de fallos menores](#) en la página 241

## Acceso al objeto FaultLog

El objeto FaultLog proporciona información de errores sobre el controlador.

Atributo	Tipo de datos	Instrucción	Descripción
MajorEvents	INT	GSV SSV	El número de fallos mayores ocurridos desde la última vez que se restableció el contador.
MajorFaultBits	DINT	GSV SSV	Los bits individuales indican el motivo del fallo mayor

			actual. Cada bit tiene un significado específico: 1 Pérdida de alimentación 3 E/S 4 Ejecución de instrucción (programa) 5 Administrador de fallos 6 Vigilancia 7 Pila 8 Cambio de modo 11 Movimiento
MinorEvents	INT	GSV SSV	El número de fallos menores ocurridos desde la última vez que se restableció el contador.
MinorFaultBits	DINT	GSV SSV	Los bits individuales indican el motivo del fallo menor actual. Cada bit tiene un significado específico: 4 - Ejecución de instrucción (programa) 6 - Vigilancia 9 - Puerto serial 10 - Módulo de almacenamiento de energía (ESM) o sistema de alimentación ininterrumpida (UPS)

### Consulte también

[Códigos y tipos de fallos mayores](#) en la página 235

[Códigos y tipos de fallos menores](#) en la página 241

## Acceso al objeto **HardwareStatus**

El objeto **HardwareStatus** se utiliza para obtener información del estado de UPS, ventiladores y temperaturas con instrucciones GSV para los proyectos del controlador CompactLogix 5480. Este objeto es compatible en las rutinas Diagrama de escalera y Texto estructurado y en las Instrucciones adicionales.

Atributo	Tipo de datos		Instrucción	Descripción
FanSpeeds	Estructura de:		GSV	Velocidad de los ventiladores.
	Number of Fans	USINT		Si el número de ventiladores admitidos por el producto es cero, entonces el dispositivo no admite ventiladores.
	Fan Speed	SINT[9] para 2 ventiladores: SINT[0] = Número de ventiladores SINT[1-4] = Velocidad del ventilador #1 SINT[5-8] = Velocidad del ventilador #2		RPM
FanStatus	Estructura de:		GSV	Indica si el ventilador tiene un error.
	Number of Fan Status Indicators	USINT		Si el número de ventiladores admitidos por el producto es cero, entonces el dispositivo no admite estados de ventilador.

Atributo	Tipo de datos		Instrucción	Descripción
	Fan Status	SINT[3] para 2 ventiladores: SINT[0] = Número de ventiladores SINT[1] = Estado de ventilador #1 SINT[2] = Estado de ventilador #2		<ul style="list-style-type: none"> <li>• 0. El ventilador no da error</li> <li>• 1. El ventilador da error</li> </ul>
TemperatureFaultLevel s	Estructura de:		GSV	Nivel de error en grados Celsius
	Number of Temperature Fault Level	USINT		Si el número del nivel de error de la temperatura es cero, entonces el dispositivo no admite los niveles de error de la temperatura.
	Temperature Fault Level	SINT[3] para 1 sensor de temperatura: SINT[0] = Niveles de error de la temperatura SINT[1-2] = Nivel de error de la temperatura #1		Temperatura en grados Celsius
Temperatures	Estructura de:		GSV	Valores de la temperatura en grados Celsius
	Number of Temperatures	USINT		Si el número de temperaturas admitidas por el producto es cero, entonces el dispositivo no admite temperaturas.
	Temperature	SINT[3] para 1 sensor de temperatura: SINT[0] = Número de temperaturas SINT[1-2] = Temperatura #1		Temperatura en grados Celsius
UPSBatteryFailure	SINT		GSV	Indica si ha fallado la batería del UPS. <ul style="list-style-type: none"> <li>• 0. La batería del UPS conectada no ha detectado errores.</li> <li>• 1. El UPS conectado ha detectado un problema con la batería conectada.</li> </ul>
UPSBuffering	SINT		GSV	Indica si el UPS está proporcionando energía desde la batería. <ul style="list-style-type: none"> <li>• 0. UPS no está proporcionando energía desde la batería.</li> <li>• 1. UPS está proporcionando energía desde la batería.</li> </ul>
UPSI inhibited	SINT		GSV	Solicita a UPS eliminar la alimentación. <ul style="list-style-type: none"> <li>• 0. El controlador no quiere que se elimine la alimentación en este momento.</li> <li>• 1. UPS va a detener el suministro de energía.</li> </ul>
UPSReady	SINT		GSV	Indica si el UPS está listo en función de: cargado $\geq$ 85%, sin fallos en el cableado, suficiente voltaje de entrada y la señal de inhibir está inactiva. <ul style="list-style-type: none"> <li>• 0. UPS no está listo</li> <li>• 1. UPS listo</li> </ul>

Atributo	Tipo de datos	Instrucción	Descripción
UPSSupported	SINT	GSV	Indica si el UPS es compatible. <ul style="list-style-type: none"> <li>• 0. Incompatible</li> <li>• 1. Compatible</li> </ul>

## Acceso al objeto Message

Se accede al objeto Message a través de instrucciones GSV/SSV. Se especifica el nombre de etiqueta del mensaje para determinar el objeto Message que desea. El objeto Message proporciona una interfaz para configurar y desencadenar las comunicaciones entre dispositivos similares. Este objeto sustituye el tipo de datos MG del procesador PLC-5.

Atributo	Tipo de datos	Instrucción	Descripción
ConnectionPath	SINT[130]	GSV SSV	Datos para configurar la ruta de conexión. Los primeros dos bytes (bajo y alto) tienen la longitud en bytes de la ruta de la conexión.
ConnectionRate	DINT	GSV SSV	Régimen del paquete solicitado de la conexión.
MessageType	SINT	GSV SSV	Especifica el tipo de mensaje. El valor tiene un significado específico: <ul style="list-style-type: none"> <li>• 0. No inicializado</li> </ul>
Port	SINT	GSV SSV	Indica el puerto por el que se debería enviar el mensaje. Cada valor tiene un significado concreto: <ul style="list-style-type: none"> <li>• 1. Backplane.</li> <li>• 2. Puerto serial.</li> </ul>
Timeout Multiplier	SINT	GSV SSV	Determina el tiempo en el que la conexión debería considerarse finalizada y cerrada. Cada valor tiene un significado concreto: <ul style="list-style-type: none"> <li>• 0. La conexión sobrepasará el tiempo de espera en 4 veces del régimen de actualización. Este es el valor predeterminado.</li> <li>• 1. La conexión sobrepasará el tiempo de espera en 8 veces del régimen de actualización</li> <li>• 2. La conexión sobrepasará el tiempo de espera en 16 veces del régimen de actualización</li> </ul>
Unconnected Timeout	DINT	GSV SSV	Tiempo de espera, en microsegundos, para todos los mensajes desconectados. El valor predeterminado es 30.000.000 microsegundos (30 s).

### Consulte también

[Códigos y tipos de fallos mayores](#) en la página 235

[Códigos y tipos de fallos menores](#) en la página 241

## Acceso al objeto Module

El objeto Module proporciona información de estado sobre un módulo. Para seleccionar un objeto Module en concreto, se establece el operando Nombre de objeto de la instrucción GSV/SSV en el nombre del módulo. El módulo específico debe estar presente en la sección de configuración E/S del organizador de controlador y debe tener un nombre de dispositivo.

Atributo	Tipo de datos	Instrucción	Descripción
EntryStatus	INT	GSV	<p>Especifica el estado actual de la entrada de asignación especificada. Los 12 bits inferiores deben enmascararse al realizar la operación de comparación. Solo son válidos los bits 12-15. Cada valor tiene un significado concreto:</p> <ul style="list-style-type: none"> <li>• <b>16#0000. En espera.</b> El controlador se está encendiendo.</li> <li>• <b>16#1000. Con fallo.</b> Fallo en alguna de las conexiones del objeto Module con su módulo asociado. Este valor no se debería usar para determinar si el módulo falla debido a que el objeto Module deja este estado periódicamente al intentar reconectar el módulo. En su lugar, prueba el estado de ejecución (16#4000). Comprueba si FaultCode no es igual a 0 para determinar si un módulo falla. Si hay un fallo, los atributos FaultCode y FaultInfo serán válidos hasta que la condición de fallo se corrija.</li> <li>• <b>16#2000. Validando.</b> El objeto Module está verificando la integridad del objeto Module antes de establecer conexiones con el módulo.</li> <li>• <b>16#3000. Conectando.</b> El objeto Module está iniciando conexiones con el módulo.</li> <li>• <b>16#4000. Ejecutando.</b> Todas las conexiones al módulo se han establecido y los datos se están transfiriendo.</li> <li>• <b>16#5000. Desactivando.</b> El objeto Module está en el proceso de desactivar todas las conexiones al módulo.</li> <li>• <b>16#6000. Inhibido.</b> El objeto Module está inhibido (el bit de inhibición en el atributo Mode está establecido).</li> <li>• <b>16#7000. Esperando.</b> El objeto primario del que depende este objeto Module no se está ejecutando.</li> <li>• <b>16#9000. Actualización de firmware.</b> Supervisor de firmware está intentando actualizar el módulo.</li> <li>• <b>16#A000. Configuración.</b> El controlador está descargando la configuración en el módulo.</li> </ul>
FaultCode	INT	GSV	Un número para identificar el fallo de módulo que pudiera ocurrir.
FaultInfo	DINT	GSV	Proporciona información específica sobre el código de fallo del objeto Module.
Firmware SupervisorStatus	INT	GSV	<p>Identifica el estado de funcionamiento actual de la función del supervisor de firmware. Cada valor tiene un significado específico:</p> <ul style="list-style-type: none"> <li>• 0. Las actualizaciones de los módulos no se están ejecutando.</li> <li>• 1. Las actualizaciones de los módulos se están ejecutando.</li> </ul>
ForceStatus	INT	GSV	<p>Especifica el estado de los forzados. Cada bit tiene un significado específico:</p> <ul style="list-style-type: none"> <li>• 0. Forzados instalados (1=sí, 0=no).</li> <li>• 1. Forzados habilitados (1=sí, 0=no).</li> </ul>
Instance	DINT	GSV	Proporciona el número de instancia de este objeto de módulo.

LEDStatus	INT	GSV	<p>Especifica el estado actual del indicador de estado E/S en la parte frontal del controlador.(1) Cada valor tiene un significado específico:</p> <ul style="list-style-type: none"> <li>• 0. Indicador de estado desactivado: No hay objetos Module configurados para el controlador. (No hay módulos en la selección de configuración E/S del organizador de controlador).</li> <li>• 1. Luz roja parpadeante: Ningún objeto Module se está ejecutando.</li> <li>• 2. Luz verde parpadeante: Hay por lo menos un objeto Module que no se está ejecutando.</li> <li>• 3. Luz verde continua: Todos los objetos Module se están ejecutando.</li> </ul> <p>No se introduce un nombre de objeto con este atributo porque este atributo aplica a toda la colección de módulos.</p>
Mode	INT	GSV SSV	<p>Especifica el modo actual del objeto Module. Cada bit tiene un significado específico:</p> <ul style="list-style-type: none"> <li>• 0. Si se establece, hace que se genere un fallo mayor en caso de que falle alguna de las conexiones del objeto Module mientras el controlador está en modo Marcha.</li> <li>• 2. Si se establece, hace que el objeto Module pase al estado Inhibido después de cerrar todas las conexiones al módulo.</li> </ul>
Path	SINT Array	GSV	<p>Especifica la ruta al módulo al que se hace referencia. Este es un nuevo atributo que comienza en el software de la versión 24. Cada byte tiene un significado específico:</p> <ul style="list-style-type: none"> <li>• 0-1. Longitud de la ruta en bytes. Si es 0, la longitud de la tabla SINT es insuficiente para mantener la ruta del módulo devuelta.</li> </ul> <p>Si la longitud de la tabla SINT es insuficiente para mantener la ruta, la tabla se pone a cero y se registra un fallo menor.</p>

(1) Los controladores 1756-L7x no tienen una pantalla de indicador de estado en la parte frontal del controlador, pero sí usan esta funcionalidad.

#### Consulte también

[Fallos de módulo: 16#0000 - 16#00ff](#) en la página 344

[Fallos de módulo: 16#0100 - 16#01ff](#) en la página 346

[Fallos de módulo: 16#0200 - 16#02ff](#) en la página 351

[Fallos de módulo: 16#0300 - 16#03ff](#) en la página 352

[Fallos de módulo: 16#0800 - 16#08ff](#) en la página 355

[Fallos de módulo: 16#fd00 - 16#fdff](#) en la página 356

[Fallos de módulo: 16#fe00 - 16#feff](#) en la página 357

[Fallos de módulo: 16#ff00 - 16#ffff](#) en la página 359

## Acceso al objeto Routine

El objeto Routine proporciona información de estado sobre una rutina. Especifique el nombre de la rutina para determinar qué objeto Routine desea.

Atributo	Tipo de datos	Instrucción dentro de la Tarea estándar	Instrucción dentro de la Tarea de seguridad	Descripción
Instance	DINT	GSV	GSV	Proporciona el número de instancia de este objeto de rutina. Los valores válidos son de 0 a 65.535.
Name	Cadena	GSV	GSV	El nombre de la rutina.
SFCPaused	INT	GSV	Ninguno	En una rutina SFC, indica si la SFC está en pausa. Cada valor tiene un significado concreto: <ul style="list-style-type: none"> <li>• 0. SFC no está en pausa.</li> <li>• 1. SFC está en pausa.</li> </ul>
SFCResuming	INT	GSV SSV	Ninguno	En una rutina SFC, indica si la SFC está reanudando la ejecución. Cada valor tiene un significado concreto: <ul style="list-style-type: none"> <li>• 0. SFC no se está ejecutando. Este atributo se establece automáticamente en 0 al final de un escaneado en el que se ejecutó el diagrama.</li> <li>• 1. SFC se está ejecutando. Los temporizadores de pasos y acciones conservarán su valor anterior si están configurados para hacerlo. Este atributo se establece automáticamente en 1 en el primer escaneado después de que un diagrama deje de estar en pausa.</li> </ul>

### Consulte también

[Códigos y tipos de fallos mayores](#) en la página 235

[Códigos y tipos de fallos menores](#) en la página 241

## Acceso al objeto Redundancy

El objeto REDUNDANCY proporciona información de estado sobre un sistema de redundancia.

Para esta información	Obtener este atributo	Tipo de datos	GSV/SV	Descripción	
Estado de redundancia de todo el chasis	ChassisRedundancyState	INT	GSV	Si	Entonces
				16#2	Primario con secundario sincronizado
				16#3	Primario con secundario descalificado
				16#4	Primario sin secundario
				16#10	Primario bloqueado para actualización

Estado de redundancia del chasis colaborador	PartnerChassis RedundancyState	INT	GSV	<b>Si</b>	<b>Entonces</b>
				16#8	Secundario sincronizado
				16#9	Secundario descalificado con primario
				16 # E	Sin colaborador
				16#12	Secundario bloqueado para actualización
Estado de redundancia del controlador	ModuleRedundancy State	INT	GSV	<b>Si</b>	<b>Entonces</b>
				16#2	Primario con secundario sincronizado
				16#3	Primario con secundario descalificado
				16#4	Primario sin secundario
				16#6	Primario con secundario sincronizando
				16#F	Primario que se está bloqueando para actualización
				16#10	Primario bloqueado para actualización
Estado de redundancia del colaborador	PartnerModule RedundancyState	INT	GSV	<b>Si</b>	<b>Entonces</b>
				16#7	Secundario sincronizando
				16#8	Secundario sincronizado
				16#9	Secundario descalificado con primario
				16 # E	Sin colaborador
				16#11	Secundario que se está bloqueando para actualización
				16#12	Secundario bloqueado para actualización
Resultados de las comprobaciones de compatibilidad con el controlador del colaborador	CompatibilityResults	INT	GSV	<b>Si</b>	<b>Entonces</b>
				0	Indeterminado
				1	Colaborador no compatible
				2	Colaborador totalmente compatible
Estado del proceso de sincronización (calificación)	Qualification InProgress	INT	GSV	<b>Si</b>	<b>Entonces</b>
				-1	La sincronización (calificación) no está en proceso
				0	Incompatible
				1...999	Para los módulos que pueden medir su porcentaje de finalización, el porcentaje de sincronización (calificación) que se completa
				50	Para los módulos que no pueden medir su porcentaje de finalización, la sincronización (calificación) está en curso
				100	La sincronización (calificación) es completa.

		DINT	GSV	<b>Si</b>	<b>Entonces</b>
La configuración del interruptor de llave del controlador y su colaborador coinciden o no coinciden	KeyswitchAlarm			0	Una de las siguientes condiciones es verdadera: Los interruptores de llave coinciden Ningún colaborador está presente
				1	Los interruptores de llave no coinciden
Posición del interruptor de llave del colaborador	PartnerKeyswitch	DINT	GSV	<b>Si</b>	<b>Entonces</b>
				0	Desconocido
				1	RUN
				2	PROG
				3	REM
Estado de los fallo menores del colaborador (si el ModuleRedundancyState indica que un colaborador está presente)	PartnerMinorFaults	DINT	GSV	<b>Este bit</b>	<b>Significa este fallo menor</b>
				1	Fallo de encendido
				3	Fallo de E/S
				4	Problema con una instrucción (programa)
				6	Superposición de tarea periódica (vigilancia)
				9	Problema con el puerto serial (no disponible para los proyectos de los controladores 1756-L7x)
				10	Batería baja o problema con el módulo de almacenamiento de energía
Modo del colaborador	PartnerMode	DINT	GSV	<b>Si</b>	<b>Entonces</b>
				16#0	Encendido
				16#1	Programa
				16#2	Marcha
				16#3	Prueba
				16#4	Con fallo
				16#5	Marcha a programa
				16#6	Prueba a programa
				16#7	Programa a marcha
				16#8	Prueba a marcha
				16#9	Marcha a prueba
				16#A	Programa a prueba
				16#B	En fallo
				16#C	Fallo a programa

En un par de chasis redundantes, identificación de un chasis específico sin tener en cuenta el estado del chasis	PhysicalChassisID	INT	GSV	Si	<b>Entonces</b>
				0	Desconocido
				1	Chasis A
				2	Chasis B
Número de ranura del módulo de redundancia (por ejemplo, 1756-RM, 1756-RM2) en el chasis	SRMSlotNumber	INT	GSV		
Tamaño de la última carga cruzada Tamaño de la última carga cruzada si tenía un chasis secundario	LastDataTransferSize	DINT	GSV	Este atributo proporciona el tamaño de los datos que se cruzaron o se habrían cruzado en el último escaneado. El tamaño en DINT (palabras de 4 bytes). Debe configurar el controlador para redundancia. No necesita un chasis secundario. Hay un chasis secundario sincronizado	
				SÍ	Esto proporciona el número de DINT que se cruzaron en el último escaneado.
				NO	Esto proporciona el número de DINT que se habrían cruzado en el último escaneado
Tamaño de la mayor carga cruzada Tamaño de la mayor carga cruzada si tenía un chasis secundario	MaxDataTransferSize	DINT	GSV SSV	El tamaño en DINT (palabras de 4 bytes). Debe configurar el controlador para redundancia. No necesita un chasis secundario. Para restablecer este valor, utilice una instrucción SSV con un valor de Source de 0. ¿Hay un chasis secundario sincronizado?	
				SÍ	Esto proporciona el mayor número de DINT que se cruzaron.
				NO	Esto proporciona el mayor número de DINT que se habrían cruzado.
Modo del colaborador	PartnerMode	DINT	GSV	Si	<b>Entonces</b>
				16#0	Encendido
				16#1	Programa
				16#2	Marcha
				16#3	Prueba
				16#4	Con fallo
				16#5	Marcha a programa
				16#6	Prueba a programa
				16#7	Programa a marcha
				16#8	Prueba a marcha
				16#9	Marcha a prueba

					16#A 16#B 16#C	Programa a prueba En fallo Fallo a programa
En un par de chasis redundantes, identificación de un chasis específico sin tener en cuenta el estado del chasis	PhysicalChassisID	INT	GSV	Si	<b>Entonces</b>	
				0 1 2	Desconocido Chasis A Chasis B	
Número de ranura del módulo 1757-SRM en el chasis	SRMSlotNumber	INT	GSV			
• Tamaño de la última carga cruzada • Tamaño de la última carga cruzada si tenía un chasis secundario	LastDataTransferSize	DINT	GSV		Este atributo proporciona el tamaño de los datos que se cruzaron o se habrían cruzado en el último escaneado. <ul style="list-style-type: none"><li>• El tamaño en DINT (palabras de 4 bytes).</li><li>• Debe configurar el controlador para redundancia.</li><li>• No necesita un chasis secundario.</li></ul> ¿Hay un chasis secundario sincronizado?	
				SÍ	Esto proporciona el número de DINT que se cruzaron en el último escaneado.	
				NO	Esto proporciona el número de DINT que se habrían cruzado en el último escaneado	
• Tamaño de la mayor carga cruzada • Tamaño de la mayor carga cruzada si tenía un chasis secundario	MaxDataTransferSize	DINT	GSV SSV		Este atributo proporciona el mayor tamaño del atributo Size de LastDataTransfer. <ul style="list-style-type: none"><li>• El tamaño en DINT (palabras de 4 bytes).</li><li>• Debe configurar el controlador para redundancia.</li><li>• No necesita un chasis secundario.</li><li>• Para restablecer este valor, utilice una instrucción SSV con un valor de Source de 0.</li></ul> ¿Hay un chasis secundario sincronizado?	
				SÍ	Esto proporciona el mayor número de DINT que se cruzaron.	
				NO	Esto proporciona el mayor número de DINT que se habrían cruzado.	

## Acceso al objeto Program

El objeto Program proporciona información de estado sobre un programa. Especifica el nombre del programa para determinar el objeto Program que desea.

Atributo	Tipo de datos	Instrucción dentro de la Tarea estándar	Instrucción dentro de la Tarea de seguridad	Descripción
DisableFlag	SINT	GSV SSV	Ninguno	Controla la ejecución de este programa. Cada valor tiene un significado concreto: <ul style="list-style-type: none"> <li>• 0. Ejecución habilitada.</li> <li>• No cero. Ejecución deshabilitada.</li> </ul>
	DINT	GSV	GSV	Un valor distinto de cero deshabilita.
LastScanTime	DINT	GSV SSV	Ninguno	Tiempo de ejecución del programa la última vez que se ejecutó. El tiempo se expresa en microsegundos.
MajorFault Record	DINT[11]	GSV SSV	GSV SSV	Registra los fallos mayores para este programa

Recomendamos que cree una estructura definida por el usuario para simplificar el acceso al atributo MajorFaultRecord:

Nombre	Tipo de datos	Estilo	Descripción	
TimeLow	DINT	Decimal	32 bits inferiores del valor de marca de tiempo de fallo	
TimeHigh	DINT	Decimal	32 bits superiores del valor de marca de tiempo de fallo	
Type	INT	Decimal	Tipo de fallo (programa, E/S, etc.)	
Código	INT	Decimal	Código único para el fallo (depende del tipo de fallo)	
Info	DINT[8]	Hexadecimal	Información específica de fallo (depende del tipo de fallo y el código)	
MaxScanTime	DINT	GSV SSV	Ninguno	Tiempo de ejecución registrado máximo para este programa. El tiempo se expresa en microsegundos.
Name	Cadena	GSV	GSV	Nombre del programa.

### Consulte también

[Códigos y tipos de fallos mayores](#) en la página 235

[Códigos y tipos de fallos menores](#) en la página 241

## Acceso al objeto de Seguridad

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

El objeto del Controlador de seguridad proporciona el estado seguro y la información sobre la firma de seguridad. Los atributos SafetyTask y SafetyFaultRecord pueden adquirir información sobre fallos no recuperables.

Consulte el [GuardLogix Controllers User Manual](#), publicación [1756-UM020](#).

Atributo	Tipo de datos	Instrucción dentro de la Tarea estándar	Instrucción dentro de la Tarea de seguridad	Descripción
SafetyLockedState	SINT	GSV	Ninguno	Indica si el controlador está bloqueado por seguridad o no.
SafetySILConfiguration	SINT	GSV	Ninguno	Especifica la configuración SIL de seguridad. <ul style="list-style-type: none"> <li>• 2 -- SIL2/PLd</li> <li>• 3 -- SIL3/PLe</li> </ul>
SafetyStatus	INT	GSV	Ninguno	Especifica el estado seguro. Cada valor tiene un significado concreto: <ul style="list-style-type: none"> <li>• 1000000000000000 -- Tarea de seguridad OK.</li> <li>• 1000000000000001 -- Tarea de seguridad inoperable.</li> <li>• 0000000000000000 -- Falta colaborador.</li> <li>• 0000000000000001 -- Colaborador no disponible.</li> <li>• 0000000000000010 -- Hardware incompatible.</li> <li>• 0000000000000011 -- Firmware incompatible.</li> </ul>
SafetySignatureExists	SINT	GSV	GSV	Indica si existe una firma de tarea de seguridad.
SafetySignatureID (Solo aplicable a Controladores Compact GuardLogix 5370 y GuardLogix 5570)	SINT	GSV	Ninguno	Número de identificación de 32 bits.
SafetySignature (Solo aplicable a Controladores Compact GuardLogix 5370 y GuardLogix 5570)	Cadena	GSV	Ninguno	El número de identificación de 32 bits incluye el número ID además del sello de hora y fecha.
SafetyTaskFaultRecord	DINT[11]	GSV	Ninguno	Registra los errores de las tareas de seguridad.
SafetySignatureIDLong (Solo aplicable a Controladores Compact GuardLogix 5380 y GuardLogix 5580)	SINT[33]	GSV	Ninguno	ID de firma de seguridad de 32 byte en matriz de byte. El primer byte tendrá el tamaño del ID de la firma de seguridad en bytes, y los otros 31 bytes corresponderán al ID de la firma.
SafetySignatureIDHex (Solo aplicable a Controladores Compact GuardLogix 5380 y GuardLogix 5580).	Cadena	GSV	Ninguno	Representación de cadena hexadecimal de 64 caracteres del ID de la firma.
SafetySignatureDateTime (Solo aplicable a Controladores Compact GuardLogix 5380 y GuardLogix 5580).	Cadena	GSV	Ninguno	Fecha y hora de la firma de seguridad en 27 caracteres, en el formato mm/dd/aaaa, hh:mm:ss.iii<AM o PM>

## Acceso al objeto SerialPort

El objeto SerialPort proporciona una interfaz al puerto de comunicación serial.

Atributo	Tipo de datos	Instrucción	Descripción
BaudRate	DINT	GSV	Especifica la velocidad en baudios. Los valores válidos son 110, 300, 600, 1200, 2400, 4800, 9600 y 19200 (predeterminado).
ComDriverID	SINT	GSV	Indica el variador específico. Cada valor tiene un significado concreto: <ul style="list-style-type: none"> <li>• 0xA2 DF1. Este es el valor predeterminado.</li> <li>• 0xA3 ASCII</li> </ul>
DataBits	SINT	GSV	Especifica el número de bits de datos por carácter. Cada valor tiene un significado concreto: <ul style="list-style-type: none"> <li>• 7. Siete bits de datos. Solo ASCII</li> <li>• 8. Ocho bits de datos. Este es el valor predeterminado.</li> </ul>
DCDDelay	INT	GSV	Especifica la cantidad de tiempo que se debe esperar para que la detección de la portadora de datos (DCD) se vuelva baja antes de producir un error en el paquete. El retardo se indica en conteos de paquetes de 1 s. El valor predeterminado es contador a 0.
Parity	SINT	GSV	Especifica la paridad. Cada valor tiene un significado concreto: <ul style="list-style-type: none"> <li>• 0. Sin paridad. Este es el valor predeterminado.</li> <li>• 1. Paridad impar. Solo ASCII</li> <li>• 2. Paridad par.</li> </ul>
RTSOffDelay	INT	GSV	Cantidad de tiempo para retardar la desconexión de la línea RTS después de que se haya transmitido el último carácter. Valores válidos: 0...32.767 Retardo en conteos de períodos de 20 ms. El valor predeterminado es 0 ms.
RTSSendDelay	INT	GSV	Cantidad de tiempo para retardar la transmisión del primer carácter de un mensaje después de activar la línea RTS. Valores válidos: 0...32.767 Retardo en conteos de períodos de 20 ms. El valor predeterminado es 0 ms.
StopBits	SINT	GSV	Especifica el número de bits de parada. Cada valor tiene un significado concreto: <ul style="list-style-type: none"> <li>• 1. Un bit de parada. Este es el valor predeterminado.</li> <li>• 2. Dos bits de parada. Solo ASCII</li> </ul>
PendingBaudRate	DINT	SSV	Valor pendiente para el atributo BaudRate.
PendingCOM DriverID	SINT	SSV	Valor pendiente para el atributo COMDriverID.
PendingDataBits	SINT	SSV	Valor pendiente para el atributo DataBits.
PendingDCD Delay	INT	SSV	Valor pendiente para el atributo DCDDelay.
PendingParity	SINT	SSV	Valor pendiente para el atributo Parity.
PendingRTSOff Delay	INT	SSV	Valor pendiente para el atributo RTSOffDelay.

PendingRTSSendDelay	INT	SSV	Valor pendiente para el atributo RTSSendDelay.
PendingStopBits	SINT	SSV	Valor pendiente para el atributo StopBits.

### Consulte también

[Códigos y tipos de fallos mayores en la página 235](#)

[Códigos y tipos de fallos menores en la página 241](#)

## Acceso al objeto Task

El objeto TASK proporciona información de estado sobre una tarea. Especifica el nombre de la tarea para determinar el objeto TASK que deseé.

Atributo	Tipo de datos	Instrucción dentro de la Tarea estándar	Instrucción dentro de la Tarea de seguridad	Descripción
DisableUpdateOutputs	DINT	GSV SSV	Ninguno	Habilita o deshabilita el proceso de salidas al final de una tarea. <ul style="list-style-type: none"> <li>Establezca el atributo a 0 para habilitar el proceso de salidas al final de la tarea.</li> <li>Establezca el atributo a 1 (o a cualquier valor diferente de cero) para deshabilitar el proceso de salidas al final de la tarea.</li> </ul>
EnableTimeOut	DINT	GSV SSV	Ninguno	Habilita o deshabilita la función de tiempo de espera de una tarea de evento. <ul style="list-style-type: none"> <li>Establezca el atributo a 0 para deshabilitar la función tiempo de espera.</li> <li>Establezca el atributo a 1 (o a cualquier valor diferente de cero) para habilitar la función tiempo de espera.</li> </ul>
InhibitTask	DINT	GSV SSV	Ninguno	Evita que la tarea se ejecute. Si la tarea está inhibida, el controlador aún realiza el pre-escaneado de la tarea cuando el controlador realiza la transición del modo Programa al modo Marcha o Prueba. <ul style="list-style-type: none"> <li>Establezca el atributo a 0 para habilitar la tarea</li> <li>Establezca el atributo a 1 (o a cualquier valor diferente de cero) para inhibir (deshabilitar) la tarea</li> </ul>
Instance	DINT	GSV	GSV	Proporciona el número de instancia de este objeto TASK. Los valores válidos son de 0 a 31.
LastScanTime	DINT	GSV SSV	Ninguno	El tiempo que tardó en ejecutarse el programa la última vez. El tiempo se expresa en microsegundos.

MaximumInterval	DINT[2]	GSV SSV	Ninguno	El intervalo de tiempo máximo entre ejecuciones sucesivas de la tarea. DINT[0] tiene los 32 bits inferiores del valor. DINT[1] tiene los 32 bits superiores del valor. Un valor 0 indica 1 o menos ejecuciones de la tarea.
MaximumScanTime	DINT	GSV SSV	Ninguno	Tiempo de ejecución registrado máximo para este programa. El tiempo se expresa en microsegundos.
MinimumInterval	DINT[2]	GSV SSV	Ninguno	El intervalo de tiempo mínimo entre ejecuciones sucesivas de la tarea. DINT[0] tiene los 32 bits inferiores del valor. DINT[1] tiene los 32 bits superiores del valor. Un valor 0 indica 1 o menos ejecuciones de la tarea.
Name	Cadena	GSV	GSV	Nombre de la tarea.
OverlapCount	DINT	GSV SSV	GSV SSV	El número de veces que la tarea se desencadenó cuando aún se estaba ejecutando. Válido para un evento o una tarea periódica. Para borrar el conteo, ponga el atributo a 0.
Priority	INT	GSV SSV	GSV	Prioridad relativa de esta tarea en comparación con las otras tareas. Los valores válidos son de 0 a 15.
Rate	DINT	GSV SSV	GSV	El intervalo de tiempo entre ejecuciones de la tarea. El tiempo se expresa en microsegundos.
StartTime	DINT[2]	GSV SSV	Ninguno	Valor de WALLCLOCKTIME cuando empezó la última ejecución de la tarea. DINT[0] tiene los 32 bits inferiores del valor. DINT[1] tiene los 32 bits superiores del valor.
Status	DINT	GSV SSV	Ninguno	Proporciona información de estado sobre la tarea. Una vez el controlador establece uno de estos bits, deben eliminarse manualmente. Para determinar si: <ul style="list-style-type: none"> <li>• una instrucción EVENT ha desencadenado la tarea (solo la tarea del evento), examinar bit 0</li> <li>• tiempo de espera ha desencadenado la tarea (solo la tarea del evento), examinar bit 1</li> <li>• ha ocurrido una superposición para esta tarea, examinar bit 2</li> </ul>
Watchdog	DINT	GSV SSV	GSV	Límite de tiempo para la ejecución de todos los programas asociados con esta tarea. El tiempo se expresa en microsegundos. Si introduce 0, se asignan los siguientes valores:

			<b>Tiempo:</b> 0,5 s 5,0 s	<b>Tipo de tarea:</b> periódica continua
--	--	--	----------------------------------	--

### Consulte también

[Códigos y tipos de fallos mayores en la página 235](#)

[Códigos y tipos de fallos menores en la página 241](#)

## Acceso al objeto TimeSynchronize

El objeto TIMESYNCHRONIZE proporciona una interfaz de protocolo industrial común (CIP) al estándar IEEE 1588 (IEC 61588) para un protocolo de sincronización de relojes de precisión para sistemas de control y medición en red. Se tiene acceso al objeto TIMESYNCHRONIZE a través de las instrucciones GSV/SSV.

Para obtener más información sobre este objeto, consulte la publicación IA-AT003 de Integrated Architecture® y técnicas de aplicación de configuraciones CIP Sync.

Atributo	Tipo de datos	Instrucción	Descripción												
ClockType	INT	GSV	<p>El tipo de reloj.</p> <table border="1"> <tr> <td><b>Bit</b></td><td><b>Tipo de reloj</b></td></tr> <tr> <td>0</td><td>Reloj ordinario</td></tr> <tr> <td>1</td><td>Reloj de límite</td></tr> <tr> <td>2</td><td>Reloj transparente entre dispositivos similares</td></tr> <tr> <td>3</td><td>Reloj transparente de extremo a extremo</td></tr> <tr> <td>4</td><td>Nodo de administración</td></tr> </table> <p>Todos los demás bits están reservados.</p>	<b>Bit</b>	<b>Tipo de reloj</b>	0	Reloj ordinario	1	Reloj de límite	2	Reloj transparente entre dispositivos similares	3	Reloj transparente de extremo a extremo	4	Nodo de administración
<b>Bit</b>	<b>Tipo de reloj</b>														
0	Reloj ordinario														
1	Reloj de límite														
2	Reloj transparente entre dispositivos similares														
3	Reloj transparente de extremo a extremo														
4	Nodo de administración														
CurrentTimeMicroseconds	LINT	GSV	Valor actual de la hora del sistema en microsegundos.												
CurrentTimeNanoseconds	LINT	GSV	Valor actual de la hora del sistema en nanosegundos.												
DomainNumber	SINT	GSV	El dominio del reloj PTP. El valor se encuentra entre 0 y 255. El valor predeterminado es 0.												
CurrentTimeMicroseconds	LINT	GSV	Valor actual de la hora del sistema en												

			microsegundos.
CurrentTimeNanoseconds	LINT	GSV	Valor actual de la hora del sistema en nanosegundos.
DomainNumber	SINT	GSV	El dominio del reloj PTP. El valor se encuentra entre 0 y 255. El valor predeterminado es 0.
GrandMasterClockInfo	Estructura	GSV	Información de propiedad sobre el reloj de gran maestro. Requiere 24 bytes de almacenamiento.

**Estructura de información del reloj de gran maestro:**

ClockIdentity	SINT[8]		
ClockClass	INT		
TimeAccuracy	INT		
OffsetScaledLogVariance	INT		
CurrentUtcOffset	INT		
TimePropertyFlags	INT		
TimeSource	INT		
Priority1	INT		
Priority2	INT		
IsSynchronized	DINT	GSV	El reloj local está sincronizado con un maestro.
			<b>Valor</b>
			<b>Significado</b>
			0 No sincronizado
			1 Sincronizado
LocalClockInfo	Estructura	GSV	Información de propiedad sobre el reloj local. Requiere 20 bytes de almacenamiento.

**Estructura de información del reloj local:**

ClockIdentity	SINT[8]		
ClockClass	INT		
TimeAccuracy	INT		
OffsetScaledLogVariance	INT		
CurrentUtcOffset	INT		
TimePropertyFlags	INT		
TimeSource	INT		
ManufactureIdentity	DINT	GSV	El IEEE OUI (Organización de identidad única) para el fabricante.
MaxOffsetFromMaster	LINT		Desplazamiento máximo del maestro en nanosegundos.

MeanPathDelayToMaster	LINT	GSV	Retardo medio de la ruta del maestro al reloj local en nanosegundos.						
NumberOfPorts	INT	GSV	El número de puertos de este reloj.						
OffsetFromMaster	LINT	GSV	La diferencia calculada entre el reloj local y el reloj maestro, en función del mensaje de sincronización más reciente, en nanosegundos.						
PTPEnable	DINT	GSV/SSV	El estado de habilitación para CIP Sync/PTP/Sincronización temporal en el dispositivo.						
			<table border="1"> <thead> <tr> <th>Valor</th><th>Significado</th></tr> </thead> <tbody> <tr> <td>0</td><td>Deshabilitar</td></tr> <tr> <td>1</td><td>Habilitado</td></tr> </tbody> </table>	Valor	Significado	0	Deshabilitar	1	Habilitado
Valor	Significado								
0	Deshabilitar								
1	Habilitado								
ParentClockInfo	Estructura	GSV	Información de propiedad sobre el reloj primario. Requiere 16 bytes de almacenamiento.						

**Estructura de información del reloj primario:**

ClockIdentity	SINT[8]		
PortNumber	INT		
ObservedOffsetScaledLogVariance	INT		
ObservedPhaseChangeRate	DINT		
PortEnableInfo	Estructura	GSV	La configuración de habilitación de cada puerto en el dispositivo. Size = 2 + (N.º de puertos habilitados x 4) Maxsize = 42 bytes

**Estructura de estado del puerto habilitado:**

NumberOfPorts	INT		El número máximo de puertos es 10.
---------------	-----	--	------------------------------------

*Estructura repetida para el número de puertos:*

PortNumber	INT	GSV	El intervalo entre sucesivos mensajes "Anuncio" emitidos por un reloj maestro en cada puerto PTP del dispositivo. Size = 2 + (N.º de puertos habilitados x 4) Maxsize = 42 bytes
PortEnable	INT		
PortLogAnnounceIntervalInfo	Estructura		

**Estructura del intervalo de anuncio del registro de puerto:**

NumberOfPorts	INT		El número máximo de puertos es 10.
---------------	-----	--	------------------------------------

*Estructura repetida para el número de puertos:*

PortNumber	INT		
PortLogAnnounceInterval	INT		

PortLogSyncIntervalInfo	Estructura	GSV	El intervalo entre sucesivos mensajes de sincronización emitidos por un maestro en cada puerto PTP del dispositivo. Size = 2 + (N.º de puertos habilitados x 4) Maxsize = 42 bytes
-------------------------	------------	-----	--

**Estructura del intervalo de sincronización del registro de puerto:**

NumberOfPorts	INT		El número máximo de puertos es 10.
---------------	-----	--	------------------------------------

*Estructura repetida para el número de puertos:*

PortNumber	INT	GSV	La dirección física y de protocolo de cada puerto del dispositivo. Size = 2 + (N.º de puertos habilitados x 36) Maxsize = 362 bytes
PortLogAnnounceInterval	INT		
PortPhysicalAddressInfo	Estructura		

**Estructura de la dirección física del puerto:**

NumberOfPorts	INT		El número máximo de puertos es 10.
---------------	-----	--	------------------------------------

*Estructura repetida para el número de puertos:*

PortNumber	INT	GSV	Perfil de cada puerto del dispositivo. Size = 2 + (N.º de puertos habilitados x 10) Maxsize = 102
Protocol	SINT[16]		
SizeOfAddress	INT		
Port Address	SINT[16]		
PortProfileIdentityInfo	Estructura		

**Estructura de identidad del perfil del puerto:**

NumberOfPorts	INT		El número máximo de puertos es 10.
---------------	-----	--	------------------------------------

*Estructura repetida para el número de puertos:*

PortNumber	INT	GSV	La dirección de red y de protocolo de cada puerto del dispositivo. Size = 2 + (N.º de puertos habilitados x 22) Maxsize = 222
ClockIdentity	SINT[8]		
PortProtocolAddressInfo	Estructura		

**Estructura de la dirección del protocolo del puerto:**

NumberOfPorts	INT		El número máximo de puertos es 10.
---------------	-----	--	------------------------------------

*Estructura repetida para el número de puertos:*

PortNumber	INT	GSV	El estado actual de cada puerto PTP en el dispositivo. Size = 2 + (N.º de puertos habilitados x 4) Maxsize = 42 bytes
NetworkProtocol	INT		
SizeOfAddress	INT		
PortAddress	SINT[16]		
PortStateInfo	Estructura		

**Estructura del estado del puerto:**

NumberOfPorts	INT		El número máximo de puertos es 10.
<i>Estructura repetida para el número de puertos:</i>			
PortNumber	INT		
PortState	INT		
Priority1	SINT	GSV/SSV	Valor Priority1 (Anulación de maestro) para el reloj local. <b>Consejo:</b> El valor es sin signo.
Priority2	SINT	GSV/SSV	Valor Priority2 (Interruptor conjuntor) para el reloj local. <b>Consejo:</b> El valor es sin signo.
ProductDescription	Estructura	GSV	Descripción del producto del dispositivo que contiene el reloj. Requiere 68 bytes de almacenamiento.

**Estructura de la descripción del producto:**

Size	DINT		
Descripción	SINT[64]		
RevisionData	Estructura	GSV	Datos de revisión del dispositivo que contiene el reloj. Requiere 36 bytes de almacenamiento.

**Estructura de datos de revisión:**

Size	DINT		
Revision	SINT[32]		
StepsRemoved	INT	GSV	El número de regiones de sincronización CIP entre el reloj local y el gran maestro (es decir, el número de relojes de límite +1)
SystemTimeAndOffset	Estructura	GSV	Hora del sistema en microsegundos y el desplazamiento al valor del reloj local.

**Tiempo del sistema y estructura de desplazamiento:**

SystemTime	LINT		
SystemOffset	LINT		
UserDescription	Estructura	GSV	Descripción del usuario del dispositivo que contiene el reloj. Requiere 132 bytes de almacenamiento.

**Estructura de la descripción del usuario:**

Size	DINT		
Descripción	SINT[128]		

## Acceso al objeto **WallClockTime**

El objeto WallClockTime proporciona una marca de tiempo que el controlador puede usar para programar.

**Con** El establecimiento del objeto WALLCLOCKTIME está limitado a una  
**sejo** actualización como máximo cada 15 segundos.  
**:**

Atributo	Tipo de datos	Instrucción	Descripción
ApplyDST	SINT	GSV SSV	Identifica si habilitar el horario de verano. Cada valor tiene un significado concreto: <ul style="list-style-type: none"> <li>• 0. No ajustar el horario de verano.</li> <li>• No cero. Ajustar el horario de verano.</li> </ul>
CSTOffset	DINT[2]	GSV SSV	Desplazamiento positivo desde el CurrentValue del objeto CST (hora coordinada del sistema). DINT[0] tiene los 32 bits inferiores del valor. DINT[1] tiene los 32 bits superiores del valor. Valor en $\mu$ s. El valor predeterminado es 0.
CurrentValue	DINT[2]	GSV SSV	Valor actual del tiempo de reloj. DINT[0] tiene los 32 bits inferiores del valor. DINT[1] tiene los 32 bits superiores del valor. El valor es el número de microsegundos que han pasado desde las 0000 h del 1 de enero de 1972. Los objetos CST y WALLCLOCKTIME están matemáticamente relacionados en el controlador. Por ejemplo, si sumas CST CurrentValue y WALLCLOCKTIME CTSTOffset, el resultado es WALLCLOCKTIME CurrentValue.
DateTime	DINT[7]	GSV SSV	Fecha y hora. Cada valor tiene un significado concreto: <ul style="list-style-type: none"> <li>• DINT[0]. Año.</li> <li>• DINT[1]. Mes (del 1 al 12).</li> <li>• DINT[2]. Día (del 1 al 31).</li> <li>• DINT[3]. Hora (de 0 a 23).</li> <li>• DINT[4]. Minutos (del 0 al 59).</li> <li>• DINT[5]. Segundos (del 0 al 59).</li> <li>• DINT[6]. Microsegundos (del 0 al 999.999).</li> </ul>
DSTAdjustment	INT	GSV SSV	El número de minutos que hay que ajustar para el horario de verano.
LocalDateTime	DINT[7]	GSV SSV	Hora local actual ya ajustada. Cada valor tiene un significado concreto: <ul style="list-style-type: none"> <li>• DINT[0]. Año.</li> <li>• DINT[1]. Mes (del 1 al 12).</li> <li>• DINT[2]. Día (del 1 al 31).</li> <li>• DINT[3]. Hora (de 0 a 23).</li> <li>• DINT[4]. Minutos (del 0 al 59).</li> <li>• DINT[5]. Segundos (del 0 al 59).</li> <li>• DINT[6]. Microsegundos (del 0 al 999.999).</li> </ul>
TimeZoneString	INT	GSV SSV	Zona horaria para el valor de tiempo.

### Consulte también

[Códigos y tipos de fallos mayores](#) en la página 235

[Códigos y tipos de fallos menores](#) en la página 241

## Objetos de seguridad GSV/SSV

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

Para las tareas de seguridad, las instrucciones GSV y SSV están más restringidas.

**Cons ejo:** Las instrucciones SSV en las tareas estándar y de seguridad no pueden establecer el bit 0 (fallo grave en error) en el atributo de modo de un módulo E/S de seguridad.

En el caso de los objetos de seguridad, la siguiente tabla muestra los atributos para los que se pueden conseguir valores gracias a la instrucción GSV, así como los atributos que se pueden establecer con la instrucción SSV en las tareas estándar y de seguridad.



Use las instrucciones GSV/SSV con cuidado. Al efectuar cambios en objetos, el controlador puede funcionar de manera inesperada o provocar lesiones en el personal.

Objeto de seguridad	Nombre de atributo	Descripción de atributo	Accesible desde la tarea de seguridad		Accesible desde la tarea estándar	
			GSV	SSV	GSV	SSV
Tarea de seguridad	Instance	Proporciona el número de instancia de este objeto de tarea. Los valores válidos son de 0 a 31.	✓		✓	
	MaximumInterval	El intervalo de tiempo máximo entre ejecuciones sucesivas de esta tarea.			✓	✓
	MaximumScanTime	Tiempo de ejecución máximo registrado (ms) para esta tarea.			✓	✓
	MinimumInterval	El intervalo de tiempo mínimo entre ejecuciones sucesivas de esta tarea.			✓	✓
	Priority	Prioridad relativa de esta tarea en comparación con las demás. Los valores válidos son de 0 a 15.	✓		✓	
	Rate	Periodo para la tarea (en ms) o valor del tiempo de espera (en ms).	✓		✓	
	Watchdog	Límite de tiempo (en ms) para la ejecución de todos los programas asociados con esta tarea.	✓		✓	

Objeto de seguridad	Nombre de atributo	Descripción de atributo	Accesible desde la tarea de seguridad	Accesible desde la tarea estándar
	<b>DisableUpdateOutputs</b>	Habilita o deshabilita el proceso de salidas al final de una tarea. <ul style="list-style-type: none"><li>• Establezca el atributo a 0 para habilitar el proceso de salidas al final de la tarea.</li><li>• Establezca el atributo a 1 (o a cualquier valor diferente de cero) para deshabilitar el proceso de salidas al final de la tarea.</li></ul>		✓
	<b>EnableTimeOut</b>	Habilita o deshabilita la función tiempo de espera de una tarea. <ul style="list-style-type: none"><li>• Establezca el atributo a 0 para deshabilitar la función tiempo de espera.</li><li>• Establezca el atributo a 1 (o a cualquier valor diferente de cero) para habilitar la función tiempo de espera.</li></ul>		✓
	<b>InhibitTask</b>	Evita que la tarea se ejecute. Si la tarea está inhibida, el controlador aún realiza el pre-escaneado de la tarea cuando el controlador realiza la transición del modo Programa al modo Marcha o Prueba. <ul style="list-style-type: none"><li>• Establezca el atributo a 0 para habilitar la tarea</li><li>• Establezca el atributo a 1 (o a cualquier valor diferente de cero) para inhibir (deshabilitar) la tarea</li></ul>		✓
	<b>LastScanTime</b>	El tiempo que tardó en ejecutarse el programa la última vez. El tiempo se expresa en microsegundos.		✓
	<b>Name</b>	El nombre de la tarea		
	<b>OverlapCount</b>	El número de veces que la tarea se desencadenó cuando aún se estaba ejecutando. Válido para un evento o una tarea periódica. Para borrar el conteo, ponga el atributo a 0.		✓

Objeto de seguridad	Nombre de atributo	Descripción de atributo	Accesible desde la tarea de seguridad	Accesible desde la tarea estándar
	<b>StartTime</b>	Valor de WALLCLOCKTIME cuando empezó la última ejecución de la tarea. DINT[0] tiene los 32 bits inferiores del valor. DINT[1] tiene los 32 bits superiores del valor.		✓
	<b>Status</b>	Proporciona información de estado sobre la tarea. Una vez el controlador establece uno de estos bits, deben eliminarse manualmente. Para determinar si: <ul style="list-style-type: none"> <li>• una instrucción EVENT ha desencadenado la tarea (solo la tarea del evento), examinar bit 0</li> <li>• tiempo de espera ha desencadenado la tarea (solo la tarea del evento), examinar bit 1</li> <li>• ha ocurrido una superposición para esta tarea, examinar bit 2</li> </ul>		✓
Programa de seguridad	Instance	Proporciona el número de instancia del objeto del programa.	✓	✓
	MajorFaultRecord	Registra los fallos mayores para este programa.	✓	✓
	MaximumScanTime	Tiempo de ejecución máximo registrado (ms) para este programa.		✓ ✓
	<b>Disable Flag</b>	Controla la ejecución de este programa. Cada valor tiene un significado concreto: <ul style="list-style-type: none"> <li>• 0. Ejecución habilitada.</li> <li>• No cero. Ejecución deshabilitada.</li> </ul>		✓
	MaximumScanTime	Tiempo de ejecución máximo registrado (ms) para este programa.		✓
	Registro de fallo menor	Registra los fallos menores para este programa.		✓
	<b>LastScanTime</b>	El tiempo que tardó en ejecutarse el programa la última vez. El tiempo se expresa en microsegundos.		✓
	Name	El nombre de la tarea.		
Rutina de seguridad	Instancia	Proporciona el número de instancia de este objeto de rutina. Los valores válidos son de 0 a 65.535.	✓	

Objeto de seguridad	Nombre de atributo	Descripción de atributo	Accesible desde la tarea de seguridad	Accesible desde la tarea estándar
Controladores de seguridad	SafetyLockedState (SINT)	Indica si el controlador está bloqueado por seguridad o no.		✓
	SafetySILConfiguration (SINT)	Especifica la configuración SIL de seguridad como: • 2 = SIL2/PLd • 3 = SIL3/PLe		✓
	SafetyStatus (INT) (Solo aplicable a Controladores Compact GuardLogix 5380 y GuardLogix 5580).	Aplicaciones configuradas para SIL3/PLe, especificar el estado de seguridad como: <ul style="list-style-type: none"><li>• Tarea de seguridad OK. (1100000000000000)</li><li>• Tarea de seguridad inoperable. (1100000000000011)</li><li>• Falta el colaborador. (0100000000000000)</li><li>• El colaborador no está disponible. (0100000000000001)</li><li>• Hardware incompatible (0100000000000010)</li><li>• Incompatible con firmware. (0100000000000011)</li></ul> <p><b>Consejo:</b> Para las aplicaciones configuradas para SIL2/PLd, los bits 15, 0 y 1 se deben ignorar si pueden ser valores distintos basados en la ranura +1 del Controlador primario. Consulte el estado anterior para ver el significado.</p> <p>Aplicaciones configuradas para SIL2/PLd, especificar la tarea de seguridad como:</p> <ul style="list-style-type: none"><li>• Tarea de seguridad OK (x100000000000xx)</li><li>• Tarea de seguridad inoperable (x10000000000001xx)</li></ul>		✓
	SafetyStatus (INT)	Especifica el estado seguro		✓

Objeto de seguridad	Nombre de atributo	Descripción de atributo	Accesible desde la tarea de seguridad	Accesible desde la tarea estándar
	(Solo aplicable a Controladores Compact GuardLogix 5370 y GuardLogix 5570).	<p>como:</p> <ul style="list-style-type: none"> <li>• Tarea de seguridad OK. (1000000000000000)</li> <li>• Tarea de seguridad inoperable. (1000000000000001)</li> <li>• Falta el colaborador. (0000000000000000)</li> <li>• El colaborador no está disponible. (0000000000000001)</li> <li>• Hardware incompatible (0000000000000010)</li> <li>• Incompatible con firmware. (0000000000000011)</li> </ul>		
	SafetySignatureExists (SINT)	Indica si existe una firma de seguridad.	✓	✓
	SafetySignatureID (DINT) (Solo aplicable a Controladores Compact GuardLogix 5370 y GuardLogix 5570)	Número de identificación de 32 bits.		✓
	SafetySignature (String) (Solo aplicable a Controladores Compact GuardLogix 5370 y GuardLogix 5570)	Número ID más sello de hora y fecha.		✓
	SafetyTaskFaultRecord (DINT)	Registra los errores de las tareas de seguridad.		✓
	SafetySignatureIDLong SINT [33] (Solo aplicable a Controladores Compact GuardLogix 5380 y GuardLogix 5580)	El primer byte es el tamaño del ID de la firma de seguridad en bytes y los otros 32 bytes corresponden a los 32 bytes del ID de la firma de seguridad.		✓
	SafetySignatureIDHex(String) (Solo aplicable a Controladores Compact GuardLogix 5380 y GuardLogix 5580)	Representación de cadena hexadecimal de 64 caracteres del ID de la firma		✓
	SafetySignatureDateTime(String) (Solo aplicable a Controladores Compact GuardLogix 5380 y GuardLogix 5580)	Fecha y hora de la firma de seguridad en 27 caracteres, en el formato mm/dd/aaaa, hh:mm:ss.iii<AM o PM>		✓

### Consulte también

[Instrucciones de entrada y salida](#) en la página 223

## Marcas de estado de monitor

El controlador es compatible con palabras clave de estado que se pueden emplear dentro de la lógica para controlar eventos específicos:

- Las palabras clave de estado *no* distinguen entre mayúsculas y minúsculas.
- Puesto que estas marcas de estado pueden cambiar con mucha rapidez, la aplicación Logix Designer *no* muestra el estado de las marcas (es decir, aun en caso de que se haya establecido una marca, no se resaltará la instrucción en la que se hace referencia a esa marca).
- *No* se puede definir una etiqueta de alias a una palabra clave.

Puede utilizar las siguientes palabras clave:

Para determinar si:	Usa:
el valor que está almacenando no queda en el destino por alguna de estas razones: • es mayor que el valor máximo para el destino • es menor que el valor mínimo para el destino <b>Importante:</b> Cada vez que S:V pasa de borrado a establecido genera un fallo menor (tipo 4, código 4)	S:V
el valor de destino de la instrucción es 0	S:Z
el valor de destino de la instrucción es negativo	S:N
una operación aritmética provoca un acarreo (positivo o negativo) que trata de usar bits que están fuera del tipo de datos Por ejemplo: • al sumar 3+9 hay un acarreo positivo de 1 • al restar 25-18 hay un acarreo negativo de 10	S:C
este es el primer escaneado normal de las rutinas en el programa actual	S:FS
se ha generado como mínimo un fallo menor: • El controlador establece este bit cuando ocurre un fallo menor debido a la ejecución del programa. • El controlador no establece este bit para fallos menores que no estén relacionados con la ejecución del programa, como batería baja.	S:MINOR

## Seleccionar el tipo de mensaje

Después de introducir la instrucción MSG y especificar la estructura MESSAGE, haga clic en la pestaña Configuración (Configuration), dentro del cuadro de diálogo Configuración de mensaje (Message Configuration), para especificar los detalles del mensaje.

La pestaña Configuración (Configuration) también contiene una casilla de verificación para establecer/borrar el bit .TO.

Los detalles que configure dependen del tipo de mensaje que seleccione.

<b>Si el dispositivo de destino es un:</b>	<b>Seleccione uno de estos tipos de mensaje:</b>
Controlador LOGIX 5000	Lectura de tabla de datos CIP (CIP data table read) Escritura de tabla de datos CIP (CIP data table write)
Módulo E/S configurado con la aplicación Logix Designer	Reconfiguración de módulo (Module Reconfigure) CIP genérico (CIP Generic)
Controlador PLC-5®	Lectura en pantalla de PLC-5 (PLC-5 typed read) Escritura en pantalla de PLC-5 (PLC-5 typed write) Lectura de rango de palabras PLC-5 (PLC-5 word range read) Escritura de rango de palabras PLC-5 (PLC-5 word range write)
Controlador SLC™ Controlador MicroLogix™	Lectura en pantalla de SLC (SLC typed read) Escritura en pantalla de SLC (SLC typed write)
Módulo de transferencia en bloques	Lectura de transferencia en bloques (block transfer read) Escritura de transferencia en bloques (block transfer write)
Procesador PLC-3®	Lectura en pantalla de PLC-3 (PLC-3 typed read) Escritura en pantalla de PLC-3 (PLC-3 typed write) Lectura de rango de palabras PLC-3 (PLC-3 word range read) Escritura de rango de palabras PLC-3 (PLC-3 word range write)
Procesador PLC-2®	Lectura no protegida PLC-2 (PLC-2 unprotected read) Escritura no protegida PLC-2 (PLC-2 unprotected write)

Debe especificar esta información de configuración:

<b>En este campo:</b>	<b>Especifique:</b>
Elemento de origen (Source Element)	Si selecciona un tipo de mensaje de lectura, el Elemento de origen (Source Element) corresponderá a la dirección de los datos que quiere leer en el dispositivo de destino. Se usa la sintaxis de direccionamiento del dispositivo de destino. Si selecciona un tipo de mensaje de escritura, la etiqueta de Source corresponderá al primer elemento de la etiqueta que desea enviar al dispositivo de destino. No es compatible con booleanos ni etiquetas de estructura E/S. Se pueden usar todos los demás tipos de datos, como INT o DINT.

Número de elementos (Number of Elements)	El número de elementos (Number of Elements) que puede leer/escribir depende del tipo de mensaje y de datos que está utilizando. En el caso de los mensajes de "rango de palabras" y "no protegidos", el tamaño del elemento está indicado en el cuadro de diálogo. En el caso de los mensajes CIP y "en pantalla", un elemento se refiere a un único elemento de la matriz que se haya especificado como el origen de una escritura o el destino de una lectura
Elemento de destino (Destination Element)	Si selecciona un tipo de mensaje de lectura, la Etiqueta de destino (Destination Element) corresponderá al primer elemento de la etiqueta en el controlador LOGIX 5000, donde desea almacenar los datos que se han leído desde el dispositivo de destino. Si selecciona un tipo de mensaje de escritura, el Elemento de destino corresponderá a la dirección de la ubicación en el dispositivo de destino donde desea escribir los datos.

### Consulte también

[Especificar mensajes CIP](#) en la página 360

[Especificar mensajes PLC-5](#) en la página 367

[Especificar mensajes SLC](#) en la página 263

[Especificar mensajes de transferencia en bloques](#) en la página 263

[Especificar mensajes PLC-3](#) en la página 366

[Especificar mensajes PLC-2](#) en la página 368

## Fallos de módulo: 16#0000 - 16#00ff

Estos son los fallos del módulo: 16#0000 - 16#00ff

Código	Cadena	Explicación y posibles causas/soluciones
16#0001	Error de conexión.	Ha fallado una conexión a un módulo.
16#0002	Recurso no disponible.	<p>Existen posibilidades:</p> <ul style="list-style-type: none"> <li>• no existen suficientes conexiones disponibles para el controlador o para el módulo de comunicaciones mediante el cual se está conectando.</li> </ul> <p>Compruebe el uso de conexión del controlador o del módulo de comunicaciones. Si se están usando todas las conexiones, intente dejar de usar alguna de ellas o añadir otro módulo para enrutar la conexión errante.</p> <ul style="list-style-type: none"> <li>• se han excedido los límites de la memoria de E/S del controlador.</li> </ul> <p>Compruebe la memoria de E/S disponible y, de ser necesario, realice cambios en el programa o etiqueta.</p> <ul style="list-style-type: none"> <li>• el módulo de destino de E/S no tiene suficientes conexiones disponibles.</li> </ul> <p>Compruebe el número de controladores que están estableciendo una conexión con este módulo de E/S y verifique que el número de conexiones se encuentra dentro de los límites del módulo de E/S.</p>

16#0005	Error de solicitud de conexión: Clase incorrecta	<p>El controlador intenta establecer una conexión con el módulo y ha recibido un error.</p> <p>Existen posibilidades:</p> <ul style="list-style-type: none"><li>• la dirección configurada para la conexión con el módulo no es correcta.</li><li>• el módulo en uso (es decir, el módulo físico) es diferente al módulo especificado en el árbol de configuración de E/S y por lo tanto está haciendo que la conexión o el servicio falle.</li></ul> <p>El fallo puede producirse incluso cuando el módulo haya pasado la prueba de codificación electrónica. Esto puede producirse cuando se usan las opciones Deshabilitar codificación o Módulo compatible en la configuración del módulo, en lugar de la opción Coincidencia exacta.</p> <p>A pesar de pasar la prueba de codificación electrónica, el módulo con el que se está conectando no tiene las mismas características o configuración que el módulo especificado en el árbol de configuración de E/S y no admite la conexión o servicio deseados.</p> <p>Compruebe el módulo en uso y verifique que coincida exactamente con el módulo especificado en el árbol de configuración de E/S de la aplicación Logix Designer.</p> <p>Si está usando un módulo 1756-DHRI0, asegúrese de que el tipo de canal seleccionado en el software (DH+ o red de E/S remota) coincida con la configuración de interruptor rotativo del módulo.</p>
16#0006	Error de solicitud de conexión: Clase incorrecta.	<p>Existen posibilidades:</p> <ul style="list-style-type: none"><li>• el búfer de respuesta es demasiado pequeño para administrar los datos de respuesta.</li><li>• el módulo en uso (es decir, el módulo físico) es diferente al módulo especificado en el árbol de configuración de E/S y por lo tanto está haciendo que la conexión o el servicio falle.</li></ul> <p>El fallo puede producirse incluso cuando el módulo haya pasado la prueba de codificación electrónica. Esto puede producirse cuando se usan las opciones Deshabilitar codificación o Módulo compatible en la configuración del módulo, en lugar de la opción Coincidencia exacta.</p> <p>A pesar de pasar la prueba de codificación electrónica, el módulo con el que se está conectando no tiene las mismas características o configuración que el módulo especificado en el árbol de configuración de E/S y no admite la conexión o servicio deseados.</p> <p>Compruebe el módulo en uso y verifique que coincida exactamente con el módulo especificado en el árbol de configuración de E/S de la aplicación Logix Designer.</p>
16#0007	Error de solicitud de conexión: Clase incorrecta.	Una solicitud de servicio no está conectada, pero debería estarlo.
16#0008	Error de solicitud de servicio: Servicio no compatible	El controlador está intentando solicitar un servicio del módulo que no está admitido por el módulo.
16#0009	Configuración de módulo rechazada: error de parámetro <b>Consejo:</b> Información adicional de fallos: Información adicional de fallos para este fallo se muestra como código hexadecimal en la ficha de Conexión (Connection).	La configuración del módulo no es válida. Es posible que la configuración del módulo se haya cambiado en el Monitor de datos o por medio del programa. Si está disponible para el módulo, acceda a la ficha Conexiones (Connections) del cuadro de diálogo Propiedades del módulo (Module Properties) para acceder al código de fallo adicional. El código de fallo adicional indica el parámetro de configuración que está causando el fallo. Quizás deba corregir varios parámetros antes de eliminar este fallo y de establecer la conexión correctamente.

16#000A	El estado de un atributo Get_Attributes_List o Set_Attributes_List no corresponde a cero.	Existen posibilidades: <ul style="list-style-type: none"> <li>• se está creando una conexión cuyo tipo de conexión no es válido.</li> <li>• un atributo de objeto o valor de etiqueta no es válido.</li> </ul> <p>Si un atributo de objeto o etiqueta no es válido, exporte el archivo de Logix Designer y luego vuelva a importarlo. Si procede, reprograme la red ControlNet después de volver importar.</p>
16#000C	Error de solicitud de servicio: Modo/estado no válido para la solicitud de servicio.	El controlador intenta obtener una petición de servicio del módulo y ha recibido un error. Primero, asegúrese de que el módulo no tenga ningún fallo.  Para un módulo de I/O, esto puede indicar que el módulo tiene una de estas condiciones: <ul style="list-style-type: none"> <li>• Comunicación limitada, pero tiene un Fallo mayor</li> <li>• Se está realizando o se tiene que realizar una actualización del firmware.</li> </ul> <p>Consulte la pestaña Información del módulo (Module Info) para determinar la causa exacta.</p>
16#000D	El objeto ya existe.	Se ha creado una instancia de asignación de E/S cuando la instancia ya está en uso.
16#000E	El valor del atributo no se puede establecer.	Se ha configurado una instrucción MSG para cambiar un valor de atributo que no se puede cambiar.
16#000F	Permiso de acceso denegado para servicio solicitado.	Se ha configurado una instrucción MSG para eliminar un objeto de asignación que no se puede eliminar.
16#0010	El modo o estado del módulo no permite que el objeto realice el servicio solicitado.	El estado del dispositivo impide que se lleve a cabo una solicitud de servicio.
16#0011	Datos de respuesta demasiado largos.	La respuesta a un mensaje tiene un tamaño de datos demasiado grande para el destino.  Cambio el destino a una etiqueta que pueda administrar el tamaño y tipo de datos de respuesta.
16#0013	Configuración del módulo rechazada: El tamaño de los datos es demasiado pequeño.	La configuración del módulo no es válida. No se han enviado suficientes datos de configuración.  Verifique que el módulo de destino es correcto.
16#0014	Atributo no definido o no compatible.	Se ha configurado una instrucción MSG para cambiar un atributo que no existe.
16#0015	Configuración del módulo rechazada: El tamaño de los datos es demasiado grande.	La configuración del módulo no es válida. Se han enviado demasiados datos de configuración.  Verifique que el módulo de destino es correcto.

## Fallos de módulo: 16#0100 - 16#01ff

Estos son los fallos del módulo: 16#0100 - 16#01ff

Código	Cadena	Explicación y posibles causas/soluciones
16#0100	Error de solicitud de conexión: Módulo en uso.	<ul style="list-style-type: none"> <li>• La conexión a la que se está accediendo ya está en uso.</li> </ul> <p>Existen posibilidades:</p> <ul style="list-style-type: none"> <li>• El controlador intenta hacer una conexión específica con un módulo y el módulo no puede aceptar más de una de estas conexiones.</li> <li>• El destino de una conexión reconoce que el propietario está intentando volver a establecer una conexión que ya se está ejecutando.</li> </ul>

16#0103	Error de solicitud de servicio: Clase de transporte CIP no admitida.	<p>Existen posibilidades:</p> <ul style="list-style-type: none"><li>• El controlador está pidiendo servicios que el módulo no acepta.</li><li>• El módulo en uso (es decir, el módulo físico) es diferente al módulo especificado en el árbol de configuración de E/S y por lo tanto está haciendo que la conexión o el servicio falle.</li></ul> <p>El fallo puede producirse incluso cuando el módulo haya pasado la prueba de codificación electrónica. Esto puede producirse cuando se usan las opciones Deshabilitar codificación o Módulo compatible en la configuración del módulo, en lugar de la opción Coincidencia exacta.</p> <p>A pesar de pasar la prueba de codificación electrónica, el módulo con el que se está conectando no tiene las mismas características o configuración que el módulo especificado en el árbol de configuración de E/S y no admite la conexión o servicio deseados.</p> <p>Compruebe el módulo en uso y verifique que coincida exactamente con el módulo especificado en el árbol de configuración de E/S de la aplicación Logix Designer.</p>
16#0106	Error de solicitud de conexión: El módulo pertenece y ha sido configurado por otro controlador. El módulo sólo puede aceptar una conexión si se está utilizando Unidifusión.	<p>Se ha producido un conflicto de propiedad para la conexión.</p> <p>Existe una de estas condiciones:</p> <ul style="list-style-type: none"><li>• La Solicitud de conexión al módulo ha sido rechazada por un conflicto de Propiedad con otro Propietario (por ejemplo, con otro Controlador). Esto puede ocurrir con módulos como los de salida, que sólo permiten que un Propietario configure y controle sus salidas.</li></ul> <p>Este fallo también puede ocurrir si se configura el módulo como de Sólo recepción y sólo admite una conexión.</p> <ul style="list-style-type: none"><li>• Si se conecta el Propietario al módulo con una conexión de Unidifusión por EtherNet/IP, fallarán otras conexiones con el módulo, puesto que el Propietario controla la conexión.</li></ul> <p>Si se conecta el Propietario al módulo con una conexión de Multidifusión por EtherNet/IP, fallarán las conexiones de Unidifusión con el módulo, puesto que el Propietario controla la conexión.</p> <p>Configure tanto la conexión de Propietario como la de Sólo recepción como conexiones de Multidifusión.</p>
16#0107	Error de solicitud de conexión: Tipo desconocido.	No se ha encontrado una conexión a la que se está accediendo.

16#0108	Error de solicitud de conexión: Tipo de conexión (Multidifusión/Unidifusión) incompatible.	<p>El controlador está pidiendo un tipo de conexión incompatible con el módulo.</p> <p>Existe una de estas condiciones:</p> <ul style="list-style-type: none"> <li>• El módulo en uso (es decir, el módulo físico) es diferente al módulo especificado en el árbol de configuración de E/S y por lo tanto está haciendo que la conexión o el servicio falle.</li> <li>• El fallo puede producirse incluso cuando el módulo haya pasado la prueba de codificación electrónica. Esto puede producirse cuando se usan las opciones Deshabilitar codificación o Codificación compatible en la configuración del módulo, en lugar de la opción Coincidencia exacta.</li> </ul> <p>A pesar de pasar la prueba de codificación electrónica, el módulo con el que se está conectando no tiene las mismas características o configuración que el módulo especificado en el árbol de configuración de E/S y no admite la conexión o servicio deseados.</p> <p>Compruebe el módulo en uso y verifique que coincida exactamente con el módulo especificado en el árbol de configuración de E/S de la aplicación Logix Designer.</p> <ul style="list-style-type: none"> <li>• Puede que haya configurado una etiqueta consumida para que utilice una conexión de Unidifusión, pero el controlador productor no admite conexiones de Unidifusión.</li> </ul>
16#0109	Error de solicitud de conexión: Tamaño de conexión no válido.  <b>Consejo:</b> Se muestra información adicional de error sobre este fallo en el nombre de etiqueta asociado al número de instancia de conexión que ha fallado.	<p>El tamaño de conexión no coincide con el esperado.</p> <p>Existen posibilidades:</p> <ul style="list-style-type: none"> <li>• el controlador intenta establecer sin éxito una conexión con el módulo. El tamaño de la conexión no es válido.</li> <li>• puede que el controlador intente conectarse con una etiqueta de un controlador originador cuyo tamaño no coincide con la etiqueta de este controlador.</li> <li>• el módulo en uso (es decir, el módulo físico) es diferente al módulo especificado en el árbol de configuración de E/S y por lo tanto está haciendo que la conexión o el servicio falle.</li> <li>• el fallo puede producirse incluso cuando el módulo haya pasado la prueba de codificación electrónica. Esto puede producirse cuando se usan las opciones Deshabilitar codificación o Codificación compatible en la configuración del módulo, en lugar de la opción Coincidencia exacta.</li> </ul> <p>A pesar de pasar la prueba de codificación electrónica, el módulo con el que se está conectando no tiene las mismas características o configuración que el módulo especificado en el árbol de configuración de E/S y no admite la conexión o servicio deseados.</p> <p>Compruebe el módulo en uso y verifique que coincida exactamente con el módulo especificado en el árbol de configuración de E/S de la aplicación Logix Designer.</p> <p>Si el módulo es un módulo 1756 ControlNet, verifique que el tamaño del chasis es correcto.</p> <p>Para adaptadores de E/S remota, verifique que el tamaño del rack y/o las densidad del rack son correctos.</p>

16#0110	Error de solicitud de conexión: Módulo no configurado.	<p>El controlador intenta establecer sin éxito una conexión de Sólo recepción con el módulo - El módulo no ha sido configurado ni conectado por un Propietario (por ejemplo, otro Controlador).</p> <p>Este controlador no es Propietario de este módulo, puesto que intenta establecer una conexión de sólo recepción, que no requiere configuración de módulo. No es posible establecer una conexión hasta que un Propietario se configure y se conecte al módulo.</p>
16#0111	Intervalo de paquete solicitado (RPI) fuera de rango.	<p>Existen posibilidades:</p> <ul style="list-style-type: none"><li>• el Intervalo de paquete solicitado (RPI) especificado no es válido para este módulo o para otro módulo que se encuentra en su ruta. Consulte la pestaña Avanzado (Advanced) para habilitar el RPI del productor.</li><li>• el módulo en uso (es decir, el módulo físico) es diferente al módulo especificado en el árbol de configuración de E/S y por lo tanto está haciendo que la conexión o el servicio falle.</li></ul> <p>El fallo puede producirse incluso cuando el módulo haya pasado la prueba de codificación electrónica. Esto puede producirse cuando se usan las opciones Deshabilitar codificación o Módulo compatible en la configuración del módulo, en lugar de la opción Coincidencia exacta.</p> <p>A pesar de pasar la prueba de codificación electrónica, el módulo con el que se está conectando no tiene las mismas características o configuración que el módulo especificado en el árbol de configuración de E/S y no admite la conexión o servicio deseados.</p> <p>Compruebe el módulo en uso y verifique que coincide exactamente con el módulo especificado en el árbol de configuración de E/S de la aplicación Logix Designer.</p> <ul style="list-style-type: none"><li>• para conexiones de Sólo recepción: el RPI establecido por el propietario de este módulo es más lento que el solicitado. Incremente el RPI solicitado o reduzca el RPI que está utilizando el propietario del controlador.</li></ul> <p>Consulte la pestaña Conexión (Connection) en el cuadro de diálogo Propiedades del módulo (Module Properties) para conocer los valores RPI válidos.</p>
16#0113	Error de solicitud de conexión: Límite de conexión de módulo superado.	<p>El número de conexiones es superior al que hay disponible en el módulo. El número de conexiones debe reducirse o el hardware debe actualizarse.</p> <p>Para reducir el número de conexiones:</p> <ul style="list-style-type: none"><li>• Cambie el Formato de comunicación del adaptador de comunicación de E/S Flex de la configuración de entrada o salida a <b>Optimización de racks</b>. Cuando se modifica el Formato de comunicación, se debe quitar el adaptador y recrearlo en el árbol de configuración de E/S.</li><li>• Si la configuración utiliza mensajería por ControlNet, secuencie los mensajes para reducir el número de mensajes ejecutados al mismo tiempo o reduzca el número de mensajes. Los mensajes (instrucciones MSG) también utilizan conexiones.</li></ul>

16#0114	Discrepancia en la codificación electrónica: El código de producto de codificación electrónica y/o el ID del proveedor no coinciden.	<p>El Código de producto del hardware del módulo real no coincide con el Código de producto del módulo creado en el software.</p> <p>La Codificación electrónica para este módulo ha fallado. Es posible que exista una desigualdad entre el módulo creado en el software y el hardware del módulo real.</p>
16#0115	Discrepancia en la codificación electrónica: El tipo de producto de Codificación electrónica no coincide.	<p>El Tipo de producto del hardware del módulo real no coincide con el Tipo de producto del módulo creado en el software.</p> <p>La Codificación electrónica para este módulo ha fallado. Es posible que exista una desigualdad entre el módulo creado en el software y el hardware del módulo real.</p>
16#0116	Discrepancia en la codificación electrónica: Las revisiones Mayor y/o Menor no son válidas o son incorrectas.	<p>Las revisiones del módulo Mayor y/o Menor no coinciden con las revisiones Mayor o Menor del módulo creado en el software.</p> <p>Compruebe que ha especificado las revisiones mayores y menores correctas si ha seleccionado Módulo compatible o la codificación Coincidencia exacta.</p> <p>La Codificación electrónica para este módulo ha fallado. Es posible que exista una desigualdad entre el módulo creado en el software y el hardware del módulo real.</p>
16#0117	<p>Error de solicitud de conexión: Punto de conexión no válido.</p> <p><b>Consejo:</b> Se muestra Información adicional de error sobre este fallo como el nombre de etiqueta asociado con el controlador para el controlador (C2C) que ha fallado.</p>	<p>La conexión se realiza a un puerto no válido o el puerto ya está en uso.</p> <p>Existe una de estas condiciones:</p> <ul style="list-style-type: none"> <li>Otro controlador es el propietario de este módulo y ha conectado con Formatos de comunicación: Módulos E/S distintos del escogido por este controlador. Compruebe que el formato de comunicaciones elegido es idéntico al elegido por el primer controlador del módulo.</li> <li>El módulo en uso (es decir, el módulo físico) es diferente al módulo especificado en el árbol de configuración de E/S y por lo tanto está haciendo que la conexión o el servicio falle.</li> </ul> <p>El fallo puede producirse incluso cuando el módulo haya pasado la prueba de codificación electrónica. Esto puede producirse cuando se usan las opciones Deshabilitar codificación o Módulo compatible en la configuración del módulo, en lugar de la opción Coincidencia exacta.</p> <p>A pesar de pasar la prueba de codificación electrónica, el módulo con el que se está conectando no tiene las mismas características o configuración que el módulo especificado en el árbol de configuración de E/S y no admite la conexión o servicio deseados.</p> <p>Compruebe el módulo en uso y verifique que coincide exactamente con el módulo especificado en el árbol de configuración de E/S de la aplicación Logix Designer.</p> <ul style="list-style-type: none"> <li>Puede que el controlador se intente conectar a una etiqueta inexistente en un controlador productor.</li> </ul>

16#0118	Configuración del módulo rechazada: Error de formato.	<p>Se ha utilizado un formato de configuración no válido. Existe una de estas condiciones:</p> <ul style="list-style-type: none"> <li>• La clase de configuración especificada no coincide con la clase admitida por el módulo.</li> <li>• El módulo no reconoce la instancia de conexión.</li> <li>• La ruta de acceso especificada para la conexión es inconsistente.</li> <li>• El módulo en uso (es decir, el módulo físico) es diferente al módulo especificado en el árbol de configuración de E/S y por lo tanto está haciendo que la conexión o el servicio falle.</li> </ul> <p>El fallo puede producirse incluso cuando el módulo haya pasado la prueba de codificación electrónica. Esto puede producirse cuando se usan las opciones Deshabilitar codificación o Módulo compatible en la configuración del módulo, en lugar de la opción Coincidencia exacta.</p> <p>A pesar de pasar la prueba de codificación electrónica, el módulo con el que se está conectando no tiene las mismas características o configuración que el módulo especificado en el árbol de configuración de E/S y no admite la conexión o servicio deseados.</p> <p>Compruebe el módulo en uso y verifique que coincida exactamente con el módulo especificado en el árbol de configuración de E/S de la aplicación Logix Designer.</p>
16#0119	Error de solicitud de conexión: Módulo sin propietario.	<p>La conexión de control no está abierta.</p> <p>Donde se requiere una conexión de Sólo recepción, la conexión de control no está abierta.</p>
16#011A	Error de solicitud de conexión: Sin recursos de conexión	<p>El controlador intenta establecer sin éxito una conexión con el módulo. Los recursos solicitados no están disponibles.</p> <p>Si el módulo es un módulo 1756 ControlNet, se pueden usar hasta cinco controladores para establecer conexiones de optimización de rack con el módulo. Verifique que no se ha sobrepasado este número.</p> <p>Si el módulo es un adaptador 1794-ACN15, 1794-ACNR15 o 1797-ACNR15, sólo se puede usar un controlador para establecer una conexión de optimización de rack con el módulo. Verifique que no se ha sobrepasado este número.</p>

## Fallos de módulo: 16#0200 - 16#02ff

Estos son los fallos del módulo: 16#0200 - 16#02ff.

Código	Cadena	Explicación y posibles causas/soluciones
16#0203	Se ha superado el tiempo de espera para la conexión.	<p>El propietario o creador reconoce que el dispositivo de destino está en la red o backplane, sin embargo, los datos o mensajes de E/S no están recibiendo respuesta. En otras palabras, se puede acceder al destino, pero su respuesta no es la esperada. Por ejemplo, este fallo puede aparecer cuando no se devuelven los paquetes de Ethernet de multidifusión.</p> <p>Cuando se produce este fallo, el controlador normalmente intenta interrumpir y volver a establecer la conexión continuamente.</p> <p>Si está utilizando módulos E/S FLEX, compruebe que está usando el dispositivo terminal correcto.</p>

16#0204	Error de solicitud de conexión: Se ha superado el tiempo de espera para la solicitud de la conexión.	<p>El controlador intenta establecer una conexión, sin embargo, el módulo de destino no responde.</p> <p>Parece también que falta el dispositivo en el backplane o red.</p> <p>Para recuperar, realice estas acciones:</p> <ul style="list-style-type: none"> <li>• Compruebe que el módulo no se haya extraído, que sigue funcionando y recibiendo alimentación.</li> <li>• Compruebe que se ha indicado el número correcto de ranura.</li> <li>• Compruebe que el módulo está conectado correctamente a la red.</li> </ul> <p>Si está usando módulos E/S FLEX, compruebe que se encuentra en uso el bloque de terminal correcto.</p>
16#0205	Error de solicitud de conexión: Parámetro no válido.	<p>Existen posibilidades:</p> <ul style="list-style-type: none"> <li>• El controlador intenta establecer una conexión con el módulo y ha recibido un error. Un parámetro tiene un error.</li> <li>• El módulo en uso (es decir, el módulo físico) es diferente al módulo especificado en el árbol de configuración de E/S y por lo tanto está haciendo que la conexión o el servicio falle.</li> </ul> <p>El fallo puede producirse incluso cuando el módulo haya pasado la prueba de codificación electrónica. Esto puede producirse cuando se usan las opciones Deshabilitar codificación o Módulo compatible en la configuración del módulo, en lugar de la opción Coincidencia exacta.</p> <p>A pesar de pasar la prueba de codificación electrónica, el módulo con el que se está conectando no tiene las mismas características o configuración que el módulo especificado en el árbol de configuración de E/S y no admite la conexión o servicio deseados.</p> <p>Compruebe el módulo en uso y verifique que coincida exactamente con el módulo especificado en el árbol de configuración de E/S de la aplicación Logix Designer.</p>
16#0206	Error de solicitud de conexión: El tamaño solicitado es demasiado grande.	<p>Existen posibilidades:</p> <ul style="list-style-type: none"> <li>• El controlador intenta establecer una conexión con el módulo y ha recibido un error. El tamaño solicitado es demasiado grande.</li> <li>• El módulo en uso (es decir, el módulo físico) es diferente al módulo especificado en el árbol de configuración de E/S y por lo tanto está haciendo que la conexión o el servicio falle.</li> </ul> <p>El fallo puede producirse incluso cuando el módulo haya pasado la prueba de codificación electrónica. Esto puede producirse cuando se usan las opciones Deshabilitar codificación o Módulo compatible en la configuración del módulo, en lugar de la opción Coincidencia exacta.</p> <p>A pesar de pasar la prueba de codificación electrónica, el módulo con el que se está conectando no tiene las mismas características o configuración que el módulo especificado en el árbol de configuración de E/S y no admite la conexión o servicio deseados.</p> <p>Compruebe el módulo en uso y verifique que coincida exactamente con el módulo especificado en el árbol de configuración de E/S de la aplicación Logix Designer.</p>

## Fallos de módulo: 16#0300 - 16#03ff

Estos son los fallos del módulo: 16#0300 - 16#03ff

Código	Cadena	Explicación y posibles causas/soluciones
--------	--------	--

16#0301	Error de solicitud de conexión: Sin memoria de búfer.	Puede existir una de las siguientes condiciones: <ul style="list-style-type: none"><li>• El controlador intenta establecer una conexión con el módulo y ha recibido un error: un módulo de la ruta de acceso no tiene memoria.</li><li>• Puede que el controlador intente conectarse con una etiqueta de un controlador originador que no está marcado como producido.</li><li>• Puede que el controlador se intente conectar con una etiqueta de un controlador originador. Puede que la etiqueta no se haya configurado para permitir suficientes consumidores.</li><li>• Reduzca el número o el tamaño de conexiones de este módulo.</li><li>• Uno de los módulos de red que se encuentran entre el módulo y el controlador puede haberse quedado sin memoria. Compruebe la configuración de la red del sistema.</li><li>• El módulo puede estar sin memoria. Compruebe la configuración del sistema y la capacidad del módulo.</li><li>• El módulo en uso (es decir, el módulo físico) es diferente al módulo especificado en el árbol de configuración de E/S y por lo tanto está haciendo que la conexión o el servicio falle.</li></ul> El fallo puede producirse incluso cuando el módulo haya pasado la prueba de codificación electrónica. Esto puede producirse cuando se usan las opciones Deshabilitar codificación o Módulo compatible en la configuración del módulo, en lugar de la opción Coincidencia exacta. A pesar de pasar la prueba de codificación electrónica, el módulo con el que se está conectando no tiene las mismas características o configuración que el módulo especificado en el árbol de configuración de E/S y no admite la conexión o servicio deseados. Compruebe el módulo en uso y verifique que coincida exactamente con el módulo especificado en el árbol de configuración de E/S de la aplicación Logix Designer.
16#0302	Error de solicitud de conexión: Fuera del ancho de banda de comunicación.	El controlador intenta establecer una conexión con el módulo y ha recibido un error. Un módulo de la ruta ha superado su capacidad de ancho de banda de comunicaciones. Aumente el Intervalo de paquete solicitado (RPI) y vuelva a configurar la red con RSNetWorx. Distribuya la carga en otro módulo de puente.
16#0303	Error de solicitud de conexión: No hay un puente disponible.	El controlador intenta establecer una conexión con el módulo y ha recibido un error. Un módulo de la ruta ha superado su capacidad de ancho de banda de comunicaciones. Distribuya la carga en otro módulo de puente.
16#0304	No se ha configurado para enviar datos programados.	El módulo ControlNet no está programado para enviar datos. Use el software RSNetWorx for ControlNet para programar o reprogramar la red ControlNet.
16#0305	Error de solicitud de conexión: La configuración de ControlNet en el controlador no es igual a la configuración del puente.	La configuración de ControlNet del controlador no es igual a la configuración el módulo de puente. Esto puede suceder porque un módulo ControlNet ha cambiado después de programar la red o porque se ha cargado un nuevo programa de control en el controlador. Use el software RSNetWorx for ControlNet para reprogramar las conexiones.

16#0306	No hay un Maestro de configuración de ControlNet (CCM) disponible.	<p>No se encuentra el Maestro de configuración de ControlNet (CCM). Los módulos 1756-CNB y PLC-5C son los únicos módulos capaces de actuar como CCM, y el CCM debe tener número de nodo 1.</p> <p>Compruebe que haya un módulo 1756-CNB o PLC-5C en el número de nodo 1 y que funcione correctamente.</p> <p>Este fallo se puede producir temporalmente cuando se enciende el sistema y se resuelve cuando se localiza el CCM.</p>
16#0311	Error de solicitud de conexión: Puerto no válido.	<p>El controlador intenta establecer una conexión con el módulo y ha recibido un error.</p> <p>Verifique que todos los módulos del árbol Configuración de E/S son los módulos correctos.</p>
16#0312	Error de solicitud de conexión: Dirección de vínculo no válida.	<p>El controlador intenta establecer una conexión con el módulo y ha recibido un error. Se ha especificado una dirección de vínculo no válida. Una dirección de vínculo puede ser un número de ranura, una dirección de red o el número y grupo inicial del chasis de E/S remoto.</p> <p>Compruebe que el número de ranura seleccionado para este módulo no sea mayor que el tamaño del rack.</p> <p>Compruebe que el número de nodo de ControlNet no sea mayor que el número máximo de nodo configurado para la red en el software RSNetWorx for ControlNet.</p>
16#0315	Error de solicitud de conexión: Tipo de segmento no válido	<p>La ruta o el tipo de segmento no son válidos.</p> <p>Existen posibilidades:</p> <ul style="list-style-type: none"> <li>• el controlador intenta establecer una conexión con el módulo y ha recibido un error. La conexión solicitada no es válida.</li> <li>• el módulo en uso (es decir, el módulo físico) es diferente al módulo especificado en el árbol de configuración de E/S y por lo tanto está haciendo que la conexión o el servicio falle.</li> </ul> <p>El fallo puede producirse incluso cuando el módulo haya pasado la prueba de codificación electrónica. Esto puede producirse cuando se usan las opciones Deshabilitar codificación o Módulo compatible en la configuración del módulo, en lugar de la opción Coincidencia exacta.</p> <p>A pesar de pasar la prueba de codificación electrónica, el módulo con el que se está conectando no tiene las mismas características o configuración que el módulo especificado en el árbol de configuración de E/S y no admite la conexión o servicio deseados.</p> <p>Compruebe el módulo en uso y verifique que coincida exactamente con el módulo especificado en el árbol de configuración de E/S de la aplicación Logix Designer.</p>
16#0317	Error de solicitud de conexión: Conexión no programada.	<p>El controlador intenta establecer una conexión ControlNet con el módulo y ha recibido un error.</p> <p>Use el software RSNetWorx for ControlNet para programar o reprogramar la conexión con este módulo.</p>
16#0318	Error de solicitud de conexión: Dirección de vínculo no válida; no se puede autodirigir.	<p>El controlador intenta establecer una conexión con el módulo y ha recibido un error: la dirección del vínculo no es válida.</p> <p>Compruebe que el módulo ControlNet asociado tenga el número de ranura y/o nodo seleccionados.</p>
16#0319	Error de solicitud de conexión: No hay recursos secundarios disponibles en chasis redundante.	<p>El controlador intenta establecer una conexión con el módulo y ha recibido un error. El módulo redundante no dispone de los recursos necesarios para aceptar la conexión.</p> <p>Reduzca el tamaño o el número de conexiones de este módulo o añada otro controlador o módulo ControlNet al sistema.</p>

16#031a	Error de solicitud de conexión: Conexión en rack rechazada.	<p>El controlador intenta establecer sin éxito una conexión directa con el módulo y ha recibido un error. Ya se ha establecido una conexión optimizada del rack para este módulo a través del 1756-CNB/R del mismo chasis.</p> <ul style="list-style-type: none"> <li>• Conéctese a este módulo mediante el 1756-CNB/R del mismo chasis.</li> <li>• Conéctese a este módulo mediante un 1756-CNB/R distinto para usar una conexión directa.</li> <li>• Cambie la primera conexión del rack optimizado a directa y después restablezca la segunda conexión directa.</li> <li>• Conéctese a este módulo desde un controlador que se encuentre en el mismo chasis que el módulo (no se conecte mediante un 1756-CNB/R).</li> </ul>
16#031e	Error de solicitud de conexión: No es posible consumir la etiqueta.	<ul style="list-style-type: none"> <li>• El controlador intenta conectarse con una etiqueta de un controlador originador y ha recibido un error.</li> <li>• El controlador intenta conectarse a una etiqueta de un controlador originador que ya están utilizando demasiados consumidores. Incremente el número máximo de consumidores de la etiqueta.</li> </ul>
16#031f	Error de solicitud de conexión: No es posible consumir la etiqueta.	No se ha encontrado ningún objeto de conexión del SC (controlador de servicio) que corresponda a una instancia de símbolo.
16#0322	Error de solicitud de conexión: Discrepancia de punto de conexión	<p>Ha ocurrido una desigualdad en el punto de conexión. Existen posibilidades:</p> <ul style="list-style-type: none"> <li>• la nueva conexión solicitada no coincide con la conexión ya existente. Compruebe los controladores que están usando la conexión y verifique que todas las configuraciones sean idénticas.</li> <li>• la conexión solicitada no es un tipo de conexión de recepción o control.</li> </ul>

**Fallos de módulo:****16#0800 - 16#08ff**

Estos son los fallos del módulo: 16#0800 - 16#08ff

Código	Cadena	Explicación y posibles causas/soluciones
16#0800	El vínculo de la red en la ruta de acceso al module está fuera de línea.	No hay interpretación disponible.
16#0801	RPI de multidifusión incompatible.	No hay interpretación disponible.
16#0810	No se dispone de datos de aplicación de destino.	<p>La aplicación de control no ha inicializado los datos que debe producir el dispositivo receptor. Esto puede producirse cuando hay conexiones de "envío de datos" configuradas en un dispositivo receptor y la aplicación de control de dicho dispositivo no ha inicializado los datos que deben producirse. Para el dispositivo receptor asociado a la conexión de "envío de datos" que informa de este error de conexión, inicie la aplicación de control y realice una escritura de datos como mínimo. Consulte la documentación del dispositivo receptor y su aplicación de control para obtener información sobre cómo hacerlo.</p>
16#0814	Error de solicitud de conexión: Desigualdad de Data Type.	Se encontró información de estado de conexión no válida.

## Fallos de módulo: 16#fd00 - 16#fdff

Los fallos del módulo: 16#fd00 - 16#fdff.

Código	Cadena	Explicación y posibles causas/soluciones
16#fd03	Error de solicitud de conexión: Falta la conexión solicitada	El controlador intenta establecer una conexión con el módulo y ha recibido un error. Este módulo requiere un determinado conjunto de conexiones y tipos de conexión, y falta uno de los tipos de conexión. <ul style="list-style-type: none"> <li>• Póngase en contacto con Soporte técnico</li> <li>• <a href="http://www.support.rockwellautomation.com">http://www.support.rockwellautomation.com</a></li> </ul>
16#fd04	Error de solicitud de conexión: Maestro CST no detectado	El controlador intenta establecer una conexión con el módulo y ha recibido un error. Este módulo requiere un CST maestro en el chasis. <ul style="list-style-type: none"> <li>• Configure un módulo (normalmente un controlador) en el chasis para que sea el maestro CST.</li> <li>• Póngase en contacto con Soporte técnico</li> <li>• <a href="http://www.support.rockwellautomation.com">http://www.support.rockwellautomation.com</a></li> </ul>
16#fd05	Error de solicitud de conexión: Eje o grupo no asignados.	El controlador intenta establecer una conexión con el módulo y ha recibido un error. Este módulo requiere la asignación de un eje o una tabla de grupo. <ul style="list-style-type: none"> <li>• Asigne un grupo o un eje.</li> <li>• Póngase en contacto con Soporte técnico</li> <li>• <a href="http://www.support.rockwellautomation.com">http://www.support.rockwellautomation.com</a></li> </ul>
16#fd06	Fallo de transición	El comando del controlador para pasar el anillo SERCOS a una nueva phase devolvió un error del módulo. Compruebe si hay Nodos de variador duplicados.
16#fd07	Régimen de datos SERCOS incorrecto	Se ha producido un error al intentar configurar el anillo SERCOS. La velocidad en baudios de todos los dispositivos debe ser la misma y debe ser admitida por todos los variadores y por el módulo SERCOS.
16#fd08	Error de Com. SERCOS	Principalmente, dos conjuntos de errores pueden producir un Com. Fallo: fallos de interfaz y físicos. Posibles orígenes de errores físicos pueden ser: <ul style="list-style-type: none"> <li>• Anillo roto</li> <li>• Conector suelto</li> <li>• Fibra óptica sucia</li> <li>• Ruido eléctrico debido a una conexión a tierra inadecuada del variador</li> <li>• Demasiados nodos en el anillo</li> </ul> Se encontraron errores de interfaz al configurar variadores de terceros. Posibles orígenes de errores de interfaz pueden ser: <ul style="list-style-type: none"> <li>• Sin SERCOS MST (Error de protocolo)</li> <li>• AT ausente (el variador no envío datos cuando se esperaba)</li> <li>• Error de temporización SERCOS en fase 3</li> <li>• Error en los datos del variador devueltos al módulo SERCOS</li> </ul>
16#fd09	Fallo de inicialización de nodo	Un intento del controlador para configurar el nodo para la operación cíclica devolvió un error.

16#fd0a	Error de atributo del eje	Se recibió una respuesta incorrecta de un módulo de movimiento.
16#fd0c	Error por gran maestro diferente	El dispositivo final posee un gran maestro diferente del controlador.
16#fd1f	Formato de protocolo de seguridad erróneo	Ocurrió un error al añadir el segmento de red de seguridad a una ruta.
16#fd20	Ninguna tarea de seguridad	No aparece ninguna tarea de seguridad ejecutándose.
16#fd22	Desigualdad de tamaño del chasis	Compruebe el número de módulos físicos de E/S de expansión que hay configurados para el controlador y luego actualice el número de módulos seleccionados de la lista de E/S de expansión en la página General del cuadro de diálogo Propiedades del controlador (Controller Properties).
16#fd23	Tamaño del chasis superado	Para verificar el número de módulos físicos de E/S de expansión que admite el controlador, abra el cuadro de diálogo Propiedades del controlador (Controller Properties) y expanda la lista de E/S de expansión en la página General. Configure el número de módulos físicos de E/S de expansión para que coincida con la selección de la lista de E/S de expansión.

## Fallos de módulo: 16#fe00 - 16#feff

Los fallos del módulo: 16#fe00 - 16#feff.

Código	Cadena	Explicación y posibles causas/soluciones
16#fe01		Se ha encontrado un formato de configuración no válido.
16#fe02	Intervalo de paquete solicitado (RPI) fuera de rango.	El Intervalo de paquete solicitado (RPI) especificado no es válido para este módulo. <ul style="list-style-type: none"><li>• Consulte la ficha Conexión (Connection) para ver una lista de los valores de RPI válidos.</li></ul>
16#fe03		No se ha establecido el punto de conexión de entrada.
16#fe04	Error de solicitud de conexión: Indicador de datos de entada no válido.	El controlador intenta establecer una conexión con el módulo y ha recibido un error.

16#fe05	Error de solicitud de conexión: Tamaño de datos de entrada no válido.	<p>Existen posibilidades:</p> <ul style="list-style-type: none"> <li>el controlador intenta establecer una conexión con el módulo y ha recibido un error.</li> <li>el módulo en uso (es decir, el módulo físico) es diferente al módulo especificado en el árbol de configuración de E/S y por lo tanto está haciendo que la conexión o el servicio falle.</li> </ul> <p>El fallo puede producirse incluso cuando el módulo haya pasado la prueba de codificación electrónica. Esto puede producirse cuando se usan las opciones Deshabilitar codificación o Módulo compatible en la configuración del módulo, en lugar de la opción Coincidencia exacta.</p> <p>A pesar de pasar la prueba de codificación electrónica, el módulo con el que se está conectando no tiene las mismas características o configuración que el módulo especificado en el árbol de configuración de E/S y no admite la conexión o servicio deseados.</p> <p>Compruebe el módulo en uso y verifique que coincida exactamente con el módulo especificado en el árbol de configuración de E/S de la aplicación Logix Designer.</p>
16#fe06		No se ha establecido el punto forzado de entrada.
16#fe07		No se ha establecido el punto de conexión de salida.
16#fe08	Error de solicitud de conexión: Indicador de datos de salida no válido.	El controlador intenta establecer una conexión con el módulo y ha recibido un error.
16#fe09	Error de solicitud de conexión: Tamaño de datos de salida no válido.	<p>Existen posibilidades:</p> <ul style="list-style-type: none"> <li>El controlador intenta establecer una conexión con el módulo y ha recibido un error.</li> <li>el módulo en uso (es decir, el módulo físico) es diferente al módulo especificado en el árbol de configuración de E/S y por lo tanto está haciendo que la conexión o el servicio falle.</li> </ul> <p>El fallo puede producirse incluso cuando el módulo haya pasado la prueba de codificación electrónica. Esto puede producirse cuando se usan las opciones Deshabilitar codificación o Módulo compatible en la configuración del módulo, en lugar de la opción Coincidencia exacta.</p> <p>A pesar de pasar la prueba de codificación electrónica, el módulo con el que se está conectando no tiene las mismas características o configuración que el módulo especificado en el árbol de configuración de E/S y no admite la conexión o servicio deseados.</p> <p>Compruebe el módulo en uso y verifique que coincida exactamente con el módulo especificado en el árbol de configuración de E/S de la aplicación Logix Designer.</p>
16#fe0a		No se ha establecido el indicador de forzado de salida.

16#fe0b	Cadena de símbolos no válida.	<p>Existen posibilidades:</p> <ul style="list-style-type: none"> <li>la etiqueta que se va a consumir en este módulo no es válida. Compruebe que la etiqueta esté marcada como producida.</li> <li>el módulo en uso (es decir, el módulo físico) es diferente al módulo especificado en el árbol de configuración de E/S y por lo tanto está haciendo que la conexión o el servicio falle.</li> </ul> <p>El fallo puede producirse incluso cuando el módulo haya pasado la prueba de codificación electrónica. Esto puede producirse cuando se usan las opciones Deshabilitar codificación o Módulo compatible en la configuración del módulo, en lugar de la opción Coincidencia exacta.</p> <p>A pesar de pasar la prueba de codificación electrónica, el módulo con el que se está conectando no tiene las mismas características o configuración que el módulo especificado en el árbol de configuración de E/S y no admite la conexión o servicio deseados.</p> <p>Compruebe el módulo en uso y verifique que coincida exactamente con el módulo especificado en el árbol de configuración de E/S de la aplicación Logix Designer.</p>
16#fe0c	Número de instancia PLC-5 no válido.	<p>El controlador intenta establecer una conexión con PLC-5 y ha recibido un error.</p> <p>Compruebe que el número de instancia especificado se haya establecido correctamente en el PLC-5.</p>
16#fe0d	No hay etiqueta en controlador similar.	<p>No se ha establecido el número de instancia de símbolo.</p>
16#fe0e	Actualización automática de firmware en curso.	<p>El módulo está siendo actualizado.</p>
16#fe0f	Fallo en la Actualización automática de firmware: Archivo de firmware incompatible con el módulo.	<p>El supervisor de firmware ha intentado actualizar un módulo no compatible.</p>
16#fe10	Fallo en la Actualización automática de firmware: Archivo de firmware no encontrado.	<p>No se puede encontrar el archivo de firmware necesario para actualizar el módulo.</p>
16#fe11	Fallo en la Actualización automática de firmware: Archivo de firmware no válido.	<p>El archivo de firmware está dañado.</p>
16#fe12	Fallo al actualizar automáticamente el firmware.	<p>Se ha producido un error mientras se actualizaba el módulo.</p>
16#fe13	Fallo en la Actualización automática de firmware: Conexiones activas detectadas.	<p>No se pudo realizar una conexión activa con el módulo de destino.</p>
16#fe14	Actualización automática de firmware pendiente: Buscando archivo NVS para identidad del módulo apropiado.	<p>El archivo de firmware está siendo leído.</p>
16#fe22		<p>El tipo de conexión de parámetros de red destino-origen no es válido.</p>
16#fe23		<p>La conexión de parámetros de red destino-origen no especifica si se permite unidifusión.</p>

## Fallos de módulo: 16#ff00 - 16#ffff

Estos son los fallos del módulo: 16#ff00 - 16#ffff.

Código	Cadena	Explicación y posibles causas/soluciones
16#ff00	Error de solicitud de conexión: No existe instancia de conexión.	<p>El controlador intenta establecer una conexión con el módulo y ha recibido un error.</p> <p>Compruebe que el módulo físico sea del mismo tipo que (o compatible con) el creado en el software.</p> <p>Si el módulo es un 1756-DHRI0 de un chasis remoto (conectado mediante una red ControlNet) compruebe que la red haya sido programada con el software RSNetWorx.</p> <p>Incluso después de que la red haya sido programada con el software RSNetWorx for ControlNet, si está en línea y si el módulo 1756-DHRI0 se configura solo para una red DH+, puede producirse un fallo de módulo #ff00 (no existe instancia de conexión). El módulo se comunica correctamente aunque se visualice un estado de fallo en el cuadro de diálogo Propiedades del módulo (Module Properties). Pase por alto el mensaje de error y el estado de fallo y continúe.</p>
16#ff01	Error de solicitud de conexión: Ruta al módulo demasiado larga.	<p>El controlador intenta establecer una conexión con el módulo y ha recibido un error.</p> <p>Compruebe que la ruta de acceso al módulo tenga una longitud válida.</p>
16#ff04		<p>La instancia de asignación del controlador remoto intentó acceder a una conexión mientras estaba en un estado no válido.</p>
16#ff08	Error de solicitud de conexión: Ruta de acceso al módulo no válida.	<p>El controlador intenta establecer una conexión con el módulo y ha recibido un error.</p> <p>Compruebe que la ruta de acceso al módulo tenga una longitud válida.</p>
16#ff0b	Configuración de módulo rechazada: formato incorrecto	<p>Existen posibilidades:</p> <ul style="list-style-type: none"> <li>• la configuración del módulo no es válida.</li> <li>• el módulo en uso (es decir, el módulo físico) es diferente al módulo especificado en el árbol de configuración de E/S y por lo tanto está haciendo que la conexión o el servicio falle.</li> </ul> <p>El fallo puede producirse incluso cuando el módulo haya pasado la prueba de codificación electrónica. Esto puede producirse cuando se usan las opciones Deshabilitar codificación o Módulo compatible en la configuración del módulo, en lugar de la opción Coincidencia exacta.</p> <p>A pesar de pasar la prueba de codificación electrónica, el módulo con el que se está conectando no tiene las mismas características o configuración que el módulo especificado en el árbol de configuración de E/S y no admite la conexión o servicio deseados.</p> <p>Compruebe el módulo en uso y verifique que coincida exactamente con el módulo especificado en el árbol de configuración de E/S de la aplicación Logix Designer.</p>
16#ff0e	Error de solicitud de conexión: No hay conexiones al puente aceptadas.	<p>El controlador intenta establecer una conexión con el módulo y ha recibido un error.</p>

## Especificar mensajes CIP

Los tipos de mensaje de escritura y lectura de tabla de datos CIP transfieren datos entre los controladores LOGIX 5000.

Seleccione este comando	Si desea
Lectura de tabla de datos CIP (CIP Data Table Read)	<p>Leer datos desde otro controlador.</p> <p>Los tipos de Source y Destination deben coincidir.</p>

Escritura de tabla de datos CIP (CIP Data Table Write)	Escribir datos en otro controlador. Los tipos de Source y Destination deben coincidir.
---	---

### Reconfigurar un módulo E/S

Utilice el mensaje de reconfiguración de módulo para enviar nueva información de configuración a un módulo E/S.

Durante la reconfiguración:

- Los módulos de entrada continúan enviando datos de entrada al controlador.
- Los módulos de salida continúan controlando sus dispositivos de salida.

Un mensaje de reconfiguración de módulo necesita estas propiedades de configuración.

En esta propiedad	Seleccione
Tipo de mensaje (Message Type)	Reconfiguración de módulo (Module Reconfigure)

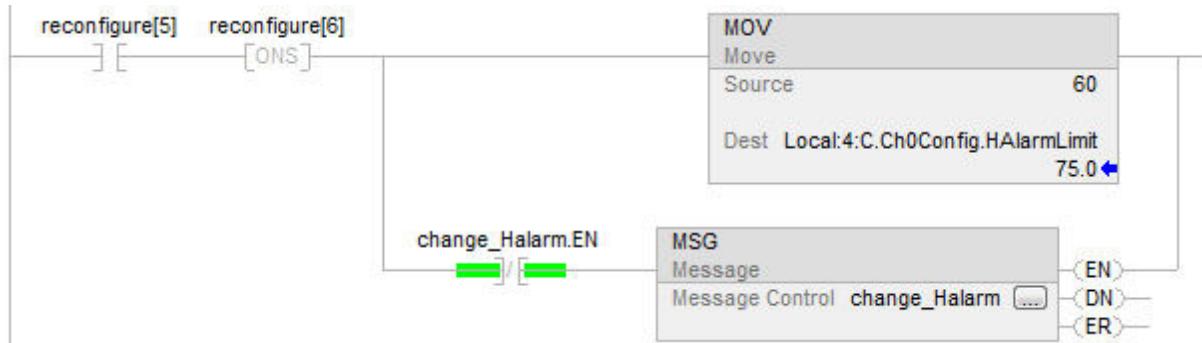
### Ejemplo

Siga estos pasos para reconfigurar un módulo E/S.

1. Cambie el miembro necesario de la etiqueta de configuración del módulo al valor nuevo.
2. Envíe un mensaje de Reconfiguración de módulo al módulo.  
Al establecer reconfigure[5], se establece la alarma alta a 60 para el módulo local en la ranura 4. A continuación, el mensaje de reconfiguración de módulo le envía al módulo el nuevo valor de alarma. La instrucción de un impulso evita que el reglón envíe múltiples mensajes al módulo mientras reconfigure[5] está activado.

**Cons** Es recomendable incluir siempre un XIO del bit MSG.EN  
**ejo:** como condición previa en serie del reglón MSG.

### Escalera de relés



### Texto estructurado

```

IF reconfigure[5] AND NOT reconfigure[6] THEN
  Local:4:C.Ch0Config.HAlarmLimit := 60;

IF NOT change_Halarm.EN THEN MSG(change_Halarm);

END_IF; END_IF;

reconfigure[6] := reconfigure[5];

```

### Especificar mensajes genéricos CIP

**Importante:** Los módulos ControlLogix tienen servicios que se pueden invocar usando una instrucción MSG y seleccionando el tipo de mensaje genérico CIP.

Si desea	En esta propiedad	Escriba o seleccione
Realizar una prueba de impulso en un módulo de salida digital	Tipo de mensaje (Message Type)	CIP genérico (CIP Generic)
	Tipo de servicio (Service Type)	Prueba de impulsos (Pulse Test)
	Origen (Source)	tag_name de tipo INT [5] Esta matriz contiene Descripción tag_name[0] Máscara de bit de puntos a probar (probar sólo un punto cada vez) tag_name[1] Reservado, dejar 0 tag_name[2] Ancho del impulso (cientos de , normalmente 20) tag_name[3] Retardo de cruce por cero para E/S ControlLogix (cientos de , normalmente 40)

		tag_name[4]	Retardo de verificación
	Destino (Destination)	En blanco	
Obtener valor de auditoría	Tipo de mensaje (Message Type)	CIP genérico (CIP Generic)	
	Tipo de servicio (Service Type)	Obtención valor de auditoría	
	Elemento de origen (Source Element)	No se puede cambiar este campo, en blanco	
	Longitud de origen (Source Length)	No se puede cambiar este campo, establecer a 0 bytes	
	Elemento de destino (Destination Element)	<p>Esta matriz contiene</p> <p>tag_name de tipo DINT[2] o LINT</p>	<p>Descripción</p> <p>Esta etiqueta contiene el Valor de auditoría para el controlador.</p> <p><b>Importante:</b> Rockwell Automation recomienda usar el tipo de datos DINT[2] para evitar limitaciones al trabajar con el tipo de datos LINT en los controladores Allen-Bradley®.</p>
Obtener eventos de controlador monitorizados para cambios	Tipo de mensaje (Message Type)	CIP genérico (CIP Generic)	
	Tipo de servicio (Service Type)	Cambios para detectar obtención (Changes to Detect Get)	
	Elemento de origen (Source Element)	No se puede cambiar este campo, en blanco	
	Longitud de origen (Source Length)	No se puede cambiar este campo, establecer a 0 bytes	
	Elemento de destino (Destination Element)	<p>Esta matriz contiene</p> <p>tag_name de tipo DINT[2] o LINT</p>	<p>Descripción</p> <p>Esta etiqueta representa una máscara de bit de los cambios monitorizados por el controlador.</p> <p><b>Importante:</b> Rockwell Automation recomienda usar el tipo de datos DINT[2] para evitar limitaciones al trabajar con tipo de datos LINT en los controladores Allen-Bradley.</p>
Establecer eventos de controlador monitorizados para cambios	Tipo de mensaje (Message Type)	CIP genérico (CIP Generic)	
	Tipo de servicio (Service Type)	Cambios para detectar establecimiento (Changes to Detect Set)	
	Elemento de origen (Source Element)	<p>Esta matriz contiene</p> <p>tag_name de tipo DINT[2] o LINT</p>	<p>Descripción</p> <p>Esta etiqueta representa una máscara de bit de los cambios monitorizados por el controlador.</p> <p><b>Importante:</b> Rockwell Automation recomienda usar el tipo de datos DINT[2] para evitar limitaciones al trabajar con tipo de datos LINT en los controladores Allen-Bradley.</p>
	Longitud de origen (Source Length)	No se puede cambiar este campo, establecer a 8 bytes	
	Elemento de destino (Destination Element)	No se puede cambiar este campo, en blanco	

Restablecer fusibles electrónicos en un módulo de salida digital	Tipo de mensaje (Message Type)	CIP genérico (CIP Generic)
	Tipo de servicio (Service Type)	Restablecer fusible electrónico (Reset Electronic Fuse)
	Origen (Source)	nombre de etiqueta de tipo DINT Esta etiqueta representa una máscara de bit de los puntos en los cuales se restablecen los fusibles.
	Destino (Destination)	Dejar en blanco
Restablecer diagnósticos enclavados en un módulo de entrada digital	Tipo de mensaje (Message Type)	CIP genérico (CIP Generic)
	Tipo de servicio (Service Type)	Restablecer diagnósticos enclavados (Reset Latched Diagnostics (I))
	Origen (Source)	tag_name de tipo DINT Esta etiqueta representa una máscara de bit de los puntos en los cuales se restablecen los diagnósticos.
Restablecer diagnósticos enclavados en un módulo de salida digital	Tipo de mensaje (Message Type)	CIP genérico (CIP Generic)
	Tipo de servicio (Service Type)	Restablecer diagnósticos enclavados (Reset Latched Diagnostics (O))
	Origen (Source)	tag_name de tipo DINT Esta etiqueta representa una máscara de bit de los puntos en los cuales se restablecen los diagnósticos.
Desenclavar la alarma de un módulo de entrada analógica	Tipo de mensaje (Message Type)	CIP genérico (CIP Generic)
	Tipo de servicio (Service Type)	Seleccione la alarma que desea desenclavar. <ul style="list-style-type: none"> <li>• Desenclavar todas las alarmas (I) (Unlatch All Alarms (I))</li> <li>• Desenclavar alarma alta analógica (I) (Unlatch Analog High Alarm (I))</li> <li>• Desenclavar alarma alta alta analógica (I) (Unlatch Analog High High Alarm (I))</li> <li>• Desenclavar alarma baja analógica (I) (Unlatch Analog Low Alarm (I))</li> <li>• Desenclavar alarma baja baja analógica (I) (Unlatch Analog Low Low Alarm (I))</li> <li>• Desenclavar alarma de régimen (I) (Unlatch Rate Alarm (I))</li> </ul>
	Instancia (Instance)	Canal de la alarma a desenclavar.
Desenclavar la alarma de un módulo de salida analógica	Tipo de mensaje (Message Type)	CIP genérico (CIP Generic)
	Tipo de servicio (Service Type)	Seleccione la alarma que desea desenclavar. <ul style="list-style-type: none"> <li>• Desenclavar todas las alarmas (O) (Unlatch All Alarms (O))</li> <li>• Desenclavar alarma alta (O) (Unlatch High Alarm (O))</li> <li>• Desenclavar alarma baja (O) (Unlatch Low Alarm (O))</li> <li>• Desenclavar alarma de rampa (O) (Unlatch Ramp Alarm (O))</li> </ul>
	Instancia (Instance)	Canal de la alarma a desenclavar.

**Obtener/establecer eventos del controlador monitorizados para los cambios en las definiciones de bits**

<b>Nombres de etiquetas</b>	<b>Tipo de datos</b>	<b>Definición de bit</b>
Obtener eventos de controlador monitorizados para cambios Establecer eventos de controlador monitorizados para cambios	DINT[0]	<p>Cada bit tiene un significado específico:</p> <ul style="list-style-type: none"> <li>0 Almacenar en medio extraíble a través de la aplicación Logix Designer</li> <li>1 Las ediciones en línea se han aceptado, probado o montado</li> <li>2 Transacción en línea de la importación parcial completada</li> <li>3 Forzados de SFC habilitados</li> <li>4 Forzados de SFC deshabilitados</li> <li>5 Forzados de SFC eliminados</li> <li>6 Forzados de SFC modificados</li> <li>7 Forzados de E/S habilitados</li> <li>8 Forzados de E/S deshabilitados</li> <li>9 Forzados de E/S eliminados</li> <li>10 Forzados de E/S cambiados</li> <li>11 Actualización de firmware desde origen desconectado</li> <li>12 Actualización de firmware a través de medio extraíble</li> <li>13 Cambio de modo a través de estación de trabajo</li> <li>14 Cambio de modo a través del cambio de modo</li> <li>15 Ha ocurrido un fallo mayor</li> <li>16 Se han eliminado los fallos mayores</li> <li>17 Los fallos mayores se han eliminado a través del cambio de modo</li> <li>18 Las propiedades de tarea se han modificado</li> <li>19 Las propiedades de programa se han modificado</li> <li>20 La fracción de tiempo del controlador se ha modificado</li> <li>21 El medio extraíble se ha eliminado</li> <li>22 Se ha introducido un medio extraíble</li> <li>23 Se ha creado una firma de seguridad</li> <li>24 Se ha eliminado una firma de seguridad</li> <li>25 Bloqueo de seguridad</li> <li>26 Desbloqueo de seguridad</li> <li>27 El valor de la etiqueta constante ha cambiado</li> <li>28 Los múltiples valores de la etiqueta constante han cambiado</li> <li>29 El atributo de la etiqueta constante se ha eliminado</li> <li>30 Etiqueta establecida como constante</li> <li>31 Se ha añadido una entrada personalizada en el registro</li> </ul>
	DINT[1]	<ul style="list-style-type: none"> <li>32 Cambio que afecta a la correlación</li> <li>33 Ayuda a proteger la firma en el establecimiento de atributos del modo Marcha</li> <li>34 Ayuda a proteger la firma en la eliminación de atributos del modo Marcha</li> <li>35...63 No utilizados</li> </ul>

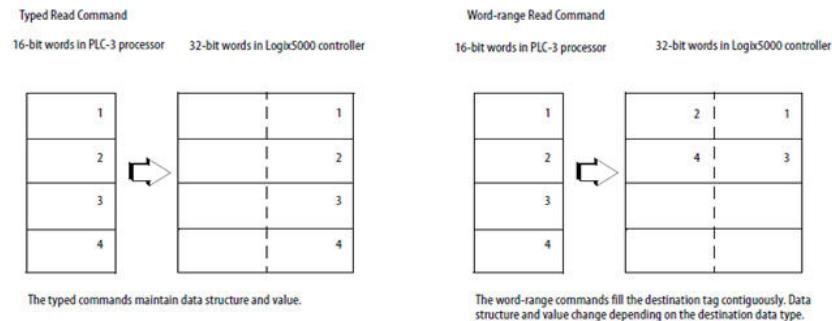
<b>Consejos:</b>	<ul style="list-style-type: none"> <li>• Al seleccionar el tipo de mensaje <b>Genérico CIP</b> (CIP Generic), se habilita la opción <b>Conexión grande</b> (Large Connection) en la pestaña <b>Comunicación</b> (Communication). Use las conexiones grandes de Genérico CIP (CIP Generic) cuando un mensaje sea mayor de 480 bytes. 500 bytes es típico, pero hay encabezados en la parte frontal del mensaje. Las conexiones grandes CIP son para mensajes de hasta 3980 bytes.</li> <li>• La casilla de verificación <b>Conexión grande</b> (Large Connection) solo se habilita cuando se marca la casilla <b>Conectado</b> (Connected) y se selecciona <b>Genérico CIP</b> (CIP Generic) como el tipo de mensaje en la pestaña <b>Configuración</b> (Configuration).</li> <li>• La opción <b>Conexión grande</b> (Large Connection) solo está disponible en la aplicación Logix Designer, versión 21.00.00 o posterior y en RSLogix 5000, versión 20.00.00 o posterior.</li> </ul>
------------------	---

## Especificar mensajes PLC-3

Los tipos de mensajes PLC-3 están diseñados para procesadores PLC-3.

Seleccione este comando:	Para:
Lectura en pantalla de PLC3 (PLC3 Typed Read)	<p>Leer entero o datos de tipo REAL.</p> <p>Para enteros, este comando lee enteros de 16 bits del procesador PLC-3 y los almacena en matrices de datos SINT, INT o DINT en el controlador LOGIX 5000 y mantiene la integridad de los datos.</p> <p>Este comando también lee datos de punto flotante desde el PLC-3 y los almacena en una etiqueta de tipo de datos REAL en el controlador LOGIX 5000.</p>
Escritura en pantalla de PLC3 (PLC3 Typed Write)	<p>Escribir entero o datos de tipo REAL.</p> <p>Este comando escribe datos SINT o INT al archivo entero PLC-3 y mantiene la integridad de los datos. Puede escribir datos DINT siempre que se ajuste a un tipo de datos INT (<math>-32.768 \geq \text{datos} \leq 32.767</math>).</p> <p>Este comando también escribe datos de tipo REAL desde el controlador LOGIX 5000 a un archivo de punto flotante PLC-3.</p>
Lectura de rango de palabras PLC3 (PLC3 Word Range Read)	<p>Leer un rango contiguo de palabras de 16 bits en la memoria del PLC-3 independientemente del tipo de datos.</p> <p>Este comando comienza en la dirección especificada como elemento de Source y lee secuencialmente el número de palabras de 16 bits solicitadas.</p> <p>Los datos del elemento de Source se almacenan, comenzando en la dirección especificada como etiqueta de Destination.</p>
Escritura de rango de palabras PLC3 (PLC3 Word Range Write)	<p>Escribir un rango contiguo de palabras de 16 bits de la memoria del LOGIX 5000, independientemente del tipo de datos, a la memoria del PLC-3.</p> <p>Este comando comienza en la dirección especificada como etiqueta de Source y lee secuencialmente el número de palabras de 16 bits solicitadas.</p> <p>Los datos de la etiqueta de Source se almacenan, comenzando en la dirección especificada como elemento de Destination, en el procesador PLC-3.</p>

Los siguientes diagramas muestran cómo difieren los comandos de pantalla y de rango de palabra. El ejemplo utiliza comandos de lectura de un procesador PLC-3 a un controlador LOGIX 5000.



## Especificar mensajes PLC-5

Se utiliza los tipos de mensaje PLC-5 para comunicarse con los controladores PLC-5.

Seleccione este comando:	Para:
Lectura en pantalla de PLC-5 (PLC-5 Typed Read)	Leer los datos de tipo entero de 16 bits, punto flotante o cadena y mantener la integridad de los datos.
Escritura en pantalla de PLC-5 (PLC-5 Typed Write)	Escribir los datos de tipo entero de 16 bits, punto flotante o cadena y mantener la integridad de los datos.
Lectura de rango de palabras PLC-5 (PLC-5 Word Range Read)	Leer un rango contiguo de palabras de 16 bits en la memoria del PLC-5 independientemente del tipo de datos. Este comando comienza en la dirección especificada como elemento de Source y lee secuencialmente el número de palabras de 16 bits solicitadas. Los datos del elemento de Source se almacenan, comenzando en la dirección especificada como etiqueta de Destination.
Escritura de rango de palabras PLC-5 (PLC-5 Word Range Write)	Escribir un rango continuo de palabras de 16 bits de la memoria del LOGIX 5000, independientemente del tipo de datos, a la memoria del PLC-5. Este comando comienza en la dirección especificada como etiqueta de Source y lee secuencialmente el número de palabras de 16 bits solicitadas. Los datos de la etiqueta de Source se almacenan, comenzando en la dirección especificada como elemento de Destination, en el procesador PLC-5.

### Tipo de datos para los mensajes de lectura y escritura en pantalla para PLC-5.

La siguiente tabla muestra los tipos de datos que se utilizarán con los mensajes de lectura y escritura en pantalla para PLC-5.

Para este tipo de datos de PLC-5:	Usa este tipo de datos LOGIX 5000:
B	INT
F	REAL
N	INT DINT (solo escriba valores DINT en un controlador PLC-5 si el valor es $\geq -32.768$ y $\leq 32.767$ ).
S	INT
ST	STRING

Los comandos de lectura y escritura en pantalla también funcionan con los procesadores SLC 5/03 (OS303 y superior), SLC 5/04 (OS402 y superior) y SLC 5/05.

## Especificar mensajes PLC-2

Los tipos de mensaje PLC-2 están designados por los procesadores PLC-2.

Seleccione este comando:	Para:
Lectura no protegida PLC2 (Unprotected Read PLC2)	Leer palabras de 16 bits de cualquier zona de la tabla de datos de PLC-2 o de un archivo compatible de otro procesador de PLC-2.
Escritura no protegida PLC2 (Unprotected Write PLC2)	Escribir palabras de 16 bits para cualquier zona de la tabla de datos de PLC-2 o para un archivo compatible de otro procesador de PLC-2.

La transferencia de mensaje usa palabras de 16 bits, así que asegúrese de que la etiqueta LOGIX 5000 almacena correctamente los datos transferidos, como en una matriz INT.

# Comparar instrucciones

## Comparar instrucciones

Las instrucciones de comparación permiten comparar valores usando expresiones o instrucciones de comparación específicas.

### Instrucciones disponibles

#### Diagrama de escalera

<a href="#">CMP</a>	<a href="#">EQU</a>	<a href="#">GEQ</a>	<a href="#">GRT</a>	<a href="#">LEQ</a>	<a href="#">LES</a>	<a href="#">LIM</a>	<a href="#">MEQ</a>	<a href="#">NEQ</a>
---------------------	---------------------	---------------------	---------------------	---------------------	---------------------	---------------------	---------------------	---------------------

#### Bloque de funciones

<a href="#">EQU</a>	<a href="#">GEQ</a>	<a href="#">GRT</a>	<a href="#">LEQ</a>	<a href="#">LES</a>	<a href="#">LIM</a>	<a href="#">MEQ</a>	<a href="#">NEQ</a>
---------------------	---------------------	---------------------	---------------------	---------------------	---------------------	---------------------	---------------------

#### Texto estructurado

No disponible

Si desea:	Utilice esta instrucción:
comparar valores basándose en una expresión	CMP
comprobar si dos valores son iguales	EQU
comprobar si un valor es mayor o igual a otro	GEQ
comprobar si un valor es mayor que otro	GRT
comprobar si un valor es menor o igual a otro	LEQ
comprobar si un valor es menor que otro	LES
comprobar si un valor está entre otros dos valores	LIM
pasar dos valores a través de una máscara y comprueba si son iguales	MEQ
comprobar si un valor no es igual a otro	NEQ

Se puede comparar valores con tipos de datos diferentes, como un entero y un número de punto flotante.

Los tipos de datos en negrita son los óptimos. Una instrucción se ejecutará a la máxima velocidad y usando la menor cantidad posible de memoria si todos los

parámetros emplean el mismo tipo de datos óptimo, normalmente DINT o REAL.

### Consulte también

[Instrucciones de cálculo/matemáticas](#) en la página 425

## Comparar (CMP)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

Se define la expresión CMP usando operadores, etiquetas y valores inmediatos. Se usa paréntesis ( ) para definir secciones en las expresiones más complejas.

La ventaja de la instrucción CMP es que permiten integrar expresiones complejas en una sola instrucción.

Al evaluar la expresión, todos los operandos del tipo no REAL se convertirán a REAL antes de realizar los cálculos si alguna de las siguientes condiciones es verdadera.

- Cualquier operando en la expresión es REAL.
- La expresión contiene SIN, COS, TAN, ASN, ACS, ATN, LN, LOG, DEG o RAD.

Hay reglas para operadores admisibles en aplicaciones de seguridad. Consulte *Operadores válidos*.

### Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

## Operando

Estos son operandos de la instrucción CMP.

**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
- Los miembros de un operando de estructura se sobrescriben.
- A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.

Existen reglas de conversión de datos para utilizar diversos tipos de datos numéricos en una instrucción. Ver *Conversión de datos*.

## Diagrama de escalera

El siguiente es el operando del Diagrama de escalera.

Operando	Tipo de datos	Formato	Descripción
Expression	SINT INT DINT REAL Tipo de cadena	inmediato etiqueta	Una expresión que consiste en etiquetas y/o valores inmediatos separados por operadores

## Formato de expresiones

Se deben proporcionar uno o dos operandos (etiquetas o valores inmediatos) para cada uno de los operadores usados en una expresión. Se usa la siguiente tabla para dar formato a los operadores y operandos dentro de una expresión.

Para operadores que operan en:	Se usa este formato:	Ejemplo
Un operando	operador(operando)	ABS(tag)
Dos operandos	operand_a operador operand_b	tag_b + 5 tag_c AND tag_d (tag_e**2) MOD (tag_f / tag_g)

## Determine el orden de operación

La instrucción realiza las operaciones dentro de una expresión en un orden estipulado, no necesariamente en el orden en que aparecen. Se puede especificar el orden de la operación agrupando los términos en paréntesis, lo que obligará a la instrucción a realizar las operaciones dentro los paréntesis antes que las suyas.

Las operaciones del mismo orden se realizarán de izquierda a derecha.

Orden	Operación
1	( )
2	ABS, ACS, ASN, ATN, COS, DEG, FRD, LN, LOG, RAD, SIN, SQR, TAN, TOD, TRN
3	**
4	- (negar), NOT
5	*, /, MOD
6	- (restar), +
7	AND
8	XOR
9	OR
10	<, <=, >, >=, =, <>

### Uso de cadenas en una expresión

Para usar cadenas de caracteres ASCII en una expresión, siga estas pautas:

- Una expresión puede comparar dos etiquetas de cadenas.
- No se pueden introducir caracteres ASCII directamente en la expresión.
- Los siguientes operadores están permitidos:

Operador	Descripción
=	Igual
<	Menor que
<=	Menor o igual que
>	Mayor que
>=	Mayor o igual que
<>	No igual

- Las cadenas son iguales si sus caracteres coinciden.
- Los caracteres ASCII distinguen entre mayúsculas y minúsculas. La mayúscula A (\$41) no es equivalente a la minúscula a (\$61).
- Lo que determina si una cadena es mayor o menor que otra son los valores hexadecimales de los caracteres.

- Cuando las dos cadenas se ordenan como en un directorio telefónico, el orden de las cadenas determina cuál es superior.

ASCII Characters	Hex Codes
1ab	\$31\$61\$62
1b	\$31\$62
A	\$41
AB	\$41\$42
B	\$42
a	\$61
ab	\$61\$62

### Afectar a las marcas de estado matemático

Controladores	Afectar a las marcas de estado matemático
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	No
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	La instrucción CMP afecta a las marcas de estado matemático si la expresión contiene un operador (como +, -, *, /) que afecte a las marcas de estado matemático.

Consulte *Marcas de estado matemático*.

### Fallos mayores/menores

No es específico para esta instrucción. Consulte *Índice a través de matrices* para ver si hay fallos de indexación de matrices.

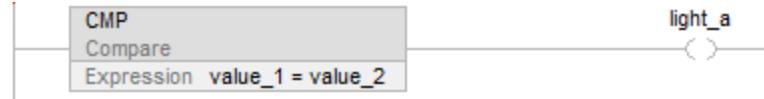
### Ejecución

#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A.
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón
La condición de entrada de reglón es verdadera	Establecer la condición de salida de reglón a condición de entrada de reglón si la expresión evalúa en falso La condición de salida de reglón se borra a falsa.
Post-escaneado	N/A.

## Ejemplo

### Diagrama de escalera



Si value\_1 es igual a value\_2, light\_a se establece en verdadero. Si value\_1 no es igual a value\_2, light\_a se establece en falso.

### Consulte también

[Comparar instrucciones](#) en la página 369

[Operadores válidos](#) en la página 422

[Índice a través de matrices](#) en la página 936

[Marcas de estado matemático](#) en la página 923

[Conversiones de datos](#) en la página 927

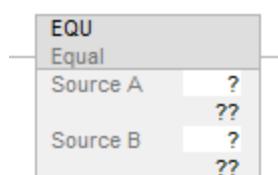
## Igual a (EQU)

Esta instrucción es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

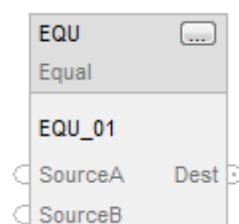
La instrucción EQU prueba si Source A es igual a Source B.

### Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones



## Texto estructurado

Esta instrucción no está disponible en texto estructurado.

**Cons** Se utiliza el operador '=' con una expresión para obtener el mismo resultado. Consulte *Structured Text Syntax* para obtener más información sobre la sintaxis de las expresiones y las asignaciones en el texto estructurado.

## Operандos

Existen reglas de conversión de datos para utilizar diversos tipos de datos numéricos en una instrucción. Consulte *Conversión de datos*.

## Diagrama de escalera

### Comparación numérica

Operando	Tipo de datos	Formato	Descripción
Source A	SINT INT DINT REAL	inmediato etiqueta	Valor que se prueba con respecto a Source B
Source B	SINT INT DINT REAL	inmediato etiqueta	Valor que se prueba con respecto a Source A

### Comparación de cadenas

Operando	Tipo de datos	Formato	Descripción
Source A	Tipo de cadena	valor literal inmediato etiqueta	Cadena que se prueba con respecto a Source B
Source B	Tipo de cadena	valor literal inmediato etiqueta	Cadena que se prueba con respecto a Source A

## Bloque de funciones

Operando	Tipo de datos	Formato	Descripción
EQU	FBD_COMPARE	etiqueta	Estructura de EQU

### Estructura de FBD\_COMPARE

Miembros de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si el valor es falso, la instrucción no se ejecuta y las salidas no se actualizan. El valor predeterminado es verdadero.
SourceA	REAL	Valor que se prueba con respecto a SourceB
SourceB	REAL	Valor que se prueba con respecto a SourceA

Miembros de salida	Tipo de datos	Descripción
EnableOut	BOOL	Indica si la instrucción está habilitada.
Dest	BOOL	Se establece en verdadero cuando SourceA es igual a SourceB. Se borra y se establece en falso cuando SourceA no es igual a SourceB.

### Descripción

Se utiliza la instrucción EQU o el operador '=' para comparar dos números o dos cadenas de caracteres ASCII.

### Afectar a las marcas de estado matemático

No

### Fallos mayores/menores

Consulte el *Diagrama de flujo de comparación de cadenas EQU* para los fallos.

Consulte *Índice a través de matrices* para ver si hay fallos de indexación de matrices.

### Ejecución

#### Diagrama de escalera

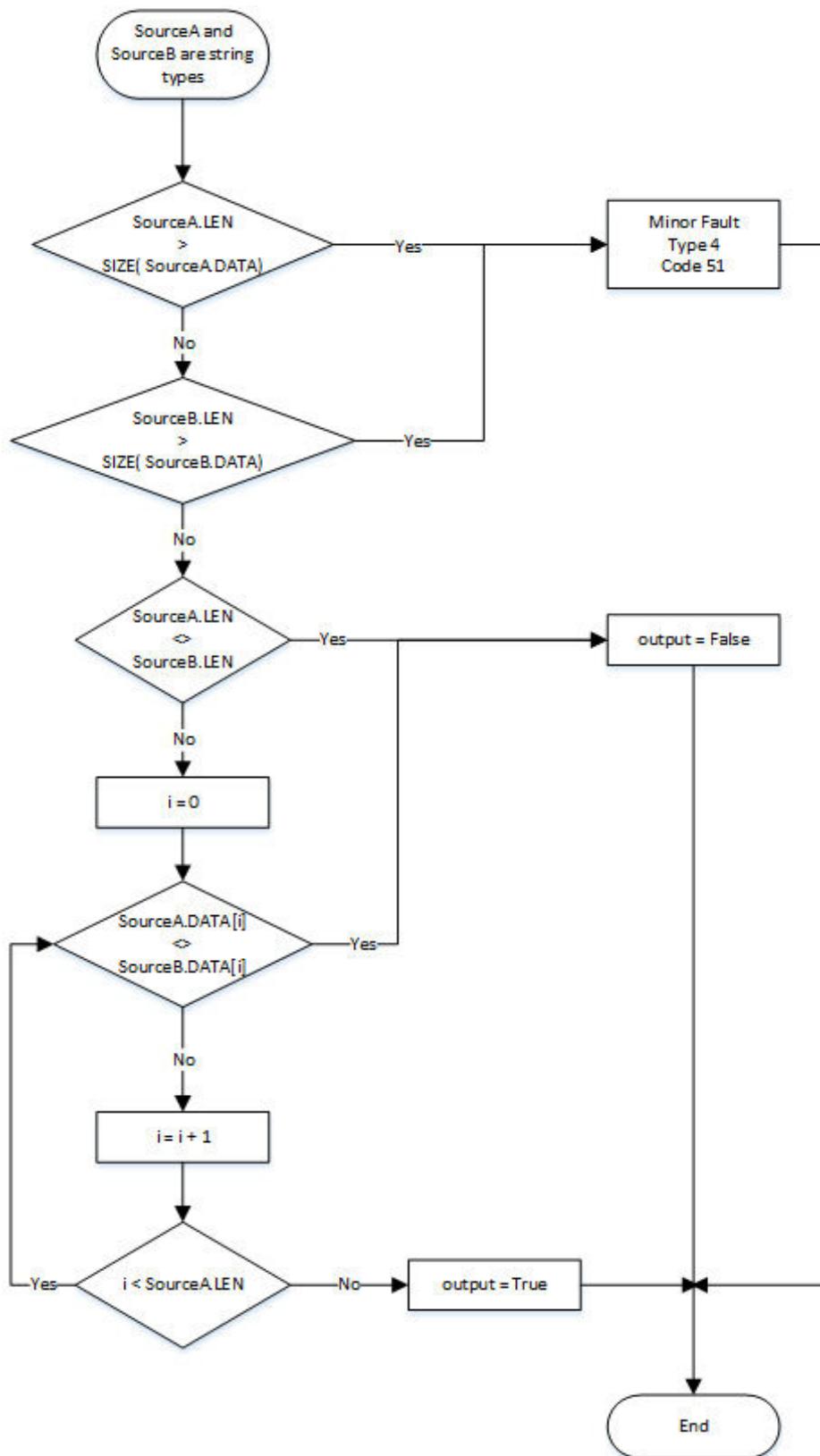
Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón

La condición de entrada de reglón es verdadera	<p><b>Comparación numérica:</b></p> <p>Si Source A y Source B no son NAN, y Source A es igual a Source B.</p> <p>Establecer la condición de salida de reglón en verdadera.</p> <p><b>else</b></p> <p>Borrar la condición de salida de reglón a falsa.</p>
	<p><b>Comparación de cadena:</b></p> <p>Consultar el diagrama de flujo de comparación de cadenas EQU.</p> <p>Si la salida es falsa</p> <p>Se borra la condición de salida de reglón a falsa de lo contrario</p> <p>Se establece la condición de salida de reglón en verdadera.</p>
Post-escaneado	N/A

### Bloque de funciones

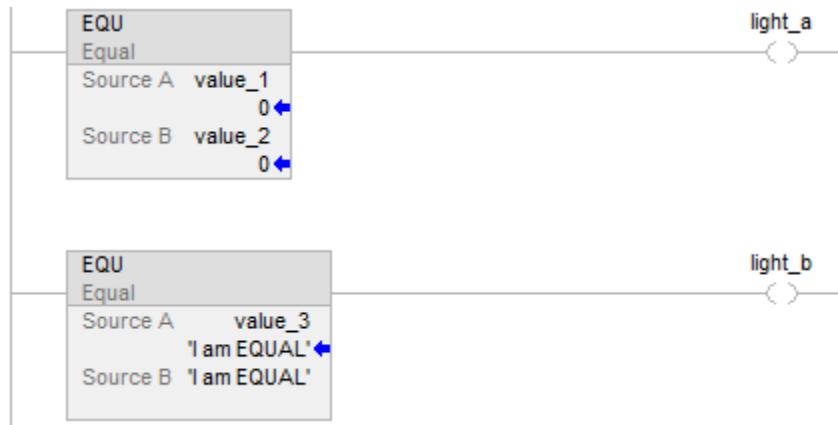
Condición/estado	Acción realizada
Pre-escaneado	N/A
EnableIn es falso	Establecer EnableOut a EnableIn
EnableIn es verdadero	<p><b>Comparación numérica:</b></p> <p>Establecer EnableOut a EnableIn</p> <p>Si SourceA y SourceB no son NAN y el SourceA es igual al SourceB .</p> <p>Establecer Destino a verdadero.</p> <p><b>else</b></p> <p>Borrar Destino a falso.</p>
Primera ejecución de instrucción	N/A
Primer escaneado de instrucción	N/A
Post-escaneado	N/A

Diagrama de flujo de comparación de cadenas EQU

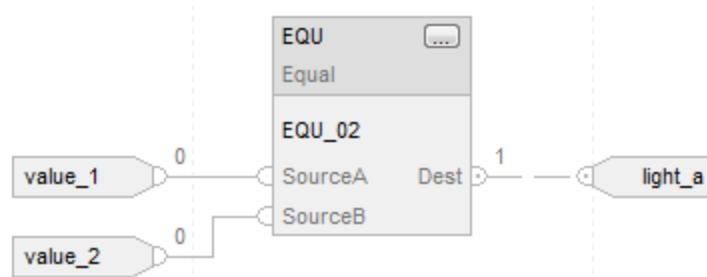


## Ejemplos

### Diagrama de escalera



### Bloque de funciones



### Texto estructurado

```

if value_1 = value_2 then
    light_a := 1;
else
    light_a := 0;
end_if;

if value_3 = 'I am EQUAL' then
    light_b := 1;
else
    light_b := 0;
end_if;

```

### Consulte también

[Sintaxis de texto estructurado](#) en la página 955

[Conversiones de datos](#) en la página 927

[Índice a través de matrices](#) en la página 936

[Comparar instrucciones](#) en la página 369

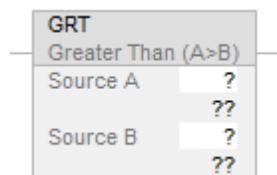
## Mayor que (GRT)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

La instrucción GRT comprueba si el Source A es mayor que el Source B.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones



#### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

**Cons** Se utiliza el operador '>' con una expresión para obtener el mismo resultado. Consulte *Structured Text Syntax* para obtener más información sobre la sintaxis de las expresiones y las asignaciones en el texto estructurado.

## Operandos

Existen reglas de conversión de datos para utilizar diversos tipos de datos numéricos en una instrucción. Consulte *Conversión de datos*.

### Diagrama de escalera

#### Comparación numérica

Operando	Tipo de datos	Formato	Descripción
Source A	SINT INT DINT REAL	inmediato etiqueta	Valor que se prueba con respecto a Source B
Source B	SINT INT DINT REAL	inmediato etiqueta	Valor que se prueba con respecto a Source A

#### Comparación de cadenas

**Cons** Los literales de cadena inmediatos solo se aplican a  
**ejo:** Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580..

Operando	Tipo de datos	Formato	Descripción
Source A	Tipo de cadena	valor literal inmediato etiqueta	Cadena que se prueba con respecto a Source B
Source B	Tipo de cadena	valor literal inmediato etiqueta	Cadena que se prueba con respecto a Source A

**Bloque de funciones**

Operando	Tipo de datos	Formato	Descripción
GRT	FBD_COMPARE	etiqueta	Estructura de GRT

**Estructura de FBD\_COMPARE**

Miembros de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si el valor es falso, la instrucción no se ejecuta y las salidas no se actualizan. El valor predeterminado es verdadero.
SourceA	REAL	Valor que se prueba con respecto a SourceB
SourceB	REAL	Valor que se prueba con respecto a SourceA

Miembros de salida	Tipo de datos	Descripción
EnableOut	BOOL	Indica si la instrucción está habilitada.
Dest	BOOL	Se establece en verdadero cuando el SourceA es mayor que el SourceB. Se borra y se establece en falso cuando el SourceA no es mayor que el SourceB.

**Descripción**

Se utiliza la instrucción GRT o el operador '>' para comparar dos números o dos cadenas de caracteres ASCII.

**Afectar a las marcas de estado matemático**

No

**Fallos mayores/menores**

Consulte el Diagrama de flujo de comparación de cadenas GRT para los fallos.

Consulte Índice a través de matrices para ver si hay fallos de indexación de matrices.

## Ejecución

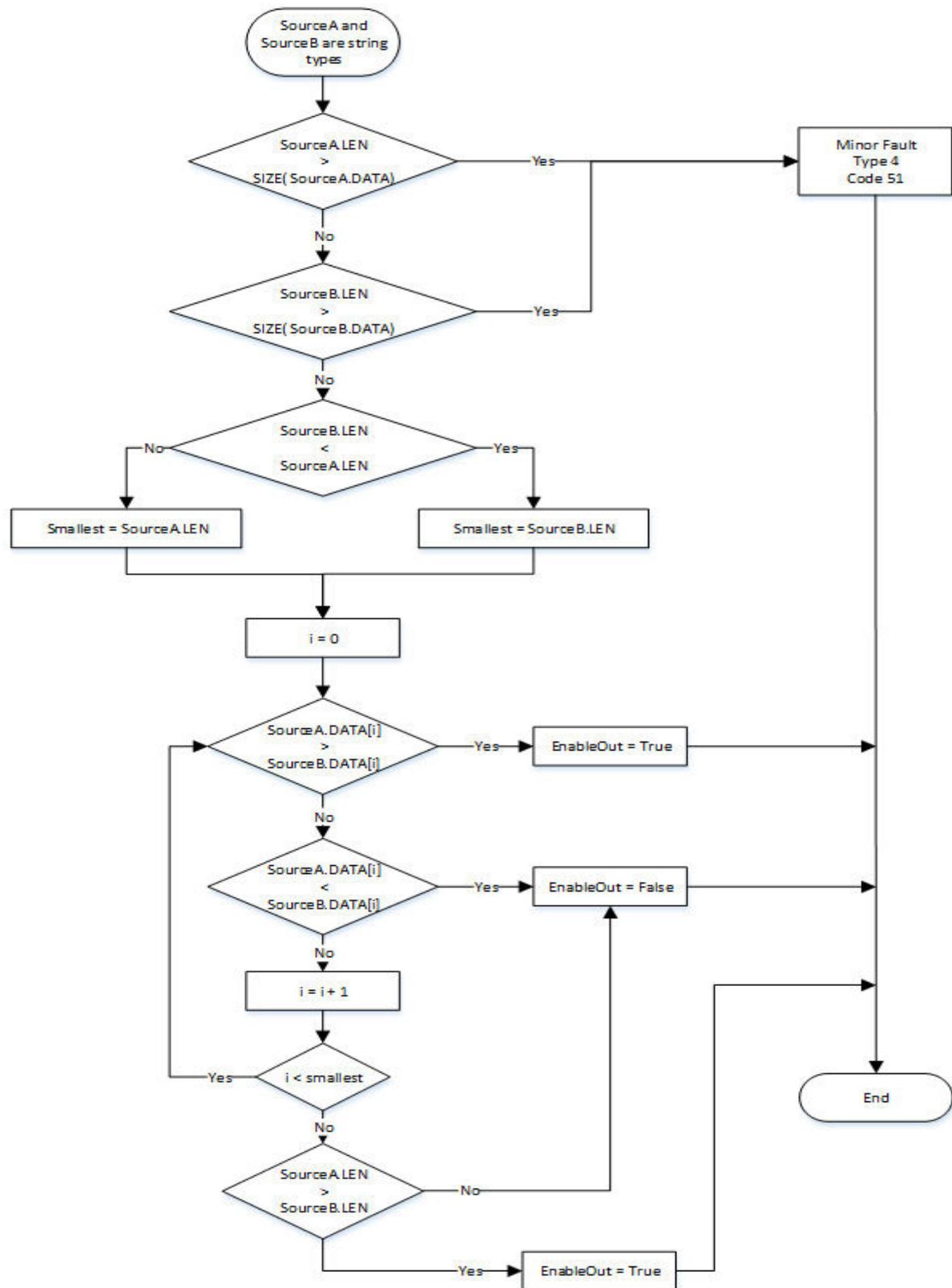
### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón
La condición de entrada de reglón es verdadera	<p><b>Comparación numérica:</b>            Si el Source A y el Source B no son NAN y el Source A es mayor que el Source B.            Establecer la condición de salida de reglón en verdadera.</p> <p><b>else</b>            Borrar la condición de salida de reglón a falsa.</p> <p><b>Comparación de cadena:</b>            Consultar el <i>Diagrama de flujo de comparación de cadenas GRT</i>            Si la salida es falsa            Borrar la condición de salida de reglón a falsa            de lo contrario            Se establece la condición de salida de reglón en verdadera.</p>
Post-escaneado	N/A

### Bloque de funciones

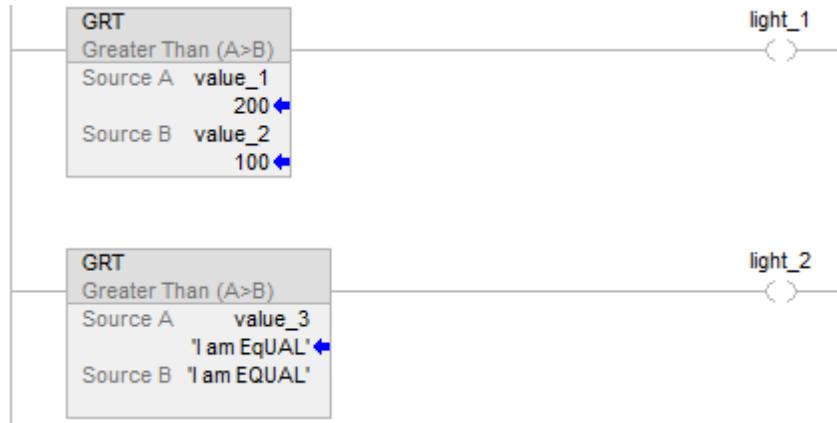
Condición/estado	Acción realizada
Pre-escaneado	N/A
EnableIn es falso	Establecer EnableOut a EnableIn
EnableIn es verdadero	<p><b>Comparación numérica:</b>            Establecer EnableOut a EnableIn            Si SourceA y SourceB no son NAN y el SourceA es mayor que el SourceB.            Establecer Destino a verdadero.</p> <p><b>else</b>            Borrar Destino a falso.</p>
Primera ejecución de instrucción	N/A
Primer escaneado de instrucción	N/A
Post-escaneado	N/A

Diagrama de flujo de comparación de cadenas GRT

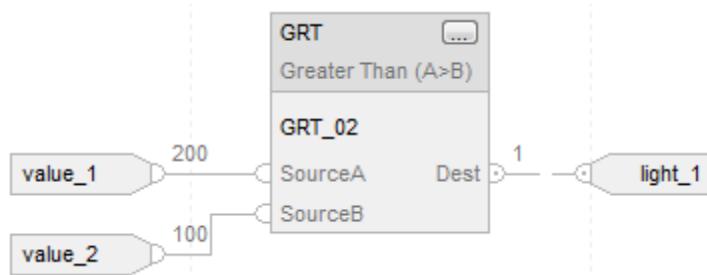


## Ejemplo

### Diagrama de escalera



### Bloque de funciones



### Texto estructurado

```

if value_1 > value_2 then
    light_1 := 1;
else
    light_1 := 0;
end_if;

if value_3 > 'I am EQUAL' then
    light_2 := 1;
else
    light_2 := 0;
end_if;
    
```

**Consulte también**

[Sintaxis de texto estructurado](#) en la página 955

[Conversiones de datos](#) en la página 927

[Índice a través de matrices](#) en la página 936

[Comparar instrucciones](#) en la página 369

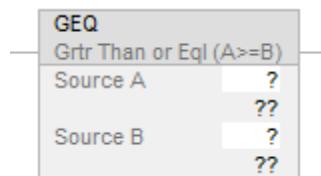
## Mayor o igual que (GEQ)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

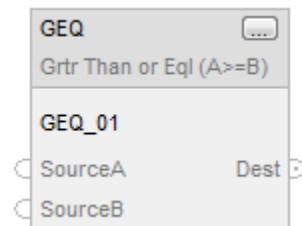
La instrucción GEQ comprueba si el Source A es igual o mayor al Source B.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones



#### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

**Cons** Se utiliza el operador '>=' con una expresión para  
**ejo:** obtener el mismo resultado. Consulte Sintaxis de texto  
 estructurado para obtener más información sobre la  
 sintaxis de las expresiones y las asignaciones en el texto  
 estructurado.

### Operandos

Existen reglas de conversión de datos para utilizar diversos tipos de datos numéricos en una instrucción. Ver Conversión de Datos.

#### Diagrama de escalera

#### Comparación numérica

Operando	Tipo de datos	Formato	Descripción
Source A	SINT INT DINT REAL	inmediato etiqueta	Valor que se prueba con respecto a Source B
Source B	SINT INT DINT REAL	inmediato etiqueta	Valor que se prueba con respecto a Source A

#### Comparación de cadenas

**Cons** Los literales de cadena inmediatos solo se aplican  
**ejo:** a Controladores CompactLogix 5380,  
 CompactLogix 5480, ControlLogix 5580, Compact  
 GuardLogix 5380 y GuardLogix 5580..

Operando	Tipo de datos	Formato	Descripción
Source A	Tipo de cadena	valor literal inmediato etiqueta	Cadena que se prueba con respecto a Source B
Source B	Tipo de cadena	valor literal inmediato etiqueta	Cadena que se prueba con respecto a Source A

#### Bloque de funciones

Operando	Tipo de datos	Formato	Descripción
GEQ	FBD_COMPARE	etiqueta	Estructura de GEQ

**Estructura de FBD\_COMPARE**

Miembros de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si el valor es falso, la instrucción no se ejecuta y las salidas no se actualizan. El valor predeterminado es verdadero.
SourceA	REAL	Valor que se prueba con respecto a SourceB
SourceB	REAL	Valor que se prueba con respecto a SourceA

Miembros de salida	Tipo de datos	Descripción
EnableOut	BOOL	Indica si la instrucción está habilitada.
Dest	BOOL	Se establece en verdadero cuando el SourceA es mayor o igual al SourceB. Se borra y pasa a falso cuando el SourceA es menor al SourceB.

**Descripción**

Se utiliza la instrucción GEQ o el operador ' $\geq$ ' para comprobar si el Source A es igual o mayor al Source B.

**Afectar a las marcas de estado matemático**

No

**Fallos mayores/menores**

Consulte el Diagrama de flujo de comparación de cadenas GEQ para los fallos.

Consulte Índice a través de matrices para ver si hay fallos de indexación de matrices.

**Ejecución****Diagrama de escalera**

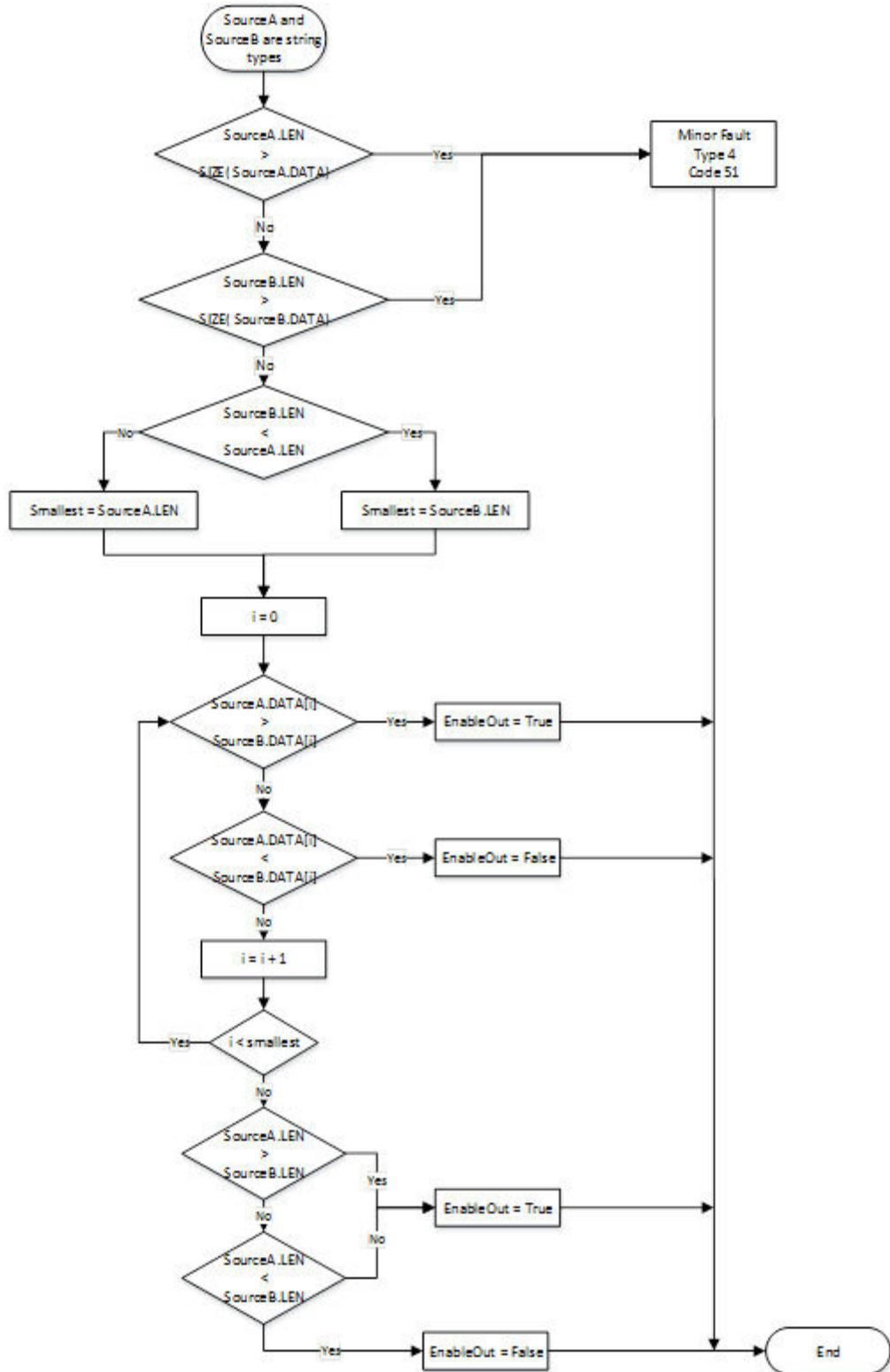
Condición/estado	Acción realizada
Pre-escaneado	N/A

La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón
La condición de entrada de reglón es verdadera	<p><b>Comparación numérica:</b>            Si el Source A y el Source B no son NAN y el Source A es mayor o igual al Source B.            Establecer la condición de salida de reglón en verdadera.</p> <p><b>else</b>            Borrar la condición de salida de reglón a falsa.</p>
	<p><b>Comparación de cadena:</b>            Consultar el Diagrama de flujo de comparación de cadenas GEQ.            Si la salida es falsa            Se borra la condición de salida de reglón a falsa            de lo contrario            Se establece la condición de salida de reglón en verdadera.</p>
Post-escaneado	N/A

### Bloque de funciones

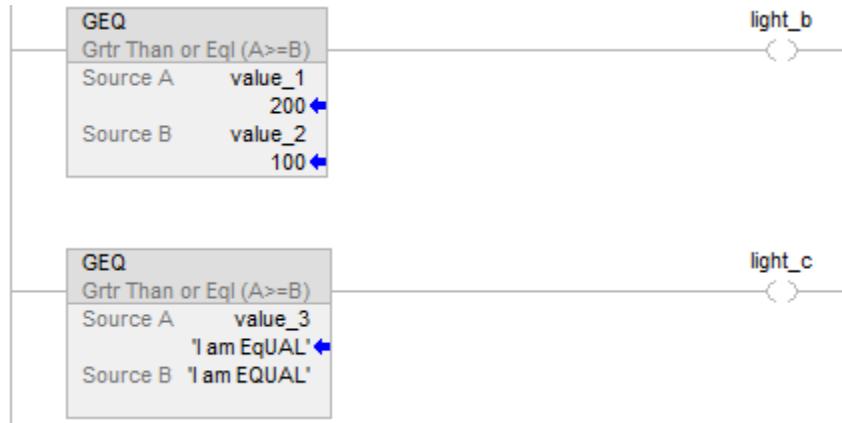
Condición/estado	Acción realizada
Pre-escaneado	N/A
EnableIn es falso	Establecer EnableOut a EnableIn
EnableIn es verdadero	<p><b>Comparación numérica:</b>            Establecer EnableOut a EnableIn            Si SourceA y SourceB no son NAN y el SourceA es igual o mayor al SourceB.            Establecer Destino a verdadero.</p> <p><b>else</b>            Borrar Destino a falso.</p>
Primera ejecución de instrucción	N/A
Primer escaneado de instrucción	N/A
Post-escaneado	N/A

Diagrama de flujo de comparación de cadenas GEQ

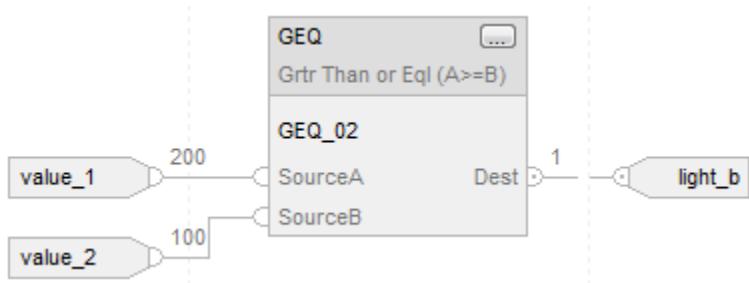


## Ejemplo

### Diagrama de escalera



### Bloque de funciones



### Texto estructurado

```

if value_1 >= value_2 then
    light_b := 1;
else
    light_b := 0;
end_if;

if value_3 >= 'I am EQUAL' then
    light_c := 1;
else
    light_c := 0;
end_if;
    
```

### Consulte también

[Sintaxis de texto estructurado](#) en la página 955

[Conversiones de datos](#) en la página 927

[Índice a través de matrices](#) en la página 936

[Comparar instrucciones](#) en la página 369

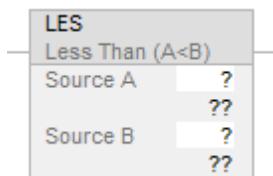
## Menor que (LES)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

La instrucción LES comprueba si el Source A es menor que el Source B.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones



#### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

**Cons** Se utiliza el operador '<' con una expresión para obtener el mismo resultado. Consulte *Structured Text Syntax* para obtener más información sobre la sintaxis de las expresiones y las asignaciones en el texto estructurado.

## Operandos

Existen reglas de conversión de datos para utilizar diversos tipos de datos numéricos en una instrucción. Consulte *Conversión de datos*.

### Diagrama de escalera

#### Comparación numérica

Operando	Tipo de datos	Formato	Descripción
Source A	SINT INT DINT REAL	inmediato etiqueta	Valor que se prueba con respecto a Source B
Source B	SINT INT DINT REAL	inmediato etiqueta	Valor que se prueba con respecto a Source A

#### Comparación de cadenas

**Cons** Los literales de cadena inmediatos solo se aplican a  
**ejo:** Controladores CompactLogix 5380, CompactLogix 5480,  
 ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580..

Operando	Tipo de datos	Formato	Descripción
Source A	Tipo de cadena	valor literal inmediato etiqueta	Cadena que se prueba con respecto a Source B
Source B	Tipo de cadena	valor literal inmediato etiqueta	Cadena que se prueba con respecto a Source A

#### Bloque de funciones

Operando	Tipo de datos	Formato	Descripción
LES	FBD_COMPARE	etiqueta	Estructura de LES

### Estructura de FBD\_COMPARE

Miembros de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si el valor es falso, la instrucción no se ejecuta y las salidas no se actualizan. El valor predeterminado es verdadero.
SourceA	REAL	Valor que se prueba con respecto a SourceB
SourceB	REAL	Valor que se prueba con respecto a SourceA

Miembros de salida	Tipo de datos	Descripción
EnableOut	BOOL	Indica si la instrucción está habilitada.
Dest	BOOL	Se establece en verdadero cuando el SourceA es menor que el SourceB. Se borra y se establece en falso cuando el SourceA no es menor que el SourceB.

### Descripción

Se utiliza la instrucción LES o el operador '<' para comparar dos números o dos cadenas de caracteres ASCII.

### Afectar a las marcas de estado matemático

No

### Fallos mayores/menores

Consulte el *Diagrama de flujo de comparación de cadenas LES* para los fallos.

Consulte *Índice a través de matrices* para ver si hay fallos de indexación de matrices.

### Ejecución

#### Diagrama de escalera

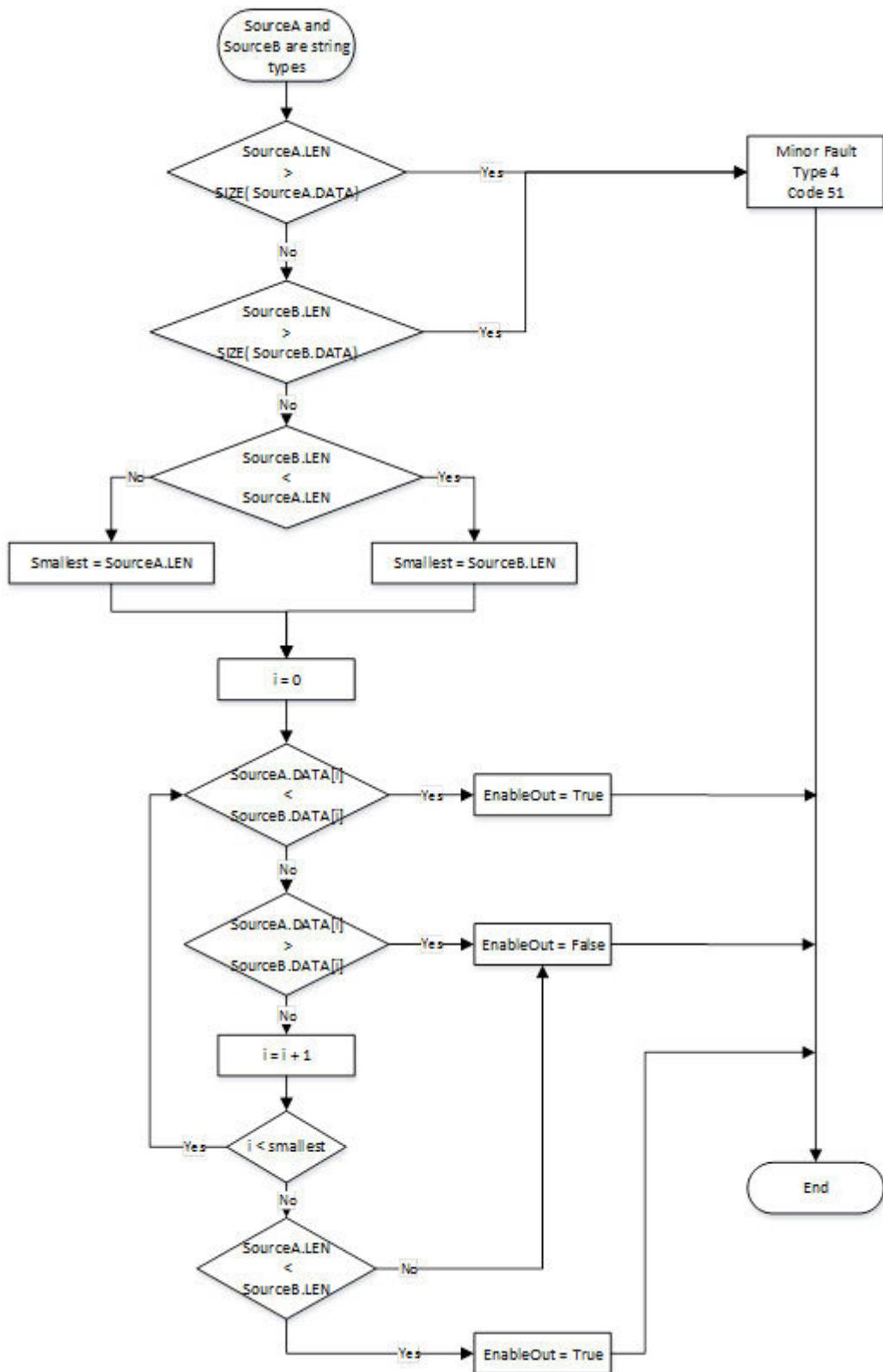
Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón

La condición de entrada de reglón es verdadera	<p><b>Comparación numérica:</b></p> <p>Si el Source A y el Source B no son NAN y el Source A es menor que el Source B.</p> <p>Establecer la condición de salida de reglón en verdadera.</p> <p><b>else</b></p> <p>Borrar la condición de salida de reglón a falsa.</p> <p><b>Comparación de cadena:</b></p> <p>Consultar el Diagrama de flujo de comparación de cadenas LES.</p> <p>Si la salida es falsa</p> <p>Se borra la condición de salida de reglón a falsa</p> <p>de lo contrario</p> <p>Se establece la condición de salida de reglón en verdadera.</p>
Post-escaneado	N/A

### Bloque de funciones

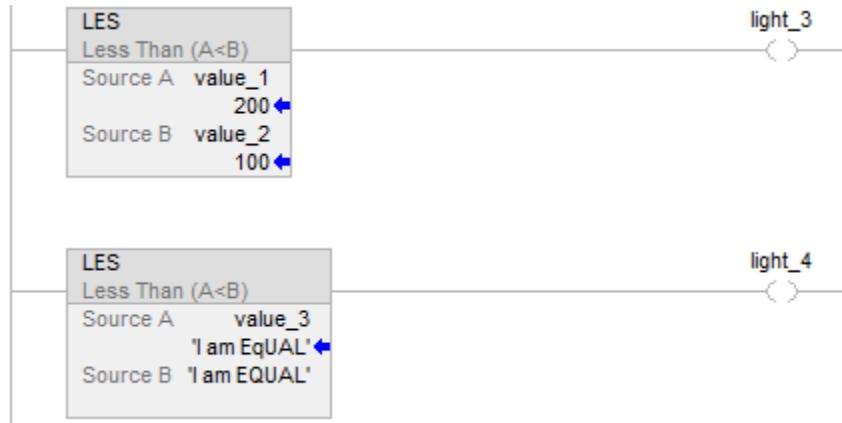
Condición/estado	Acción realizada
Pre-escaneado	N/A
EnableIn es falso	Establecer EnableOut a EnableIn
EnableIn es verdadero	<p><b>Comparación numérica:</b></p> <p>Establecer EnableOut a EnableIn</p> <p>Si SourceA y SourceB no son NAN y el SourceA es menor que el SourceB.</p> <p>Establecer Destino a verdadero.</p> <p><b>else</b></p> <p>Borrar Destino a falso.</p>
Primera ejecución de instrucción	N/A
Primer escaneado de instrucción	N/A
Post-escaneado	N/A

Diagrama de flujo de comparación de cadenas LES

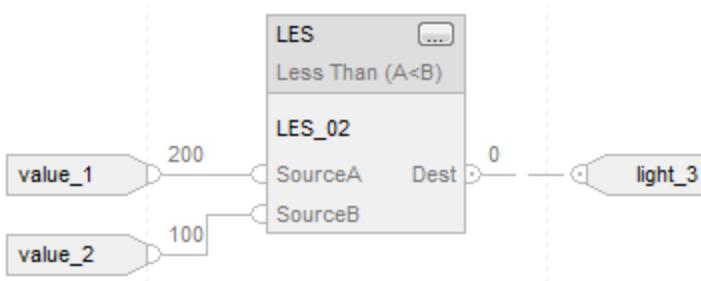


## Ejemplo

### Diagrama de escalera



### Bloque de funciones



### Texto estructurado

```

if value_1 < value_2 then
    light_3 := 1;
else
    light_3 := 0;
end_if;

if value_3 < 'I am EQUAL' then
    light_4 := 1;
else
    light_4 := 0;
end_if;
    
```

### Consulte también

[Sintaxis de texto estructurado](#) en la página 955

[Conversiones de datos](#) en la página 927

[Índice a través de matrices](#) en la página 936

[Comparar instrucciones](#) en la página 369

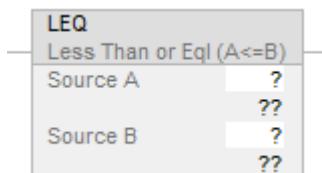
## Menor o igual a (LEQ)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

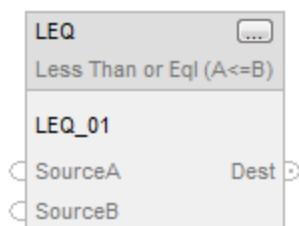
La instrucción LEQ comprueba si el Source A es igual o menor al Source B.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones



#### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

**Cons** Se utiliza el operador '`<=`' con una expresión para obtener el mismo resultado. Consulte *Structured Text Syntax* para obtener más información sobre la sintaxis de las expresiones y las asignaciones en el texto estructurado.

### Operandos

Existen reglas de conversión de datos para utilizar diversos tipos de datos numéricos en una instrucción. Consulte *Conversión de datos*.

### Diagrama de escalera

#### Comparación numérica

Operando	Tipo de datos	Formato	Descripción
Source A	SINT INT DINT REAL	inmediato etiqueta	Valor que se prueba con respecto a Source B
Source B	SINT INT DINT REAL	inmediato etiqueta	Valor que se prueba con respecto a Source A

#### Comparación de cadenas

**Cons** Los literales de cadena inmediatos solo se aplican a  
**ejo:** Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580..

Operando	Tipo de datos	Formato	Descripción
Source A	Tipo de cadena	valor literal inmediato etiqueta	Cadena que se prueba con respecto a Source B
Source B	Tipo de cadena	valor literal inmediato etiqueta	Cadena que se prueba con respecto a Source A

### Bloque de funciones

Operando	Tipo de datos	Formato	Descripción
LEQ	FBD_COMPARE	etiqueta	Estructura de LEQ

**Estructura de FBD\_COMPARE**

Miembros de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si el valor es falso, la instrucción no se ejecuta y las salidas no se actualizan. El valor predeterminado es verdadero.
SourceA	REAL	Valor que se prueba con respecto a SourceB
SourceB	REAL	Valor que se prueba con respecto a SourceA

Miembros de salida	Tipo de datos	Descripción
EnableOut	BOOL	Indica si la instrucción está habilitada.
Dest	BOOL	Se establece en verdadero cuando el SourceA es menor o igual al SourceB. Se borra y se establece en falso cuando el SourceA es mayor que el SourceB.

**Descripción**

Se utiliza la instrucción LEQ o el operador ' $\leq$ ' para comprobar si el Source A es igual o menor al Source B.

**Afectar a las marcas de estado matemático**

No

**Fallos mayores/menores**

Consulte el *Diagrama de flujo de comparación de cadenas LEQ* para los fallos.

Consulte *Índice a través de matrices* para ver si hay fallos de indexación de matrices.

**Ejecución****Diagrama de escalera**

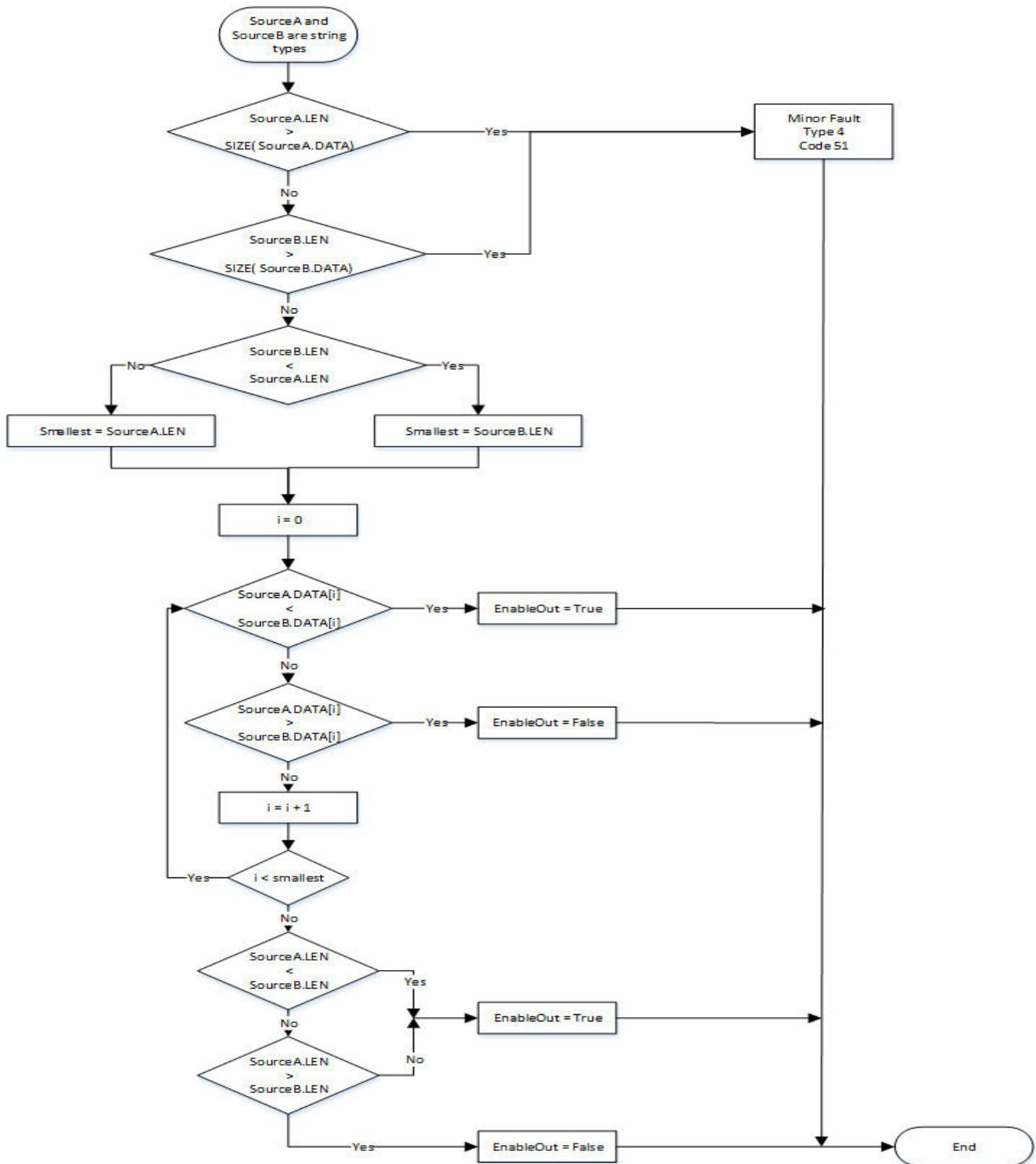
Condición/estado	Acción realizada
Pre-escaneado	N/A

La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón
La condición de entrada de reglón es verdadera	<p><b>Comparación numérica:</b>            Si el Source A y el Source B no son NAN y el Source A es menor o igual al Source B.</p> <p>Establecer la condición de salida de reglón en verdadera.</p> <p><b>else</b></p> <p>Borrar la condición de salida de reglón a falsa.</p>
	<p><b>Comparación de cadena:</b>            Consultar el <i>Diagrama de flujo de comparación de cadenas LEQ</i>.</p> <p>Si la salida es falsa            Se borra la condición de salida de reglón a falsa            de lo contrario            Se establece la condición de salida de reglón en verdadera.</p>

### Bloque de funciones

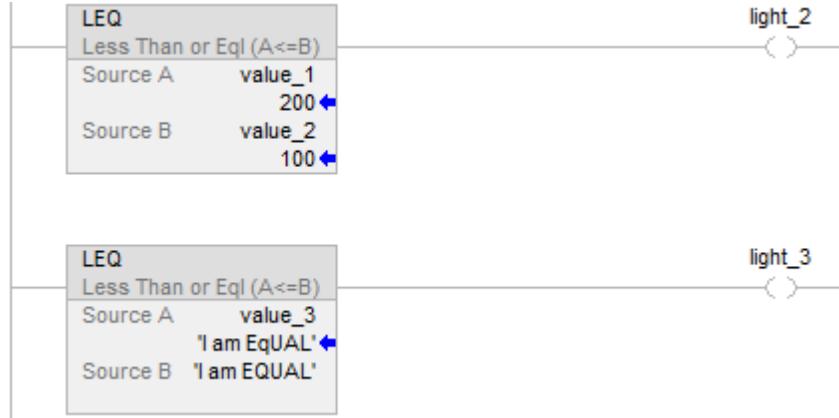
Condición/estado	Acción realizada
Pre-escaneado	N/A
EnableIn es falso	Establecer EnableOut a EnableIn
EnableIn es verdadero	<p><b>Comparación numérica:</b>            Establecer EnableOut a EnableIn            Si SourceA y SourceB no son NAN y el SourceA es igual o menor al SourceB.            Establecer Destino a verdadero.</p> <p><b>else</b></p> <p>Borrar Destino a falso.</p>
Primera ejecución de instrucción	N/A
Primer escaneado de instrucción	N/A
Post-escaneado	N/A

Diagrama de flujo de comparación de cadenas LEQ

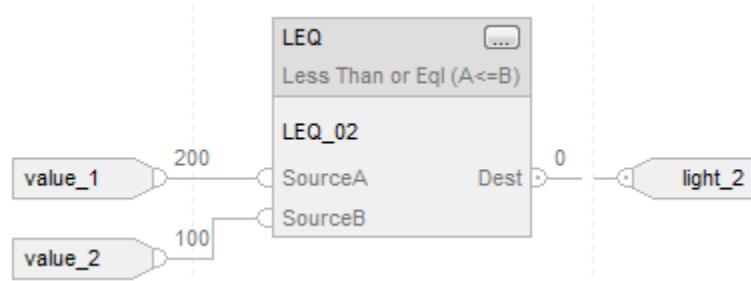


## Ejemplo

### Diagrama de escalera



### Bloque de funciones



### Texto estructurado

```

if value_1 <= value_2 then
    light_2 := 1;
else
    light_2 := 0;
end_if;

if value_3 <= 'I am EQUAL' then
    light_3 := 1;
else
    light_3 := 0;

```

end\_if;

#### Consulte también

[Sintaxis de texto estructurado](#) en la página 955

[Conversiones de datos](#) en la página 927

[Índice a través de matrices](#) en la página 936

[Comparar instrucciones](#) en la página 369

## Límite (LIM)

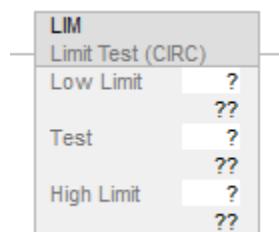
Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción LIM comprueba si el valor de Test está dentro o fuera del rango de Límite bajo o Límite alto, tal y como se describe en el Diagrama de flujo LIM (verdadero).

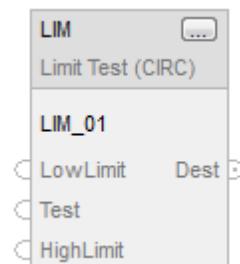
Si hay algún operando es No un número (NAN), .EnableOut se borra a falso.

#### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones



### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

### Operандos

Existen reglas de conversión de datos para utilizar diversos tipos de datos numéricos en una instrucción. Consulte *Conversión de datos*.

### Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción
Low Limit	SINT INT DINT REAL	inmediato etiqueta	Valor del límite bajo
Prueba	SINT INT DINT REAL	inmediato etiqueta	Valor a comprobar con respecto a los límites.
High Limit	SINT INT DINT REAL	inmediato etiqueta	Valor del límite alto

**Cons** Una etiqueta SINT o INT se convierte al valor DINT por **ejo:** extensión de signo.

### Bloque de funciones

Operando	Tipo de datos	Formato	Descripción
LIM	FBD_LIMIT	etiqueta	Estructura de LIM

### Estructura de FBD\_LIMIT

Miembros de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si el valor es falso, la instrucción no se ejecuta y las salidas no se actualizan. El valor predeterminado es verdadero.
LowLimit	REAL	Valor del límite bajo.

Prueba	REAL	Valor a comprobar con respecto a los límites.
HighLimit	REAL	Valor del límite alto

Miembros de salida	Tipo de datos	Descripción
EnableOut	BOOL	Indica si la instrucción está habilitada.
Dest	BOOL	Establece en verdadero cuando la prueba del límite es verdadera. Se borra a falso cuando la prueba del límite es falsa.

### Descripción

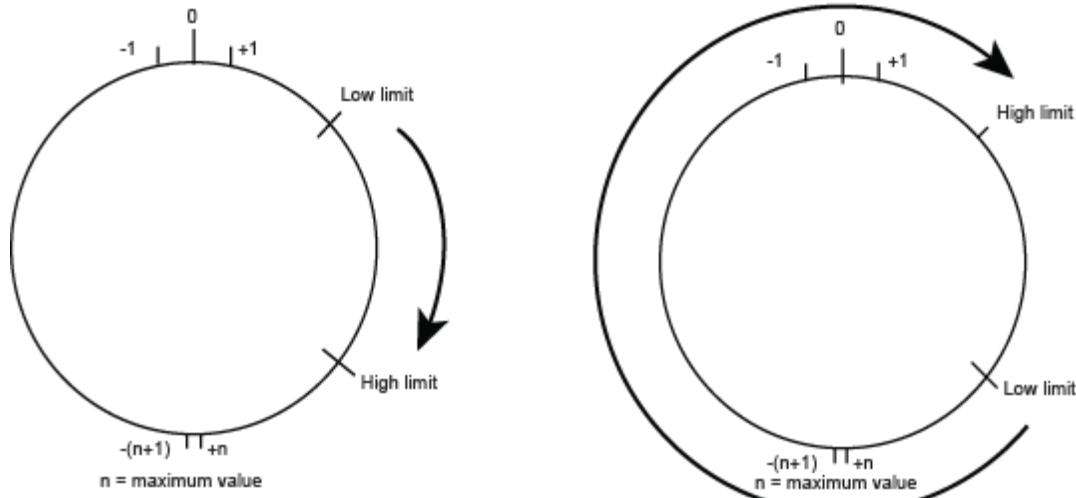
La instrucción LIM comprueba si el valor de Test está dentro o fuera del rango de Límite bajo Límite alto.

Low Limit  $\leq$  High Limit

Low Limit  $\geq$  High Limit

The instruction is true if the test value is equal to or between the low and high limit.

The instruction is true if the test value is equal to outside the low and high limit



Si el Límite bajo es :	Y si el valor de prueba es:	Entonces EnableOut es:
$< 0 = \text{Límite alto}$	igual o entre límites no igual o fuera de límites	verdadero falso

> Límite alto	igual o fuera de límites no igual o dentro de límites	verdadero falso
---------------	--	--------------------

Los enteros firmados realizarán la transición del número positivo máximo al número negativo máximo cuando el bit más significativo sea verdadero. Por ejemplo, en los enteros de 16 bits (INT type), el entero positivo máximo es 32.767, que estará representado en hexadecimal como 16#7FFF (los bits de 0 a 14 son verdaderos). Si ese número se aumenta en 1, el resultado será 16#8000 (el bit 15 es verdadero). Para los enteros firmados, el hexadecimal 16#8000 es igual al decimal -32.768. El incremento a partir de este punto hasta establecer los 16 bits terminará en 16#FFFF, que es igual a -1 en decimal.

Esto se puede mostrar como una línea de números en círculo. La instrucción LIM empieza en el Límite bajo y aumenta en sentido horario hasta que llega al Límite alto. Cualquier valor de Test en el rango en sentido horario desde el Límite bajo hasta el Límite alto establece EnableOut en verdadero. Cualquier valor de Test en el rango en sentido horario desde el Límite alto hasta el Límite bajo borra EnableOut a falso.

Si hay algún operando es No un número (NAN), .EnableOut se borra a falso.

#### Afectar a las marcas de estado matemático

No

#### Fallos mayores/menores

No es específico para esta instrucción. Consulte *Índice a través de matrices* para ver si hay fallos de indexación de matrices.

#### Ejecución

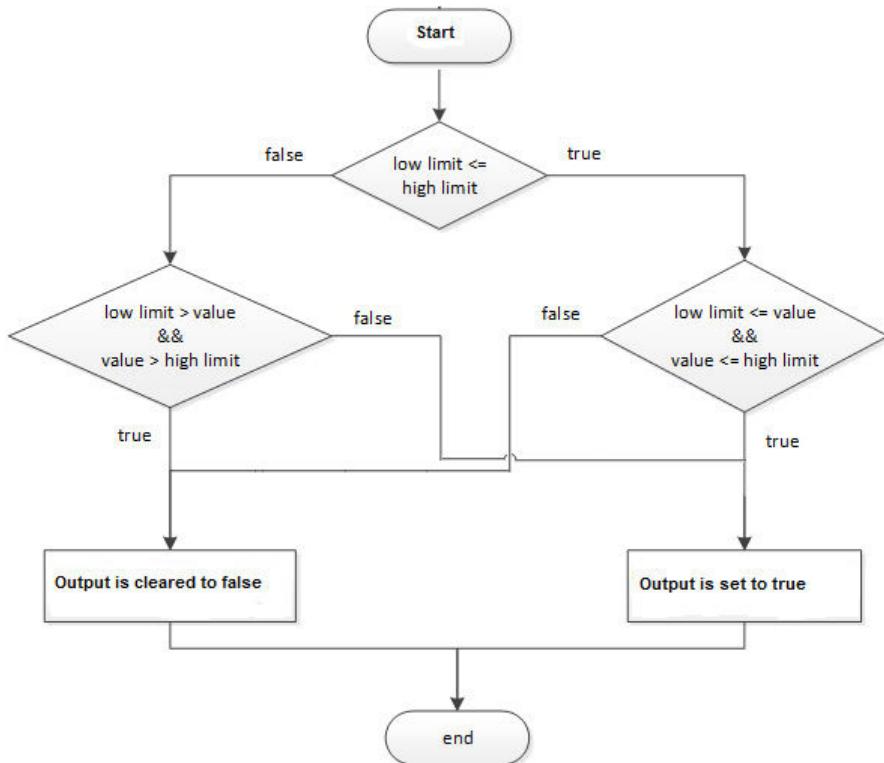
#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón.
La condición de entrada de reglón es verdadera	Consultar el <i>Diagrama de flujo LIM (Verdadero)</i> Si la salida es verdadera Se establece la condición de salida de reglón en verdadera. De lo contrario Se borra la condición de salida de reglón a falsa.
Post-escaneado	N/A

### Bloque de funciones

Condición/estado	Acción realizada
Pre-escaneado	N/A
EnableIn es falso	Establecer EnableOut a EnableIn.
EnableIn es verdadero	Establecer EnableOut a EnableIn. Consultar Diagrama de flujo LIM (Verdadero) Dest= salida
Primera ejecución de instrucción	N/A
Primer escaneado de instrucción	N/A
Post-escaneado	N/A

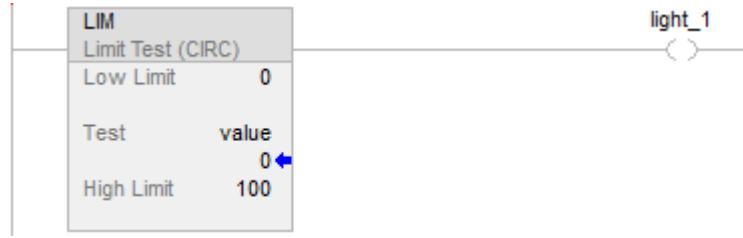
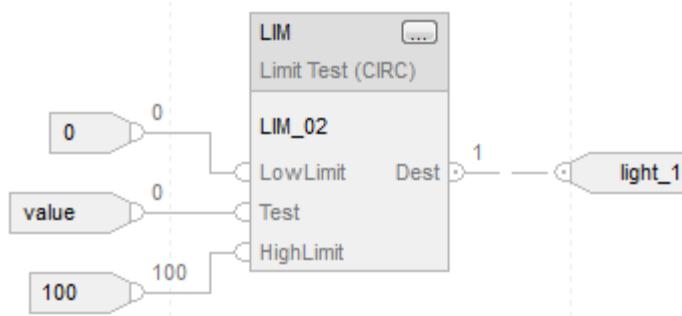
### Diagrama de flujo LIM (Verdadero)



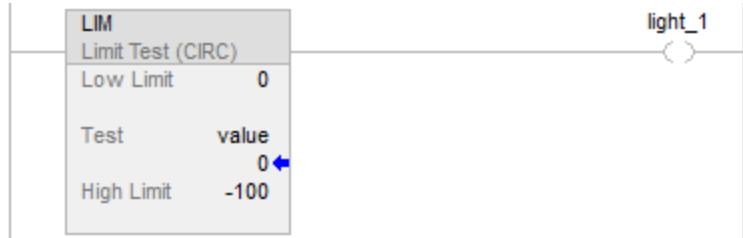
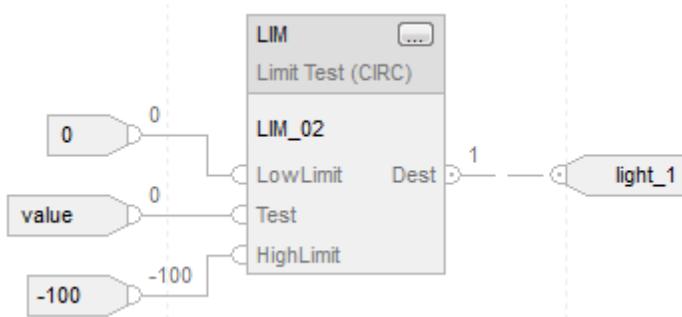
### Ejemplos

#### Ejemplo 1: Límite bajo <= Límite alto

Cuando el valor de Test es igual o mayor que el Límite bajo y el valor de Test es menor o igual al Límite alto se establecerá light\_1.

**Diagrama de escalera****Bloque de funciones****Ejemplo 2: Límite bajo > Límite alto**

Cuando valor < o = a 0 o valor < o = a -100, establece light\_1 en verdadero. Si valor < 0 o valor > -100, borra light\_1 a falso.

**Diagrama de escalera****Bloque de funciones**

**Consulte también**

[Comparar instrucciones](#) en la página 369

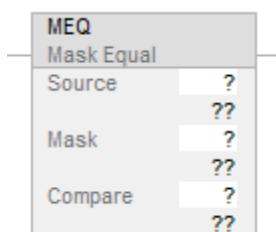
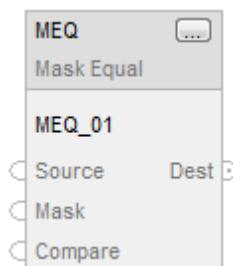
[Conversiones de datos](#) en la página 927

[Índice a través de matrices](#) en la página 936

**Máscara igual a (MEQ)**

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción MEQ pasa los valores de Source y Compare a través de una Mask y compara los resultados.

**Idiomas disponibles****Diagrama de escalera****Bloque de funciones****Texto estructurado**

Esta instrucción no está disponible en texto estructurado.

**Operандos**

Existen reglas de conversión de datos para utilizar diversos tipos de datos numéricos en una instrucción. Consulte *Conversión de datos*.

### Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción
Source	SINT INT DINT	inmediato etiqueta	Valor que se comprueba con respecto a Compare.
Mask	SINT INT DINT	inmediato etiqueta	Qué Bits a bloquear o pasar.
Compare	SINT INT DINT	inmediato etiqueta	Valor que se prueba con respecto a Source.
Una etiqueta SINT o INT se convierte al valor DINT mediante relleno de ceros			

### Bloque de funciones

Operando	Tipo de datos	Formato	Descripción
MEQ	FBD_MASK_EQUAL	etiqueta	Estructura de MEQ

### Estructura de FBD\_MASK\_EQUAL

Miembros de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si el valor es falso, la instrucción no se ejecuta y las salidas no se actualizan. El valor predeterminado es verdadero.
Source	DINT	Valor que se comprueba con respecto a Compare.
Mask	DINT	Define qué bits a bloquear, como máscara.
Compare	DINT	Valor que se prueba con respecto a Source.

Miembros de salida	Tipo de datos	Descripción
EnableOut	BOOL	Indica si la instrucción se ejecutó sin fallos al ser habilitada.

Dest	BOOL	Establece en verdadero cuando el resultado es verdadero. Borra a falso cuando el resultado es falso.
------	------	--

### Descripción

Un "1" en la máscara significa que el bit de datos se ha pasado. Un "0" en la máscara significa que el bit de datos se ha bloqueado. Normalmente, los valores de Source, Mask y Compare son del mismo tipo de datos.

Si se emplea un tipo de dato SINT o INT, la instrucción rellena los bits más altos con ceros, de manera que queden del mismo tamaño que el tipo de datos DINT.

### Introducción de un valor de máscara inmediato

Al introducir una máscara, el software de programación la establece por defecto a valores decimales. Para introducir una máscara con otro formato, ponga el valor con el valor prefijo.

Prefijo	Descripción
16#	hexadecimal (por ejemplo, 16#0F0F)
8#	octal (por ejemplo, 8#16)
2#	binario (por ejemplo, 2#00110011)

### Afectar a las marcas de estado matemático

No

### Fallos mayores/menores

No es específico para esta instrucción. Consulte *Índice a través de matrices* para ver si hay fallos de indexación de matrices.

### Ejecución

#### Diagrama de escalera

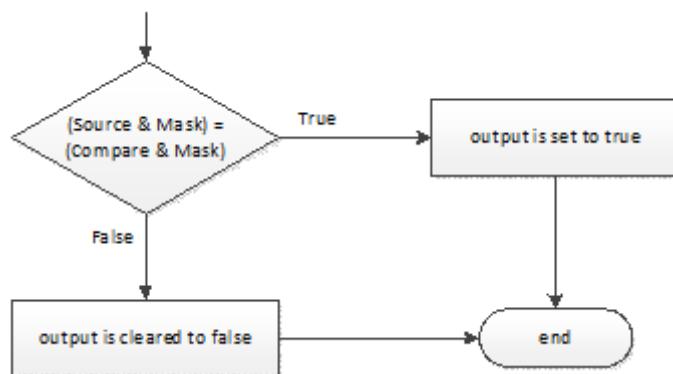
Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón.

La condición de entrada de reglón es verdadera	Consultar el Diagrama de Flujo MEQ (Verdadero) Si la salida es verdadera Se establece la condición de salida de reglón en verdadera. De lo contrario Se borra la condición de salida de reglón a falsa
Post-escaneado	N/A

### Bloque de funciones

Condición/estado	Acción realizada
Pre-escaneado	N/A
EnableIn es falso	Establecer EnableOut a EnableIn.
EnableIn es verdadero	Establecer EnableOut a EnableIn. Consultar el <i>Diagrama de Flujo MEQ (Verdadero)</i> . Si la salida es verdadera Se establece Dest en verdadero de lo contrario Borrar Dest a falso
Primera ejecución de instrucción	N/A
Primer escaneado de instrucción	N/A
Post-escaneado	N/A

### Diagrama de flujo MEQ (verdadero)



## Ejemplos

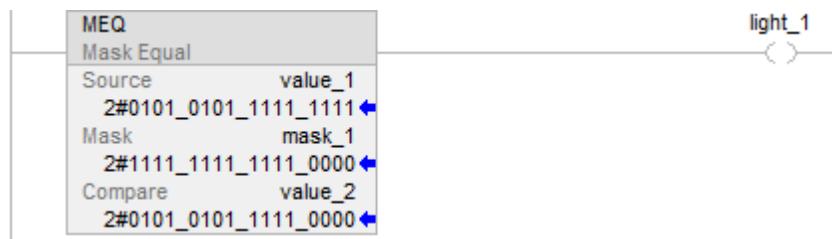
### Ejemplo 1

Si value\_1 enmascarado es igual a value\_2 enmascarado, se establece light\_1 en verdadero. Si value\_1 enmascarado no es igual a value\_2 enmascarado, se borra light\_1 a falso.

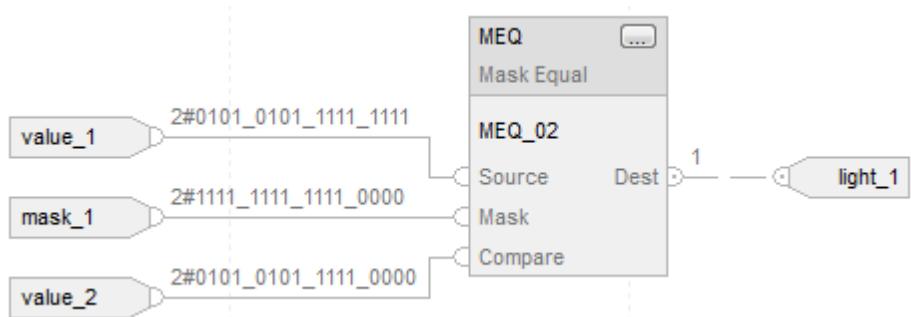
Este ejemplo muestra que los valores mascaraos son iguales. Un 0 en la máscara evita que la instrucción compare ese bit (indicado por una x en el ejemplo).

### Diagrama de escalera

value_1    0101010111111111 mask_1    1111111111110000 Masked    01010101111111xx	value_2    0101010111110000 mask_1    1111111111110000 Masked    01010101111111xx
---	---



### Bloque de funciones

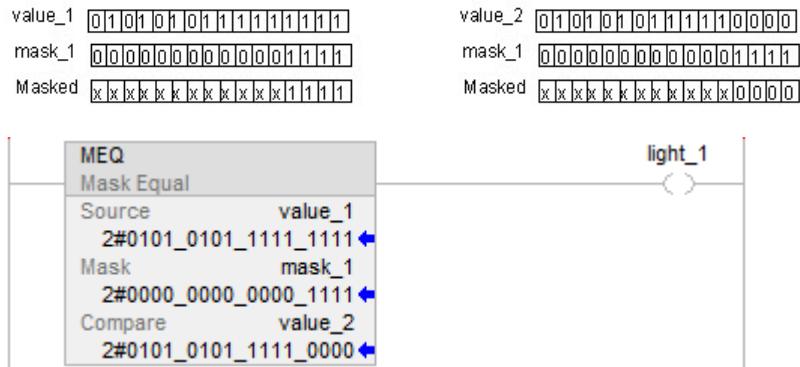


### Ejemplo 2

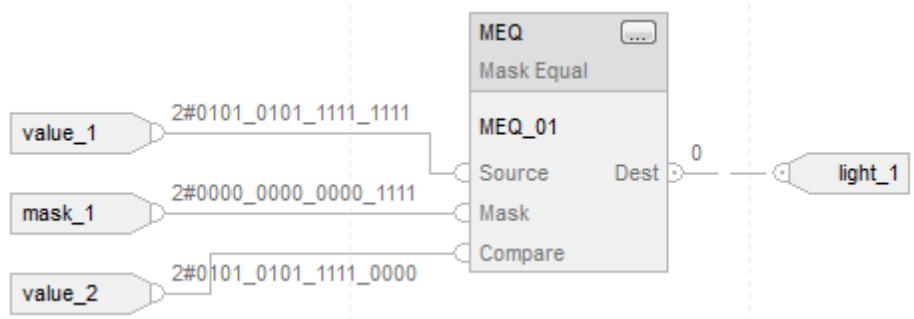
Si value\_1 enmascarado es igual a value\_2 enmascarado, se establece light\_1 en verdadero. Si value\_1 enmascarado no es igual a value\_2 enmascarado, se borra light\_1 a falso.

Este ejemplo muestra que los valores mascaraos no son iguales. Un 0 en la máscara evita que la instrucción compare ese bit (indicado por una x en el ejemplo).

### Diagrama de escalera



### Bloque de funciones



### Consulte también

[Índice a través de matrices](#) en la página 936

[Comparar instrucciones](#) en la página 369

[Conversiones de datos](#) en la página 927

[¿Qué es el relleno de ceros?](#) en la página 423

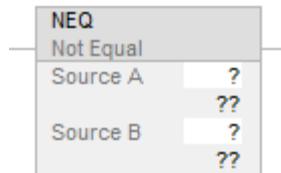
## No igual a (NEQ)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

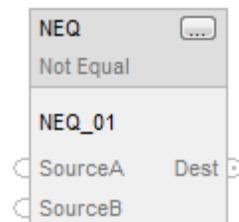
La instrucción NEQ prueba si Source A no es igual a Source B.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones



#### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

**Cons** Utilice el operador '<>' con una expresión para obtener el mismo resultado. Consulte *Structured Text Syntax* para obtener más información sobre la sintaxis de las expresiones y las asignaciones en el texto estructurado.

#### Operандos

Existen reglas de conversión de datos para utilizar diversos tipos de datos numéricos en una instrucción. Consulte *Conversión de datos*.

#### Diagrama de escalera

#### Comparación numérica

Operando	Tipo de datos	Formato	Descripción
Source A	SINT INT DINT REAL	inmediato etiqueta	Valor que se prueba con respecto a Source B
Source B	SINT INT DINT REAL	inmediato etiqueta	Valor que se prueba con respecto a Source A

## Comparación de cadenas

**Cons** Los literales de cadena inmediatos solo se aplican a  
**ejo:** Controladores CompactLogix 5380, CompactLogix  
 5480, ControlLogix 5580, Compact GuardLogix  
 5380 y GuardLogix 5580..

Operando	Tipo de datos	Formato	Descripción
Source A	Tipo de cadena	valor literal inmediato etiqueta	Cadena que se prueba con respecto a Source B
Source B	Tipo de cadena	valor literal inmediato etiqueta	Cadena que se prueba con respecto a Source A

## Bloque de funciones

Operando	Tipo de datos	Formato	Descripción
NEQ	FBD_COMPARE	etiqueta	Estructura de NEQ

## Estructura de FBD\_COMPARE

Miembros de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si el valor es falso, la instrucción no se ejecuta y las salidas no se actualizan. El valor predeterminado es verdadero.
SourceA	REAL	Valor que se comprueba con respecto a SourceB.
SourceB	REAL	Valor que se comprueba con respecto a SourceA.

Miembros de salida	Tipo de datos	Descripción
EnableOut	BOOL	Indica si la instrucción está habilitada.
Dest	BOOL	Establece en verdadero cuando SourceA no es igual a SourceB. Se borra a falso cuando SourceA es igual a SourceB.

## Descripción

Se utiliza la instrucción NEQ o el operador '<>' para comparar dos números o dos cadenas de caracteres ASCII.

## Afectar a las marcas de estado matemático

No

## Fallos mayores/menores

Consulte el *Diagrama de flujo de comparación de cadenas NEQ* para los fallos.

Consulte *Índice a través de matrices* para ver si hay fallos de indexación de matrices.

## Ejecución

### Diagrama de escalera

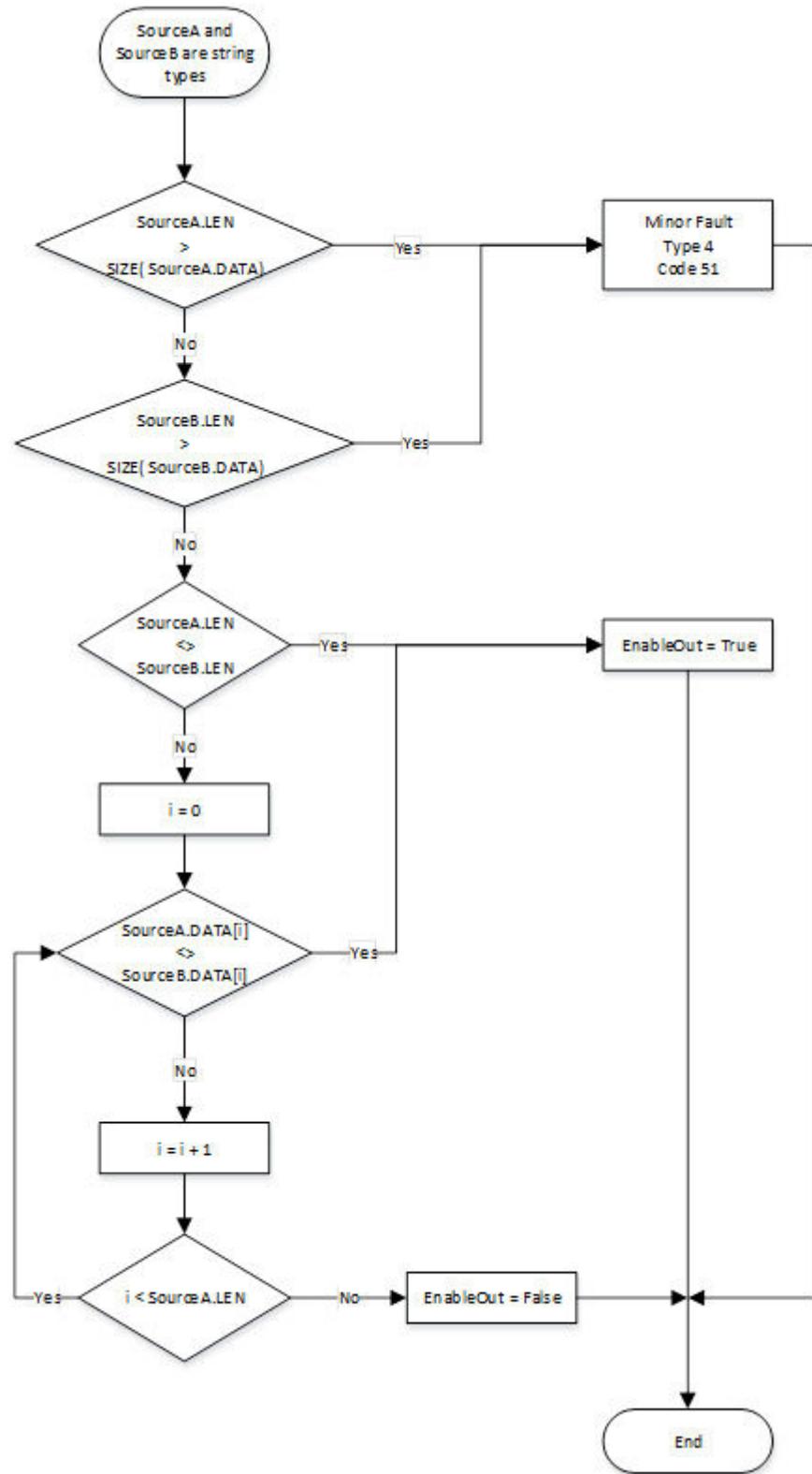
Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón
La condición de entrada de reglón es verdadera	<b>Comparación numérica:</b> Si el Source A o el Source B es NAN y el Source A no es igual a Source B. Establecer la condición de salida de reglón en verdadera. <b>else</b> Borrar la condición de salida de reglón a falsa. <b>Comparación de cadena:</b> Consultar el <i>Diagrama de flujo de comparación de cadenas NEQ</i> . Si la salida es falsa Se borra la condición de salida de reglón a falsa de lo contrario Se establece la condición de salida de reglón en verdadera.
Post-escaneado	N/A

## Bloque de funciones

Condición/estado	Acción realizada
Pre-escaneado	N/A
EnableIn es falso	Establecer EnableOut a EnableIn

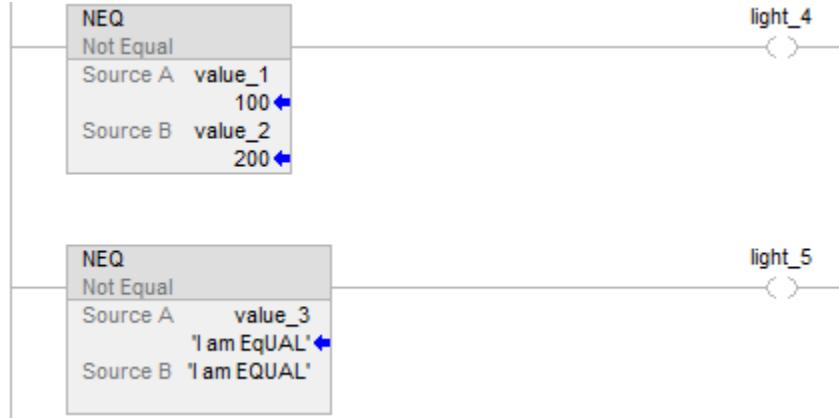
EnableIn es verdadero	<b>Comparación numérica:</b> Establecer EnableOut a EnableIn Si el SourceA o el SourceB es NAN o si el SourceA no es igual a SourceB. Establecer Destino a verdadero. <b>else</b> Borrar Destino a falso.
Primera ejecución de instrucción	N/A
Primer escaneado de instrucción	N/A
Post-escaneado	N/A

Diagrama de flujo de comparación de cadenas NEQ

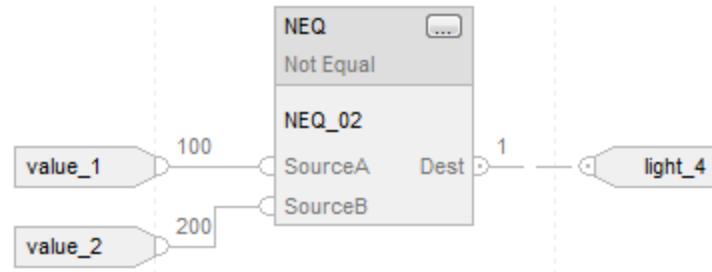


## Ejemplo

### Diagrama de escalera



### Bloque de funciones



### Texto estructurado

```

if value_1 <> value_2 then
    light_4 := 1;
else
    light_4 := 0;
end_if;

if value_3 <> 'I am EQUAL' then
    light_5 := 1;
else
    light_5 := 0;
end_if;
    
```

### Consulte también

[Sintaxis de texto estructurado](#) en la página 955

[Conversiones de datos](#) en la página 927

[Índice a través de matrices](#) en la página 936

[Comparar instrucciones](#) en la página 369

## Operadores válidos

A continuación se muestran los operadores válidos.

Operador	Descripción	Óptimo
+	sumar	DINT, REAL
-	restar/negar	DINT, REAL
*	multiplicar	DINT, REAL
/	dividir	DINT, REAL
=	igual	DINT, REAL
<	menor que	DINT, REAL
<=	menor o igual que	DINT, REAL
>	mayor que	DINT, REAL
>=	mayor o igual que	DINT, REAL
<>	No igual	DINT, REAL
**	exponente (x a y)	DINT, REAL
ABS	valor absoluto	DINT, REAL
ACS	arcocoseno	REAL
AND	Y a nivel de bits	DINT
ASN	arcoseno	REAL
ATN	arcotangente	REAL
COS	coseno	REAL
DEG	radianes a grados	DINT, REAL
FRD	BCD a entero	DINT
LN	logaritmo natural	REAL
LOG	logaritmo de base 10	REAL
MOD	módulo-división	DINT, REAL
NOT	complemento a nivel de bits	DINT
OR	O a nivel de bits	DINT
RAD	grados a radianes	DINT, REAL
SIN	seno	REAL
SQR	raíz cuadrada	DINT, REAL
TAN	tangente	REAL
TOD	entero a BCD	DINT
TRN	truncar	DINT, REAL

XOR	O exclusivo a nivel de bits	DINT
-----	-----------------------------	------

## ¿Qué es el relleno de ceros?

Hay dos formas de convertir un tipo entero más pequeño a uno más grande:

- Relleno de ceros
- Extensión de signo

El método empleado depende de la instrucción que está utilizando el operando.

Para el relleno de ceros, todos los bits por encima del rango del tipo más pequeño se llenan con 0.

Por ejemplo, SINT: 16#87 = -121 convertido a DINT da como resultado  
16#00000087 = 135

Para la extensión de signo, todos los bits por encima del rango del tipo más pequeño se llenan con el bit de signo del tipo más pequeño.

Por ejemplo: SINT: 16#87 = -121 convertido a DINT da como resultado  
16#FFFFF87 = -121

### Consulte también

[Máscara igual a \(Mask Equal To, MEQ\)](#) en la página 410



# **InSTRUCCIONES DE CÁLCULO/MATEMÁTICAS**

## **InSTRUCCIONES DE CÁLCULO/MATEMÁTICAS**

### **S**

Las instrucciones de cálculo/matemáticas evalúan las operaciones aritméticas usando una expresión o una instrucción aritmética específica.

#### **InSTRUCCIONES DISPONIBLES**

##### **Diagrama de escalera**

<a href="#">CPT</a>	<a href="#">ADD</a>	<a href="#">SUB</a>	<a href="#">MUL</a>	<a href="#">DIV</a>	<a href="#">MOD</a>	<a href="#">SQR</a>	<a href="#">SQRT</a>	<a href="#">NEG</a>	<a href="#">ABS</a>
---------------------	---------------------	---------------------	---------------------	---------------------	---------------------	---------------------	----------------------	---------------------	---------------------

##### **BLOQUE DE FUNCIONES**

<a href="#">ADD</a>	<a href="#">SUB</a>	<a href="#">MUL</a>	<a href="#">DIV</a>	<a href="#">MOD</a>	<a href="#">SQR</a>	<a href="#">SQRT</a>	<a href="#">NEG</a>	<a href="#">ABS</a>
---------------------	---------------------	---------------------	---------------------	---------------------	---------------------	----------------------	---------------------	---------------------

##### **TEXTO ESTRUCTURADO**

<a href="#">SQR</a>	<a href="#">SQRT</a>	<a href="#">ABS</a>
---------------------	----------------------	---------------------

<b>Si desea:</b>	<b>Utilice esta instrucción:</b>
evaluar una expresión	CPT
sumar dos valores	ADD
restar dos valores	SUB
multiplicar dos valores	MUL
dividir dos valores	DIV
determinar el resto después de dividir dos valores	MOD
calcular la raíz cuadrada de un valor	SQR
coger el signo opuesto de un valor	NEG
coger el valor absoluto de un valor	ABS

Se pueden mezclar tipos de datos, pero es posible que se pierda precisión, que haya errores de redondeo y que la instrucción tarde más tiempo en ejecutarse. Compruebe el bit S:V para ver si el resultado se ha truncado.

Los tipos de datos en negrita son los óptimos. Una instrucción se ejecutará a la máxima velocidad y la menor cantidad posible de memoria si todos los parámetros emplean el mismo tipo de datos óptimo, normalmente DINT o REAL.

Una instrucción de cálculo/matemática se ejecuta una vez en cada escaneado, siempre y cuando la condición de entrada de reglón sea verdadera. Si desea evaluar la expresión solo una vez, utilice cualquier instrucción de un impulso para desencadenar la instrucción.

### Consulte también

[Comparar instrucciones](#) en la página 369

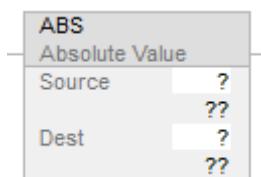
## Valor absoluto (ABS)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

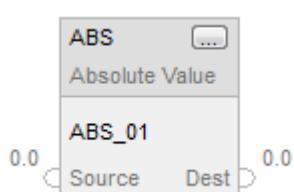
Cuando está habilitada, la instrucción ABS y el operador obtienen el valor absoluto de Source. La instrucción almacena el resultado en Dest mientras que el operador solo devuelve el resultado. Se indicará un desbordamiento si el resultado es el valor entero negativo máximo (por ejemplo, -128 para SINT, -32.768 para INT y -2.147.483.648 para DINT).

### Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones



## Texto estructurado

Esta instrucción no está disponible en texto estructurado.

**Cons ejo:** Utilice ABS como el operador en una expresión para calcular el mismo resultado. Consulte Sintaxis de texto estructurado para obtener más información sobre la sintaxis de las expresiones y las asignaciones en el texto estructurado.

## Operандos

**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
- Los miembros de un operando de estructura se sobrescriben.
- A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.

Existen reglas de conversión de datos para utilizar diversos tipos de datos numéricos en una instrucción. Ver Conversión de Datos.

## Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción
Source	SINT INT DINT REAL	inmediato etiqueta	Valor de la que se extrae el valor absoluto.
Dest	SINT INT DINT REAL	etiqueta	Etiqueta para almacenar el resultado de la instrucción.

## Bloque de funciones

Operando	Tipo de datos	Formato	Descripción
ABS	FBD_MATH_A DVANCED	etiqueta	Estructura de ABS

**Estructura de FBD\_MATH\_ADVANCED**

Miembros de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si el valor es falso, la instrucción no se ejecuta y las salidas no se actualizan. El valor predeterminado es verdadero.
Source	REAL	Valor de la que se extrae el valor absoluto.

Miembros de salida	Tipo de datos	Descripción
EnableOut	BOOL	Indica si la instrucción se ejecutó sin fallos al ser habilitada.
Dest	REAL	Resultado de la instrucción.

**Afectar a las marcas de estado matemático**

Controladores	Afectar a las marcas de estado matemático
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Condicional
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí

Consulte las Marcas de estado matemático.

**Fallos mayores/menores**

No es específico para esta instrucción. Consulte Índice a través de matrices para ver si hay fallos de indexación de matrices.

## Ejecución

### Diagrama de escalera

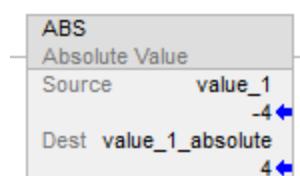
Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón.
La condición de entrada de reglón es verdadera	Establecer la condición de salida de reglón a condición de entrada de reglón. Dest = valor absoluto de Source
Post-escaneado	N/A

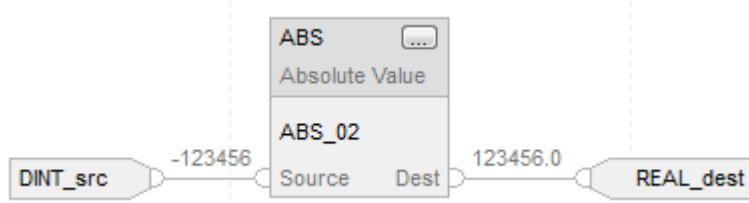
### Bloque de funciones

Condición/estado	Acción realizada
Pre-escaneado	N/A
EnableIn es falso.	Establecer EnableOut a EnableIn.
EnableIn es verdadero	Dest = valor absoluto de Source Si se produce un desbordamiento Borrar EnableOut a falso. else Se establece EnableOut a verdadero.
Primera ejecución de instrucción	N/A
Primer escaneado de instrucción	N/A
Post-escaneado	N/A

## Ejemplos

### Diagrama de escalera



**Bloque de funciones****Texto estructurado**

```
DINT_dest := ABS(DINT_src);
```

**Consulte también**

[Instrucciones de cálculo](#) en la página 425

[Sintaxis de texto estructurado](#) en la página 955

[Índice a través de matrices](#) en la página 936

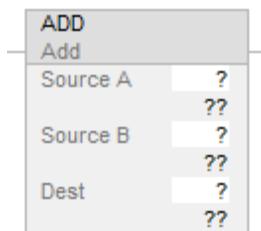
[Marcas de estado matemático](#) en la página 923

[Conversiones de datos](#) en la página 927

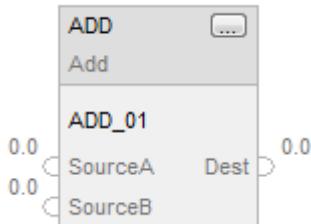
**Sumar (ADD)**

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

La instrucción ADD y el operador '+' añaden Source A al Source B.

**Idiomas disponibles****Diagrama de escalera**

## Bloque de funciones



## Texto estructurado

Esta instrucción no está disponible en texto estructurado.

**Cons ejo:** Se utiliza el operador '+' en una expresión para computar el mismo resultado. Consulte Sintaxis de texto estructurado para obtener más información sobre la sintaxis de las expresiones y las asignaciones en el texto estructurado.

## Operандos

**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
- Los miembros de un operando de estructura se sobrescriben.
- A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.

Existen reglas de conversión de datos para utilizar diversos tipos de datos numéricos en una instrucción. Consulte *Conversión de datos*.

## Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción
Source A	SINT INT DINT REAL	inmediato etiqueta	Valor a añadir a Source B
Source B	SINT INT DINT REAL	inmediato etiqueta	Valor a añadir a Source A
Dest	SINT INT DINT REAL	etiqueta	Etiqueta para almacenar el resultado de la instrucción

**Bloque de funciones**

Operando	Tipo de datos	Formato	Descripción
ADD	FBD_MATH	etiqueta	Estructura de ADD

**Estructura de FBD\_MATH**

Miembros de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si el valor es falso, la instrucción no se ejecuta y las salidas no se actualizan. El valor predeterminado es verdadero.
Source A	REAL	Valor a añadir a SourceB.
Source B	REAL	Valor a añadir a SourceA.

Miembros de salida	Tipo de datos	Descripción
EnableOut	BOOL	Indica si la instrucción se ejecutó sin fallos al ser habilitada.
Dest	REAL	Resultado de la instrucción.

**Descripción**

Cuando se habilita, la instrucción ADD y el operador '+' añaden el Source A al Source B. Además, ADD almacena el resultado en el Dest, mientras que '+' solo devuelve el resultado.

**Afectar a las marcas de estado matemático**

Controladores	Afectar a las marcas de estado matemático
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Condicional
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí

Consulte *Marcas de estado matemático*.

### Fallos mayores/menores

No es específico para esta instrucción. Consulte *Índice a través de matrices* para ver si hay fallos de indexación de matrices.

### Ejecución

#### Diagrama de escalera

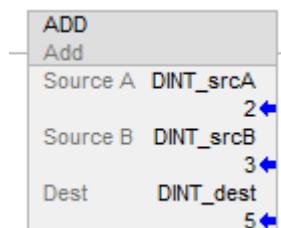
Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón
La condición de entrada de reglón es verdadera	Establecer la condición de salida de reglón a la condición de entrada de reglón, Dest =Source A + Source B
Post-escaneado	N/A

#### Bloque de funciones

Condición/estado	Acción realizada
Pre-escaneado	N/A
EnableIn es falso	Establecer EnableOut a EnableIn
EnableIn es verdadero	Dest = SourceA + SourceB Si se produce un desbordamiento Borrar EnableOut a falso else Establecer EnableOut a verdadero
Primera ejecución de instrucción	N/A
Primer escaneado de instrucción	N/A
Post-escaneado	N/A

### Ejemplo

#### Diagrama de escalera



### Bloque de funciones



### Texto estructurado

DINT\_dest := DINT\_srcA + DINT\_srcB;

### Consulte también

[Instrucciones de cálculo](#) en la página 425

[Sintaxis de texto estructurado](#) en la página 955

[Índice a través de matrices](#) en la página 936

[Conversiones de datos](#) en la página 927

[Marcas de estado matemático](#) en la página 923

## Calcular (CPT)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

Cuando está habilitada, la instrucción CPT evalúa la expresión y pone el resultado en Dest.

La ejecución de una instrucción CPT es ligeramente más lenta y usa más memoria que la de otras instrucciones de cálculo/matemáticas. La ventaja de las instrucciones CPT es que permiten integrar expresiones complejas en una sola instrucción.

Al evaluar la expresión, todos los operandos del tipo nonREAL se convertirán a REAL antes de realizar los cálculos si alguna de las siguientes condiciones es verdadera.

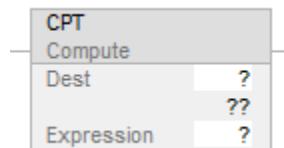
- Cualquier operando en la expresión es REAL.
- La expresión contiene SIN, COS, TAN, ASN, ACS, ATN, LN, LOG, DEG o RAD.

- Dest es REAL

Hay reglas para operadores admisibles en aplicaciones de seguridad. Consulte los *Operadores válidos*.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

#### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

#### Operандos

**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
- Los miembros de un operando de estructura se sobrescriben.
- A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.

Existen reglas de conversión de datos para utilizar diversos tipos de datos numéricos en una instrucción. Consulte *Conversión de datos*.

#### Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción
Dest	SINT INT DINT REAL	etiqueta	Etiqueta para almacenar el resultado

Expression	SINT INT DINT REAL	inmediato etiqueta	Una expresión que consiste en etiquetas y/o valores inmediatos separados por operadores
------------	-----------------------------	-----------------------	---

### Formato de expresiones

Se deben proporcionar uno o dos operandos (etiquetas o valores inmediatos) para cada uno de los operadores usados en una expresión. Se usa la siguiente tabla para dar formato a los operadores y operandos dentro de una expresión.

Para operadores que operan en:	Se usa este formato:	Ejemplo
Un operando	operador(operando)	ABS(tag)
Dos operandos	operand_a operador operand_b	tag_b + 5 tag_c AND tag_d (tag_e**2) MOD (tag_f / tag_g)

### Determine el orden de operación

La instrucción realiza las operaciones dentro de una expresión en un orden estipulado, no necesariamente en el orden en que aparecen. Se puede especificar el orden de la operación agrupando los términos en paréntesis, lo que obligará a la instrucción a realizar las operaciones dentro los paréntesis antes de otras operaciones.

Las operaciones del mismo orden se realizarán de izquierda a derecha.

Orden	Operación
1	( )
2	ABS, ACS, ASN, ATN, COS, DEG, FRD, LN, LOG, RAD, SIN, SQR, TAN, TOD, TRN
3	**
4	- (negar), NOT
5	*, /, MOD
6	- (restar), +
7	AND
8	XOR
9	OR

### Afectar a las marcas de estado matemático

Controladores	Afectar a las marcas de estado matemático
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Condicional
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí

Consulte *Marcas de estado matemático*.

### Fallos mayores/menores

No es específico para esta instrucción. Consulte *Índice a través de matrices* para ver si hay fallos de indexación de matrices.

### Ejecución

#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón
La condición de entrada de reglón es verdadera	Se establece la condición de salida de reglón al valor de la condición de entrada de reglón La instrucción evaluará la expresión y colocará el resultado en el Dest.
Post-escaneado	N/A

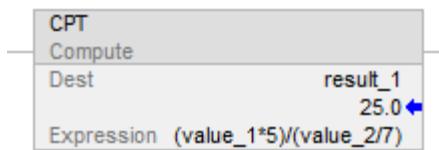
## Ejemplos

### Diagrama de escalera

#### Ejemplo 1

Cuando está habilitada, la instrucción CPT evalúa value\_1 multiplicada por 5 y divide ese resultado por el resultado de dividir value\_2 por 7. Finalmente, coloca el resultado final en result\_1.

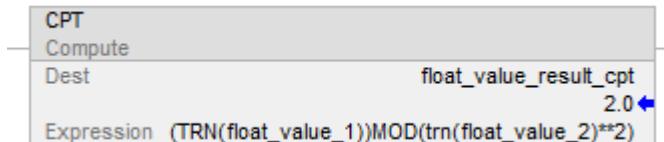
result_1	25.0	Float	REAL
+ value_1	10	Decimal	DINT
+ value_2	14	Decimal	DINT



#### Ejemplo 2

Cuando está habilitada, la instrucción CPT trunca float\_value\_1 y float\_value\_2 a un múltiplo de dos y divide el valor truncado de float\_value\_1 por ese resultado. A continuación, almacena el resto de la división en float\_value\_result\_cpt.

### Diagrama de escalera



float_value_result_cpt	2.0	Float	REAL
float_value_1	10.5	Float	REAL
float_value_2	2.5	Float	REAL

### Consulte también

[Instrucciones de cálculo](#) en la página 425

[Operadores válidos](#) en la página 422

[Índice a través de matrices](#) en la página 936

[Marcas de estado matemático](#) en la página 923

[Conversiones de datos](#) en la página 927

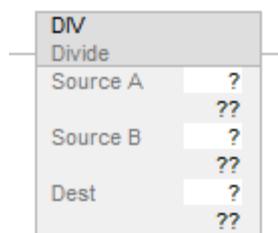
## Dividir (DIV)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

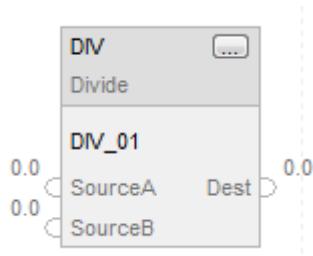
La instrucción DIV y el operador '/' dividen Source A por Source B.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones



#### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

**Cons ejo:** Se utiliza el operador '/' en una expresión para obtener el mismo resultado. Consulte Sintaxis de texto estructurado para obtener más información sobre la sintaxis de las expresiones y las asignaciones en el texto estructurado.

## Operandos

**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
- Los miembros de un operando de estructura se sobrescriben.
- A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.

Existen reglas de conversión de datos para utilizar diversos tipos de datos numéricos en una instrucción. Consulte *Conversión de datos*.

## Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción
Source A	SINT INT DINT REAL	inmediato etiqueta	Valor del dividendo
Source B	SINT INT DINT REAL	inmediato etiqueta	Valor del divisor
Dest	SINT INT DINT REAL	etiqueta	Etiqueta para almacenar el resultado de la instrucción

## Bloque de funciones

Operando	Tipo de datos	Formato	Descripción
DIV	FBD_MATH	etiqueta	Estructura de DIV

### Estructura de FBD\_MATH

Miembros de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si el valor es falso, la instrucción no se ejecuta y las salidas no se actualizan. El valor predeterminado es verdadero.
Source A	REAL	Valor del dividendo
Source B	REAL	Valor del divisor

Miembros de salida	Tipo de datos	Descripción
EnableOut	BOOL	Indica si la instrucción se ejecutó sin fallos al ser habilitada.
Dest	REAL	Resultado de la instrucción.

### Descripción

Cuando está habilitada, la instrucción DIV y el operador '/' dividen Source A por Source B. Además, DIV almacena el resultado en Dest mientras que '/' simplemente devuelve el resultado.

### Afectar a las marcas de estado matemático

Controladores	Afectar a las marcas de estado matemático
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Condicional
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí

Consulte *Marcas de estado matemático*.

**Fallos mayores/menores**

Ocurrirá un fallo menor si:	Tipo de fallo	Código de fallo
Source_B = 0	4	4

Consulte *Índice a través de matrices* para ver si hay fallos de indexación de matrices.

**Ejecución****Diagrama de escalera**

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón
La condición de entrada de reglón es verdadera	Establecer la condición de salida de reglón a condición de entrada de reglón Dest = Source A / Source B <sup>1,2</sup>
Post-escaneado	N/A

**Bloque de funciones**

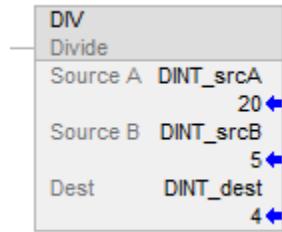
Condición/estado	Acción realizada
Pre-escaneado	N/A
EnableIn es falso	Establecer EnableOut a EnableIn
EnableIn es verdadero	Dest = SourceA / SourceB <sup>1,2</sup> Si se produce un desbordamiento Borrar EnableOut a falso else Establecer EnableOut a verdadero
Primera ejecución de instrucción	N/A
Primer escaneado de instrucción	N/A
Post-escaneado	N/A

<sup>1</sup> Si Source B es 0, el resultado es Source A y se genera un fallo menor.

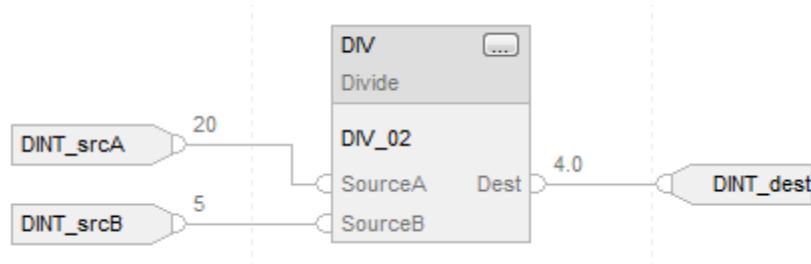
<sup>2</sup> Para destino de número entero y operandos de origen, el resultado se trunca.

## Ejemplos

### Diagrama de escalera



### Bloque de funciones



### Texto estructurado

DINT\_dst := DINT\_srcA / DINT\_srcB;

### Consulte también

[Sintaxis de texto estructurado](#) en la página 955

[Índice a través de matrices](#) en la página 936

[Marcas de estado matemático](#) en la página 923

[Conversiones de datos](#) en la página 927

[Instrucciones de cálculo](#) en la página 425

## Módulo (MOD)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

La instrucción MOD divide Source A por Source B y coloca el resto en Dest.

## Idiomas disponibles

### Diagrama de escalera

MOD	
Modulo	
Source A	?
	??
Source B	?
	??
Dest	?
	??

### Bloque de funciones



### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

**Cons** Se utiliza MOD como un operador en una expresión para **ejo:** obtener el mismo resultado. Consulte Sintaxis de texto estructurado para obtener más información sobre la sintaxis de las expresiones y las asignaciones en el texto estructurado.

### Operандos

Los siguientes son operandos de la instrucción MOD.

**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
- Los miembros de un operando de estructura se sobrescriben.
- A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.

Existen reglas de conversión de datos para utilizar diversos tipos de datos numéricos en una instrucción. Consulte *Conversión de datos*.

### Diagrama de escalera

Estos son operandos del diagrama de escalera.

Operando	Tipo	Formato	Descripción
Source A	SINT INT DINT REAL	inmediato etiqueta	Valor del dividendo.
Source B	SINT INT DINT REAL	inmediato etiqueta	Valor del divisor.
Dest	SINT INT DINT REAL	etiqueta	Etiqueta para almacenar el resultado de la instrucción.

### Bloque de funciones

Estos son operandos del bloque de funciones.

Operando	Tipo	Formato	Descripción
MOD	FBD_MATH	etiqueta	Estructura de MOD

### Estructura de FBD\_MATH

Estos son los miembros de entrada y salida.

Miembros de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si el valor es falso, la instrucción no se ejecuta y las salidas no se actualizan. El valor predeterminado es verdadero.
SourceA	REAL	Valor del dividendo.
SourceB	REAL	Valor del divisor.

Miembros de salida	Tipo de datos	Descripción
EnableOut	BOOL	Indica si la instrucción se ejecutó sin fallo cuando se habilitó.
Dest	REAL	Resultado de la instrucción.

**Descripción**

El algoritmo utilizado es:

$$\text{Dest} = \text{Source A} - (\text{truncate}(\text{Source A} / \text{Source B}) * \text{Source B})$$

**Afectar a las marcas de estado matemático**

Controladores	Afectar a las marcas de estado matemático
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Condicional
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí

Consulte *Marcas de estado matemático*.

**Fallos mayores/menores**

Ocurrirá un fallo menor si:	Tipo de fallo	Código de fallo
Source B = 0	4	4

Consulte *Índice a través de matrices* para ver si hay fallos de indexación de matrices.

**Ejecución****Diagrama de escalera**

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón
La condición de entrada de reglón es verdadera	Establecer la condición de salida de reglón a condición de entrada de reglón Dest está establecido (para el resto) como se describe en la sección Descripción.
Post-escaneado	N/A

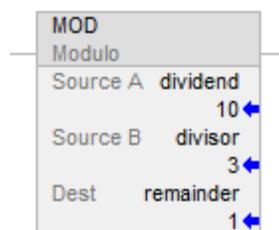
### Bloque de funciones

Condición/estado	Acción realizada
Pre-escaneado	N/A
EnableIn es falso	Establecer EnableOut a EnableIn
EnableIn es verdadero	Dest está establecido (para el resto) como se describe en la sección Descripción. Si ocurre un desbordamiento Borrar EnableOut a falso else Establecer EnableOut a verdadero
Primera ejecución de instrucción	N/A
Primer escaneado de instrucción	N/A
Post-escaneado	N/A

**Cons** Si Source B es 0, el resultado es 0 y se genera un fallo  
**ejo:** menor.

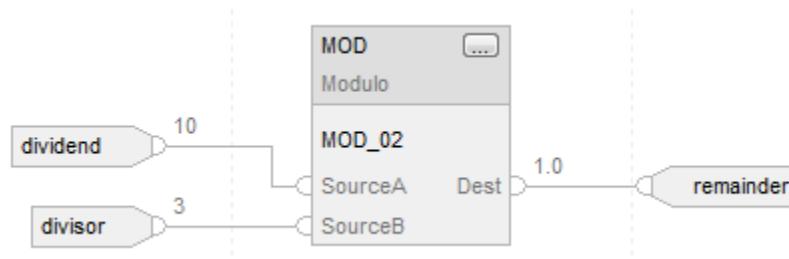
### Ejemplos

#### Diagrama de escalera



Divide el dividendo por el divisor y coloca el resto en el remainder. En este ejemplo, 3 entra en 10, tres veces, con un resto de 1.

### Bloque de funciones



### Texto estructurado

resto := dividendo MOD divisor;

### Consulte también

[Sintaxis de texto estructurado](#) en la página 955

[Índice a través de matrices](#) en la página 936

[Marcas de estado matemático](#) en la página 923

[Conversiones de datos](#) en la página 927

[Instrucciones de cálculo](#) en la página 425

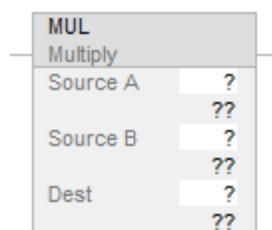
## Multiplicar (MUL)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

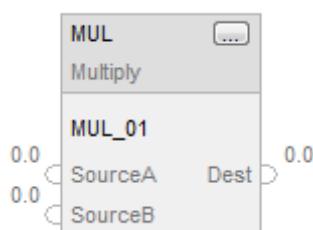
La instrucción MUL y el operador '\*' multiplican Source A por Source B.

### Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones



## Texto estructurado

Esta instrucción no está disponible en texto estructurado.

**Cons ejo:** Se utiliza el operador '\*' en una expresión para obtener el mismo resultado. Consulte Sintaxis de texto estructurado para obtener más información sobre la sintaxis de las expresiones y las asignaciones en el texto estructurado.

## Operандos

**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
- Los miembros de un operando de estructura se sobrescriben.
- A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.

Existen reglas de conversión de datos para utilizar diversos tipos de datos numéricos en una instrucción. Consulte *Conversión de datos*.

## Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción
Source A	SINT INT DINT REAL	inmediato etiqueta	Valor del multiplicando
Source B	SINT INT DINT REAL	inmediato etiqueta	Valor del multiplicador
Dest	SINT INT DINT REAL	etiqueta	Etiqueta para almacenar el resultado de la instrucción

## Bloque de funciones

Operando	Tipo de datos	Formato	Descripción
MUL	FBD_MATH	etiqueta	Estructura de MUL

**Estructura de FBD\_MATH**

Miembros de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si el valor es falso, la instrucción no se ejecuta y las salidas no se actualizan. El valor predeterminado es verdadero.
Source A	REAL	Valor del multiplicando
Source B	REAL	Valor del multiplicador

Miembros de salida	Tipo de datos	Descripción
EnableOut	BOOL	Indica si la instrucción se ejecutó sin fallos al ser habilitada.
Dest	REAL	Resultado de la instrucción

**Descripción**

Cuando está habilitada, la instrucción MUL y el operador '\*' multiplican Source A por Source B. Además, MUL almacena el resultado en Dest mientras que '\*' simplemente devuelve el resultado.

**Afectar a las marcas de estado matemático**

Controladores	Afectar a la marca de estado matemático
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Condicional
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí

Consulte *Marcas de estado matemático*.

**Fallos mayores/menores**

No es específico para esta instrucción. Consulte *Índice a través de matrices* para ver si hay fallos de indexación de matrices.

## Ejecución

### Diagrama de escalera

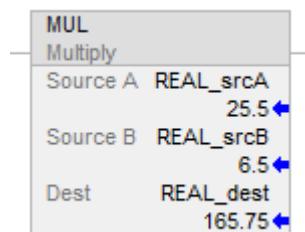
Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón
La condición de entrada de reglón es verdadera	Establecer la condición de salida de reglón a condición de entrada de reglón Dest = Source A x Source B
Post-escaneado	N/A

### Bloque de funciones

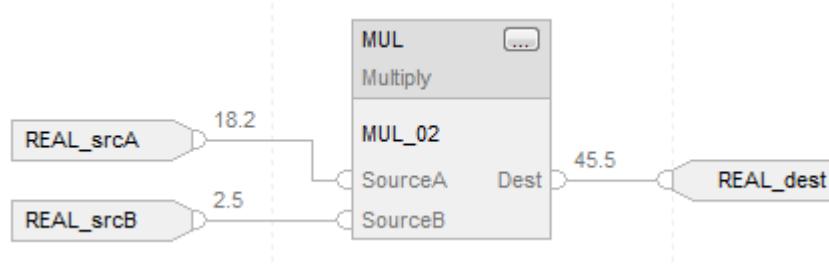
Condición/estado	Acción realizada
Pre-escaneado	N/A
EnableIn es falso	Establecer EnableOut a EnableIn
EnableIn es verdadero	Dest = SourceA x SourceB Si se produce un desbordamiento Borrar EnableOut a falso else Establecer EnableOut a verdadero
Primera ejecución de instrucción	N/A
Primer escaneado de instrucción	N/A
Post-escaneado	N/A

## Ejemplos

### Diagrama de escalera



### Bloque de funciones



### Texto estructurado

```
REAL_dest := REAL_srcA * REAL_srcB;
```

### Consulte también

[Sintaxis de texto estructurado](#) en la página 955

[Índice a través de matrices](#) en la página 936

[Marcas de estado matemático](#) en la página 923

[Conversiones de datos](#) en la página 927

[Instrucciones de cálculo](#) en la página 425

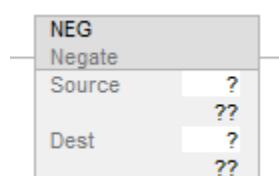
## Cambiar signo (NEG)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

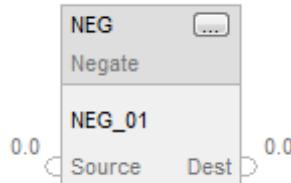
Cuando está habilitada, la instrucción NEG y el operador restan el valor de Source de cero. La instrucción almacena el resultado en Dest mientras que el operador solo devuelve el resultado. Se indicará un desbordamiento si el resultado es el valor entero negativo máximo por ejemplo, -128 para SINT, -32.768 para INT y -2.147.483.648 para DINT.

### Idiomas disponibles

### Diagrama de escalera



## Bloque de funciones



### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

**Cons** Se utiliza el operador '' en una expresión para obtener el  
**ejo:** mismo resultado. Consulte Sintaxis de texto estructurado  
para obtener más información sobre la sintaxis de las  
expresiones y las asignaciones en el texto estructurado.

### Operандos

**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
- Los miembros de un operando de estructura se sobrescriben.
- A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.

Existen reglas de conversión de datos para utilizar diversos tipos de datos numéricos en una instrucción. Consulte *Conversión de datos*.

### Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción
Source	SINT INT DINT REAL	inmediato etiqueta	Valor a negar
Dest	SINT INT DINT REAL	etiqueta	Etiqueta para almacenar el resultado de la instrucción.

**Bloque de funciones**

Operando	Tipo de datos	Formato	Descripción
NEG	FBD_MATH_ADVANCED	etiqueta	Estructura de NEG

**Estructura de FBD\_MATH**

Miembros de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si el valor es falso, la instrucción no se ejecuta y las salidas no se actualizan. El valor predeterminado es verdadero.
Source	REAL	Valor a negar.

Miembros de salida	Tipo de datos	Descripción
EnableOut	BOOL	Indica si la instrucción se ejecutó sin fallos al ser habilitada.
Dest	REAL	Resultado de la instrucción

**Afectar a las marcas de estado matemático**

Controladores	Afectar a las marcas de estado matemático
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Condicional
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí

Consulte *Marcas de estado matemático*.

### Fallos mayores/menores

No es específico para esta instrucción. Consulte *Índice a través de matrices* para ver si hay fallos de indexación de matrices.

### Ejecución

#### Diagrama de escalera

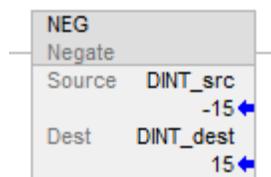
Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón.
La condición de entrada de reglón es verdadera	Establecer la condición de salida de reglón a condición de entrada de reglón. Dest = 0 - Source.
Post-escaneado	N/A

#### Bloque de funciones

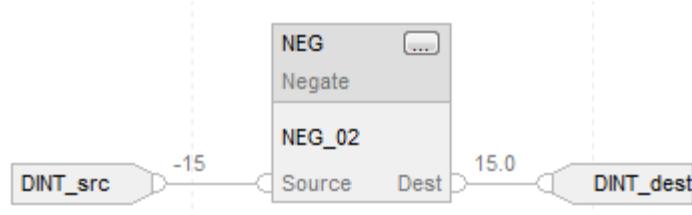
Condición/estado	Acción realizada
Pre-escaneado	N/A
EnableIn es falso	Establecer EnableOut a EnableIn.
EnableIn es verdadero	Dest = 0 - Source. Si se produce un desbordamiento Borrar EnableOut a falso else Establecer EnableOut a verdadero
Primera ejecución de instrucción	N/A
Primer escaneado de instrucción	N/A
Post-escaneado	N/A

### Ejemplos

#### Diagrama de escalera



### Bloque de funciones



### Texto estructurado

DINT\_dest := -DINT\_src;

### Consulte también

[Instrucciones de cálculo](#) en la página 425

[Sintaxis de texto estructurado](#) en la página 955

[Índice a través de matrices](#) en la página 936

[Marcas de estado matemático](#) en la página 923

[Conversiones de datos](#) en la página 927

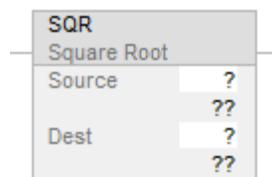
## Raíz cuadrada (SQR/SQRT)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

La instrucción SQR calcula la raíz cuadrada de Source y coloca el resultado en Dest.

### Idiomas disponibles

### Diagrama de escalera



## Bloque de funciones



## Texto estructurado

Esta instrucción no está disponible en texto estructurado.

**Cons ejo:** Se utiliza SQRT como un operador en una expresión para obtener el mismo resultado. Consulte Sintaxis de texto estructurado para obtener más información sobre la sintaxis de las expresiones y las asignaciones en el texto estructurado.

## Operандos

**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
- Los miembros de un operando de estructura se sobrescriben.
- A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.

Existen reglas de conversión de datos para utilizar diversos tipos de datos numéricos en una instrucción. Consulte *Conversión de datos*.

## Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción
Source	SINT INT DINT REAL	inmediato etiqueta	Calcula la raíz cuadrada de este valor.
Dest	SINT INT DINT REAL	etiqueta	Etiqueta para almacenar el resultado de la instrucción.

**Bloque de funciones**

Operando	Tipo de datos	Formato	Descripción
SQR D	FBD_MATH_ADVANCE	etiqueta	Estructura de SQR

**Estructura de FBD\_MATH\_ADVANCED**

Miembros de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si el valor es falso, la instrucción no se ejecuta y las salidas no se actualizan. El valor predeterminado es verdadero.
Source	REAL	Encuentra la raíz cuadrada de este valor.

Miembros de salida	Tipo de datos	Descripción
EnableOut	BOOL	Indica si la instrucción se ejecutó sin fallo cuando se habilitó.
Dest	REAL	Resultado de la instrucción.

**Descripción**

Si Dest no es REAL, la instrucción manipula la parte fraccionaria del resultado de la siguiente manera:

Si Source es:	(Para Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570) La parte fraccionaria del resultado:	Ejemplo	(Para Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.) La parte fraccionaria del resultado:	Ejemplo
SINT, INT, DINT	Truncar	Source	DINT	3
		Dest	DINT	1
REAL	Redondear	Source	REAL	3,0

<b>Si Source es:</b>	<b>(Para Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570) La parte fraccionaria del resultado:</b>	<b>Ejemplo</b>			<b>(Para Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.) La parte fraccionaria del resultado:</b>	<b>Ejemplo</b>		
		Dest	DINT	2		Dest	DINT	2

Si Source es negativo, la instrucción toma el valor absoluto de Source antes de calcular la raíz cuadrada.

Para Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570, si Source es un tipo de datos entero y Dest es un tipo de datos entero, la instrucción trunca el resultado. Por ejemplo, si el valor entero de Source es 3, el resultado es 1,732 y el valor de Dest se convierte en 1.

Si Source es un tipo de datos real y Dest es un tipo de datos entero, la instrucción redondea el resultado. Por ejemplo, si el valor real de Source es 3,0, el resultado es 1,732 y el valor de Dest se convierte en 2.

SQR se usa como operador en expresiones de diagrama de escalera; SQRT se usa como operador en las instrucciones de Texto estructurado.

#### Afectar a las marcas de estado matemático

<b>Controladores</b>	<b>Afectar a las marcas de estado matemático</b>
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Condicional
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí

Consulte *Marcas de estado matemático*.

**Fallos mayores/menores**

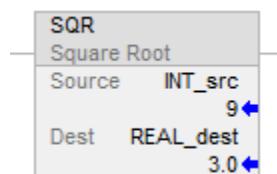
No es específico para esta instrucción. Consulte *Índice a través de matrices* para ver si hay fallos de indexación de matrices.

**Ejecución****Diagrama de escalera**

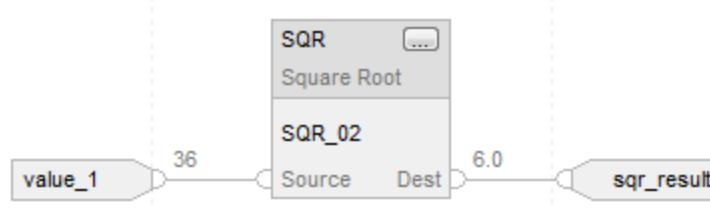
Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón.
La condición de entrada de reglón es verdadera	Establecer la condición de salida de reglón a condición de entrada de reglón. Dest = raíz cuadrada de Source.
Post-escaneado	N/A

**Bloque de funciones**

Condición/estado	Acción realizada
Pre-escaneado	N/A
EnableIn es falso	Establecer EnableOut a EnableIn.
EnableIn es verdadero	Dest. = raíz cuadrada de Source. Si se produce un desbordamiento Borrar EnableOut a falso else Establecer EnableOut a verdadero
Primera ejecución de instrucción	N/A
Primer escaneado de instrucción	N/A
Post-escaneado	N/A

**Ejemplo****Diagrama de escalera**

## Bloque de funciones



## Texto estructurado

```
REAL_dest := SQRT(INT_src);
```

### Consulte también

[Instrucciones de cálculo](#) en la página 425

[Sintaxis de texto estructurado](#) en la página 955

[Índice a través de matrices](#) en la página 936

[Conversiones de datos](#) en la página 927

[Marcas de estado matemático](#) en la página 923

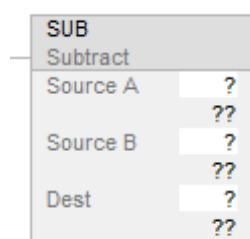
## Restar (SUB)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

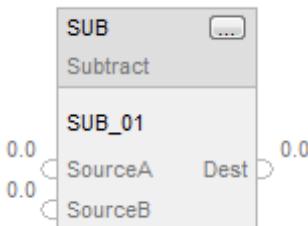
La instrucción SUB y el operador '-' resta Source B de Source A.

### Idiomas disponibles

#### Diagrama de escalera



## Bloque de funciones



### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

**Cons ejo:** Se utiliza el operador '-' en una expresión para calcular el mismo resultado. Consulte *Structured Text Syntax* para obtener más información sobre la sintaxis de las expresiones y las asignaciones en el texto estructurado.

### Operандos

**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
- Los miembros de un operando de estructura se sobrescriben.
- A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.

Existen reglas de conversión de datos para utilizar diversos tipos de datos numéricos en una instrucción. Consulte *Conversión de datos*.

### Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción
Source A	SINT INT DINT REAL	inmediato etiqueta	Valor al que restarle el valor de Source B.
Source B	SINT INT DINT REAL	inmediato etiqueta	Valor a restar de Source A
Destination	SINT INT DINT REAL	etiqueta	Etiqueta para almacenar el resultado de la instrucción

### Bloque de funciones

Operando	Tipo de datos	Formato	Descripción
SUB	FBD_MATH	etiqueta	Estructura de SUB

### Estructura de FBD\_MATH

Miembros de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si el valor es falso, la instrucción no se ejecuta y las salidas no se actualizan. El valor predeterminado es verdadero.
Source A	REAL	Valor al que restarle el valor de SourceB.
Source B	REAL	Valor a restar de SourceA.

Miembros de salida	Tipo de datos	Descripción
EnableOut	BOOL	Indica si la instrucción se ejecutó sin fallos al ser habilitada.
Dest	REAL	Resultado de la instrucción.

### Descripción

Cuando está habilitada, la instrucción SUB y el operador '-' resta Source B de Source A. Además, SUB almacena el resultado en Dest mientras que '-' simplemente devuelve el resultado.

### Afectar a las marcas de estado matemático

Controladores	Afectar a las marcas de estado matemático
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Condicional

Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí
---	----

Consulte *Marcas de estado matemático*.

#### Fallos mayores/menores

No es específico para esta instrucción. Consulte *Índice a través de matrices* para ver si hay fallos de indexación de matrices.

#### Ejecución

##### Diagrama de escalera

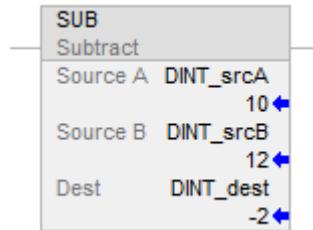
Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón
La condición de entrada de reglón es verdadera	Establecer la condición de salida de reglón a condición de entrada de reglón Dest = Source A - Source B
Post-escaneado	N/A

##### Bloque de funciones

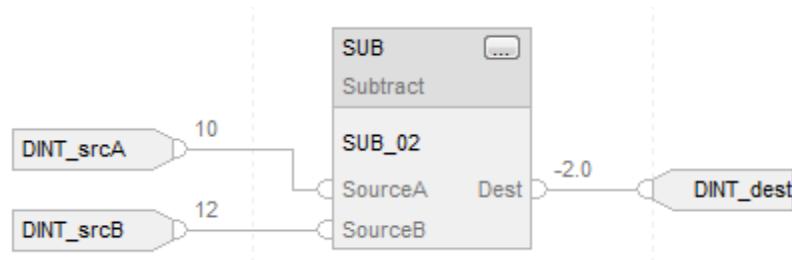
Condición/estado	Acción realizada
Pre-escaneado	N/A
EnableIn es falso	Establecer EnableOut a EnableIn
EnableIn es verdadero	Dest = SourceA - SourceB Si se produce un desbordamiento Borrar EnableOut a falso else Establecer EnableOut a verdadero
Primera ejecución de instrucción	N/A
Primer escaneado de instrucción	N/A
Post-escaneado	N/A

## Ejemplos

### Diagrama de escalera



### Bloque de funciones



### Texto estructurado

```
DINT_dest := DINT_srcA - DINT_srcB;
```

### Consulte también

[Índice a través de matrices](#) en la página 936

[Marcas de estado matemático](#) en la página 923

[Conversiones de datos](#) en la página 927

[Sintaxis de texto estructurado](#) en la página 955

[Instrucciones de cálculo](#) en la página 425



# Instrucciones de movimiento/lógicas

## Instrucciones de movimiento/lógicas

Las instrucciones de movimiento modifican y mueven los bits.

### Instrucciones disponibles

#### Diagrama de escalera

<a href="#">MOV</a>	<a href="#">MVM</a>	<a href="#">AND</a>	<a href="#">OR</a>	<a href="#">XOR</a>	<a href="#">NOT</a>	<a href="#">SWPB</a>	<a href="#">CLR</a>	<a href="#">BTD</a>
---------------------	---------------------	---------------------	--------------------	---------------------	---------------------	----------------------	---------------------	---------------------

#### Bloque de funciones

<a href="#">MVMT en la página 512</a>	<a href="#">AND</a>	<a href="#">OR</a>	<a href="#">XOR</a>	<a href="#">NOT</a>	<a href="#">BTDT</a>	<a href="#">BAND</a>	<a href="#">BXOR</a>
---	---------------------	--------------------	---------------------	---------------------	----------------------	----------------------	----------------------

<a href="#">BNOT</a>	<a href="#">BOR</a>
----------------------	---------------------

#### Texto estructurado

<a href="#">MVMT en la página 512</a>	<a href="#">SWPB</a>	<a href="#">BTDT</a>
---	----------------------	----------------------

Si desea:	Utilice esta instrucción:
Copiar un valor o mover cadenas	MOV
Copiar una parte específica de un entero	MVM
Copiar una parte específica de un entero en un bloque de funciones	MVMT
Mover bits dentro de un entero o entre enteros	BTD

Mover bits dentro de un entero o entre enteros en un bloque de funciones	BTDT
Borrar un valor	CLR
Reordenar los bytes de una etiqueta INT, DINT o REAL	SWPB

Las instrucciones lógicas realizan operaciones lógicas en bits.

Si desea:	Utilice esta instrucción:
Realizar una operación Y a nivel de bits	AND
Realizar una operación O a nivel de bits	OR
Realizar una operación O exclusivo a nivel de bits	XOR
Realizar una operación No a nivel de bits	NOT

Puede mezclar tipos de datos, pero puede que se produzca una pérdida de precisión y errores de redondeo y la instrucción tarda más en ejecutarse. Compruebe el bit S:V para ver si el resultado se ha truncado.

Los tipos de datos que aparecen en **negrita** indican tipos de datos óptimos. Una instrucción se ejecutará a la máxima velocidad y la menor cantidad posible de memoria si todos los parámetros emplean el mismo tipo de datos óptimo, normalmente DINT o REAL.

Una instrucción de movimiento/lógica se ejecuta una vez cada vez que se escanea la instrucción, siempre que el valor de condición de entrada de reglón sea verdadero. Si quiere que solo se evalúe una vez la expresión, utilice cualquier instrucción de un impulso para desencadenar la instrucción de movimiento/lógica.

#### Consulte también

[Instrucciones de conversión matemática](#) en la página 805

[Instrucciones de entrada y salida](#) en la página 223

[Instrucciones Para/Dividir](#) en la página 705

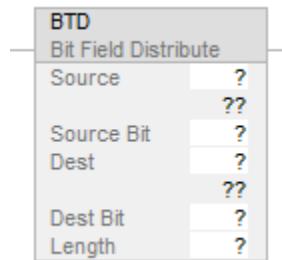
## Distribuir campo de bits con receptor (BTD)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción BTD copia los bits especificados del Source, cambia los bits a la posición correspondiente y escribe los bits en Destination.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

#### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

#### Operандos

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción. Consulte *Conversión de datos*.

#### Diagrama de escalera

Operando	Tipo	Formato	Descripción
Source	SINT INT DINT	inmediato etiqueta	Etiqueta que contiene los bits a mover
Source bit	DINT	inmediato (0-31)	Número del bit (número de bit más bajo) desde el que comenzar el mover Debe estar dentro del rango válido para el tipo de datos de Source.
Destination	SINT INT DINT	etiqueta	Etiqueta a la cual mover los bits
Destination bit	DINT	inmediato (0-31)	El número del bit al que se deberían mover los datos debe estar dentro del rango válido para el tipo de datos de Destination.

Length	DINT	inmediato (1-32)	Número de bits a mover
--------	------	------------------	------------------------

### Descripción

Cuando está habilitada, la instrucción BTD copia un grupo de bits de Source a Destination. El grupo de bits se identifica mediante el Source bit (número de bit más bajo de Source) y Length (número de bits a copiar). El Destination bit identifica el número de bit más bajo por el que comenzar en Destinación. Source permanece sin cambios.

Si la longitud del campo de bits se extiende más allá de Destination, la instrucción no guarda los bits adicionales. Cualquier bit adicional no se ajusta a la siguiente palabra.

Una etiqueta SINT o INT se convierte al valor DINT mediante relleno de ceros

### Afectar a las marcas de estado matemático

No

### Fallos mayores/menores

No es específico para esta instrucción. Consulte los *Atributos comunes* para fallos relacionados con el operando.

### Ejecución

#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa.	N/A
La condición de entrada de reglón es verdadera.	La instrucción copia y desplaza los bits de Source a Destination.
Post-escaneado	N/A

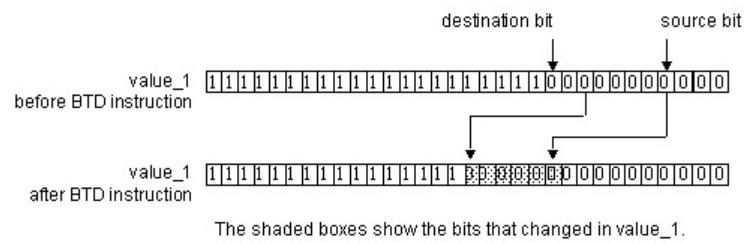
## Ejemplos

### Ejemplo 1

#### Diagrama de escalera

BTD	
Bit Field Distribute	
Source	value_1 2#1111_1111_1111_1111_1111_1000_0000_0000
Source Bit	3
Dest	value_1 2#1111_1111_1111_1111_1111_1000_0000_0000
Dest Bit	10
Length	6

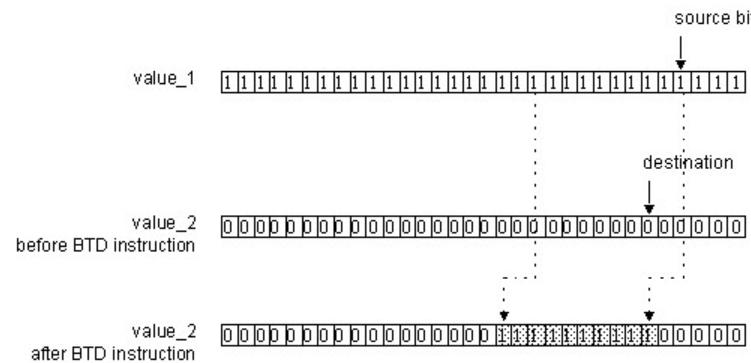
Cuando se habilita, la instrucción BTD mueve los bits dentro de value\_1.



### Ejemplo 2

BTD	
Bit Field Distribute	
Source	value_1 2#1111_1111_1111_1111_1111_1111_1111_1111
Source Bit	3
Dest	value_2 2#0000_0000_0000_0000_0000_0000_0000_0000
Dest Bit	5
Length	10

Cuando se habilita, la instrucción BTD mueve 10 bits de value\_1 a value\_2.



The shaded boxes show the bits that changed in value\_2.

### Consulte también

[Instrucciones de movimiento](#) en la página 467

[Borrar \(CLR\)](#) en la página 507

[Atributos comunes](#) en la página 923

[Conversiones de datos](#) en la página 927

[Mover con máscara \(MVM\)](#) en la página 509

## Distribuir campo de bits con receptor (BTDT)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

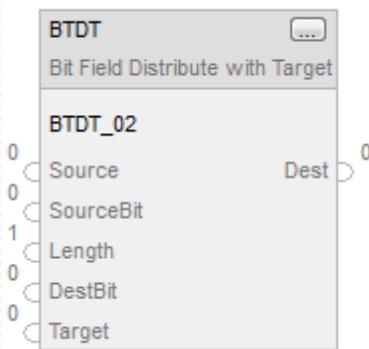
La instrucción BTDT copia primero el Target a Destination. A continuación, la instrucción copia los bits especificados del Source, cambia los bits a la posición correspondiente y escribe los bits en Destination. Target y Source se quedan igual.

### Idiomas disponibles

### Diagrama de escalera

Esta instrucción no está disponible en un diagrama de escalera.

### Bloque de funciones



### Texto estructurado

```
BTDT(BTDT_tag);
```

### Operando

### Bloque de funciones

Operando	Tipo	Formato	Descripción
BTDT tag	FBD_BIT_FIELD_DISTRIBUTE	estructura	Estructura de BTDT

### Texto estructurado

Parámetro de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Si no está activada, la instrucción no se ejecuta y las salidas no se actualizan. Si está activado, la instrucción se ejecuta. Está establecido de forma predeterminada.
Source	DINT	Valor de entrada que contiene los bits a mover a Destination. Válido = cualquier entero
SourceBit	DINT	La posición del bit en Source (número de bit más bajo desde el que empezar el desplazamiento). Válido = 0-31
Length	DINT	Número de bits a mover. Válido= 1-32
DestBit	DINT	La posición del bit en Dest (número de bit más bajo desde el que empezar a copiar bits). Válido = 0-31
Target	DINT	Valor de entrada a mover a Dest antes de mover bits de Source. Válido = cualquier entero

Parámetro de salida	Tipo de datos	Descripción
EnableOut	BOOL	Indica si la instrucción está habilitada.
Dest	DINT	Resultado de la operación de movimiento de bits.

Consulte *Sintaxis de texto estructurado* para obtener información sobre la sintaxis de las expresiones dentro de texto estructurado.

### Descripción

Cuando es verdadero, la instrucción BTDT copia primero el valor de Target en Destination y copia un grupo de bits de Source a Destination. El grupo de bits se identifica mediante el Source bit (número de bit más bajo del grupo) y Length (número de bits a copiar). El Destination bit identifica el número de bit más bajo por el que comenzar en Destination. Source y Target se quedan igual.

Si la longitud del campo de bits se extiende más allá de Destination, la instrucción no guarda los bits adicionales. Cualquier bit adicional no se ajusta a la siguiente palabra.

### Afectar a las marcas de estado matemático

Controladores	Marca de estado matemático afectado
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Sí
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	No

Consulte *Marcas de estado matemático*.

### Fallos mayores/menores

No es específico para esta instrucción. Consulte los *Atributos comunes* para fallos relacionados con el operando.

## Ejecución

## Bloque de funciones

<b>Condición/estado</b>	<b>Acción realizada</b>
Pre-escaneado	Los bits EnableIn y EnableOut se borran a falsos.
Tag.EnableIn es falso	Los bits EnableIn y EnableOut se borran a falsos.
Tag.EnableIn es verdadero	Los bits EnableIn y EnableOut se establecen en verdaderos. La instrucción se ejecuta.
Primera ejecución de instrucción	N/A
Primer escaneado de instrucción	N/A
Post-escaneado	Los bits EnableIn y EnableOut se borran a falsos.

## **Texto estructurado**

<b>Condición/estado</b>	<b>Acción realizada</b>
Pre-escaneado	Consultar Pre-escaneado en la tabla Bloque de funciones.
Ejecución normal	Consultar Tag.EnableIn es verdadero en la tabla Bloque de funciones.
Post-escaneado	Consultar Post-escaneado en la tabla Bloque de funciones.

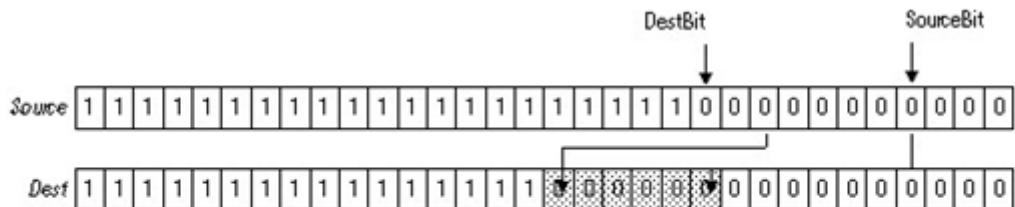
## Ejemplo

## Paso 1

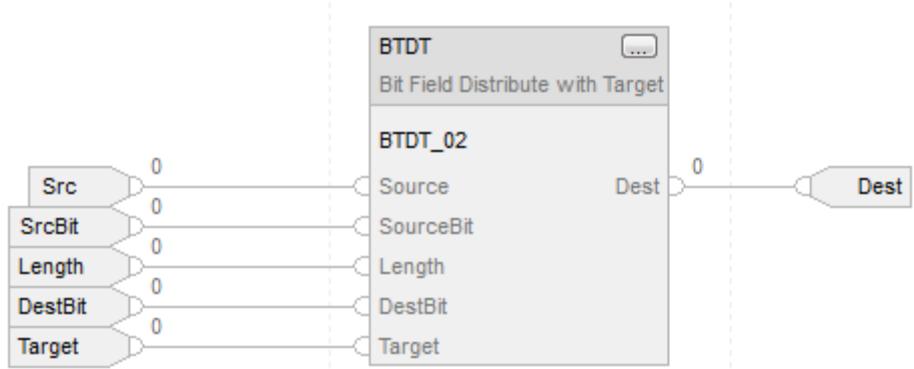
El controlador copia el valor de Target en Dest.

**Paso 2**

SourceBit y Length especifican cuál bits de Source se deben copiar en Dest. Comenzando por DestBit, Source y Target se quedan igual.



The shaded boxes show the bits that changed.

**Bloque de funciones****Texto estructurado**

```

BTDT_01.Source := sourceSTX;
BTDT_01.SourceBit := source_bitSTX;
BTDT_01.Length := LengthSTX;
BTDT_01.DestBit := dest_bitSTX;
BTDT_01.Target := TargetSTX;
BTDT(BTDT_01);
distributed_value := BTDT_01.Dest;

```

**Consulte también**

[Atributos comunes](#) en la página 923

[Marcas de estado matemático](#) en la página 923

[Sintaxis de texto estructurado](#) en la página 955

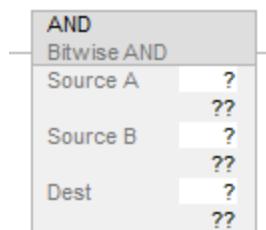
## Y a nivel de bits (AND)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

La instrucción AND realiza una operación Y a nivel de bits utilizando los bits de Source A y Source B y coloca el resultado en Dest.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones



#### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

**Cons** Se utiliza el operador AND (o '&') en una expresión para  
**ejo:** obtener el mismo resultado. Consulte *Structured Text Syntax* para obtener más información sobre la sintaxis de las expresiones y las asignaciones en el texto estructurado.

## Operандos

**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
- Los miembros de un operando de estructura se sobrescriben.
- A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.

Existen reglas de conversión de datos para utilizar diversos tipos de datos numéricos en una instrucción. Consulte *Conversión de datos*.

## Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción
Source A	SINT INT DINT REAL	inmediato etiqueta	Valor a AND con Source B. <b>Consejo:</b> Si el tipo de datos es REAL, el valor de entrada se convertirá a DINT, lo que puede causar un desbordamiento.
Source B	SINT INT DINT REAL	inmediato etiqueta	Valor a AND con Source A. <b>Consejo:</b> Si el tipo de datos es REAL, el valor de entrada se convertirá a DINT, lo que puede causar un desbordamiento.
Dest	SINT INT DINT REAL	etiqueta	Etiqueta para almacenar el resultado de la instrucción. <b>Consejo:</b> Si el tipo de datos es REAL, el valor DINT resultante se convertirá en REAL.

**Cons** La instrucción AND funciona con DINT. Los operandos  
**ejo:** de origen INT o SINT se convierten a DINT rellenando los bits superiores con ceros.

## Bloque de funciones

Operando	Tipo de datos	Formato	Descripción
AND	FBD_LOGICAL	etiqueta	Estructura de AND

### Estructura de FBD\_LOGICAL

Miembros de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si el valor es falso, la instrucción no se ejecuta y las salidas no se actualizan. El valor predeterminado es verdadero.
SourceA	DINT	Valor a AND con SourceB.
SourceB	DINT	Valor a AND con SourceA.

Miembros de salida	Tipo de datos	Descripción
EnableOut	BOOL	Indica si la instrucción se ejecutó sin fallo cuando se habilitó.
Dest	DINT	Resultado de la instrucción.

### Descripción

Cuando está habilitada, la instrucción evalúa la operación Y a nivel de bits: Dest = A AND B

Si el bit en Source A es:	Y el bit en Source B es:	El bit en Dest es:
0	0	0
0	1	0
1	0	0
1	1	1

### Afectar a las marcas de estado matemático

Controladores	Afectar a las marcas de estado matemático
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Condicional
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí

Consulte *Marcas de estado matemático*.

**Fallos mayores/menores**

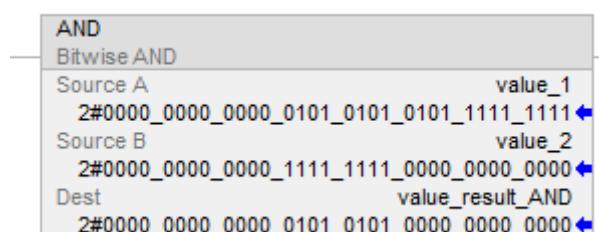
No es específico para esta instrucción. Consulte *Índice a través de matrices* para ver si hay fallos de indexación de matrices.

**Ejecución****Diagrama de escalera**

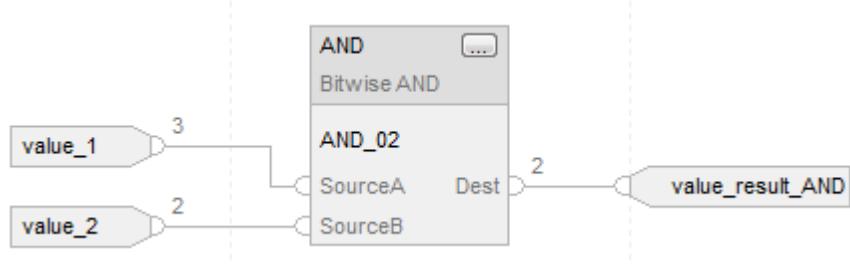
Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón
La condición de entrada de reglón es verdadera	Establecer la condición de salida de reglón a condición de entrada de reglón Dest se establece como se describe en la sección Descripción.
Post-escaneado	N/A

**Bloque de funciones**

Condición/estado	Acción realizada
Pre-escaneado	N/A
EnableIn es falso	Establecer EnableOut a EnableIn
EnableIn es verdadero	Establecer EnableOut a EnableIn Dest se establece como se describe en la sección Descripción.
Primera ejecución de instrucción	N/A
Primer escaneado de instrucción	N/A
Post-escaneado	N/A

**Ejemplos****Diagrama de escalera**

## Bloque de funciones



## Texto estructurado

```
value_result_and := value_1 AND value_2;
```

## Consulte también

[Sintaxis de texto estructurado](#) en la página 955

[Índice a través de matrices](#) en la página 936

[Marcas de estado matemático](#) en la página 923

[Conversiones de datos](#) en la página 927

[Instrucciones de movimiento](#) en la página 467

## O exclusivo a nivel de bits (XOR)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

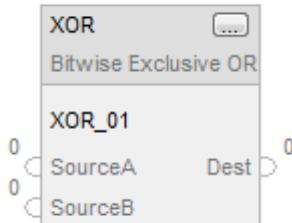
La instrucción XOR realiza una operación O exclusivo a nivel de bits usando los bits de Source A y de Source B y coloca el resultado en Dest.

## Idiomas disponibles

### Diagrama de escalera



## Bloque de funciones



## Texto estructurado

Esta instrucción no está disponible en texto estructurado.

**Consejo:** Se utiliza XOR como un operador en una expresión para calcular el mismo resultado. Consulte *Structured Text Syntax* para obtener más información sobre la sintaxis de las expresiones y las asignaciones en el texto estructurado.

## Operандos

**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
- Los miembros de un operando de estructura se sobrescriben.
- A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.

Existen reglas de conversión de datos para utilizar diversos tipos de datos numéricos en una instrucción. Consulte *Conversión de datos*.

## Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción
Source A	SINT INT DINT REAL	inmediato etiqueta	Valor a XOR con Source B.  <b>Consejo:</b> Si el tipo de datos es REAL, el valor de entrada se convertirá a DINT (lo que puede causar un desbordamiento).

Source B	SINT INT DINT REAL	inmediato etiqueta	Valor a XOR con Source A.  <b>Consejo:</b> Si el tipo de datos es REAL, el valor de entrada se convertirá a DINT (lo que puede causar un desbordamiento).
Dest	SINT INT DINT REAL	etiqueta	Etiqueta para almacenar el resultado de la instrucción.  <b>Consejo:</b> Si el tipo es REAL, el valor DINT resultante se convertirá en REAL.

**Cons** La instrucción XOR utiliza valores DINT. Los operandos **ejo:** de origen INT o SINT se convierten a DINT rellenando los bits superiores con ceros.

### Bloque de funciones

Operando	Tipo de datos	Formato	Descripción
XOR	FBD_LOGICAL	etiqueta	Estructura de XOR

### Estructura de FBD\_LOGICAL

Miembros de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si el valor es falso, la instrucción no se ejecuta y las salidas no se actualizan. El valor predeterminado es verdadero.
SourceA	DINT	Valor al que aplicar XOR con SourceB.
SourceB	DINT	Valor al que aplicar XOR con SourceA.

Miembros de salida	Tipo de datos	Descripción
EnableOut	BOOL	Indica si la instrucción se ejecutó sin fallos al ser habilitada.
Dest	DINT	Resultado de la instrucción.

## Descripción

Si está habilitada, la instrucción evalúa la operación O exclusivo a nivel de bits:

$$\text{Dest} = \text{Source A XOR Source B}$$

Si el bit de Source A es:	Y el bit en Source B es:	El bit de Dest es:
0	0	0
0	1	1
1	0	1
1	1	0

## Afectar a las marcas de estado matemático

Controladores	Afectar a las marcas de estado matemático
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Condicional
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí

Consulte *Marcas de estado matemático*.

## Fallos mayores/menores

No es específico para esta instrucción. Consulte *Índice a través de matrices* para ver si hay fallos de indexación de matrices.

## Ejecución

### Diagrama de escalera

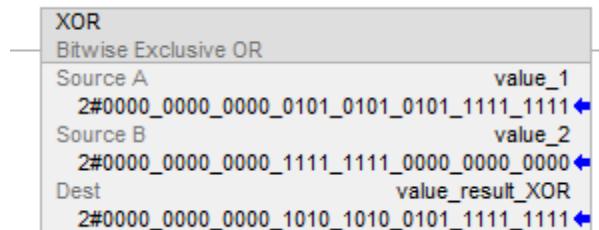
Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón
La condición de entrada de reglón es verdadera	Establecer la condición de salida de reglón a condición de entrada de reglón Dest se establece como se describe en la sección Descripción.
Post-escaneado	N/A

### Bloque de funciones

Condición/estado	Acción realizada
Pre-escaneado	N/A
EnableIn es falso	Establecer EnableOut a EnableIn
EnableIn es verdadero	Establecer EnableOut a EnableIn Dest se establece como se describe en la sección Descripción.
Primera ejecución de instrucción	N/A
Primer escaneado de instrucción	N/A
Post-escaneado	N/A

### Ejemplos

#### Diagrama de escalera



### Bloque de funciones



### Texto estructurado

```
value_result_XOR := value_1 XOR value_2;
```

### Consulte también

[Sintaxis de texto estructurado](#) en la página 955

[Índice a través de matrices](#) en la página 936

[Marcas de estado matemático](#) en la página 923

[Instrucciones de movimiento](#) en la página 467

[Conversiones de datos](#) en la página 927

## NO a nivel de bits (NOT)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

La instrucción NOT realiza una inversión a nivel de bits de Source y coloca el resultado en Dest.

### Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones



### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

**Consejo:** Utiliza NOT como un operador en una expresión para calcular el mismo resultado. Consulte *Structured Text Syntax* para obtener más información sobre la sintaxis de las expresiones y las asignaciones en el texto estructurado.

## Operandos

**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
- Los miembros de un operando de estructura se sobrescriben.
- A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.

Existen reglas de conversión de datos para utilizar diversos tipos de datos numéricos en una instrucción. Consulte *Conversión de datos*

## Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción
Source	SINT INT DINT REAL	inmediato etiqueta	Valor sobre el cual se aplicará NO  <b>Consejo:</b> Si el tipo de datos es REAL, el valor de entrada se convertirá a DINT (lo que puede causar un desbordamiento).
Dest	SINT INT DINT REAL	etiqueta	Etiqueta para almacenar el resultado de la instrucción.  <b>Consejo:</b> Si el tipo es REAL, el valor DINT resultante se convertirá en REAL.

**Cons** La instrucción NOT funciona con DINT. Los operandos  
**ejo:** de origen INT o SINT se convierten a DINT rellenando  
los bits superiores con ceros.

### Bloque de funciones

Operand o	Tipo de datos	Formato	Descripción
NOT	FBD_CONVERT	etiqueta	Estructura de NOT

### Estructura de FBD\_CONVERT

Miembros de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si el valor es falso, la instrucción no se ejecuta y las salidas no se actualizan. El valor predeterminado es verdadero.
Source	DINT	Valor sobre el cual se aplicará NO

Miembros de salida	Tipo de datos	Descripción
EnableOut	BOOL	Indica si la instrucción se ejecutó sin fallos al ser habilitada.
Dest	DINT	Resultado de la instrucción

### Descripción

Si está habilitada, la instrucción evalúa la operación NO a nivel de bits:

$$\text{Dest} = \text{NOT Source}$$

Si el bit en Source es:	El bit en Dest es:
0	1
1	0

### Afectar a las marcas de estado matemático

Controladores	Afectar a las marcas de estado matemático
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Condicional
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí

Consulte *Marcas de estado matemático*.

### Fallos mayores/menores

No es específico para esta instrucción. Consulte *Índice a través de matrices* para ver si hay fallos de indexación de matrices.

### Ejecución

#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón
La condición de entrada de reglón es verdadera	Establecer la condición de salida de reglón a condición de entrada de reglón Dest se establece como se describe en la sección Descripción.
Post-escaneado	N/A

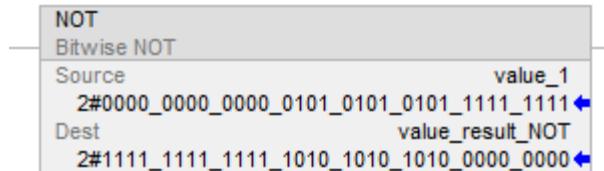
#### Bloque de funciones

Condición/estado	Acción realizada
Pre-escaneado	N/A
EnableIn es falso	Establecer EnableOut a EnableIn
EnableIn es verdadero	Establecer EnableOut a EnableIn Dest se establece como se describe en la sección Descripción.
Primera ejecución de instrucción	N/A
Primer escaneado de instrucción	N/A

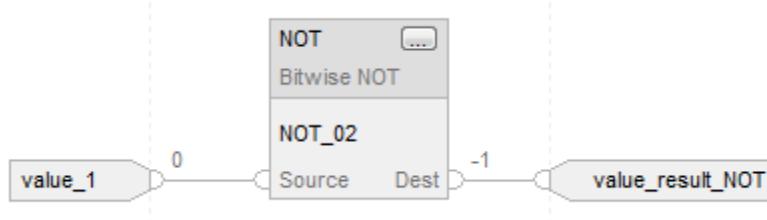
Post-escaneado	N/A
----------------	-----

## Ejemplos

### Diagrama de escalera



### Bloque de funciones



### Texto estructurado

```
value_result_not := NOT value_1;
```

### Consulte también

[Sintaxis de texto estructurado](#) en la página 955

[Índice a través de matrices](#) en la página 936

[Marcas de estado matemático](#) en la página 923

[Conversiones de datos](#) en la página 927

[Instrucciones de movimiento](#) en la página 467

## O a nivel de bits (OR)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

La instrucción OR realiza una operación O a nivel de bits utilizando los bits de Source A y Source B y coloca el resultado en Dest.

## Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones



### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

**Cons** Utilice OR como un operador en una expresión para  
**ejo:** calcular el mismo resultado. Consulte *Structured Text Syntax* para obtener más información sobre la sintaxis de las expresiones y las asignaciones en el texto estructurado.

### Operандos

**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
- Los miembros de un operando de estructura se sobrescriben.
- A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.

Existen reglas de conversión de datos para utilizar diversos tipos de datos numéricos en una instrucción. Consulte *Conversión de datos*.

### Diagrama de escalera

Operando	Tipo	Formato	Descripción
Source A	SINT INT DINT REAL	inmediato etiqueta	Valor a OR con Source B. <b>Consejo:</b> Si el tipo de datos es REAL, el valor de entrada se convertirá a DINT (lo que puede causar un desbordamiento).
Source B	SINT INT DINT REAL	inmediato etiqueta	Valor a OR con Source A. <b>Consejo:</b> Si el tipo de datos es REAL, el valor de entrada se convertirá a DINT (lo que puede causar un desbordamiento).
Dest	SINT INT DINT REAL	etiqueta	Etiqueta para almacenar el resultado de la instrucción. <b>Consejo:</b> Si el tipo es REAL, el valor DINT resultante se convertirá en REAL.

**Cons** La instrucción OR utiliza valores DINT. Los operandos **ejo:** de origen INT o SINT se convierten a DINT rellenando los bits superiores con ceros.

### Bloque de funciones

Operando	Tipo	Formato	Descripción
OR	FBD_LOGICA L	etiqueta	Estructura de OR

### Estructura de FBD\_LOGICAL

Miembros de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si el valor es falso, la instrucción no se ejecuta y las salidas no se actualizan. El valor predeterminado es verdadero.
SourceA	DINT	Valor a OR con SourceB.
SourceB	DINT	Valor a OR con SourceA.

Miembros de salida	Tipo de datos	Descripción
EnableOut	BOOL	Indica si la instrucción se ejecutó sin fallo cuando se habilitó.

Miembros de salida	Tipo de datos	Descripción
Dest	DINT	Resultado de la instrucción.

### Descripción

Si está habilitada, la instrucción evalúa la operación O a nivel de bits:

$$\text{Dest} = \text{Source A} \text{ O } \text{Source B}$$

Si el bit de Source A es:	Y el bit en Source B es:	El bit de Dest es:
0	0	0
0	1	1
1	0	1
1	1	1

### Afectar a las marcas de estado matemático

Controladores	Afectar a la marca de estado matemático
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Condicional
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí

Consulte *Marcas de estado matemático*.

### Fallos mayores/menores

No es específico para esta instrucción. Consulte *Índice a través de matrices* para ver si hay fallos de indexación de matrices.

### Ejecución

#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón

La condición de entrada de reglón es verdadera	Establecer la condición de salida de reglón a condición de entrada de reglón Dest se establece como se describe en la sección Descripción.
Post-escaneado	N/A

### Bloque de funciones

Condición/estado	Acción realizada
Pre-escaneado	N/A
EnableIn es falso	Establecer EnableOut a EnableIn
EnableIn es verdadero	Establecer EnableOut a EnableIn Dest se establece como se describe en la sección Descripción.
Primera ejecución de instrucción	N/A
Primer escaneado de instrucción	N/A
Post-escaneado	N/A

### Ejemplos

#### Diagrama de escalera

```

OR
Bitwise Inclusive OR
Source A           value_1
2#0000_0000_0101_0101_0101_1111_1111
Source B           value_2
2#0000_0000_1111_1111_0000_0000_0000
Dest              value_result_or
2#0000_0000_1111_1111_0101_1111_1111

```

### Bloque de funciones



### Texto estructurado

```
value_result_or := value_1 OR value_2;
```

### Consulte también

[Sintaxis de texto estructurado](#) en la página 955

[Índice a través de matrices](#) en la página 936

[Marcas de estado matemático](#) en la página 923

[Conversiones de datos](#) en la página 927

[Instrucciones de movimiento](#) en la página 467

## Y booleano (BAND)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

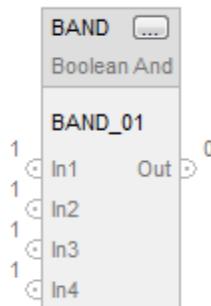
La instrucción BAND realiza una operación AND de forma lógica como hasta ocho entradas booleanas. Para realizar una operación AND bit a bit, consulte *Y a nivel de bits (AND)*.

### Idiomas disponibles

### Diagrama de escalera

Esta instrucción no está disponible en el diagrama de escalera.

### Bloque de funciones



### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

## Operandos

### Bloque de funciones

Operando	Tipo	Formato	Descripción
BAND tag	FBD_BOOLEAN_AN D	estructura	Estructura BAND

### Estructura de FBD\_BOOLEAN\_AND

Parámetro de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si se borra, la instrucción no se ejecuta y las salidas no se actualizan. Está establecido de forma predeterminada.
In1	BOOL	Primera entrada booleana. Se establece en 1 en la primera descarga.
In2	BOOL	Segunda entrada booleana. Se establece en 1 en la primera descarga.
In3	BOOL	Tercera entrada booleana. Se establece en 1 en la primera descarga.
In4	BOOL	Cuarta entrada booleana. Se establece en 1 en la primera descarga.
In5	BOOL	Quinta entrada booleana. Se establece en 1 en la primera descarga.
In6	BOOL	Sexta entrada booleana. Se establece en 1 en la primera descarga.
In7	BOOL	Séptima entrada booleana. Se establece en 1 en la primera descarga.
In8	BOOL	Octava entrada booleana. Se establece en 1 en la primera descarga.

Parámetro de salida	Tipo de datos	Descripción
EnableOut	BOOL	Indica si la instrucción está habilitada.
Out	BOOL	La salida de la instrucción.

## Descripción

La instrucción BAND realiza una operación AND como hasta en ocho entradas booleanas. Si no se usa una entrada, se establece de forma predeterminada (1).

$\text{Out} = \text{In1 AND In2 AND In3 AND In4 AND In5 AND In6 AND In7 AND In8}$

---

**Importante:** Si durante una edición elimina un cable de entrada de la instrucción BAND, debe asegurarse de que se haya establecido (1) la entrada asociada.

---

## Afectar a las marcas de estado matemático

No

## Fallos mayores/menores

No es específico para esta instrucción. Consulte los *Atributos comunes* para fallos relacionados con el operando.

## Ejecución

### Bloque de funciones

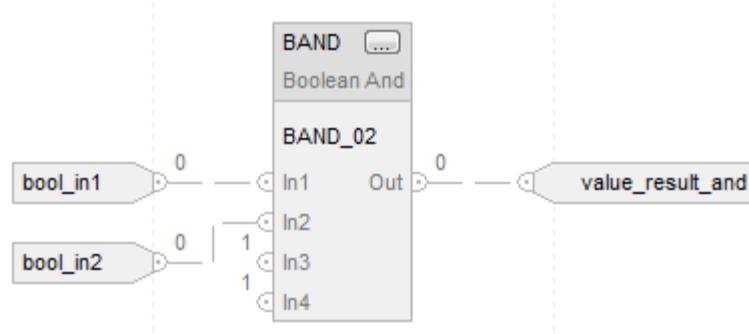
Condición/estado	Acción realizada
Pre-escaneado	Los bits EnableIn y EnableOut se borran a falsos.
Tag.EnableIn es falso	Los bits EnableIn y EnableOut se borran a falsos.
Tag.EnableIn es verdadero	Los bits EnableIn y EnableOut se establecen en verdaderos. La instrucción se ejecuta
Primera ejecución de instrucción	N/A
Primer escaneado de instrucción	N/A
Post-escaneado	Los bits EnableIn y EnableOut se borran a falsos.

## Ejemplo

Este ejemplo realiza una operación AND con bool\_in1 y bool\_in2 y coloca el resultado en value\_result\_and.

Si bool_In1 es:	Si bool_In2 es:	Entonces, value_result_and es:
0	0	0

0	1	0
1	0	0
1	1	1

**Bloque de funciones****Consulte también**

[Y a nivel de bits \(AND\)](#) en la página 477

[Atributos comunes](#) en la página 923

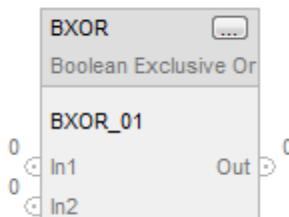
## O exclusivo booleano (BXOR)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción BXOR realiza una operación OR exclusivo con dos entradas booleanas.

**Idiomas disponibles****Diagrama de escalera**

Esta instrucción no está disponible en el diagrama de escalera.

**Bloque de funciones**

### Texto estructurado

Esta instrucción no está disponible para texto estructurado.

### Operandos

#### Bloque de funciones

Operando	Tipo	Formato	Descripción
BXOR tag	FBD_BOOLEAN_XOR	Estructura	Estructura BXOR

#### Estructura de FBD\_BOOLEAN\_XOR

Parámetro de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si se borra, la instrucción no se ejecuta y las salidas no se actualizan. Está establecido de forma predeterminada.
In1	BOOL	Primera entrada booleana. Su valor predeterminado es borrado.
In2	BOOL	Segunda entrada booleana. Su valor predeterminado es borrado.

Parámetro de salida	Tipo de datos	Descripción
EnableOut	BOOL	Indica si la instrucción está habilitada.
Out	BOOL	La salida de la instrucción.

### Descripción

La instrucción BXOR realiza una operación OR exclusivo con dos entradas booleanas.

$$\text{Out} = \text{In1 XOR In2}$$

### Afectar a las marcas de estado matemático

No

### Fallos mayores/menores

No es específico para esta instrucción. Consulte los *Atributos comunes* para fallos relacionados con el operando.

### Ejecución

#### Bloque de funciones

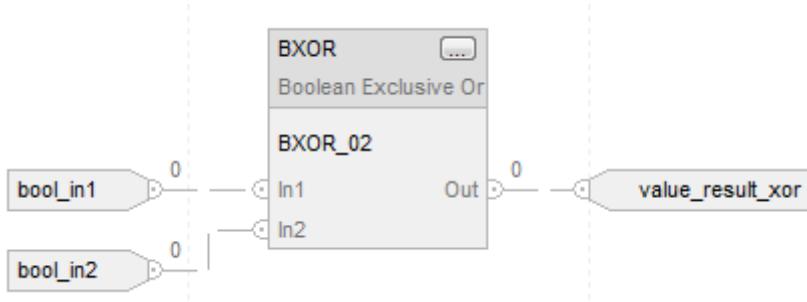
Condición/estado	Acción realizada
Pre-escaneado	Los bits EnableIn y EnableOut se borran a falsos.
Tag.EnableIn es falso	Los bits EnableIn y EnableOut se borran a falsos.
Tag.Enable-in es verdadero	Los bits EnableIn y EnableOut se establecen en verdaderos. La instrucción se ejecuta.
Primer escaneado de instrucción	N/A
Primer valor verdadero de la instrucción	N/A
Post-escaneado	Los bits EnableIn y EnableOut se borran a falsos.

### Ejemplo

Este ejemplo realiza una operación OR exclusivo con bool\_in1 y bool\_in2 y coloca el resultado en value\_result\_xor.

Si bool_In1 es:	Si bool_In2 es:	Entonces, value_result_xor es:
0	0	0
0	1	1
1	0	1
1	1	0

#### Bloque de funciones



## Consulte también

[Atributos comunes](#) en la página 923

### NO booleano (BNOT)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

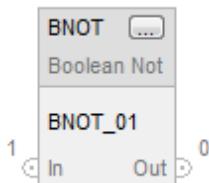
La instrucción BNOT complementa una entrada booleana. Para realizar un NOT bit a bit, consulte *No a nivel de bits (NOT)*.

#### Idiomas disponibles

#### Diagrama de escalera

Esta instrucción no está disponible en el diagrama de escalera.

#### Bloque de funciones



#### Texto estructurado

Esta instrucción no está disponible para texto estructurado.

#### Operandos

#### Bloque de funciones

Operando	Tipo	Formato	Descripción
BNOT tag	FBD_BOOLEAN_NOT	estructura	Estructura de BNOT

**Estructura de FBD\_BOOLEAN\_NOT**

Parámetro de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si se borra, la instrucción no se ejecuta y las salidas no se actualizan. Está establecido de forma predeterminada.
In	BOOL	Entrada de la instrucción. Establecida en 1 en la primera descarga.

Parámetro de salida	Tipo de datos	Descripción
EnableOut	BOOL	Indica si la instrucción está habilitada.
Out	BOOL	La salida de la instrucción.

**Descripción**

La instrucción BNOT complementa una entrada booleana.

$$\text{Out} = \text{NOT In}$$

**Afectar a las marcas de estado matemático**

No

**Fallos mayores/menores**

No es específico para esta instrucción. Consulte *Atributos de comandos* para los fallos relacionados con operandos.

**Ejecución****Bloque de funciones**

Condición/estado	Acción realizada
Pre-escaneado	Los bits EnableIn y EnableOut se borran a falsos.
Tag.EnableIn es falso	Los bits EnableIn y EnableOut se borran a falsos.
Tag.EnableIn es verdadero	Los bits EnableIn y EnableOut se establecen en verdaderos.

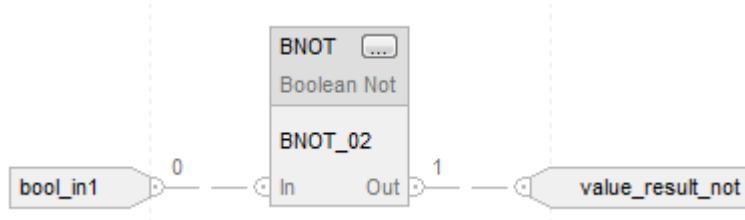
Primera ejecución de instrucción	N/A
Primer escaneado de instrucción	N/A
Post-escaneado	Los bits EnableIn y EnableOut se borran a falsos.

### Ejemplos

Este ejemplo copia bool\_in1 en BNOT\_01.in, complementa BNOT\_01.in colocando el resultado en BNOT\_01.out y copia BNOT\_01.out en value\_result\_not.

Si bool_in1 es:	Entonces, value_result_not es:
0	1
1	0

### Bloque de funciones



### Consulte también

[NO a nivel de bits \(NOT\)](#) en la página 486

[Atributos comunes](#) en la página 923

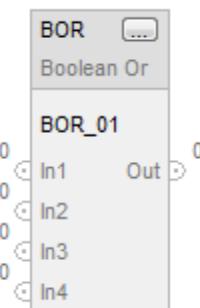
## O booleano (BOR)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción BOR realiza una operación OR de manera lógica como hasta ocho entradas booleanas. Para realizar una operación OR bit a bit, consulte *O a nivel de bits (OR)*.

**Idiomas disponibles****Diagrama de escalera**

Esta instrucción no está disponible en el diagrama de escalera.

**Bloque de funciones****Texto estructurado**

Esta instrucción no está disponible en texto estructurado.

**Operандos****Bloque de funciones**

Operando	Tipo	Formato	Descripción
BOR tag	FBD_BOOLEAN_OR	estructura	Estructura BOR

**Estructura de FBD\_BOOLEAN\_OR**

Parámetro de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si se ha borrado, la instrucción no se ejecuta y las salidas no se actualizan. Se establece en 0 en la primera descarga.
In1	BOOL	Primera entrada booleana. Se establece en 0 en la primera descarga.
In2	BOOL	Segunda entrada booleana. Se establece en 0 en la primera descarga.
In3	BOOL	Tercera entrada booleana. Se establece en 0 en la primera descarga.

In4	BOOL	Cuarta entrada booleana. Se establece en 0 en la primera descarga.
In5	BOOL	Quinta entrada booleana. Se establece en 0 en la primera descarga.
In6	BOOL	Sexta entrada booleana. Se establece en 0 en la primera descarga.
In7	BOOL	Séptima entrada booleana. Se establece en 0 en la primera descarga.
In8	BOOL	Octava entrada booleana. Se establece en 0 en la primera descarga.

Parámetro de salida	Tipo de datos	Descripción
EnableOut	BOOL	Indica si la instrucción está habilitada.
Out	BOOL	La salida de la instrucción.

### Descripción

La instrucción BOR realiza una operación OR como hasta ocho entradas booleanas. Si no se usa una entrada, se borra de forma predeterminada (0).

Out = In1 OR In2 OR In3 OR In4 OR In5 OR In6 OR In7 OR In8

---

**Importante:** Si quita un cable de entrada de la instrucción BOR durante una edición, debe asegurarse de que se haya borrado la entrada asociada (0).

---

### Afectar a las marcas de estado matemático

No

### Mayores Menores Fallos

No es específico para esta instrucción. Consulte los *Atributos comunes* para fallos relacionados con el operando.

## Ejecución

### Bloque de funciones

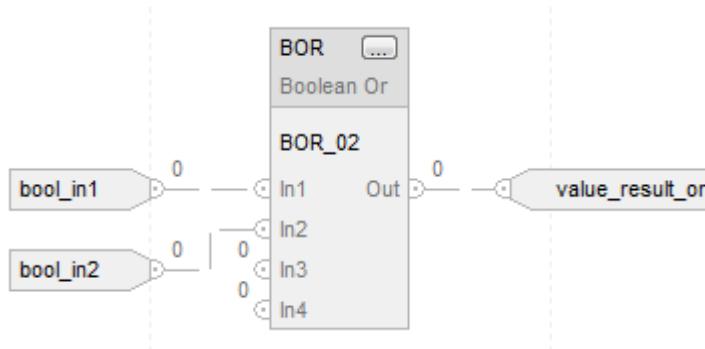
Condición/estado	Acción realizada
Pre-escaneado	Los bits EnableIn y EnableOut se borran a falsos.
Tag.EnableIn es falso	Los bits EnableIn y EnableOut se borran a falsos.
Tag.EnableIn es verdadero	Los bits EnableIn y EnableOut se establecen en verdaderos. La instrucción se ejecuta.
Primera ejecución de instrucción	N/A
Primer escaneado de instrucción	N/A
Post-escaneado	Los bits EnableIn y EnableOut se borran a falsos.

### Ejemplos

Este ejemplo realiza una operación OR con bool\_in1 y bool\_in2 y coloca el resultado en value\_result\_or.

Si bool_in1 es:	Si bool_in2 es:	Entonces, value_result_or es:
0	0	0
0	1	1
1	0	1
1	1	1

### Bloque de funciones



### Consulte también

[O a nivel de bits \(OR\)](#) en la página 490

[Atributos comunes](#) en la página 923

## Borrar (CLR)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción CLR borra todos los bits de Dest.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

#### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

#### Operandos

**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
- Los miembros de un operando de estructura se sobrescriben.
- A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.

La instrucción CLR admite tipo de datos elementales. Consulte *Tipo de datos*.

#### Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción
Dest	SINT INT DINT REAL	etiqueta	Etiqueta a borrar.

**Afectar a las marcas de estado matemático**

Controladores	Afectar a las marcas de estado matemático
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Condicional
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí

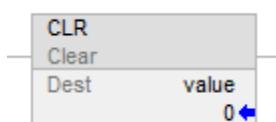
Consulte *Marcas de estado matemático*.

**Fallos mayores/menores**

No es específico para esta instrucción. Consulte *Índice a través de matrices* para ver si hay fallos de indexación de matrices.

**Ejecución****Diagrama de escalera**

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón.
La condición de entrada de reglón es verdadera	Establecer la condición de salida de reglón a condición de entrada de reglón. Establecer Dest en 0.
Post-escaneado	N/A

**Ejemplo****Diagrama de escalera**

### Consulte también

[Instrucciones de movimiento](#) en la página 467

[Índice a través de matrices](#) en la página 936

[Tipos de datos](#) en la página 931

[Marcas de estado matemático](#) en la página 923

## Mover con máscara (MVM)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

La instrucción MVM copia el valor de Source en Destination y permite que se aplique una máscara a partes de los datos.

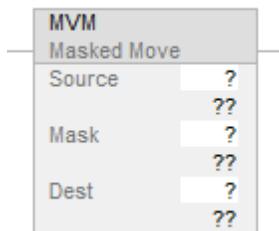
La instrucción MVM utiliza Mask para dejar pasar o bloquear bits de datos de Source. Un "1" en la máscara significa que se deja pasar el bit de datos; Un "0" en la máscara se bloquea el bit de datos.

Si mezcla tipos de datos enteros, la instrucción rellena los bits superiores de los tipos de datos enteros más pequeños con ceros para que tengan el mismo tamaño que el tipo de datos más grande.

### Introducción de un valor de máscara inmediato

Cuando se introduce una máscara, el software de programación pasa de forma predeterminada a usar valores decimales. Para introducir una máscara con otro formato, ponga el valor con el valor prefijo.

Prefijo	Descripción
16#	Hexadecimal (por ejemplo, 16#0F0F)
8#	Octal (por ejemplo, 8#16)
2#	Binario (por ejemplo, 2#00110011)

**Idiomas disponibles****Diagrama de escalera****Bloque de funciones**

Esta instrucción no está disponible en bloque de funciones.

**Texto estructurado**

Esta instrucción no está disponible en texto estructurado.

**Operандos**

**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
- Los miembros de un operando de estructura se sobrescriben.
- A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción. Consulte *Conversión de datos*.

**Diagrama de escalera**

Operando	Tipo de datos	Formato	Descripción
Source	SINT INT DINT	inmediato etiqueta	Valor a mover
Mask	SINT INT DINT	inmediato etiqueta	Qué bits bloquear o dejar pasar
Dest	SINT INT DINT	etiqueta	Etiqueta para almacenar el resultado

### Afectar a las marcas de estado matemático

Controladores	Afectar a las marcas de estado matemático
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	No
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí

### Fallos mayores/menores

Controladores	Ocurrirá un fallo menor si:	Tipo de fallo	Código de fallo
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	La función está habilidad y se detecta un desbordamiento	4	4
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	N/A	N/A	N/A

Consulte *Índice a través de matrices* para ver si hay fallos de indexación de matrices.

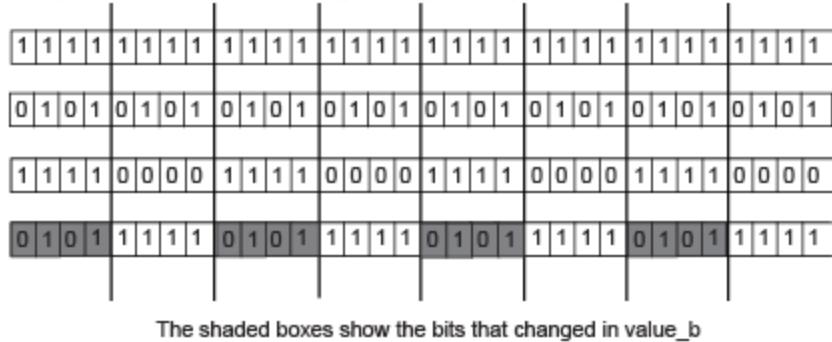
### Ejecución

#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	La instrucción pasa el Source a través de Mask y copia el resultado en Destination. Los bits de Destination a los que no se les ha aplicado la máscara se quedan igual.
Post-escaneado	N/A

## Ejemplo

### Diagrama de escalera



Fila 1: value\_b antes de MVM

Fila 2: value\_a

Fila 3: mask\_2

Fila 4: value\_b después de MVM

MVM Masked Move	
Source	value_a
2#0101_0101_0101_0101_0101_0101_0101_0101	↑
Mask	mask_2
2#1111_0000_1111_0000_1111_0000_1111_0000	↑
Dest	value_b
2#1111_1111_1111_1111_1111_1111_1111_1111	↑

Copia los datos de value\_a a value\_b, al mismo tiempo que permite que se aplique una máscara a los datos (un 0 aplica una máscara a los datos de value\_a).

### Consulte también

[Instrucciones de movimiento](#) en la página 467

[Conversiones de datos](#) en la página 927

[Índice a través de matrices](#) en la página 936

## Mover con máscara con receptor (MVMT)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

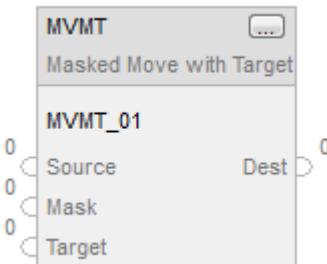
La instrucción MVMT copia el valor de Source en Destination y permite que se aplique una máscara a partes de los datos.

### Idiomas disponibles

#### Diagrama de escalera

Esta instrucción no está disponible en el diagrama de escalera.

#### Bloque de funciones



#### Texto estructurado

```
MVMT(MVMT_tag);
```

#### Operандos

#### Texto estructurado

Variable	Tipo	Formato	Descripción
MVMT tag	FBD_MASKED_MOVE	Estructura	Estructura de MVMT

Consulte *Sintaxis de texto estructurado* para obtener información sobre la sintaxis de las expresiones dentro de texto estructurado.

#### Bloque de funciones

Operando	Tipo	Formato	Descripción
MVMT tag	FBD_MASKED_MOVE	Estructura	Estructura de MVMT

### Estructura de FBD\_MASKED\_MOVE

Parámetro de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Si no está activada, la instrucción no se ejecuta y las salidas no se actualizan. Si está activado, la instrucción se ejecuta. Está establecido de forma predeterminada.
Source	DINT	Valor de entrada para mover a Destination en función del valor de Mask. Válido = cualquier entero
Mask	DINT	Mask de bits para mover de Source a Dest. Si todos los bits están establecidos en, los bits correspondientes se mueven de Source a Dest. Todos los bits que se establecen en cero hacen que los bits correspondientes no se muevan de Source a Dest. Válido = cualquier entero
Target	DINT	Valor de entrada a mover a Dest antes de mover bits de Source a través de Mask. Válido = cualquier entero

Parámetro de salida	Tipo de datos	Descripción
EnableOut	BOOL	Indica si la instrucción está habilitada.
Dest	DINT	Resultado de la operación de movimiento con máscara.

#### Descripción

Si está habilitada, la instrucción MVMT usa una Mask para dejar pasar o bloquear bits de datos de Source. Un "1" en la máscara significa que el bit de datos se ha pasado. Un "0" en la máscara significa que el bit de datos se ha bloqueado.

Si mezcla tipos de datos enteros, la instrucción rellena los bits superiores de los tipos de datos enteros más pequeños con ceros para que tengan el mismo tamaño que el tipo de datos más grande.

#### Introducir un valor de máscara inmediato usando una Reference de entrada

Cuando se introduce una máscara, el software de programación pasa de forma predeterminada a usar valores decimales. Si quiere introducir una máscara usando otro formato, coloque el prefijo adecuado antes del valor.

Prefijo	Descripción
16#	hexadecimal (por ejemplo, 16#0F0F)
8#	octal (por ejemplo, 8#16)
2#	binario (por ejemplo, 2#00110011)

### Afectar a las marcas de estado matemático

Controladores	Afectar a las marcas de estado matemático
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	No
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí para la salida

### Fallos mayores/menores

No es específico para esta instrucción. Consulte los *Atributos comunes* para fallos relacionados con el operando.

### Ejecución

#### Bloque de funciones

Condición/estado	Acción realizada
Pre-escaneado	Los bits EnableIn y EnableOut se borran a falsos.
Tag.EnableIn es falso	Los bits EnableIn y EnableOut se borran a falsos.
Tag.EnableIn es verdadero	Los bits EnableIn y EnableOut se establecen en verdaderos. La instrucción se ejecuta.
Primera ejecución de instrucción	N/A
Primer escaneado de instrucción	N/A
Post-escaneado	Los bits EnableIn y EnableOut se borran a falsos.

### Texto estructurado

Condición/estado	Acción realizada
Pre-escaneado	Consultar Pre-escaneado en la tabla Bloque de funciones.
Ejecución normal	Consultar Tag.EnableIn es verdadero en la tabla Bloque de funciones.
Post-escaneado	Consultar Post-escaneado en la tabla Bloque de funciones.

### Ejemplos

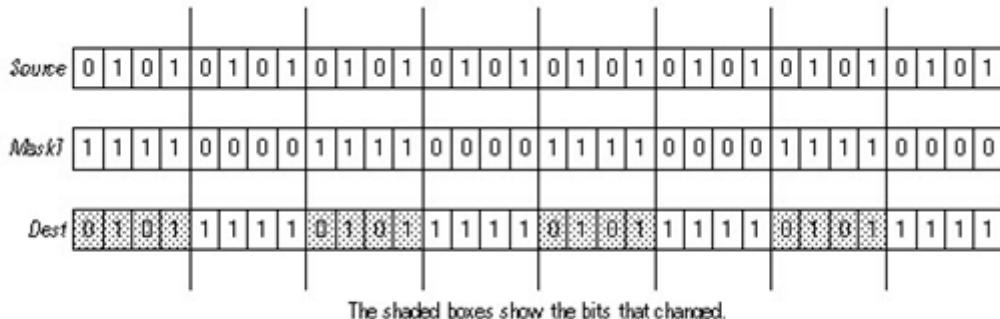
#### Paso 1

El controlador copia el valor de Target en Dest.

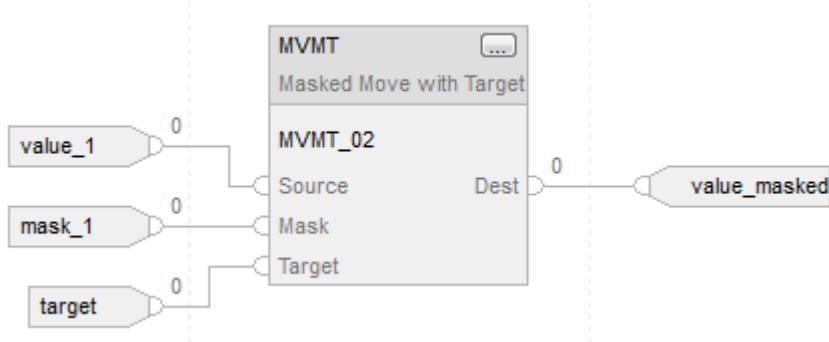


#### Paso 2

La instrucción aplica una máscara a Source y lo compara con Dest. Cualquier cambio que sea necesario, se realiza en Dest, que se convierte en un parámetro de entrada para value\_masked. Source y Target se quedan igual. Un 0 en la máscara impide que la instrucción compare ese bit.



### Bloque de funciones



### Texto estructurado

```
MVMT_01.Source := value_1;  
MVMT_01.Mask := mask_1;  
MVMT_01.Target := target;
```

```
MVMT(MVMT_01);
```

```
value_masked := MVMT_01.Dest;
```

### Consulte también

[Mover con máscara \(MVM\)](#) en la página 509

[Conversiones de datos](#) en la página 927

[Sintaxis de texto estructurado](#) en la página 955

[Atributos comunes](#) en la página 923

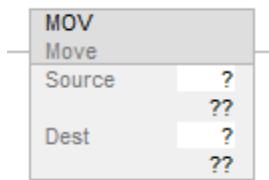
## Mover (MOV)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

La instrucción MOV mueve una copia de Source a Dest. Source permanece sin cambios.

## Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

**Cons** Utilice una asignación ":=>" con una expresión para  
**ejo:** obtener el mismo resultado. Consulte *Structured Text Syntax* para obtener más información sobre la sintaxis de las expresiones y las asignaciones en el texto estructurado.

## Operандos

**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
- Los miembros de un operando de estructura se sobrescriben.
- A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.

Existen reglas de conversión de datos para utilizar diversos tipos de datos numéricos en una instrucción. Consulte *Conversión de datos*.

### Diagrama de escalera

### Numérico

Operando	Tipo de datos	Formato	Descripción
Source	SINT INT DINT REAL	inmediato etiqueta	Valor a mover

Dest	SINT INT DINT REAL	etiqueta	Etiqueta para almacenar el resultado
------	-----------------------------	----------	--------------------------------------

Cadena (solo para Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.)

Operando	Tipo de datos	Formato	Descripción
Source	Tipo de cadena	inmediato etiqueta	Cadena a mover
Dest	Tipo de cadena	etiqueta	Etiqueta para almacenar el resultado

#### Afectar a las marcas de estado matemático

Controladores	Afectar a las marcas de estado matemático
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Condicional
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí

Consulte *Marcas de estado matemático*.

#### Fallos mayores/menores

No es específico para esta instrucción. Consulte *Índice a través de matrices* para ver si hay fallos de indexación de matrices.

#### Ejecución

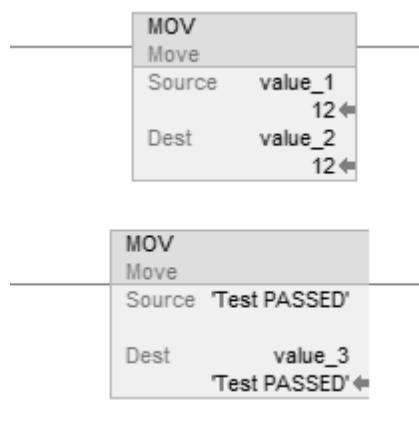
#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón.

La condición de entrada de reglón es verdadera	Establecer la condición de salida de reglón a condición de entrada de reglón. La instrucción copia el valor de Source en Dest. <b>Operando de cadena:</b> Si Source.LEN > SIZE( Dest.DATA) La cadena se trunca a lo que encaje S:V está establecido.
Post-escaneado	N/A

### Ejemplos

#### Diagrama de escalera



#### Texto estructurado

```
value_2 := value_1;
value_3 := 'Test PASSED';
```

#### Consulte también

[Sintaxis de texto estructurado](#) en la página 955

[Conversiones de datos](#) en la página 927

[Marcas de estado matemático](#) en la página 923

[Índice a través de matrices](#) en la página 936

[Instrucciones de movimiento](#) en la página 467

## Intercambiar byte (SWPB)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción SWPB reorganiza el orden de los bytes de Source. Coloca el resultado en Destination.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

#### Texto estructurado

SWPB(Source, Order Mode, Dest);

#### Operандos

**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
- Los miembros de un operando de estructura se sobrescriben.
- A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.

Existen reglas de conversión de datos para utilizar diversos tipos de datos numéricos en una instrucción. Consulte *Conversión de datos*.

#### Diagrama de escalera y Texto estructurado

Operando	Tipo de datos	Formato	Descripción
Source	INT DINT	etiqueta	Etiqueta que contiene los bytes que se deben reorganizar.
Order Mode		elemento de lista	Este operando especifica cómo reordenar los bytes. Consulte la tabla Modo Orden.
Dest	INT DINT	etiqueta	Etiqueta para almacenar los bytes en un nuevo orden. Consulte la tabla Dest.

Si selecciona el modo de orden HIGH/LOW, intodúzcalo como HIGHLOW (sin la barra inclinada). Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

### Modo Orden

Si el valor de Source es un	Y quiere cambiar los bytes a este patrón (cada letra representa un byte diferente)	Entonces, selecciona
INT	AB => BA	Cualquiera opción
DINT	ABCD => DCBA	REVERSE
	ABCD => CDAB	WORD
	ABCD => BADC	HIGH/LOW

### Dest

Si el valor de Source es un	Entonces, el valor de Destination debe ser un
INT	INT, DINT Si el valor de Destinación es un DINT, se extiende el signo del resultado después de que se intercambien los bytes.
DINT	DINT

### Afectar a las marcas de estado matemático

No

### Fallos mayores/menores

No es específico para esta instrucción. Consulte *Índice a través de matrices* para ver si hay fallos de indexación de matrices.

### Ejecución

#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	La instrucción reordena los bytes especificados.
Post-escaneado	N/A

### Texto estructurado

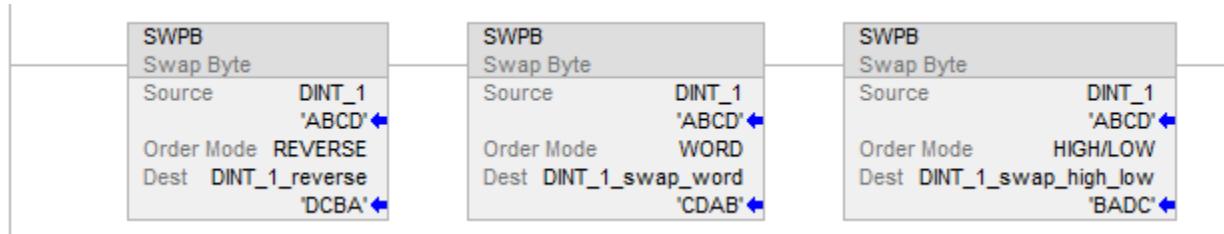
Condición/estado	Acción realizada
Pre-escaneado	Consultar Pre-escaneado en la tabla Diagrama de escalera.
Ejecución normal	Consultar La condición de entrada de reglón es verdadera en la tabla Diagrama de escalera.
Post-escaneado	Consultar Post-escaneado en la tabla Diagrama de escalera.

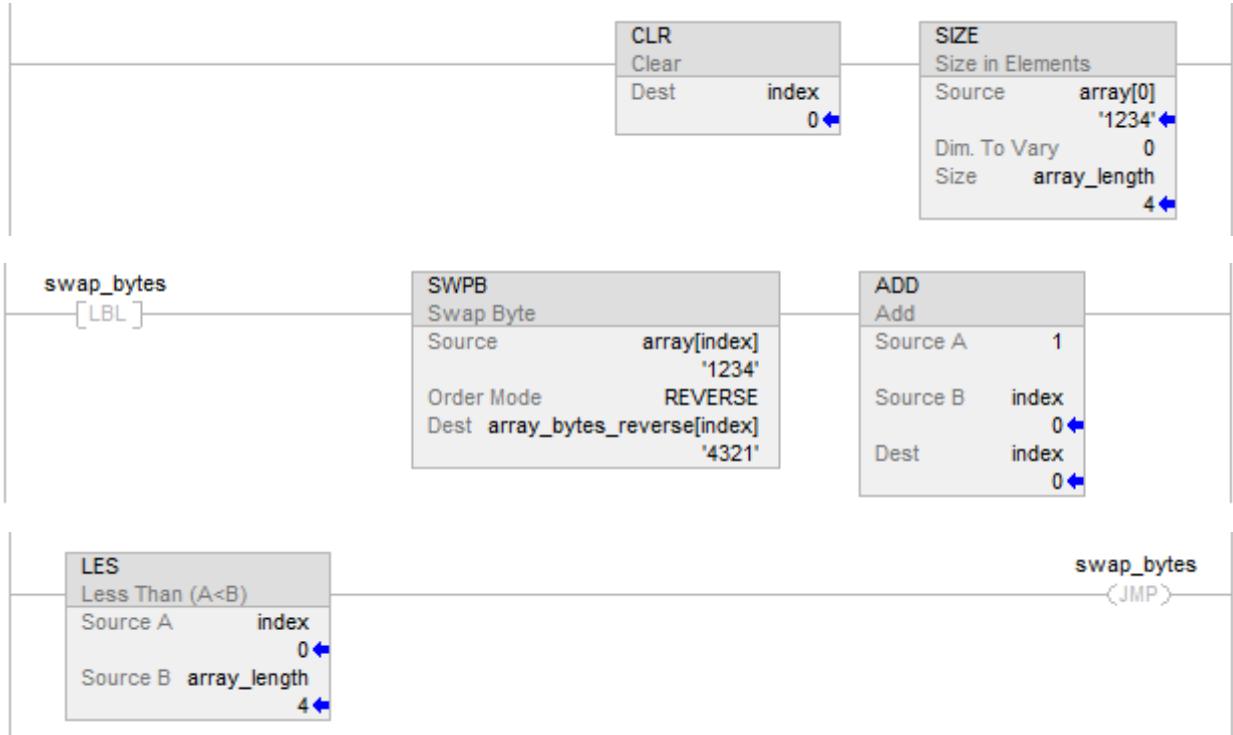
### Ejemplos

#### Ejemplo 1: intercambiar los bytes de una etiqueta DINT

Las tres instrucciones SWPB reordenan los bytes de DINT\_1 conforme a un modo de ordenación diferente. El estilo de visualización es ASCII y cada carácter representa un byte. Cada instrucción coloca los bytes en el nuevo orden en un Destination diferente.

#### Diagrama de escalera



**Ejemplo 2: intercambiar los bytes en todos los elementos de una matriz****Diagrama de escalera****Ejemplo 3: SWPB en texto estructurado****Texto estructurado**

```

index := 0;
SIZE (array[0],0,array_length);
REPEAT
    SWPB(array[index],REVERSE,a rray_bytes_reverse[index]);
    index := index + 1;
UNTIL(index >= array_length)END_REPEAT;

```

**Consulte también**

[Instrucciones de movimiento](#) en la página 467

[Índice a través de matrices](#) en la página 936

[Conversiones de datos](#) en la página 927

[Sintaxis de texto estructurado](#) en la página 955



## **Matriz (Archivo)/Instrucciones misceláneas**

### **Matriz (Archivo)/Instrucciones misceláneas**

Las instrucciones de archivo/miscelánea operan en matrices de datos.

#### **Instrucciones disponibles**

#### **Diagrama de escalera**

<a href="#">FAL</a>	<a href="#">FSC</a>	<a href="#">COP</a>	<a href="#">CPS</a>	<a href="#">FLL</a>	<a href="#">AVE</a>
---------------------	---------------------	---------------------	---------------------	---------------------	---------------------

#### **Bloque de funciones**

No disponible

#### **Texto estructurado**

<a href="#">SIZE</a>	<a href="#">FSC</a>	<a href="#">COP</a>	<a href="#">CPS</a>
----------------------	---------------------	---------------------	---------------------

<b>Si desea:</b>	<b>Utilice esta instrucción:</b>
Realizar operaciones aritméticas, lógicas, de desplazamiento y funciones en valores de matrices.	FAL
Buscar y comparar valores de matrices	FSC
Copiar el contenido de una matriz en otra matriz	COP
Copiar uno o varios valores de Source en Destination	CPS
Llenar una matriz con datos específicos	FLL
Calcular la media de una matriz de valores	AVE
Ordenar una dimensión de datos de matriz en orden ascendente	SRT
Calcular la desviación estándar de una matriz de valores	STD

Averiguar el tamaño de una dimensión de una matriz	SIZE
--	------

Se pueden mezclar tipos de datos, pero es posible que se pierda precisión, que haya errores de redondeo y que la instrucción tarde más tiempo en ejecutarse. Compruebe el bit S:V para ver si el resultado se ha truncado.

Los tipos de datos que aparecen en **negrita** indican tipos de datos óptimos. Una instrucción se ejecutará a la máxima velocidad y la menor cantidad posible de memoria si todos los parámetros emplean el mismo tipo de datos óptimo, normalmente DINT o REAL.

### Seleccionar el Modo de funcionamiento

En el caso de las instrucciones FAL y FSC, el modo indica al controlador cómo distribuir la operación de matriz.

Si desea:	Seleccione este modo:
operar en todos los elementos especificados de una matriz antes de pasar a la siguiente instrucción	Modo Todos
distribuir una operación de matriz a lo largo de varios escaneados introducir el número de elementos sobre los que se debe operar en cada escaneado (1-2147483647)	Modo Numérico
manipular un elemento de la matriz cada vez que el valor de condición de entrada de reglón pase de falso a verdadero.	Modo Incremental

### Consulte también

[Modo Todos](#) en la página 595

[Modo Numérico](#) en la página 597

[Modo Incremental](#) en la página 599

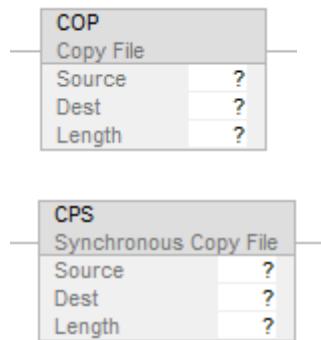
## Copiar archivo (COP), Copiar archivo sincrónico (CPS)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

Las instrucciones COP y CPS copian los valores de Source en Dest. Source permanece sin cambios.

## Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

### Texto estructurado

COP(Source,Dest,Length);

CPS(Source,Dest,Length);

### Operandos

**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
- Los miembros de un operando de estructura se sobrescriben.
- A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.

### Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción
Source	SINT INT DINT LINT REAL Tipo de cadena estructura	etiqueta	Elemento inicial para copiar

Dest	SINT INT DINT LINT REAL Tipo de cadena estructura	etiqueta	Elemento inicial a sobrescribir por Source
Length	SINT INT DINT	inmediato etiqueta	Número de elementos de Destination para copiar

**Texto estructurado**

Operando	Tipo de datos	Formato	Descripción
Source	SINT INT DINT LINT REAL Tipo de cadena estructura	etiqueta	Elemento inicial para copiar
Dest	SINT INT DINT LINT REAL Tipo de cadena estructura	etiqueta	Elemento inicial a sobrescribir por Source
Length	SINT INT DINT	inmediato etiqueta	Número de elementos de Destination para copiar

Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

**Afectar a las marcas de estado matemático**

No

**Fallos mayores/menores**

No es específico para esta instrucción. Consulte *Índice a través de matrices* para ver si hay fallos de indexación de matrices.

## Ejecución

### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón.
La condición de entrada de reglón es verdadera	Establecer la condición de salida de reglón a condición de entrada de reglón. La instrucción copia los datos.
Post-escaneado	N/A

### Texto estructurado

Condición/estado	Acción realizada
Pre-escaneado	Consultar Pre-escaneado en la tabla de diagrama de escalera
Ejecución normal	Consultar La condición de entrada de reglón es verdadera en la tabla Diagrama de escalera.
Post-escaneado	Consultar Post-escaneado en la tabla Diagrama de escalera.

Durante la ejecución de las instrucciones COP y CPS, otras acciones del controlador pueden intentar interrumpir la operación de copia y cambiar el origen:

Si el origen o el destino es:	Y necesita:	Entonces selecciona:	Notas
<ul style="list-style-type: none"> <li>• etiqueta producida</li> <li>• etiqueta consumida</li> <li>• datos de E/S</li> <li>• datos que otra tarea puede sobrescribir</li> </ul>	Evitar que los datos de origen cambien durante la operación de copia	CPS	Las tareas que intentan interrumpir una instrucción CPS se retardan hasta que se realiza la instrucción. Para calcular el tiempo de ejecución de la instrucción CPS, consulte el Manual del usuario del sistema ControlLogix, publicación 1756-UM001.
	Permitir que los datos de origen cambien durante la operación de copia	COP	
Ninguno de los anteriores	----->	COP	

Las instrucciones COP y CPS funcionan en la memoria continua y realizan una copia de memoria de byte a byte directa.

Cuando Source y Dest tienen diferentes tipos de datos, el número de bytes copiados es igual al menor entre:

- La cantidad solicitada que es igual a la longitud x (el número de bytes en un elemento de destino)
- El número de bytes en la etiqueta de destino
- Para Controladores Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 o GuardLogix 5580s: el número de bytes en la etiqueta de origen

**Cons** El final de la etiqueta de origen o destino se define como **ejo**: el último byte de la etiqueta de base. Si la etiqueta es una estructura, el final de la etiqueta es el último byte del último elemento de la estructura. Esto significa que las instrucciones COP y CPS podrían escribir más allá del final de una matriz de miembros, pero nunca escribirán más allá del final de la etiqueta de base.

---

**Importante:** Prueba y confirma que la instrucción no cambia los datos que no debe cambiar.

---

### Ejemplos

#### Ejemplo 1

Copiar una matriz.

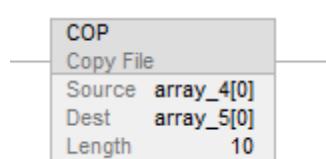
Cuando está habilitada, la instrucción COP copia 40 bytes de array\_4 a array\_5.

array\_4 es un DINT (4 bytes por elemento) y contiene 10 elementos (tamaño total = 40 bytes).

array\_5 es un DINT (4 bytes por elemento) y contiene 10 elementos (tamaño total = 40 bytes).

Length indica que se deben copiar 10 elementos de destino para que se copien 40 bytes.

#### Diagrama de escalera



### Texto estructurado

COP(array\_4[0],array\_5[0],10);

#### Ejemplo 2

Copiar una estructura.

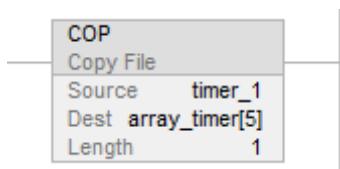
Cuando está habilitada, la instrucción COP copia la estructura timer\_1 en el elemento 5 de array\_timer.

timer\_1 es un TIMER (tamaño total = 12 bytes).

array\_timer es un TIMER (12 bytes por elemento) y contiene 10 elementos (tamaño total = 120 bytes).

Longitud indica un elemento de destino, de forma que se copian 12 bytes.

### Diagrama de escalera



### Texto estructurado

COP(timer\_1,array\_timer[5],1);

#### Ejemplo 3

Copiar los datos de la matriz evitando que se modifiquen los datos hasta que completa la copia.

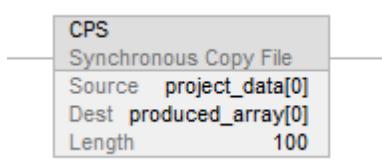
La matriz project\_data (100 elementos) almacena una variedad de valores que cambian en diferentes momentos de la aplicación. Para enviar una imagen completa de project\_data en una instancia a tiempo a otro controlador, la instrucción CPS copia project\_data a produced\_array. Mientras la instrucción CPS copia los datos, ninguna actualización de E/S u otras tareas pueden cambiar los datos. La etiqueta produced\_array produce los datos en una red ControlNet para su consumo por parte de otros controladores.

project\_data es un DINT (4 bytes por elemento) y contiene 100 elementos (tamaño total = 400 bytes).

produced\_array es un DINT (4 bytes por elemento) y contiene 100 elementos (tamaño total = 400 bytes).

Length indica 100 elementos de destino, de forma que se copian 400 bytes.

#### Diagrama de escalera



#### Texto estructurado

```
CPS(project_data[0],produced_array[0],100);
```

#### Ejemplo 4

Copiar los datos en una etiqueta producida evitando que se envíen los datos hasta que completa la copia.

Local:0:I.Data almacena los datos de entrada para la red DeviceNet que está conectada al módulo 1756-DNB en la ranura 0. Para sincronizar las entradas con la aplicación, la instrucción CPS copia los datos de entrada a input\_buffer.

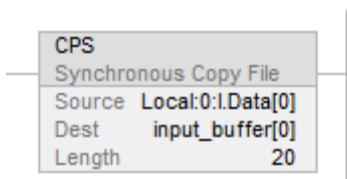
Mientras la instrucción CPS copia los datos, ninguna actualización de E/S puede cambiar los datos. A medida que la aplicación se ejecuta, utiliza para sus entradas los datos de entrada en input\_buffer.

Local:0:I.Data es un DINT (4 bytes por elemento) y contiene 2 elementos (tamaño total = 8 bytes).

input\_buffer es un DINT (4 bytes por elemento) y contiene 20 elementos (tamaño total = 80 bytes).

Length indica que se deben copiar 20 elementos de destino ( $4 \times 20 = 80$  bytes). Sin embargo, el origen solo puede proporcionar 8 bytes, de forma que se copian 8 bytes.

#### Diagrama de escalera

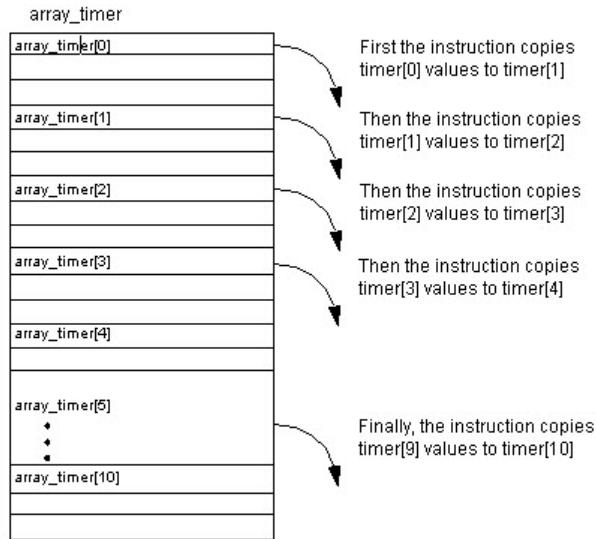


#### Texto estructurado

```
CPS(Local:0:I.Data[0], input_buffer[0], 20);
```

## Ejemplo 5

Iniciar una estructura de matriz, inicializar el primer elemento y utilizar COP para replicarlo al resto de la matriz.

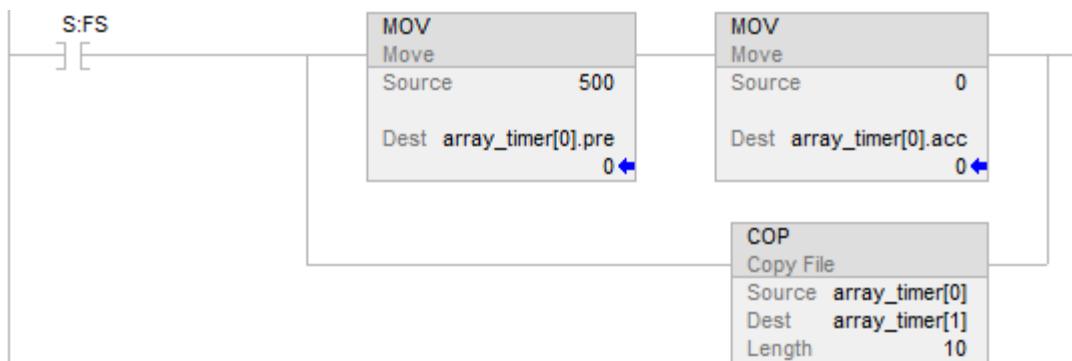


Este ejemplo inicializa una estructura de matriz o temporizador. Cuando están habilitadas, las instrucciones MOV inicializan los valores .PRE y .ACC del primer elemento de `array_timer`. Cuando está habilitada, la instrucción COP copia un bloque continuo de bytes, comenzando en `array_timer[0]`. La longitud es de nueve estructuras de temporizador.

`array_timer` es un TIMER (12 bytes por elemento) y contiene 15 elementos (tamaño total = 180 bytes).

Length indica 10 elementos de destino, de forma que se copian 120 bytes.

### Diagrama de escalera



### Texto estructurado

IF S:FS THEN

```
array_timer[0].pre := 500;  
array_timer[0].acc := 0;  
COP(array_timer[0],array_timer[1],10);  
END_IF;
```

### Ejemplo 6

Copiar matrices de diferentes tamaños.

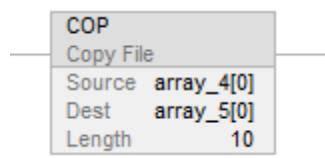
Cuando está habilitada, la instrucción COP copia bytes de SINT array\_6 a DNT array\_7.

array\_6 es un SINT (1 byte por elemento) y contiene 5 elementos (tamaño total = 5 bytes).

array\_7 es un DINT (4 bytes por elemento) y contiene 10 elementos (tamaño total = 40 bytes).

Length indica que se deben copiar 20 elementos de destino ( $4 \times 20 = 80$  bytes). Sin embargo, el destino solo puede aceptar 40 bytes y el origen solo puede proporcionar 5 bytes, de forma que se copian 5 bytes.

### Diagrama de escalera



### Texto estructurado

```
COP(array_4[0],array_5[0],10);
```

### Consulte también

[Índice a través de matrices](#) en la página 936

[Instrucciones de archivos/miscláneas](#) en la página 527

[Instrucciones de movimiento/lógicas](#) en la página 467

[Sintaxis de texto estructurado](#) en la página 955

## Aritmética y lógica de archivo (FAL)

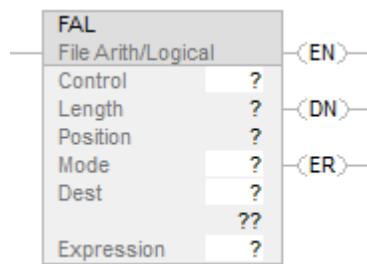
Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact

GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

La instrucción FAL realiza operaciones de copia, aritmética, lógica y función en los datos almacenados en una matriz. Cuando el valor de condición de entrada de reglón de la instrucción FAL pasa de falso a verdadero, la expresión proporcionada se ejecutará a lo largo del modo de iteración especificado.

### Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

### Operандos

**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
- Los miembros de un operando de estructura se sobrescriben.
- A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.

Existen reglas de conversión de datos para utilizar diversos tipos de datos numéricos en una instrucción. Ver Conversión de Datos.

### Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción
Control	CONTROL	Etiqueta	Estructura de control para la operación
Length	DINT	Inmediato	Número de elementos en la matriz a manipular
Position	DINT	Inmediato	Desplazamiento en una matriz El valor inicial suele ser 0
Mode	DINT	Inmediato	Muestra cómo distribuir la operación Selecciona INC, ALL o introduce un número en el rango de 1 a 2147483647
Expression	SINT INT DINT REAL	Inmediato Etiqueta	Una expresión que consta de etiquetas y/o valores inmediatos separados por operadores.
Destination	SINT INT DINT REAL	Etiqueta	El valor de Expression se almacenará en Destination.

### Estructura de CONTROL

Mnemónico	Tipo de datos	Descripción
.EN	BOOL	El bit de habilitación indica que la instrucción FAL está habilitada.
.DN	BOOL	El bit de efectuado se establece cuando la instrucción ha operado en el último elemento (.POS = .LEN).
.ER	BOOL	Si se produce un desbordamiento, las dos plataformas establecerán el valor de .ER y dejarán de ejecutar la instrucción. Los siguientes controladores generarán un desbordamiento: <ul style="list-style-type: none"><li>• CompactLogix 5370</li><li>• ControlLogix 5570</li></ul>
.LEN	DINT	La longitud especifica el número de elementos en la matriz en los que opera la instrucción FAL.
.POS	DINT	La posición contiene la posición del elemento actual al que tiene acceso la instrucción.

Consulte Síntaxis de texto estructurado[2] para obtener más información sobre la sintaxis de expresiones dentro de texto estructurado.

El valor de la expresión se almacena en la etiqueta de destino. Si se produce un desbordamiento, establecerá el bit ER y detendrá la ejecución. Una vez la

instrucción FAL completa todas las iteraciones configuradas, se establece el bit .DN.

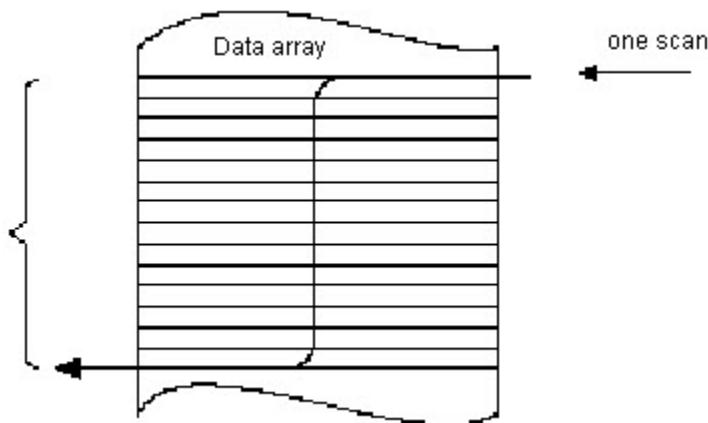
### Seleccionar el modo de funcionamiento

En el caso de las instrucciones FAL, el modo indica al controlador cómo distribuir la operación de matriz.

Si:	Seleccione este modo:
Operando en todos los elementos especificados de una matriz antes de pasar a la siguiente instrucción.	Todos
Distribuyendo una operación de matriz a lo largo de varios escaneados. Introducir el número de elementos para operar por escaneado (1-2147483647).	Numérico
Manipulando un elemento de la matriz cada vez que el valor de EnableIn pasa de falso a verdadero.	Incremental

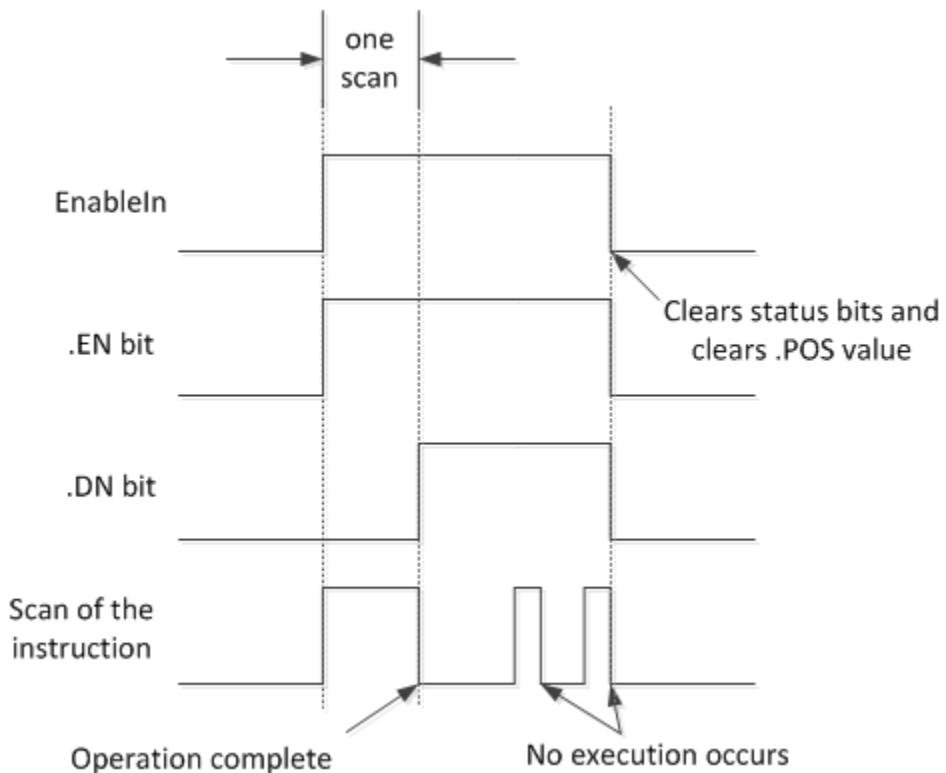
### Modo Todos

En modo Todos, la instrucción opera en todos los elementos especificados de la matriz antes de seguir a la siguiente instrucción. La operación comienza cuando EnableIn de la instrucción realiza una transición de falso a verdadero. El valor de posición (.POS) en la estructura de control apunta al elemento de la matriz que la instrucción está utilizando actualmente. La operación se detiene cuando el valor de .POS es igual o superior al valor de .LEN y cuando se produce un desbordamiento en la expresión y el bit .ER se establece en verdadero.



El siguiente diagrama de temporización muestra la relación entre los bits de estado y la operación de la instrucción. Cuando finaliza la ejecución de la instrucción, el bit .DN es verdadero. El bit .EN, el bit .ER y el valor .POS se borran cuando

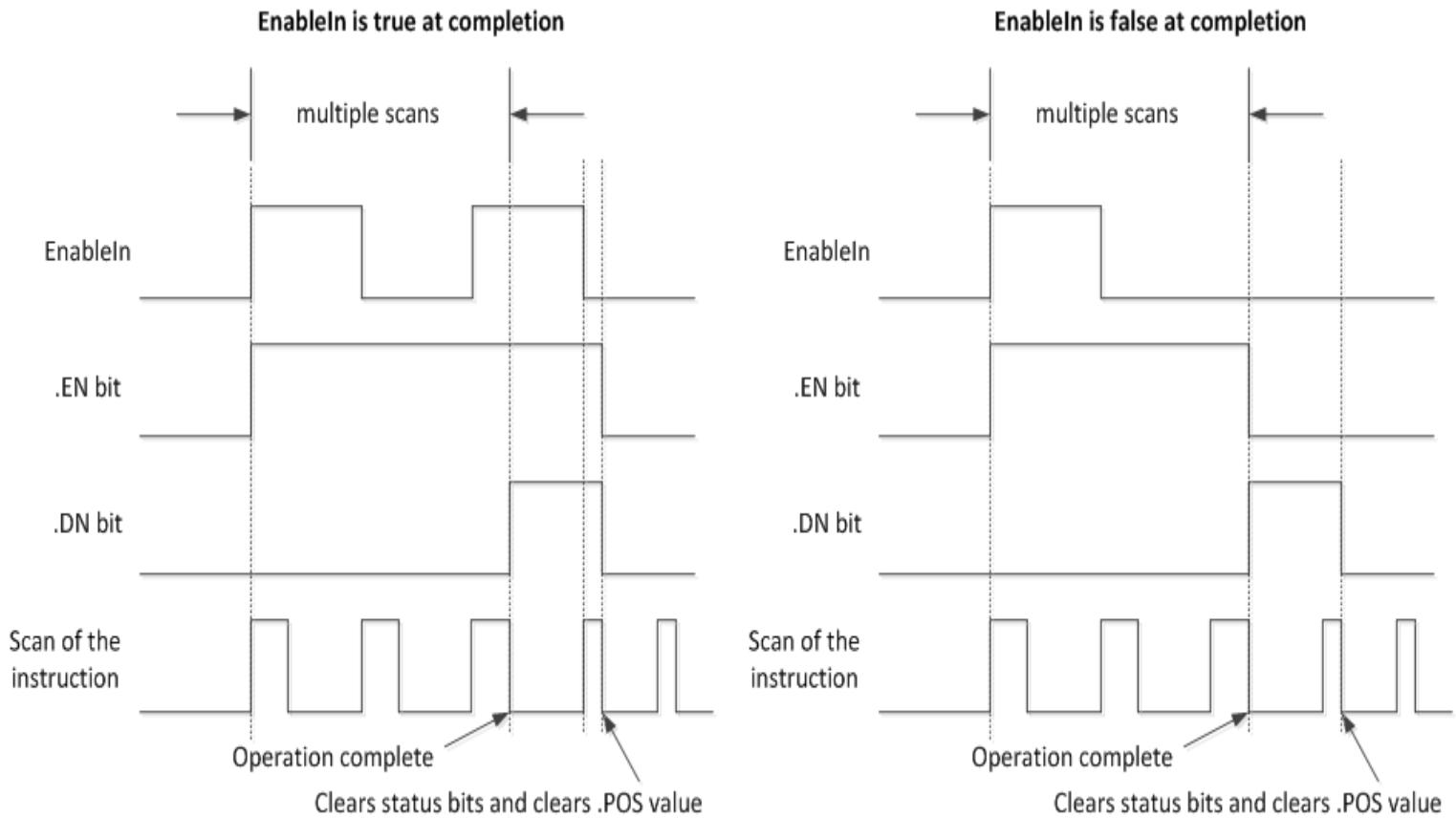
EnableIn es falso. Solo entonces puede desencadenarse otra ejecución de la instrucción mediante una transición de falso a verdadero de EnableIn.



### Modo Numérico

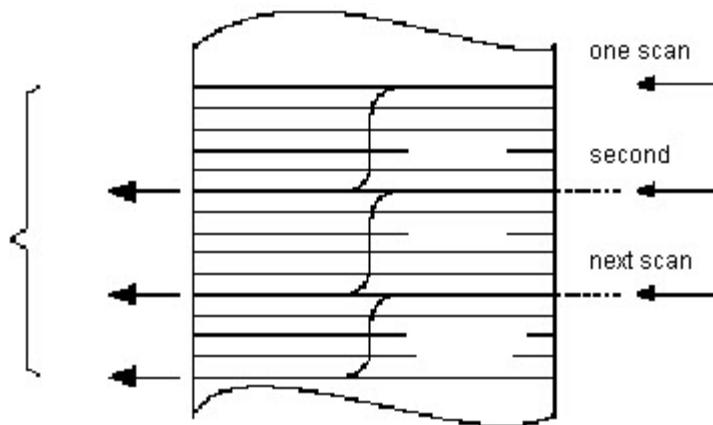
El modo Numérico distribuye el funcionamiento de la matriz sobre una serie de escaneados. Se utiliza este modo cuando trabaje con datos que no sean críticos o cuando el volumen de datos sea grande. Se introduce el número de elementos a operar en cada escaneado, que reduce el tiempo de escaneado.

La ejecución se desencadena cuando EnableIn realiza una transición de falso a verdadero. Una vez desencadenada, la instrucción se ejecuta cada vez que se escanea con el número de escaneados necesarios para completar la operación sobre toda la matriz. Cuando está desencadenada, EnableIn puede cambiar repetidamente sin interrumpir la ejecución de la instrucción.



Evita usar los resultados de una instrucción de archivo que opere en el modo numérico hasta que se haya establecido el bit .DN.

El siguiente diagrama de temporización muestra la relación entre los bits de estado y la operación de la instrucción. Cuando se haya completado la ejecución de la instrucción, se establece el bit .DN.

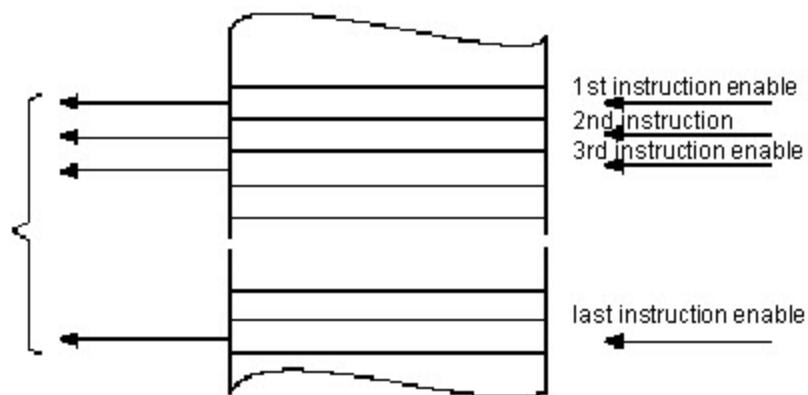


Si EnableIn es verdadero al finalizar, los bits .EN y .DN son verdaderos hasta que EnableIn pase a falso. Cuando EnableIn se vuelve falso, estos bits y el valor .POS se borran.

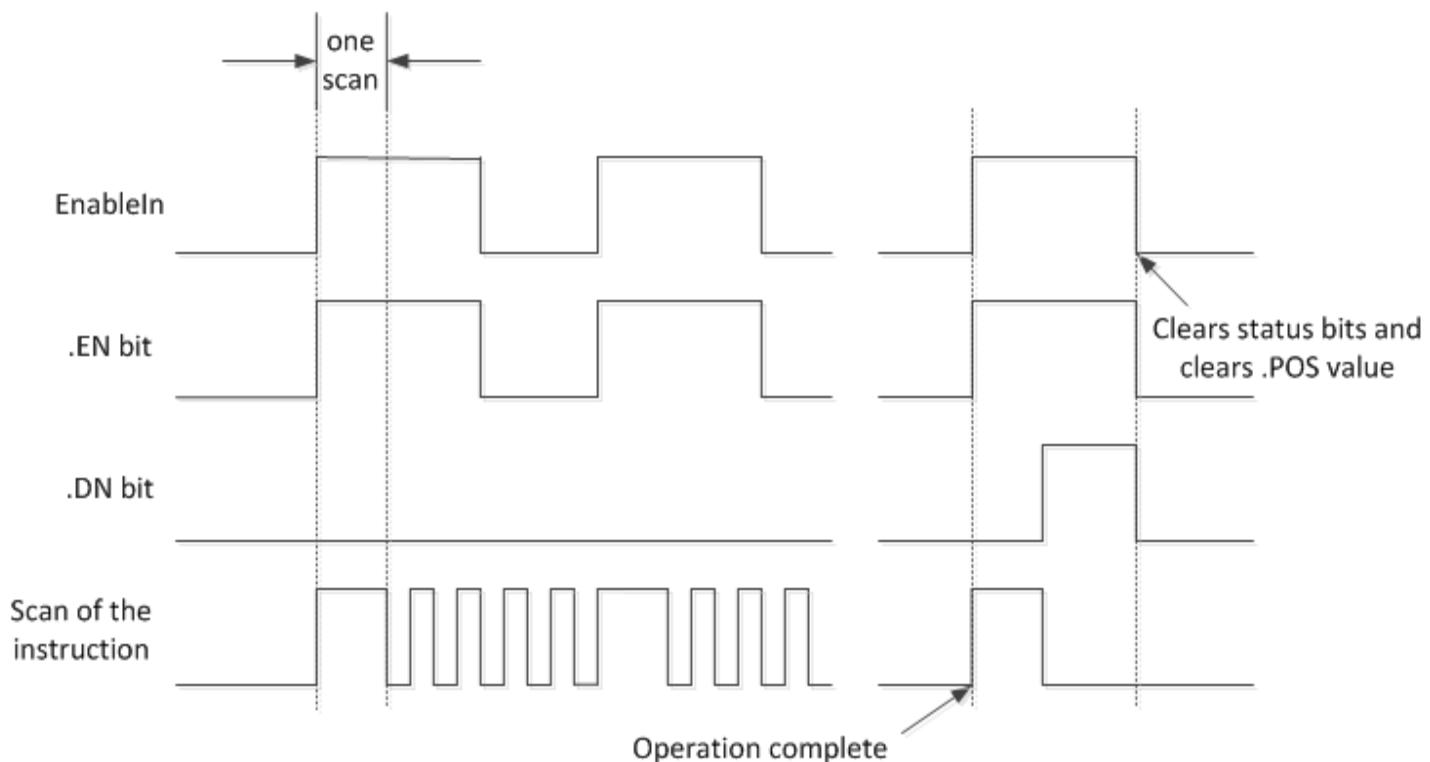
Si EnableIn es falso al finalizar, el bit .EN se borra inmediatamente. Después de borrarse el bit .EN, en el siguiente escaneado se borran el bit .DN y el valor .POS.

### Modo Incremental

El modo Incremental manipula un elemento de la matriz cada vez que EnableIn pase de falso a verdadero.



El siguiente diagrama de temporización muestra la relación entre los bits de estado y la operación de la instrucción. La ejecución se produce solo en un escaneado en el que EnableIn pasa de falso a verdadero. Cada vez que esto ocurre, solo se manipula un elemento de la matriz. Si el valor de EnableIn sigue siendo verdadero durante más de un escaneado, la instrucción solo se ejecuta durante el primer escaneado.



El bit .EN se establece cuando el valor de EnableIn es verdadero. El bit .DN se establece cuando se ha manipulado el último elemento de la matriz. Cuando se ha manipulado el último elemento y el valor de EnableIn pasa a falso, se borran los valores del bit .EN, del bit .DN y de .POS.

La diferencia entre el modo Incremental y el modo Numérico a un régimen de un elemento por escaneado es:

El modo Numérico con cualquier cantidad de elementos por escaneado requiere solo una transición de falso a verdadero de EnableIn para iniciar la ejecución. La instrucción continúa ejecutando el número de elementos especificado en cada escaneado hasta su finalización, independientemente del estado de EnableIn.

El modo Incremental requiere que EnableIn cambie de falso a verdadero para manipular un elemento de la matriz.

### Operadores válidos

Operador	Descripción	Óptimo
+	Sumar	DINT, REAL

-	Restar/negativo	DINT, REAL
*	Multiplicar	DINT, REAL
/	Dividir	DINT, REAL
=	Igual	DINT, REAL
<	Menor que	DINT, REAL
<=	Menor o igual que	DINT, REAL
>	Mayor que	DINT, REAL
>=	Mayor o igual que	DINT, REAL
<>	No igual	DINT, REAL
**	Exponente (x a y)	DINT, REAL
ABS	Valor absoluto	DINT, REAL
ACS	Arcocoseno	REAL
AND	Y a nivel de bits	DINT
ASN	Arcoseno	REAL
ATN	Arcotangente	REAL
COS	Coseno	REAL
DEG	Radianes a grados	DINT, REAL
FRD	BCD a entero	DINT
LN	Logaritmo natural	REAL
LOG	Logaritmo de base 10	REAL
MOD	División de módulo	DINT, REAL
NOT	NO a nivel de bits	DINT
OR	O a nivel de bits	DINT
RAD	Grados a radianes	DINT, REAL
SIN	Seno	REAL
SQR	Raíz cuadrada	DINT, REAL
TAN	Tangente	REAL
TOD	Entero a BCD	DINT
TRN	Truncar	DINT, REAL
XOR	O exclusivo a nivel de bits	DINT

### Expresiones de formato

Para cada operador que utilice en una expresión, debe proporcionar uno o dos operandos (etiquetas o valores inmediatos). Se usa la siguiente tabla para dar formato a los operadores y operandos dentro de una expresión.

Para operadores que operan en:	Se usa este formato:	Ejemplo
Un operando	operador(operando)	ABS(tag)

Dos operandos	operand_a operador operand_b	tag_b + 5 tag_c AND tag_d (tag_e**2) MOD (tag_f / tag_g)
---------------	------------------------------------	--

### Determine el orden de operación

Las operaciones que escribe en la expresión se realizan mediante la instrucción en un orden prescrito, no necesariamente en el orden en que las escribe. Puede anular el orden de operación agrupando los términos entre paréntesis, forzando la instrucción a realizar una operación entre paréntesis antes de otras operaciones.

Las operaciones del mismo orden se realizarán de izquierda a derecha.

Orden	Operación
1	( )
2	ABS, ACS, ASN, ATN, COS, DEG, FRD, LN, LOG, RAD, SIN, SQR, TAN, TOD, TRN
3	**
4	- (negar), NOT
	* , /, MOD
6	- (restar), +
7	AND
8	XOR
9	OR

### Afectar a las marcas de estado matemático

Controladores	Afectar a las marcas de estado matemático
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	No
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí

### Fallos mayores/menores

Se producirá un fallo mayor si:	Tipo de fallo	Código de fallo
.POS < 0 o .LEN < 0	4	21

Consulte Índice a través de matrices para ver si hay fallos de indexación de matrices.

## Ejecución

### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	Consultar el Diagrama de flujo de FAL (Condición de salida de reglón es falsa)
La condición de entrada de reglón es verdadera	Consultar el Diagrama de flujo de FAL (Condición de salida de reglón es verdadera)
Post-escaneado	N/A

### Diagrama de flujo de FAL (Condición de salida de reglón es falsa)

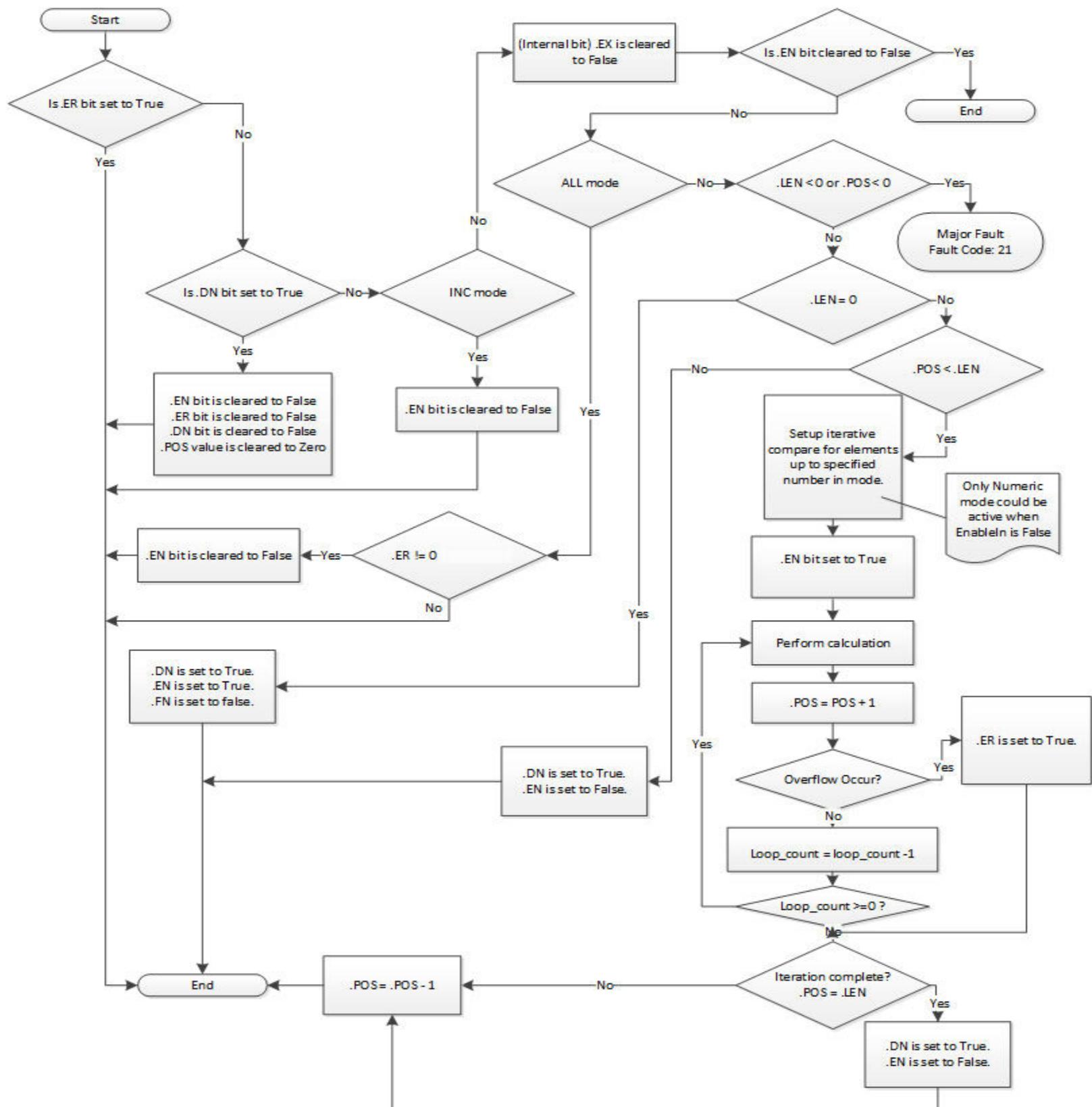


Diagrama de flujo de FAL (Condición de salida de reglón es verdadera)

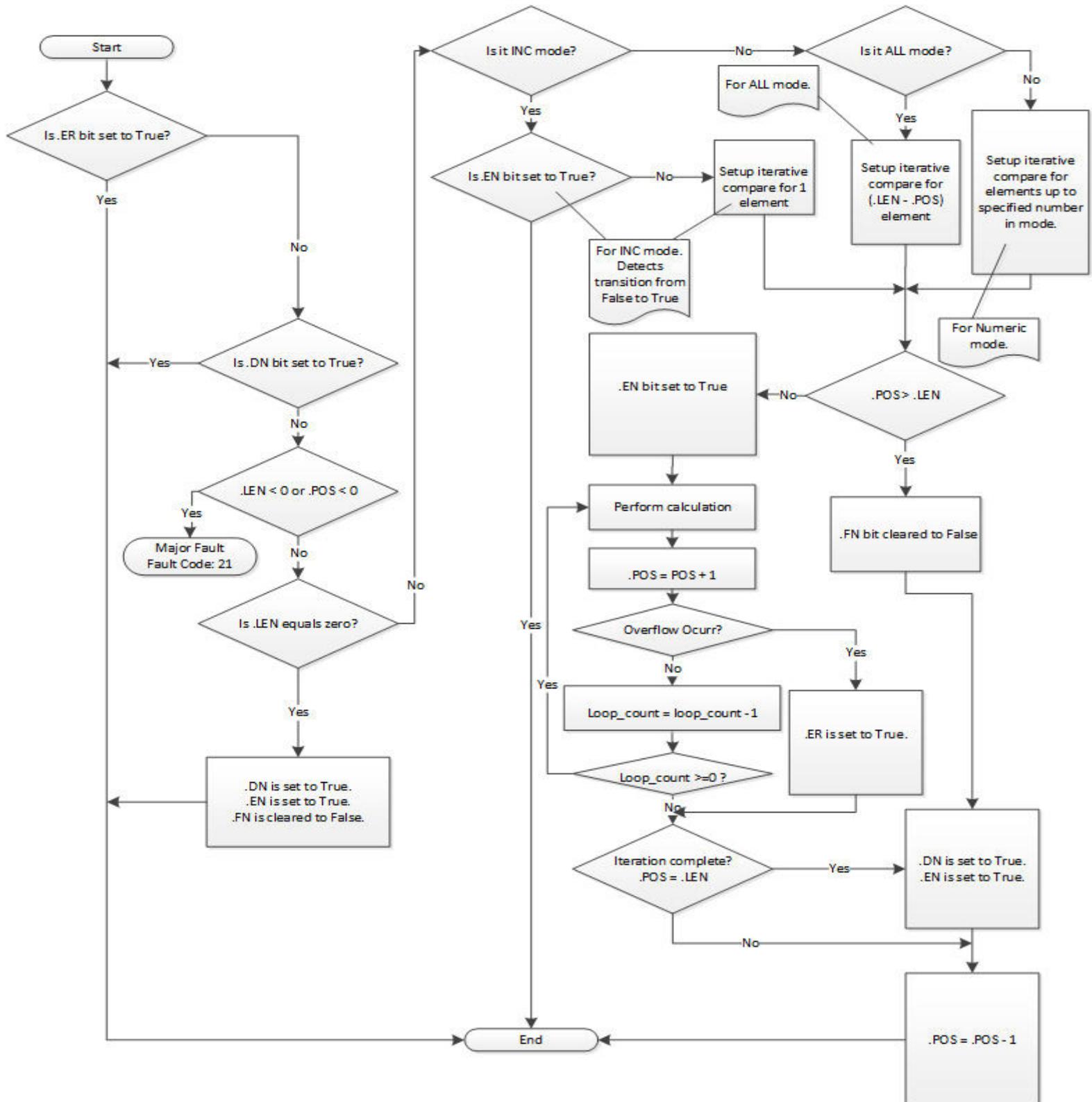


Diagrama de flujo de FAL (Modo Todos)

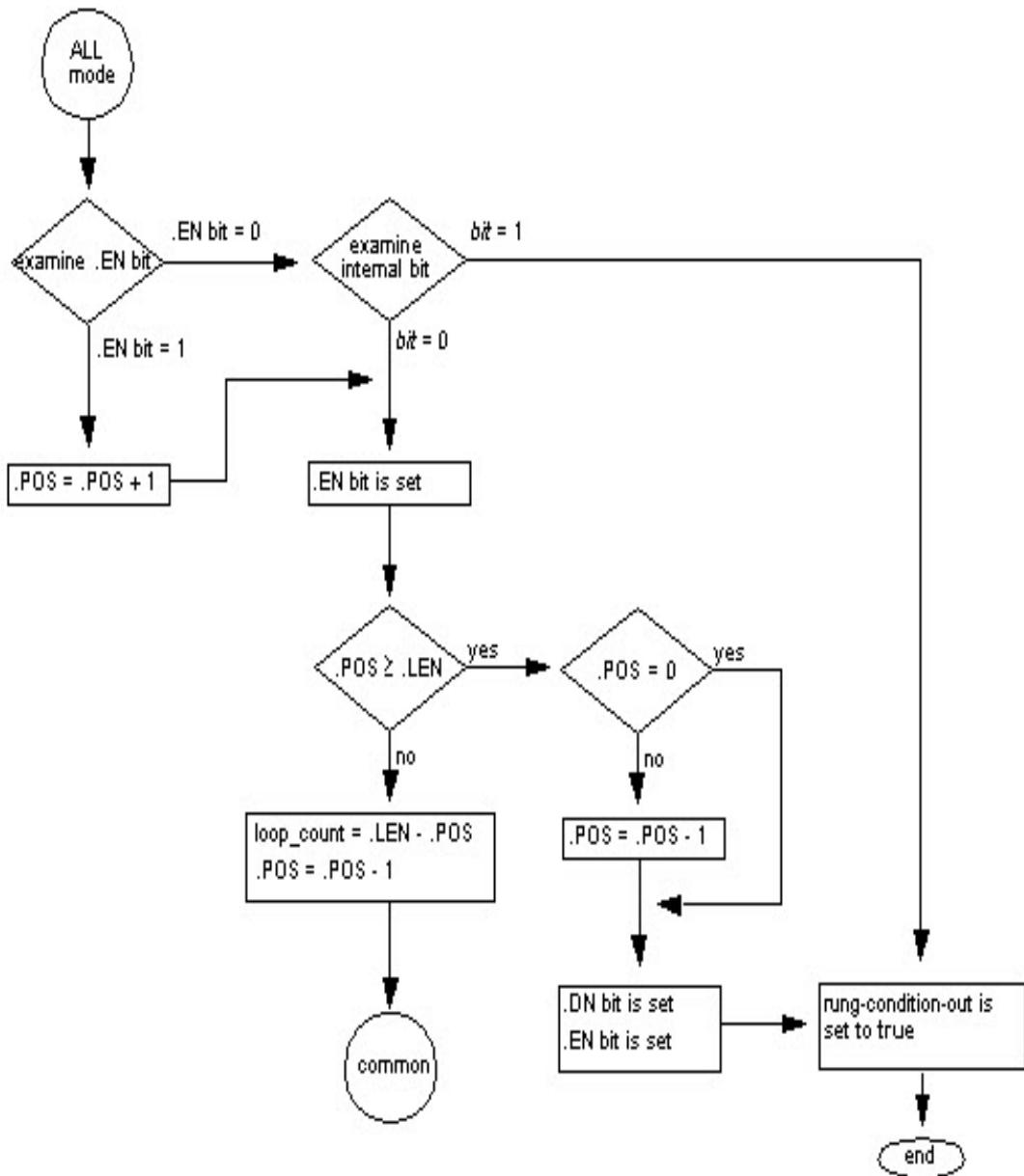


Diagrama de flujo de FAL (Modo Numérico)

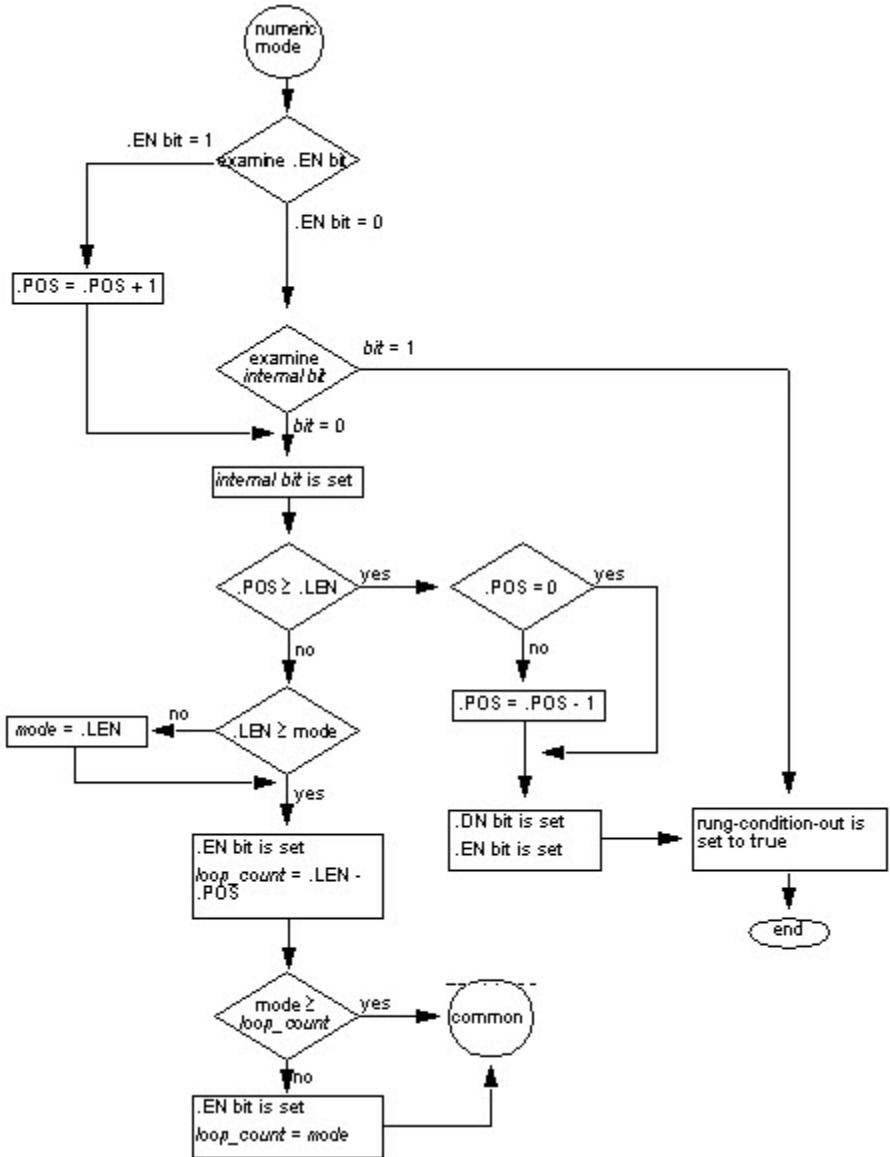
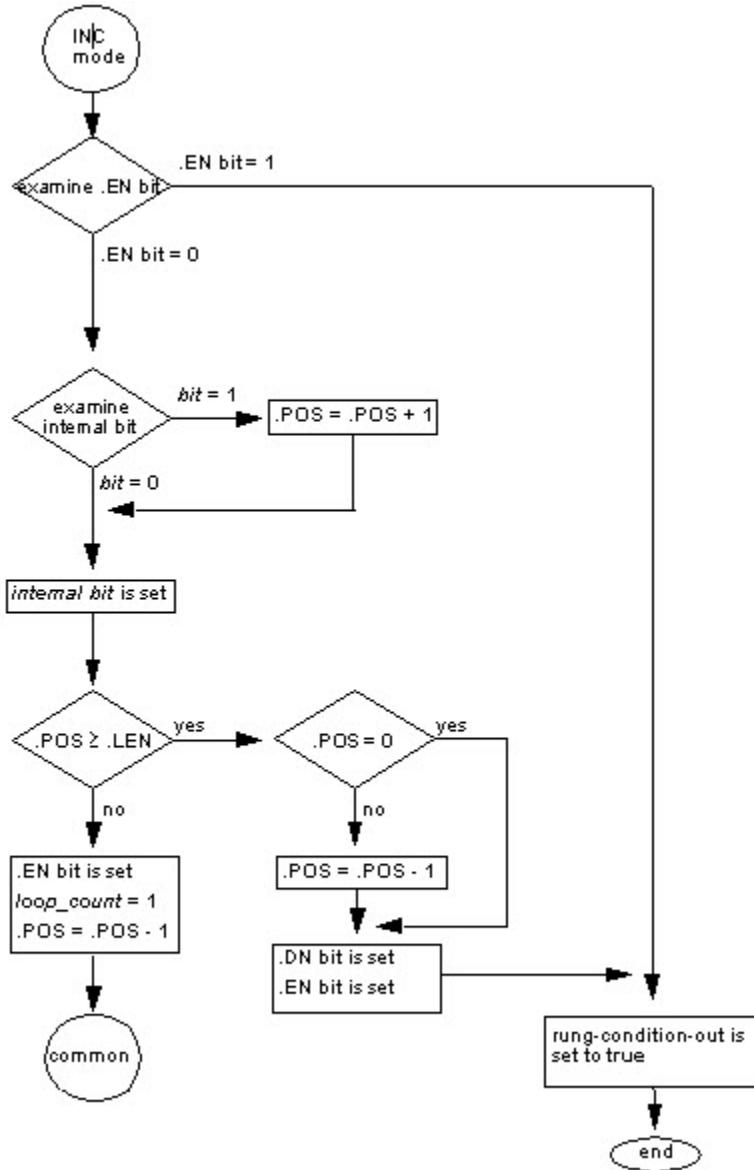


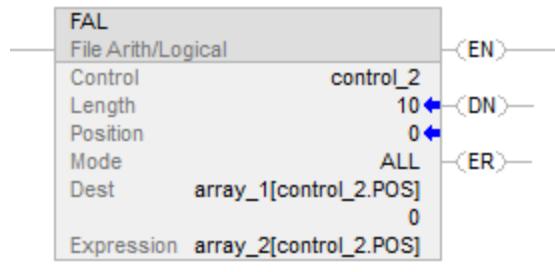
Diagrama de flujo de FAL (Modo Incremental)



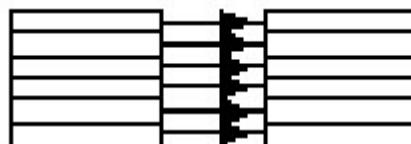
## Ejemplos

### Ejemplo 1

De matriz a matriz.

**Diagrama de escalera**

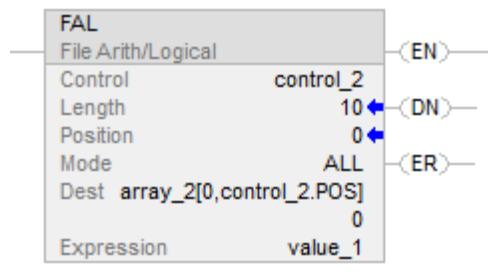
Cuando se habilita, la instrucción FAL copia cada elemento de array\_2 en la misma posición dentro de array\_1.



Expression: array\_2[control\_2.pos]      Destination: array\_1[control\_2.pos]

**Ejemplo 2**

Copia de elemento a matriz.

**Diagrama de escalera**

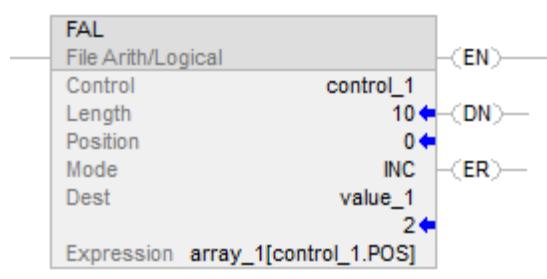
Cuando está habilitada, la instrucción FAL copia value\_1 dentro de las primeras 10 posiciones de la segunda dimensión de array\_2.



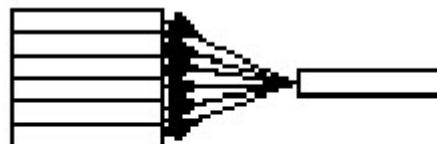
Expression: value\_1      Destination: array\_2[0,control\_2.pos]

**Ejemplo 3:**

Copia de matriz a elemento.



Cada vez que se habilita la instrucción FAL, copia el valor actual de array\_1 en value\_1. La instrucción FAL usa el modo incremental, de modo que solo se copia un valor de matriz cada vez que se habilita la instrucción. La próxima vez que se habilita la instrucción, esta sobrescribirá value\_1 utilizando el siguiente valor de array\_1.



Expression:  
array\_1[control\_1.POS]

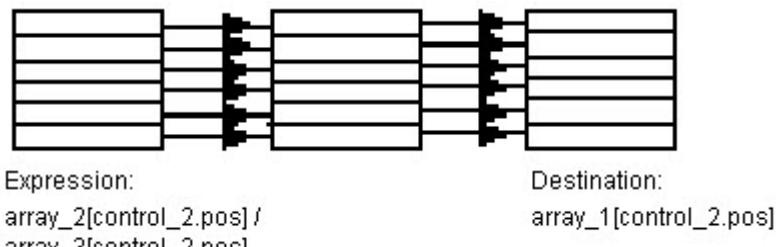
Destination:  
value\_1

**Ejemplo 4:**

Operación aritmética: matriz / matriz a matriz

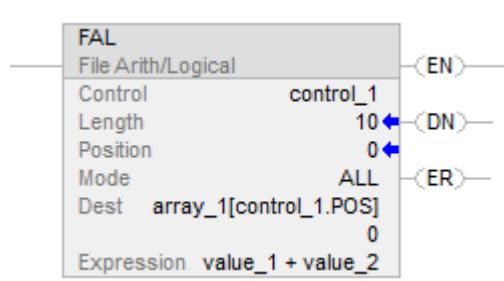


Cuando se habilita, la instrucción FAL divide el valor en la posición actual de array\_2 por el valor que haya en la posición actual de array\_3 y almacena el resultado en la posición actual de array\_1.

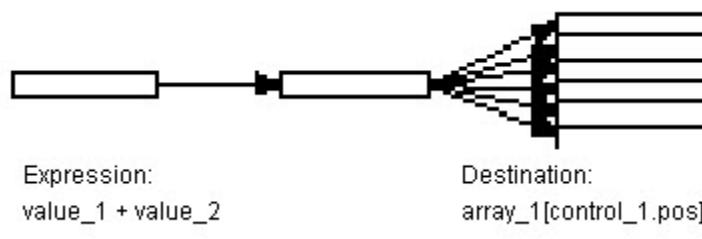


### Ejemplo 5:

Operación aritmética: matriz / matriz a matriz

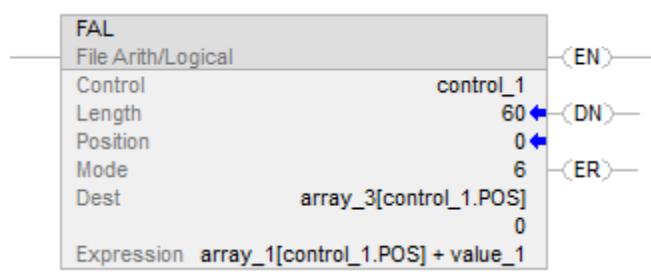


Cuando se habilita, la instrucción FAL añade value\_1 y value\_2 y almacena el resultado en la posición actual de array\_1.

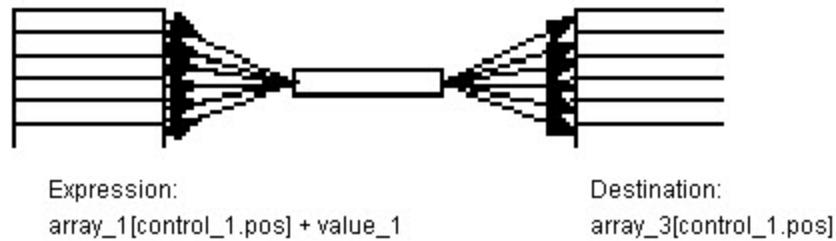


### Ejemplo 6:

Operación aritmética: matriz / elemento a matriz

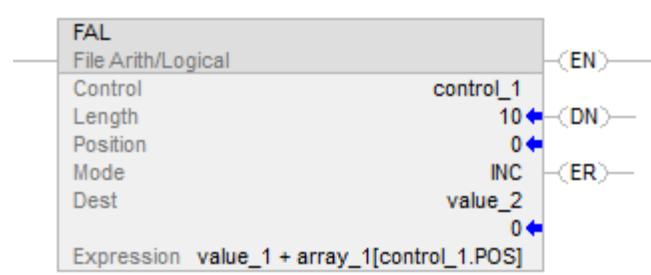


Cuando se habilita, la instrucción FAL añade el valor de la posición actual de array\_1 a value\_1 y almacena el resultado en la posición actual de array\_3. La instrucción debe ejecutarse 10 veces para que se manipulen completamente el array\_1 y array\_3.

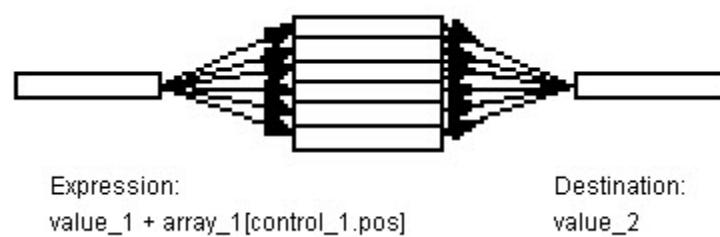


### Ejemplo 7:

Operación aritmética: (elemento + matriz) a elemento

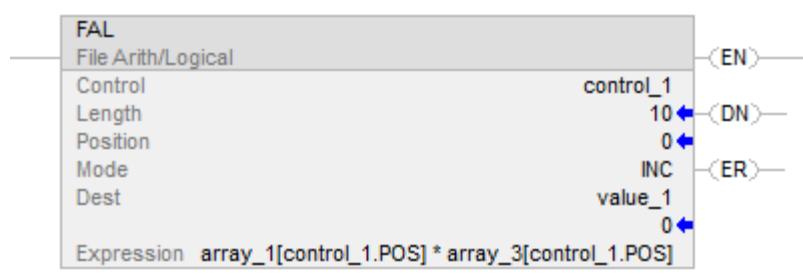


Cada vez que se habilita la instrucción FAL, añade value\_1 al valor actual de array\_1 y almacena el resultado en value\_2. La instrucción FAL usa el modo incremental, de modo que solo se añade un valor de matriz a value\_1 cada vez que se habilita la instrucción. La próxima vez que se habilita la instrucción, esta sobrescribirá value\_2.

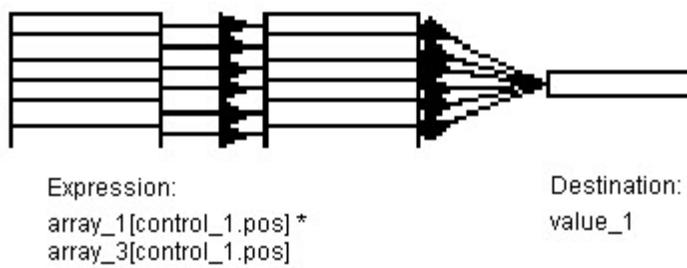


**Ejemplo 8:**

Operación aritmética: (elemento \* matriz) a elemento



Cuando se habilita la instrucción FAL, multiplica el valor actual de array\_1 por el valor actual de array\_3 y almacena el resultado en value\_1. La instrucción FAL usa el modo incremental, de modo que solo se multiplica un par de valores de matriz cada vez que se habilita la instrucción. La próxima vez que se habilita la instrucción, esta sobrescribirá value\_1.

**Consulte también**

[Instrucciones de archivos/miscláneas](#) en la página 527

[Operadores válidos](#) en la página 422

[Índice a través de matrices](#) en la página 936

[Conversiones de datos](#) en la página 927

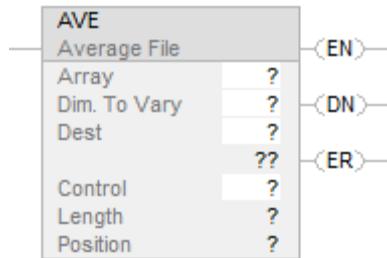
## Promedio de archivo (AVE)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

La instrucción AVE calcula la media de un conjunto de valores.

## Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

### Operандos

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción. Consulte Conversión de datos.

### Diagrama de escalera

Operando	Tipo	Formato	Descripción
Array Tag	SINT INT DINT REAL	etiqueta	Encuentra el promedio de los valores en esta matriz. Especifica el primer elemento del grupo de elementos para promediar No se utiliza CONTROL.POS en el subíndice
Dimension to vary	DINT	inmediato (0, 1, 2)	Qué dimensión a usar El orden de las dimensiones es: array[0,1,2].
Destination	SINT INT DINT REAL	etiqueta	Resultado de la operación.
Control	CONTROL	etiqueta	Estructura de control para la operación
Length	DINT	inmediato	Número de elementos de la matriz a promediar

Position	DINT	inmediato	Desplazamiento en la matriz especificada que identifica el elemento actual al que está accediendo la instrucción. El valor inicial es normalmente 0
----------	------	-----------	--

### Descripción

La instrucción AVE calcula el promedio de un conjunto de valores.

---

**Importante:** Asegúrese de que Length no cause que la instrucción exceda la Dimension to vary especificada. Si esto sucede, Destination será incorrecto. Para obtener más información, consulte Visualización de una matriz como un bloque de memoria.

---

Si ocurre un desbordamiento durante la evaluación de la expresión, las instrucciones leen más allá del final de una matriz, la instrucción establece el bit ER y detiene la ejecución

### Afectar a las marcas de estado matemático

Controladores	Afectar a las marcas de estado matemático
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Condicional, consulte Marcas de estado matemático.
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí

### Fallos mayores/menores

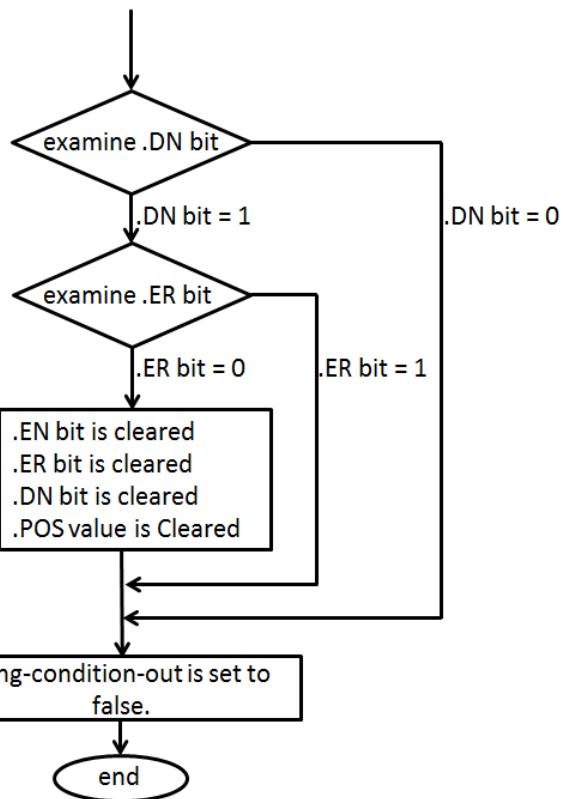
No es específico para esta instrucción. Consulte los Atributos comunes para los fallos relacionados con los operandos.

### Ejecución

#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	El bit .EN se borra. El bit .DN se borra. Si el bit .ER es cero durante el pre-escaneado, todos los bits de control (.DN, .EN, .EU, .EM, .UL, .IN y .FD) se borrarán a cero.

La condición de entrada de reglón es falsa.	Consulte el Diagrama de flujo de AVE (Falso)
La condición de entrada de reglón es verdadera.	La instrucción AVE calcula el promedio al agregar todos los elementos especificados en la matriz y dividir por el número de elementos.
Post-escaneado	N/A.

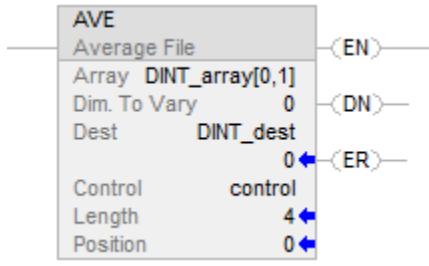
**Diagrama de flujo de AVE (Falso).****Ejemplo 1**

		dimension 1				
		0	1	2	3	4
dimension 0	0	20	19	18	17	16
	1	15	14	13	12	11
	2	10	9	8	7	6
	3	5	4	3	2	1

$$AVE = \frac{19 + 14 + 9 + 4}{4} = \frac{46}{4} = 11.5$$

dint\_ave = 12

### Diagrama de escalera



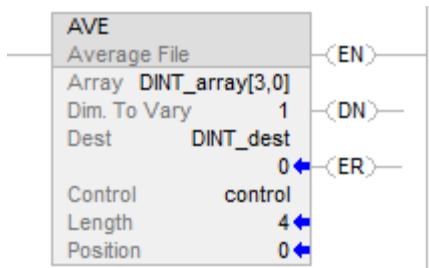
### Ejemplo 2

		dimension 1				
		0	1	2	3	4
dimension 0	0	20	19	18	17	16
	1	15	14	13	12	11
	2	10	9	8	7	6
	3	3	4	3	2	1

$$AVE = \frac{5 + 4 + 3 + 2 + 1}{5} = \frac{15}{5} = 3$$

dint\_ave = 3

### Diagrama de escalera



### Consulte también

[Atributos comunes](#) en la página 923

[Marcas de estado matemático](#) en la página 923

[Conversiones de datos](#) en la página 927

## Llenar archivo (FLL)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

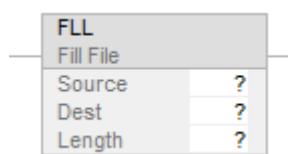
La instrucción FLL rellena un bloque de memoria con el valor de origen proporcionado. Source permanece sin cambios.

Si la matriz de destino es de tipo SINT, INT, DINT o REAL y el tipo de valor de origen es diferente, el valor de origen se convertirá en el tipo de destino antes de que se almacene. Los tipos de entero menores se convertirán en mayores mediante extensión de signo.

Si la matriz de destino es una estructura, el valor de origen se escribirá sin haberse convertido.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

#### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

#### Operандos

**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
- Los miembros de un operando de estructura se sobrescriben.
- A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.

Existen reglas de conversión de datos para utilizar diversos tipos de datos numéricos en una instrucción. Ver Conversión de Datos.

#### Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción
Source	SINT INT DINT REAL	inmediato etiqueta	Elemento a copiar

Destination	SINT INT DINT REAL estructura	etiqueta	Elemento inicial que se sobrescribirá mediante el valor de Source.
Length	DINT INT SINT	inmediato etiqueta	Número de elementos de destino a llenar.

El número de bytes rellenos es el menor de:

- Cantidad solicitada = Length x (número de bytes en un elemento de destino)
- El número de bytes en la etiqueta de destino

**Cons** El final de la etiqueta de destino se define como el último byte de la etiqueta de base. Si la etiqueta es una estructura, el final de la etiqueta es el último byte del último elemento de la estructura. Esto significa que la instrucción FLL podría escribir más allá del final de una matriz de miembros, pero nunca lo hará más allá de la etiqueta de base. Prueba y confirma que la instrucción FLL no cambia datos que no deban cambiarse.

Para obtener los mejores resultados posibles, Source y Destination deberían ser del mismo tipo. Se usa FLL para llenar una estructura con una constante, como cero.

Si inicializa una estructura, asegúrese de tener una instancia que contenga los valores iniciales y use COP para replicarla. FLL se puede utilizar, por ejemplo, para llenar de cero toda la estructura.

Si Source es:	Y el valor de Destination es:	Source se convierte en:
SINT, INT, DINT o REAL	SINT	SINT
SINT, INT, DINT o REAL	INT	INT
SINT, INT, DINT o REAL	DINT	DINT
SINT, INT, DINT o REAL	REAL	REAL

La conversión de enteros mayores en enteros menores producirá el truncado (se desechan los bits altos). Una vez que se convierte el origen, se escribe en el destino N veces, donde N = conteo de bytes. La extensión de signo resulta al realizar la conversión de enteros menores a enteros mayores. Los números de tipo REAL se redondean cuando se convierten en enteros.

### Afectar a las marcas de estado matemático

No

### Fallos mayores/menores

No es específico para esta instrucción. Consulte Índice a través de matrices para ver si hay fallos de indexación de matrices.

### Ejecución

#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	La instrucción rellena la memoria.
Post-escaneado	N/A

### Ejemplo

La instrucción FLL copia el número de elementos del destino especificados mediante Length del operando de origen de tipo DINT\_src en un destino de tipo REAL\_dest.

#### Diagrama de escalera



### Consulte también

[Instrucciones de archivos/miscláneas](#) en la página 527

[Índice a través de matrices](#) en la página 936

[Conversiones de datos](#) en la página 927

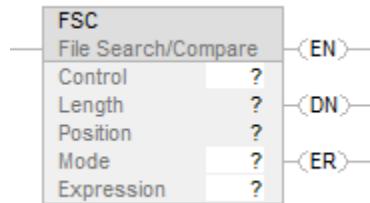
## Búsqueda y comparación de archivos (FSC)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

La instrucción FSC compara valores en una matriz, elemento por elemento.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

#### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

#### Operандos

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción. Consulte Conversión de datos.

#### Diagrama de escalera

Operando	Tipo	Formato	Descripción
Control	CONTRO L	Etiqueta	Estructura de control para la operación
Length	DINT	Inmediato	Número de elementos en la matriz a manipular
Position	DINT	Inmediato	Desplazamiento en una matriz El valor inicial suele ser 0
Mode	DINT	Inmediato	Cómo distribuir la operación Selecciona INC, ALL o introduce un número en el rango de 1 a 2147483647
Expression	SINT INT DINT REAL STRING	Inmediato Etiqueta	Una expresión que consiste en etiquetas y/o valores inmediatos separados por operadores

## Estructura de CONTROL

Mnemónico	Tipo de datos	Descripción
.EN	BOOL	El bit de habilitación indica que la instrucción FSC está habilitada.
.DN	BOOL	El bit de efectuado se establece cuando la instrucción ha operado en el último elemento (.POS = .LEN).
.ER	BOOL	El bit de error no se modifica.
.IN	BOOL	El bit de inhibición indica que la instrucción FSC detectó una comparación verdadera. Debe borrar este bit para continuar la operación de búsqueda.
.FD	BOOL	El bit de encontrado indica que la instrucción FSC detectó una comparación verdadera.
.LEN	DINT	La longitud especifica el número de elementos de la matriz en el que opera la instrucción.
.POS	DINT	La posición contiene la posición del elemento actual al que tiene acceso la instrucción.

### Descripción

Cuando EnableIn de la instrucción FSC realiza una transición de falso a verdadero, la expresión se evalúa sobre el modo especificado de iteración.

Si el resultado de la evaluación es verdadero, la instrucción establece el bit .FD, y el bit .POS refleja la posición de la matriz donde la instrucción encontró la comparación verdadera. La instrucción establece el bit .IN para evitar más iteraciones.

### Seleccionar el modo de funcionamiento

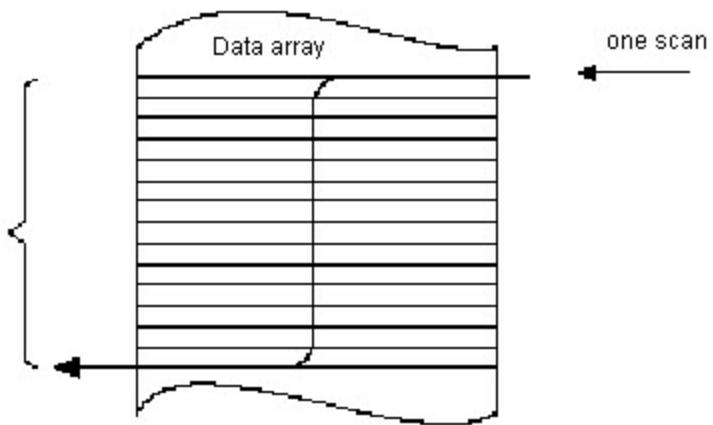
Para las instrucciones del FSC, el modo le indica al controlador cómo distribuir la operación de la matriz.

Si desea:	Seleccione este modo:
Operar en todos los elementos especificados en una matriz antes de continuar con la siguiente instrucción.	Todos
Distribuir el funcionamiento de la matriz sobre una serie de escaneados. Introducir el número de elementos para operar por escaneado (1-2147483647).	Numérico

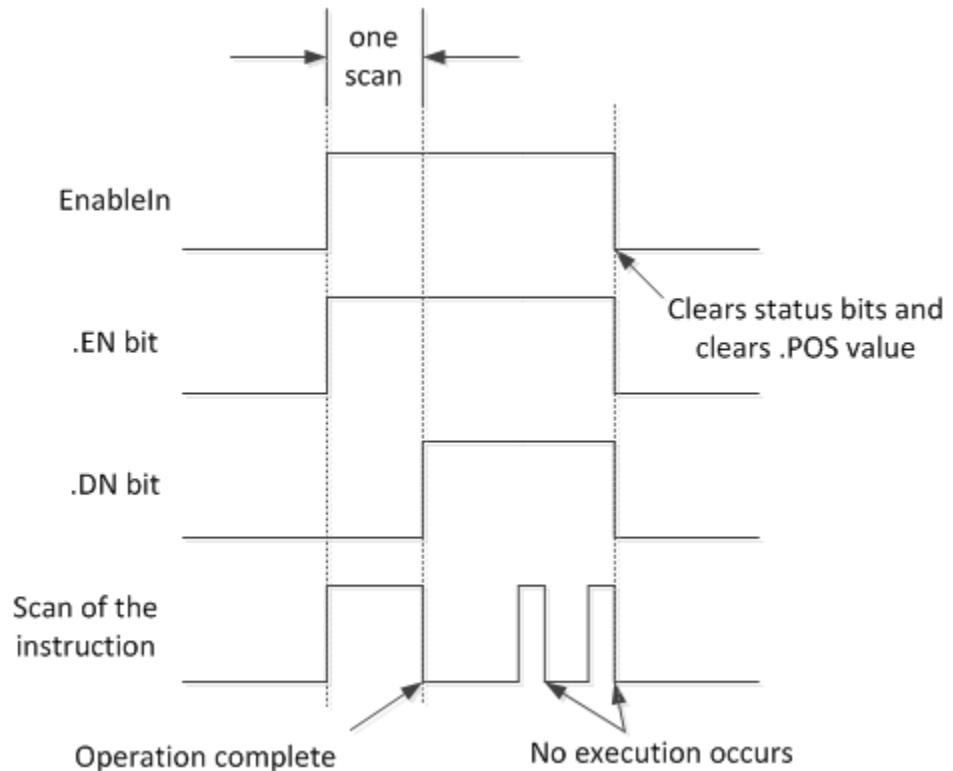
Manipular un elemento de la matriz cada vez que EnableIn pase de falso a verdadero.	Incremental
---	-------------

### Modo Todos

En el modo Todos, todos los elementos especificados en la matriz se utilizan antes de continuar con la siguiente instrucción. La operación comienza cuando EnableIn de la instrucción realiza una transición de falso a verdadero. El valor de posición (.POS) en la estructura de control apunta al elemento de la matriz que la instrucción está utilizando actualmente. La operación se detiene bajo dos condiciones. Cuando el valor .POS es igual o superior al valor .LEN, Y cuando la expresión se evalúa como verdadera.



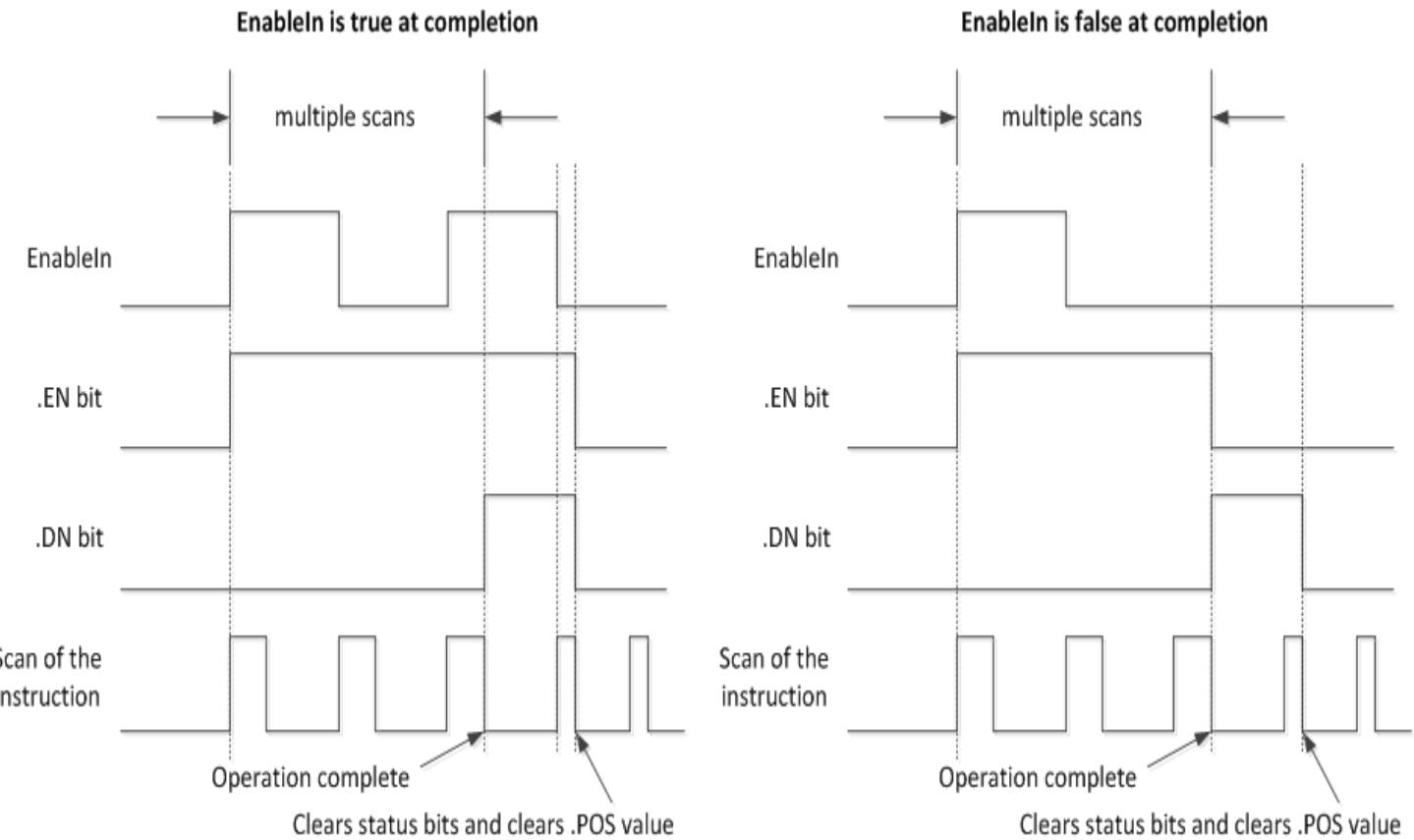
El siguiente diagrama de temporización muestra la relación entre los bits de estado y la operación de la instrucción. Cuando finaliza la ejecución de la instrucción, el bit .DN es verdadero. El bit .EN y el valor .POS se borran cuando EnableIn es falso. Solo entonces puede desencadenarse otra ejecución de la instrucción mediante una transición de falso a verdadero de EnableIn.



### Modo Numérico

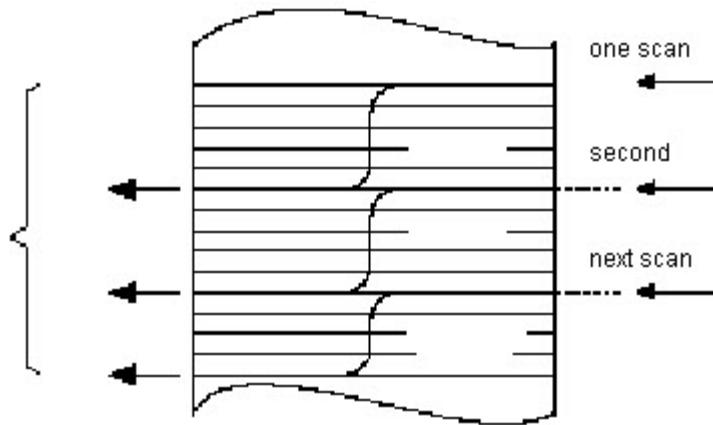
El modo Numérico distribuye el funcionamiento de la matriz sobre una serie de escaneados. Este modo es útil cuando se trabaja con datos que no son críticos para el tiempo o grandes cantidades de datos. Se introduce el número de elementos para operar durante cada escaneado, lo que mantiene el tiempo de escaneado más corto.

La ejecución se desencadena cuando EnableIn realiza una transición de falso a verdadero. Una vez desencadenada, la instrucción se ejecuta cada vez que se escanea con el número de escaneados necesarios para completar la operación sobre toda la matriz. Cuando está desencadenada, EnableIn puede cambiar repetidamente sin interrumpir la ejecución de la instrucción.



Evite utilizar los resultados de una instrucción de archivo que funcione en modo numérico hasta que el bit .DN o .IN sea verdadero.

El siguiente diagrama de temporización muestra la relación entre los bits de estado y la operación de la instrucción. Cuando finaliza la ejecución de la instrucción, el bit .DN es verdadero.

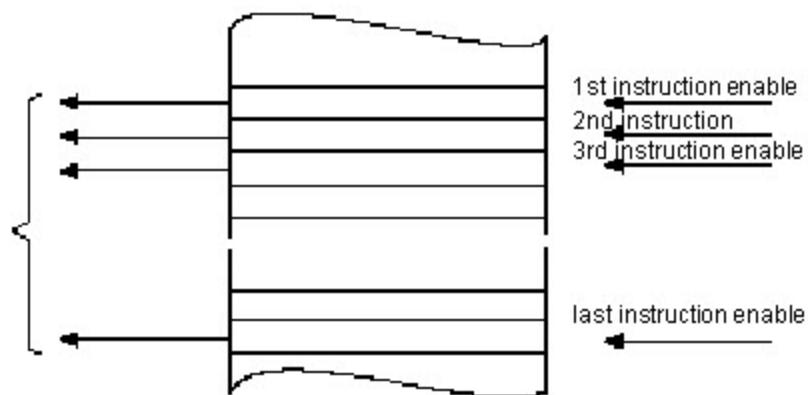


Si EnableIn es verdadero al finalizar, los bits .EN y .DN son verdaderos hasta que EnableIn pase a falso. Cuando EnableIn se vuelve falso, estos bits y el valor .POS se borran.

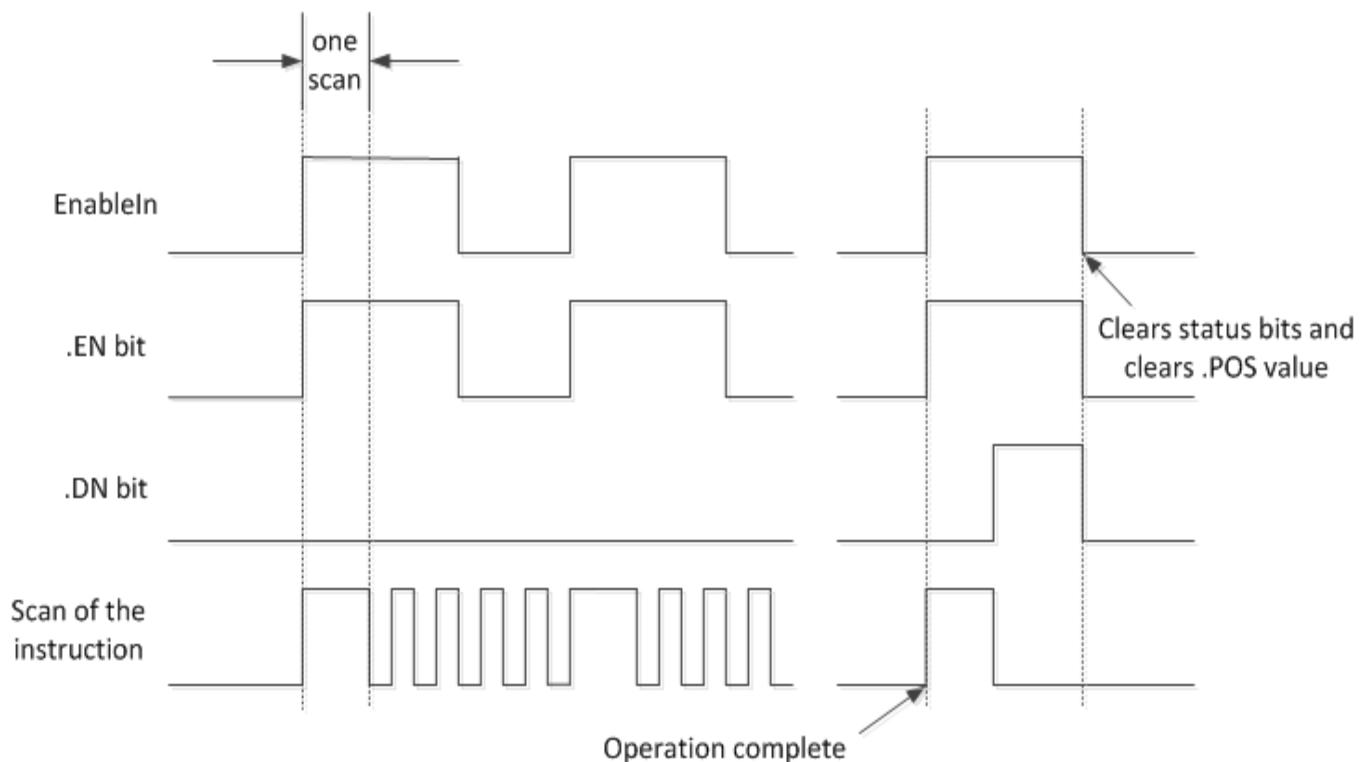
Si EnableIn es falso al finalizar, el bit .EN se borra inmediatamente. Después de borrarse el bit .EN, en el siguiente escaneado se borran el bit .DN y el valor .POS.

### Modo Incremental

El modo Incremental manipula un elemento de la matriz cada vez que EnableIn pase de falso a verdadero.



El siguiente diagrama de temporización muestra la relación entre los bits de estado y la operación de la instrucción. La ejecución se produce solo en un escaneado en el que EnableIn pasa de falso a verdadero. Cada vez que esto ocurre, solo se manipula un elemento de la matriz. Si EnableIn permanece verdadero durante más de un escaneado, la instrucción solo se ejecuta durante el primer escaneado.



El bit **.EN** se establece cuando la condición de entrada de reglón es verdadera. El bit **.DN** se establece cuando se ha manipulado el último elemento de la matriz. Cuando se ha manipulado el último elemento y la condición de entrada de reglón se vuelve falsa, se borran el bit **.EN**, el bit **.DN** y el valor **.POS**.

La diferencia entre el modo Incremental y el modo Numérico a un régimen de un elemento por escaneado es:

El modo Numérico con cualquier cantidad de elementos por escaneado requiere solo una transición de falso a verdadero de EnableIn para iniciar la ejecución. La instrucción continúa ejecutando el número de elementos especificado en cada escaneado hasta su finalización, independientemente del estado de EnableIn.

El modo Incremental requiere que EnableIn cambie de falso a verdadero para manipular un elemento de la matriz.

### Operadores válidos

Operador	Descripción	Permitido en seguridad
+	Sumar	X
-	Restar/negativo	X
*	Multiplicar	X
/	Dividir	X
=	Igual	X
<	Menor que	X
<=	Menor o igual que	X
>	Mayor que	X
>=	Mayor o igual que	X
<>	No igual	X
**	Exponente (x a y)	
ABS	Valor absoluto	
ACS	Arcocoseno	
AND	Y a nivel de bits	X
ASN	Arcoseno	
ATN	Arcotangente	
COS	Coseno	
DEG	Radianes a grados	
FRD	BCD a entero	
LN	Logaritmo natural	
LOG	Logaritmo de base 10	
MOD	División de módulo	
NOT	NO a nivel de bits	X
OR	O a nivel de bits	X
RAD	Grados a radianes	
SIN	Seno	
SQR	Raíz cuadrada	
TAN	Tangente	
TOD	Entero a BCD	
TRN	Truncar	

XOR	O exclusivo a nivel de bits	X
-----	-----------------------------	---

### Expresiones de formato

Para cada operador que utilice en una expresión, debe proporcionar uno o dos operandos (etiquetas o valores inmediatos). Se usa la siguiente tabla para dar formato a los operadores y operandos dentro de una expresión.

Para operadores que operan en:	Se usa este formato:	Ejemplo
Un operando	operador(operando)	ABS(tag)
Dos operandos	operand_a operador operand_b	tag_b + 5 tag_c AND tag_d (tag_e**2) MOD (tag_f / tag_g)

### Determine el orden de operación

Las operaciones que escribe en la expresión se realizan mediante la instrucción en un orden prescrito, no necesariamente en el orden en que las escribe. Puede anular el orden de operación agrupando los términos entre paréntesis, forzando la instrucción a realizar una operación entre paréntesis antes de otras operaciones.

Las operaciones del mismo orden se realizarán de izquierda a derecha.

Orden	Operación
1	( )
2	ABS, ACS, ASN, ATN, COS, DEG, FRD, LN, LOG, RAD, SIN, SQR, TAN, TOD, TRN
3	**
4	- (negar), NOT
5	*, /, MOD
6	- (restar), +
7	AND
8	XOR
9	OR
10	<, <=, >, >=, =, <>

### Uso de cadenas en una expresión

Para usar cadenas de caracteres ASCII en una expresión, siga estas pautas:

Una expresión puede comparar dos etiquetas de cadenas.

No puede introducir caracteres ASCII directamente en la expresión.

Solo se permiten los siguientes operadores:

Operador	Descripción
=	Igual
<	Menor que
<=	Menor o igual que
>	Mayor que
>=	Mayor o igual que
<>	No igual

Las cadenas son iguales si sus caracteres coinciden.

Los caracteres ASCII distinguen entre mayúsculas y minúsculas. La mayúscula A (\$41) no es equivalente a la minúscula a (\$61).

Lo que determina si una cadena es mayor o menor que otra son los valores hexadecimales de los caracteres.

Cuando las dos cadenas se ordenan como en un directorio telefónico, el orden de las cadenas determina cuál es superior.

ASCII Characters	Hex Codes
1ab	\$31\$61\$62
1b	\$31\$62
A	\$41
AB	\$41\$42
B	\$42
a	\$61
ab	\$61\$62

#### Afectar a las marcas de estado matemático

Controladores	Afectar a las marcas de estado matemático
ControlLogix 5580	No
CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570	Sí

#### Fallos mayores/menores

Se producirá un fallo mayor si:	Tipo de fallo	Código de fallo
.POS < 0 o .LEN < 0	4	21

Consulte los Atributos comunes para los fallos relacionados con los operandos.  
Consulte Índice a través de matrices para ver si hay fallos de indexación de matrices.

### Ejecución

#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	Consultar el diagrama de flujo de FSC (la condición de salida de reglón es falsa)
La condición de entrada de reglón es verdadera	Consultar el diagrama de flujo de FSC (la condición de salida de reglón es verdadera)
Post-escaneado	N/A

Diagrama de flujo de FSC (la condición de salida de reglón es falsa)

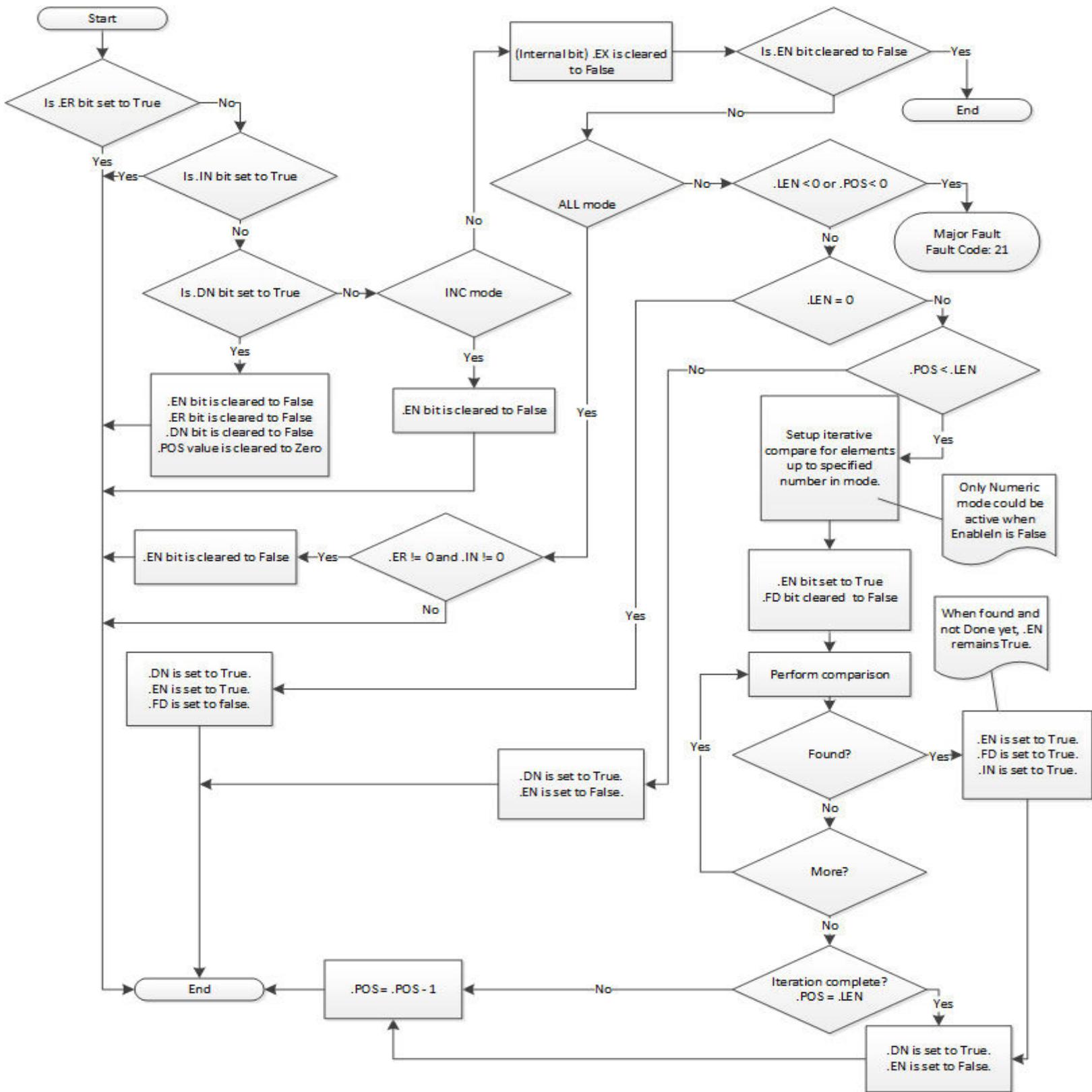
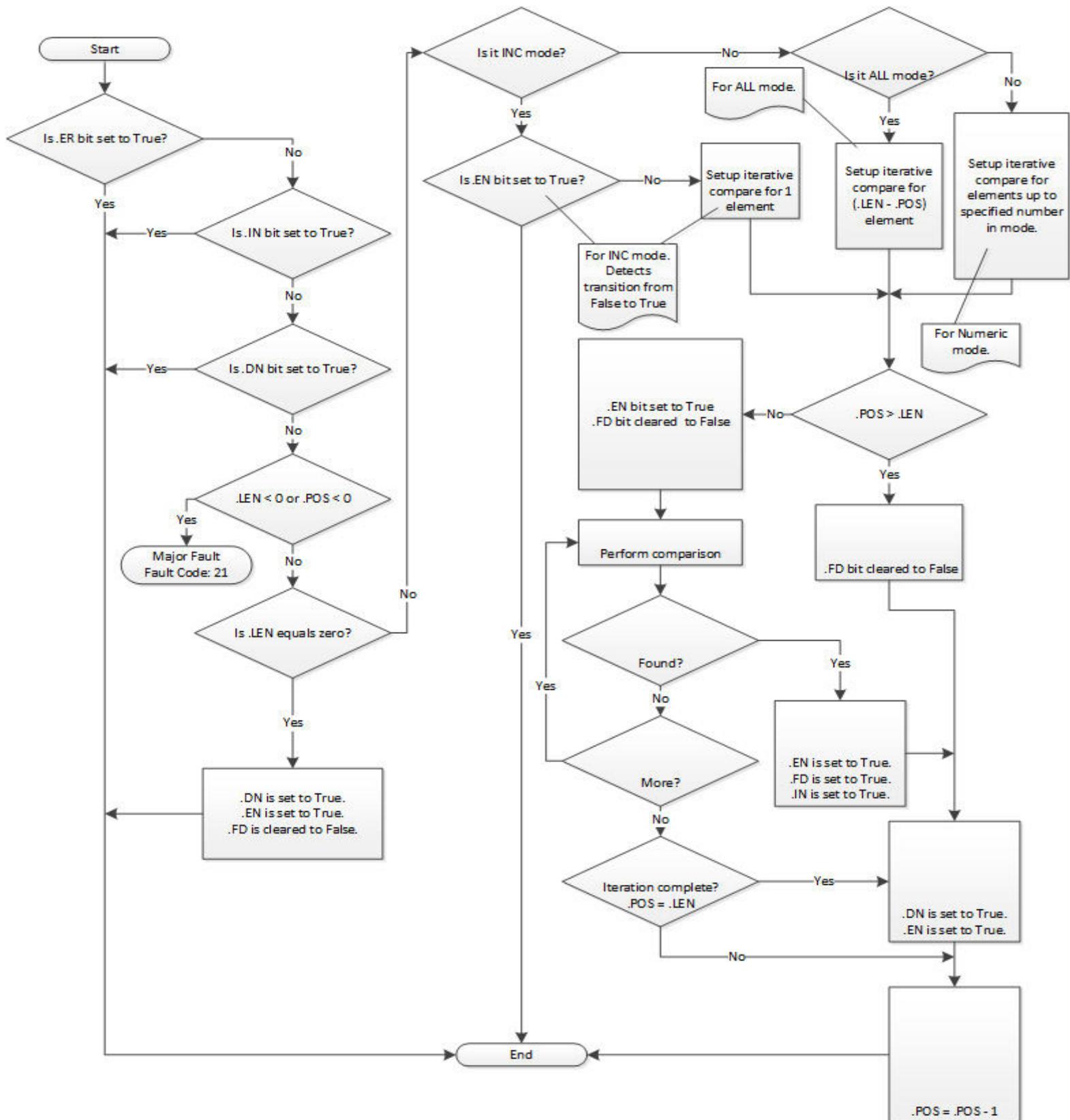


Diagrama de flujo de FSC (la condición de salida de reglón es verdadera)



## Diagrama de flujo de FSC (subflujo común de FSC)

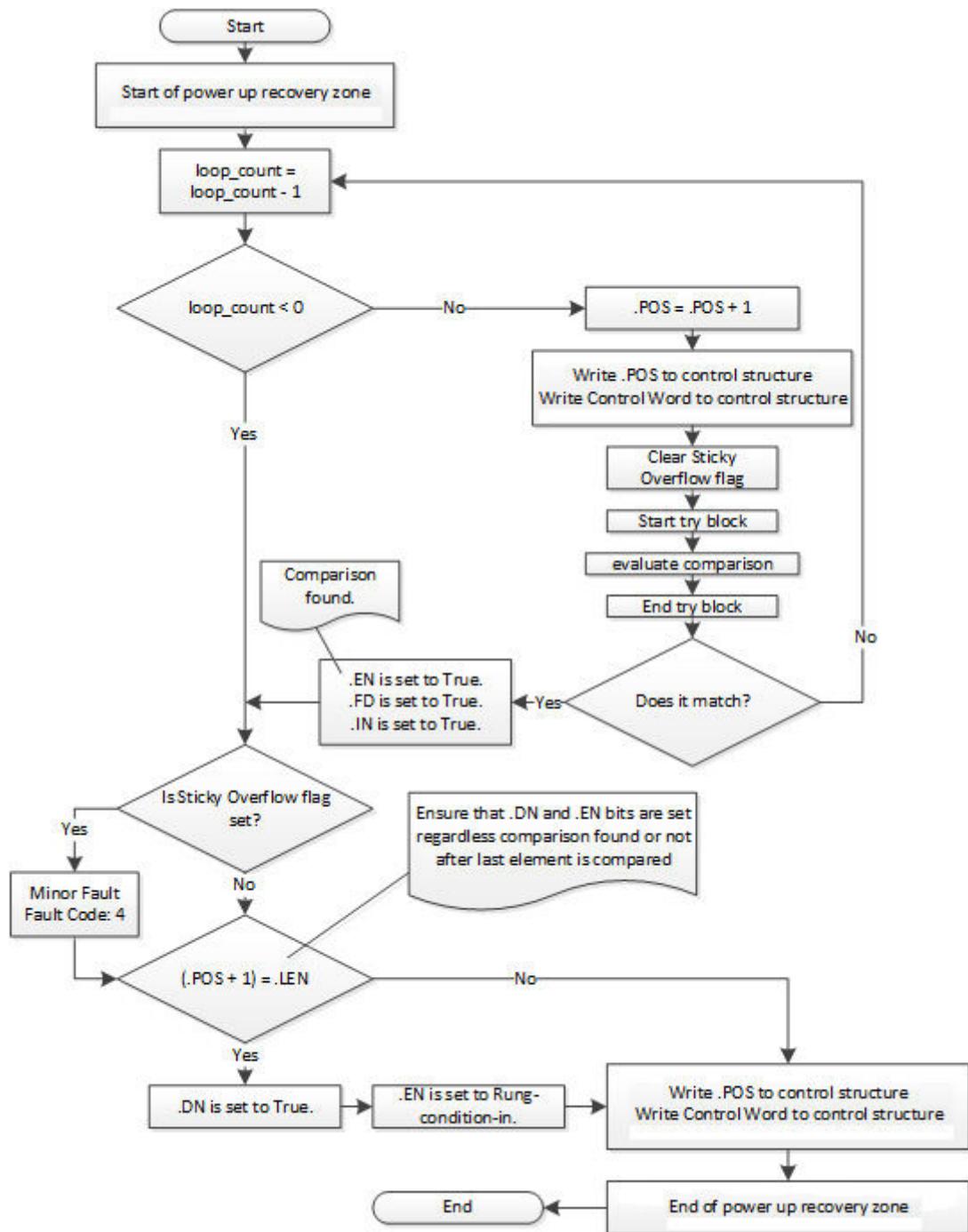
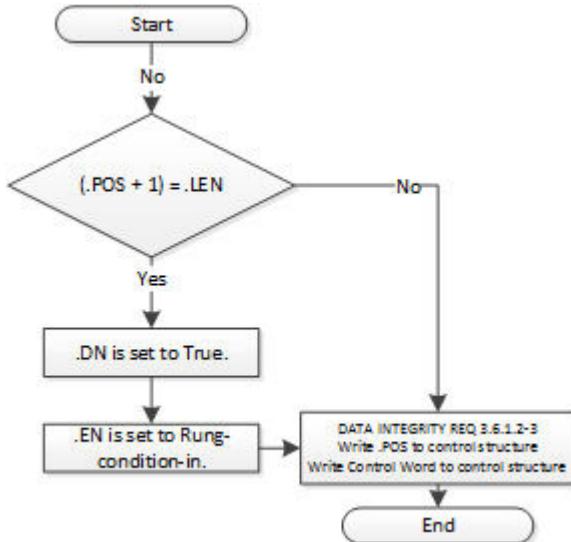
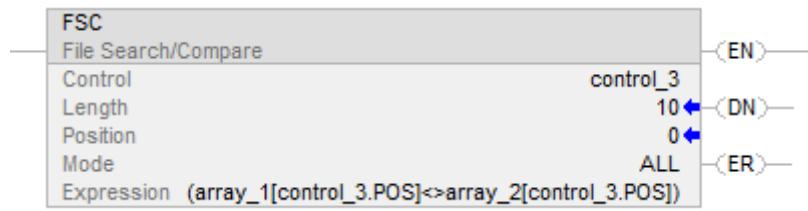


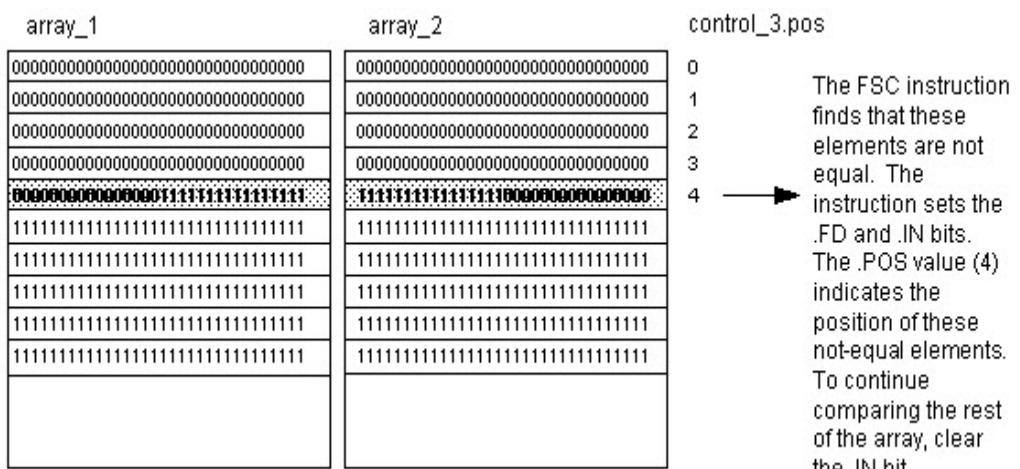
Diagrama de flujo de FSC (subflujo de excepción común de FSC)

**Ejemplos****Ejemplo 1**

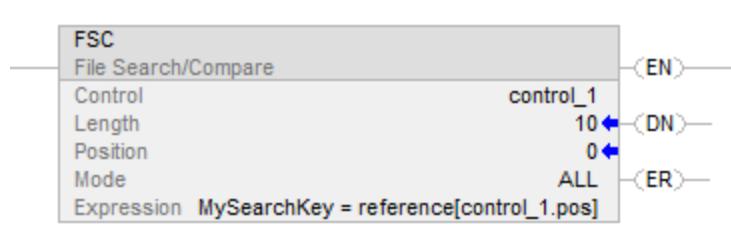
Buscar entre dos matrices DINT elementos que no son iguales.

**Diagrama de escalera**

Cuando está habilitada, la instrucción FSC compara cada uno de los 10 primeros elementos de array\_1 con los elementos correspondientes de array\_2. Cuando se encuentra un elemento que no es igual, se establecen los bits FD e IN. POS identifica la ubicación de los elementos no iguales. Borre el bit IN para buscar el resto de la matriz.

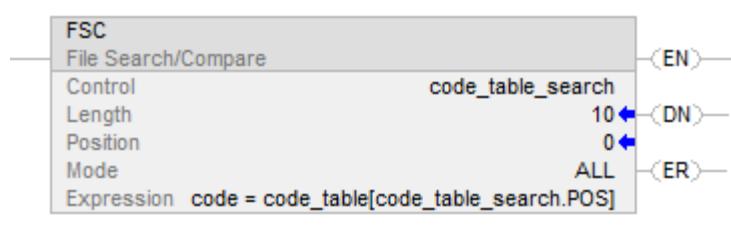


### Ejemplo 2



Buscar una coincidencia en una matriz de estructuras.

### Ejemplo 3



Buscar una cadena en una matriz de cadenas.

Cuando está habilitada, la instrucción FSC compara los caracteres en code con 10 elementos de code\_table.

### Consulte también

[Instrucciones de archivos/miscláneas](#) en la página 527

[CMP](#) en la página 370

[FAL](#) en la página 536

[Índice a través de matrices](#) en la página 936

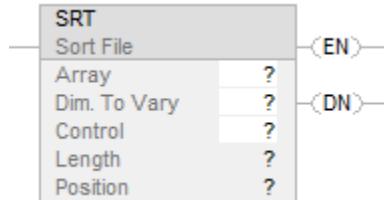
## Clasificación de archivo (SRT)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

La instrucción SRT ordena un conjunto de valores en una dimensión (Dim to vary) de la Array en orden ascendente.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

#### Texto estructurado

```
SRT(Array,Dimtovary,Control);
```

#### Operандos

#### Diagrama de escalera

Operando	Tipo	Formato	Descripción
Array	SINT INT DINT REAL	Etiqueta de matriz	la matriz que se va a ordenar. especifica el primer elemento del grupo de elementos a ordenar.

Dimension to vary	DINT	Inmediato (0, 1, 2)	la dimensión que se va a usar. el orden de las dimensiones es: array[0,1,2]
Control	CONTROL	Etiqueta	estructura de control para la operación
Length	DINT	Inmediato	número de elementos de la matriz a ordenar
Position	DINT	Inmediato	elemento actual de la matriz el valor inicial suele ser 0

### Texto estructurado

Operando	Tipo	Formato	Descripción
Array	SINT INT DINT REAL	Etiqueta de matriz	la matriz que se va a ordenar. especifica el primer elemento del grupo de elementos a ordenar.
Dimension to vary	DINT	Inmediato (0, 1, 2)	la dimensión que se va a usar. el orden de las dimensiones es: array[0,1,2]
Control	CONTROL	Etiqueta	estructura de control para la operación
Length	DINT	Inmediato	Número de elementos de la matriz a ordenar. A los valores de Length y Position especificados se accede desde los miembros .LEN y .POS de la estructura CONTROL.
Position	DINT	Inmediato	elemento actual de la matriz el valor inicial suele ser 0 A los valores de Length y Position especificados se accede desde los miembros .LEN y .POS de la estructura CONTROL.

Consulte Sintaxis de texto estructurado para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

### Estructura de CONTROL

Mnemónico	Tipo de datos	Descripción
.EN	BOOL	El bit de habilitación indica que la instrucción SRT está habilitada.
.DN	BOOL	El bit de efectuado se establece cuando la instrucción ha operado en el último de la Array.
.ER	BOOL	El bit de error se establece cuando .LEN < 0 o .POS < 0. Además, cualquiera de estas condiciones genera un fallo grave. Si se ha establecido .ER, la instrucción no se ejecuta.
.LEN	DINT	La palabra longitud especifica el número de elementos en la matriz en los que opera la instrucción.

.POS	DINT	La palabra posición identifica el elemento actual al que está accediendo la instrucción.
------	------	--

### Descripción

La instrucción SRT ordena un conjunto de valores en una dimensión (Dim to vary) de la Array en orden ascendente.

---

**Importante:** Comprueba y confirma que la instrucción no modifica datos que usted no desea que sean modificados.

---

La instrucción SRT opera en la memoria continua de datos. Solo para Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570, el ámbito de la instrucción se ve limitado por la etiqueta de base. La instrucción SRT no escribirá datos fuera de la etiqueta de base, pero puede pasar los límites de los miembros. Si especifica una matriz que sea miembro de una estructura y la longitud supera el tamaño de una matriz, debe probar y conformar que la instrucción SRT no modifique datos que usted no quiera que sean modificados.

En Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580., los datos se ven limitados por el miembro especificado.

En esta instrucción de transición, la escalera de relé cambia el valor de condición de entrada de reglón de falso a verdadero para que la instrucción se ejecute.

### Afectar a las marcas de estado matemático

Controladores	Afectar a las marcas de estado matemático
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	No
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí

### Fallos mayores/menores

Se producirá un fallo mayor si:	Tipo de fallo	Código de fallo
.POS < 0 o .LEN < 0	4	21
Dimension to vary > número de dimensiones	4	20
Length > final de la matriz	4	20

Consulte *Atributos comunes* para los fallos relacionados con los operandos

### Ejecución

#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A.
La condición de entrada de reglón es falsa	El bit .EN se borra a falso El bit .EN se borra a falso El bit .DN se borra a falso
La condición de entrada de reglón es verdadera	La instrucción se ejecuta
Post-escaneado	N/A.

#### Texto estructurado

Condición/estado	Acción realizada
Pre-escaneado	Consultar Pre-escaneado en la tabla de diagrama de escalera
Ejecución normal	Dado que esta instrucción requiere que se realice una transición, se ejecuta en falso y después en verdadero. Para obtener información detallada, vea la tabla Diagrama de escalera.
Post-escaneado	Consultar Post-escaneado en la tabla Diagrama de escalera.

### Ejemplos

#### Ejemplo 1

Ordenar DINT\_array, que es DINT[4,5].

		dimension 1							dimension 1				
		0	1	2	3	4			0	1	2	3	4
dimension 0	dimension 0	20	19	18	17	16	dimension 0	20	19	18	17	16	
		15	14	13	12	11		15	14	13	12	11	
0	dimension 0	10	9	8	7	6	1	10	9	8	7	6	
1		5	4	3	2	1	2	5	4	3	2	1	

### Diagrama de escalera



### Texto estructurado

```
IF sort1 then
```

```
    control_1.LEN := 4;
```

```
    control_1.POS := 0;
```

```
    SRT (DINT_array[0,2],0,control_1);
```

```
END_IF;
```

### Ejemplo 2

Ordenar DINT\_array, que es DINT[4,5].

		Before							After						
		dimension 1							dimension 1						
		subscripts	0	1	2	3	4			subscripts	0	1	2	3	4
dimension 0	0		20	19	18	17	16			0	20	19	18	17	16
	1		15	14	13	12	11			1	15	14	13	12	11
	2		10	9	8	7	6			2	6	7	8	9	10
	3		5	4	3	2	1			3	5	4	3	2	1

### Diagrama de escalera



### Texto estructurado

```
ctrl.LEN := 4;
```

```
ctrl.POS := 0;
```

```
SRT(DINT_array[0,2],0,ctrl);
```

### Consulte también

[Instrucciones de archivos/miscláneas](#) en la página 527

[Promedio de archivo \(AVE\)](#) en la página 556

[Conversiones de datos](#) en la página 927

[Atributos comunes](#) en la página 923

[Sintaxis de texto estructurado](#) en la página 955

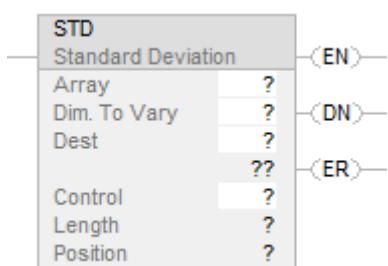
## Desviación estándar de archivo (STD)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

La instrucción STD calcula la desviación estándar de un conjunto de valores en una dimensión de la Array y almacena el resultado en el Destination.

### Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

### Operandos

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción. Consulte Conversión de datos.

### Diagrama de escalera

Operando	Tipo	Formato	Descripción
Array	SINT INT DINT REAL	etiqueta de matriz	Averigua la desviación estándar de los valores de esta matriz. Especifica el primer elemento del grupo de elementos a usar a la hora de calcular la desviación estándar.
Dimension to vary	DINT	inmediato (0, 1, 2)	la dimensión que se va a usar. el orden de las dimensiones es: array[0,1,2]
Destination	REAL	etiqueta	Resultado de la operación.
Control	CONTROL	etiqueta	Estructura de control para la operación
Length	DINT	inmediato	Número de elementos de la matriz a usar a la hora de calcular la desviación estándar.
Position	DINT	inmediato	Desplazamiento en la matriz especificada que identifica el elemento actual al que está accediendo la instrucción. El valor inicial es normalmente 0

### Estructura de CONTROL

Mnemónico	Tipo de datos	Descripción
.EN	BOOL	El bit de habilitación indica que la instrucción STD está habilitada.
.DN	BOOL	El bit de efectuado se establece cuando la instrucción ha operado en el último de la Array.
.ER	BOOL	El bit de error se establece cuando la instrucción genera un desbordamiento. La instrucción deja de ejecutarse hasta que el programa borra el valor del bit .ER. el valor de .POS almacena la posición del elemento que produjo el desbordamiento.
.LEN	DINT	La palabra longitud especifica el número de elementos en la matriz en los que opera la instrucción.
.POS	DINT	La palabra posición es un desplazamiento dentro de la matriz especificada que identifica el elemento actual al que accede la instrucción.

### Descripción

La desviación estándar se calcula conforme a esta fórmula:

$$\text{Standard Deviation} = \sqrt{\frac{\sum_{i=1}^N [(X_{(start+i)} - AVE)^2]}{(N - 1)}}$$

Donde:

start = subíndice dimensión a variar del operando de matriz

xi = elemento variable de la matriz

N = número de elementos especificados en la matriz

$$\text{AVE} = \frac{\sum_{i=1}^{N*} x_{(start+i)}}{N}$$

---

**Importante:** Asegúrese de que Length no cause que la instrucción exceda la Dimension to vary especificada. Si esto sucede, Destination será incorrecto.

---

Si se produce un desbordamiento durante la evaluación de la expresión o si las instrucciones leen más allá del final de una matriz, la instrucción establece el bit ER y detiene la ejecución.

#### Afectar a las marcas de estado matemático

Controladores	Afectar a las marcas de estado matemático
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Condicional, basado en el lenguaje de programación. Consulte las Marcas de estado matemático.
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí

#### Fallos mayores/menores

Se producirá un fallo mayor si:	Tipo de fallo	Código de fallo
.POS < 0 o .LEN < 0	4	21
Dimension to vary > número de dimensiones	4	20

Consulte los Atributos comunes para fallos relacionados con el operando.

## Ejecución

### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	El bit .EN se borra. El bit .DN se borra. El bit .ER se borra.
La condición de entrada de reglón es falsa	El bit .EN se borra. El bit .ER se borra. El bit .DN se borra. El valor de .POS se borra. El valor de condición de salida de pedaño es falso.
La condición de entrada de reglón es verdadera	Internamente, la instrucción usa una instrucción FAL para calcular la media: Expresión = cálculo de desviación estándar Modo = ALL
Post-escaneado	N/A.

### Ejemplos

#### Ejemplo 1

Calcular la desviación estándar de arrayDint, que es DINT[4,5].

		dimension 1				
		dimension 1				
		0	1	2	3	4
dimension 0	0	20	19	18	17	16
	1	15	14	13	12	11
	2	10	9	8	7	6
	3	5	4	3	2	1

$$STD = \sqrt{\frac{(16 - 8.5)^2 + (11 - 8.5)^2 + (6 - 8.5)^2 + (1 - 8.5)^2}{(4 - 1)}} = 6.454972$$

$$real\_std = 6.454972$$

### Diagrama de escalera

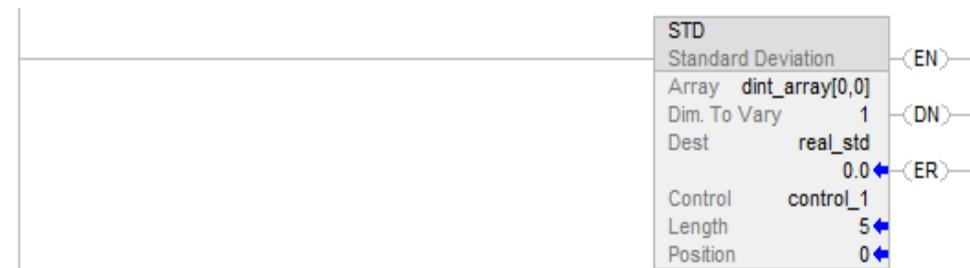


### Ejemplo 2

Calcular la desviación estándar de `dint_array`, que es `DINT[4,5]`.

		dimension 1				
		0	1	2	3	4
dimension 0	0	29	19	18	37	16
	1	15	14	13	12	11
	2	10	9	8	7	6
	3	5	4	3	2	1

### Diagrama de escalera



### Consulte también

[Instrucciones de archivos/miscláneas](#) en la página 527

[AVE](#) en la página 556

[Atributos comunes](#) en la página 923

[Marcas de estado matemático](#) en la página 923

[Conversiones de datos](#) en la página 927

## Tamaño en elementos (SIZE)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción SIZE averigua el número de elementos (tamaño) en la dimensión designada de la matriz Source o el operando de cadena y coloca el resultado en el operando Size. La instrucción averigua el tamaño de una dimensión de una matriz.

La instrucción opera en:

- Matrices
- Matrices dentro de una estructura
- Matrices que forman parte de una matriz mayor
- Etiquetas de cadena

### Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

### Texto estructurado

```
SIZE(Source,Dimtovary,Size);
```

## Operandos

**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
- Los miembros de un operando de estructura se sobrescriben.
- A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.

Existen reglas de conversión de datos para utilizar diversos tipos de datos numéricos en una instrucción. Consulte *Conversión de datos*.

## Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción	
Source	SINT INT DINT REAL estructura Tipo de cadena	Etiqueta de matriz	Primer elemento de la matriz en el que tiene que operar la instrucción Durante la verificación no se aceptan etiquetas que no sean matrices	
Dimension to Vary	DINT	inmediato (0, 1, 2)	Dimensión a usar:	
			Para un tamaño de:	Introduzca :
			primera dimensión	0
			segunda dimensión	1
			tercera dimensión	2
Size	SINT INT DINT REAL	etiqueta	Etiqueta para almacenar el número de elementos en la dimensión especificada de la matriz	

Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

## Afectar a las marcas de estado matemático

No

### Fallos mayores/menores

No es específico para esta instrucción. Consulte *Índice a través de matrices* para ver si hay fallos de indexación de matrices.

### Ejecución

#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	Establecer la condición de salida de reglón a condición de entrada de reglón.
La condición de entrada de reglón es verdadera	Establecer la condición de salida de reglón a condición de entrada de reglón. La instrucción se ejecuta.
Post-escaneado	N/A

#### Texto estructurado

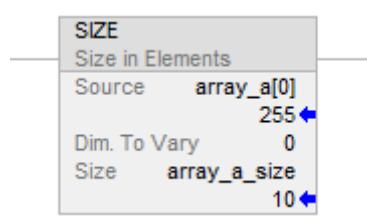
Condición/estado	Acción realizada
Pre-escaneado	Consultar Pre-escaneado en la tabla de diagrama de escalera
Ejecución normal	Consultar La condición de entrada de reglón es verdadera en la tabla Diagrama de escalera.
Post-escaneado	Consultar Post-escaneado en la tabla Diagrama de escalera.

### Ejemplos

#### Ejemplo 1

Averigua el número de elementos en la dimensión 0 (primera dimensión) de matriz\_a. Almacene el tamaño en matriz\_a\_size. En este ejemplo, la dimensión 0 de array\_a tiene 10 elementos.

#### Diagrama de escalera



### Texto estructurado

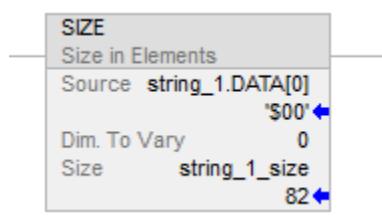
```
SIZE(array_a,0,array_a_size);
```

### Ejemplo 2

Averigua el número de elementos en el miembro DATA de cadena\_1, que es una cadena. Almacena el tamaño en cadena\_1\_tamaño.

En este ejemplo, el miembro DATA de cadena\_1 tiene 82 elementos. La cadena usa el tipo de datos STRING predeterminado. Dado que cada elemento contiene un carácter, cadena\_1 puede contener hasta 82 caracteres.

### Diagrama de escalera



### Texto estructurado

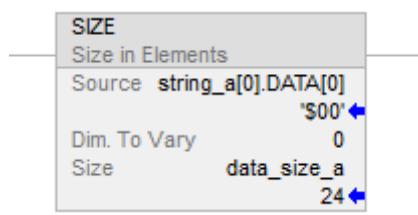
```
SIZE(string_1.DATA[0],0,string_1_size);
```

### Ejemplo 3

Cadena\_a es una matriz de estructuras de cadena. La instrucción SIZE averigua el número de elementos en el miembro DATA de la estructura de cadena y almacena el tamaño en \_tamaño\_a.

En este ejemplo, el miembro DATA tiene 24 elementos. La estructura de cadena tiene una longitud especificada por el usuario de 24.

### Diagrama de escalera



### Texto estructurado

```
SIZE(string_a[0].DATA[0],0,data_size_a);
```

**Consulte también**

[Instrucciones de archivos/miscláneas](#) en la página 527

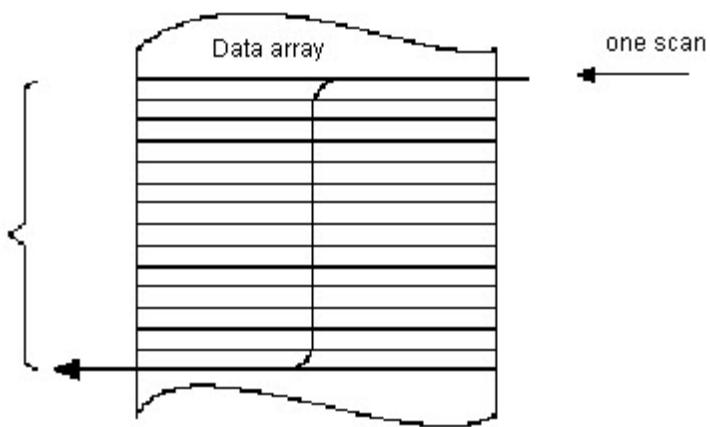
[Índice a través de matrices](#) en la página 936

[Conversiones de datos](#) en la página 927

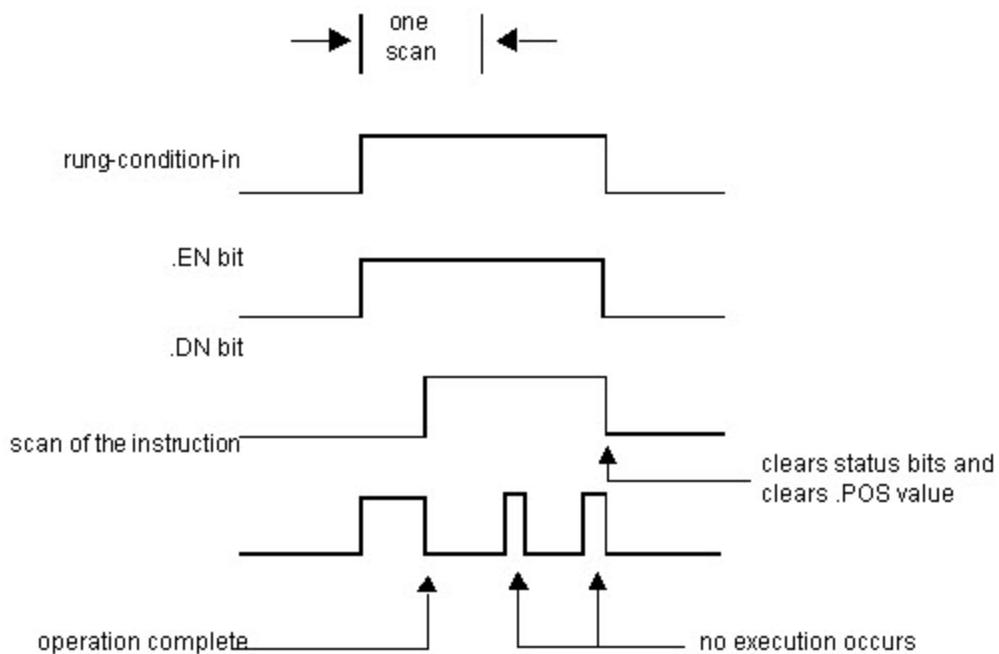
[Sintaxis de texto estructurado](#) en la página 955

## Modo Todos

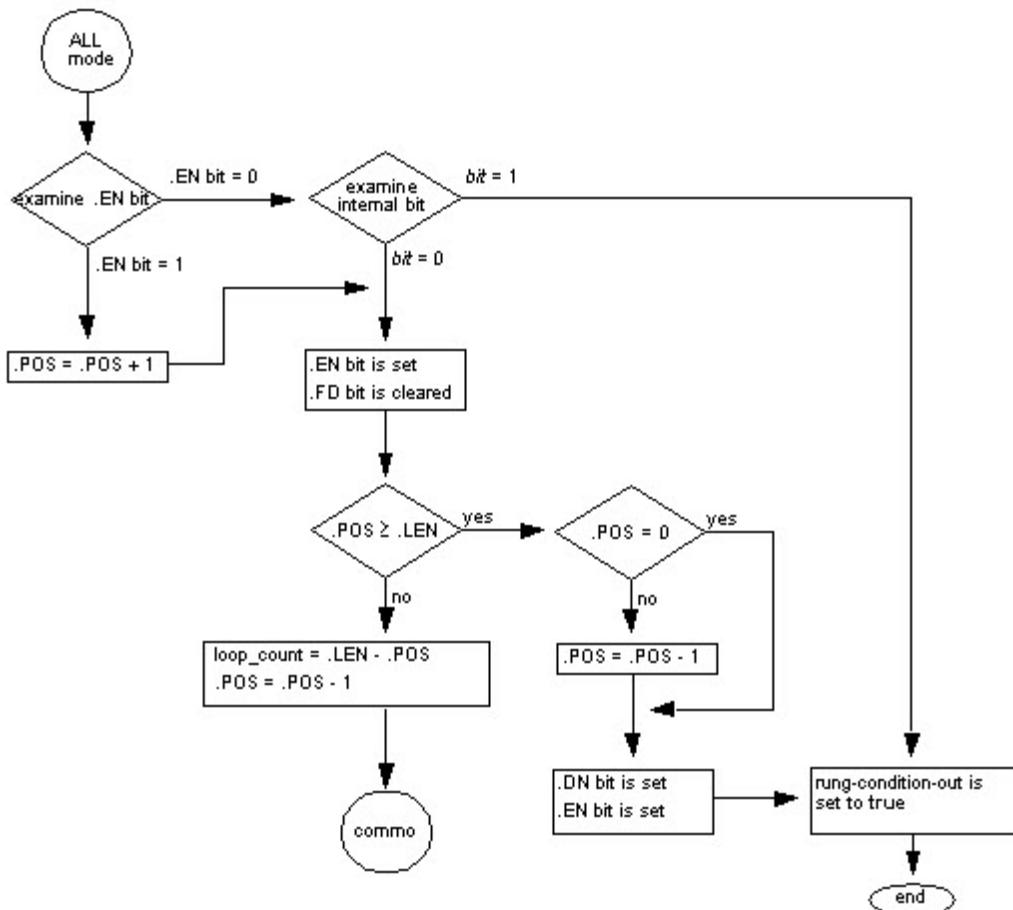
En el modo Todos, todos los elementos especificados en la matriz se utilizan antes de continuar con la siguiente instrucción. La operación comienza cuando el valor de condición de entrada de peldaño de la instrucción pase de falso a verdadero. El valor de posición (.POS) en la estructura de control apunta al elemento de la matriz que la instrucción está utilizando actualmente. La operación se detiene cuando el valor de .POS es igual al valor de .LEN.



El siguiente diagrama de temporización muestra la relación entre los bits de estado y la operación de la instrucción. Cuando se haya completado la ejecución de la instrucción, se establece el bit .DN. El bit .DN, el bit .EN y el valor .POS se borran cuando el valor de la condición de entrada de peldaño es falso. Solo entonces se puede desencadenar otra ejecución de la instrucción mediante una transición de falso a verdadero de la condición de entrada de peldaño



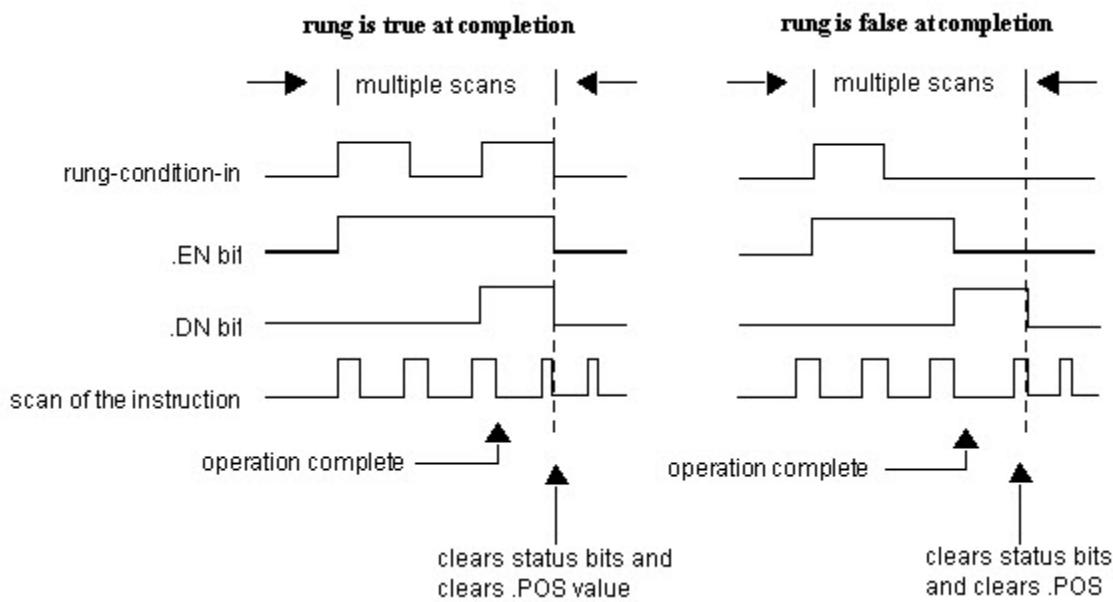
## Diagrama de flujo del modo Todos (FSC)



## Modo Numérico

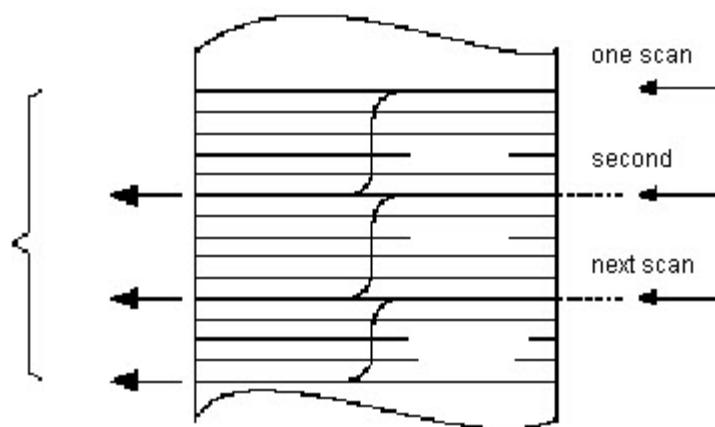
El modo Numérico distribuye el funcionamiento de la matriz sobre una serie de escaneados. Este modo es útil cuando se trabaja con datos que no son críticos para el tiempo o grandes cantidades de datos. Se introduce el número de elementos para operar durante cada escaneado, lo que mantiene el tiempo de escaneado más corto.

La ejecución desencadena cuando el valor de condición de entrada de peldaño pasa de falso a verdadero. Una vez desencadenada, la instrucción se ejecuta cada vez que se escanea con el número de escaneados necesarios para completar la operación sobre toda la matriz. Una vez desencadenada, el valor de condición de entrada de peldaño puede cambiar repetidamente sin interrumpir la ejecución de la instrucción.



Evita usar los resultados de una instrucción de archivo que opere en el modo numérico hasta que se haya establecido el bit .DN.

El siguiente diagrama de temporización muestra la relación entre los bits de estado y la operación de la instrucción. Cuando se haya completado la ejecución de la instrucción, se establece el bit .DN.

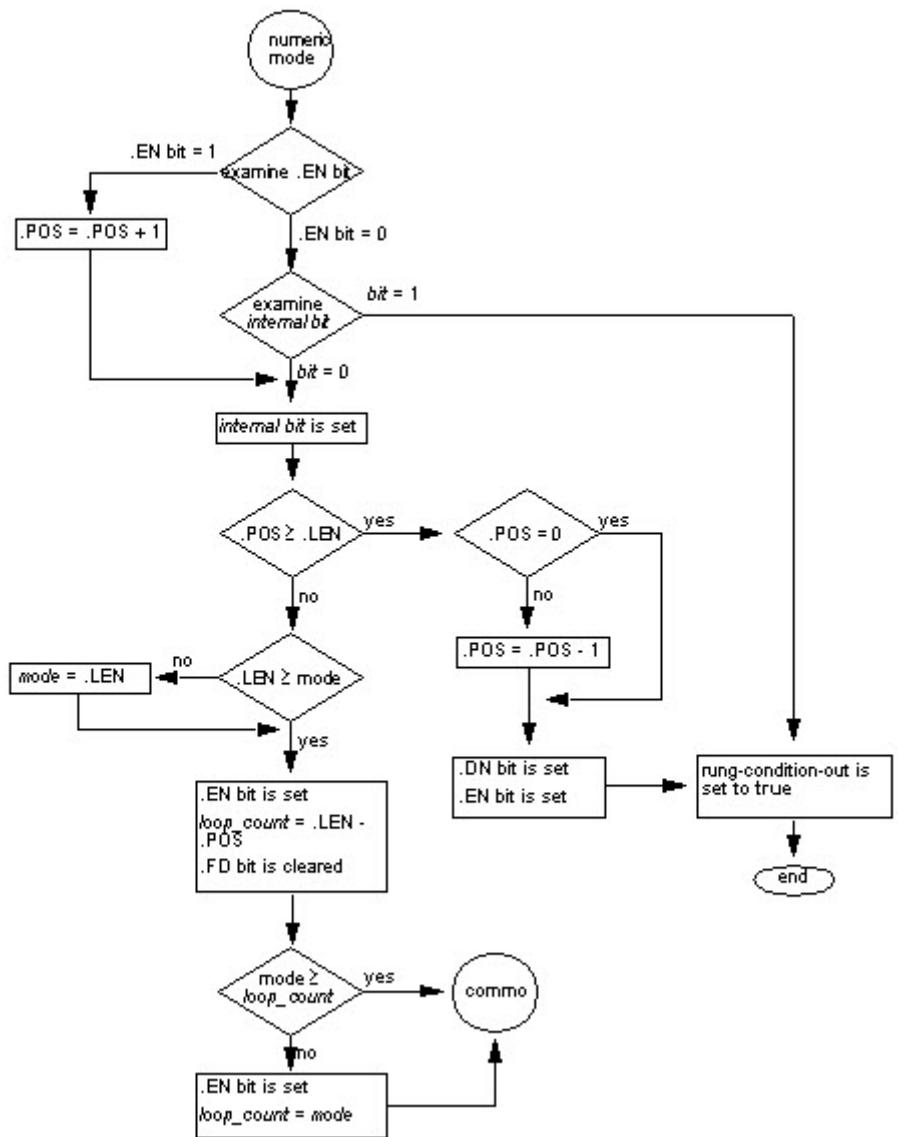


Si el valor de condición de entrada de peldaño es verdadero al completar, se establecen los bits .EN y .DN hasta que el valor de la condición de entrada de

peldaño a falso. Cuando el valor de condición de entrada de peldaño a falso, se borran estos bit y el valor de .POS.

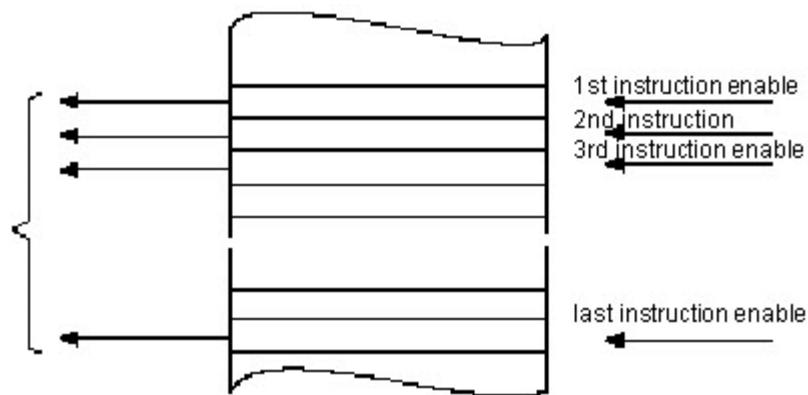
Si el valor de condición de entrada de peldaño es falso al completarse, el bit .EN se borra de inmediato. Un escaneado después de que se haya borrado el bit .EN, se borra el bit .DN y el valor .POS borrado

## Diagrama de flujo del modo Numérico (FSC)

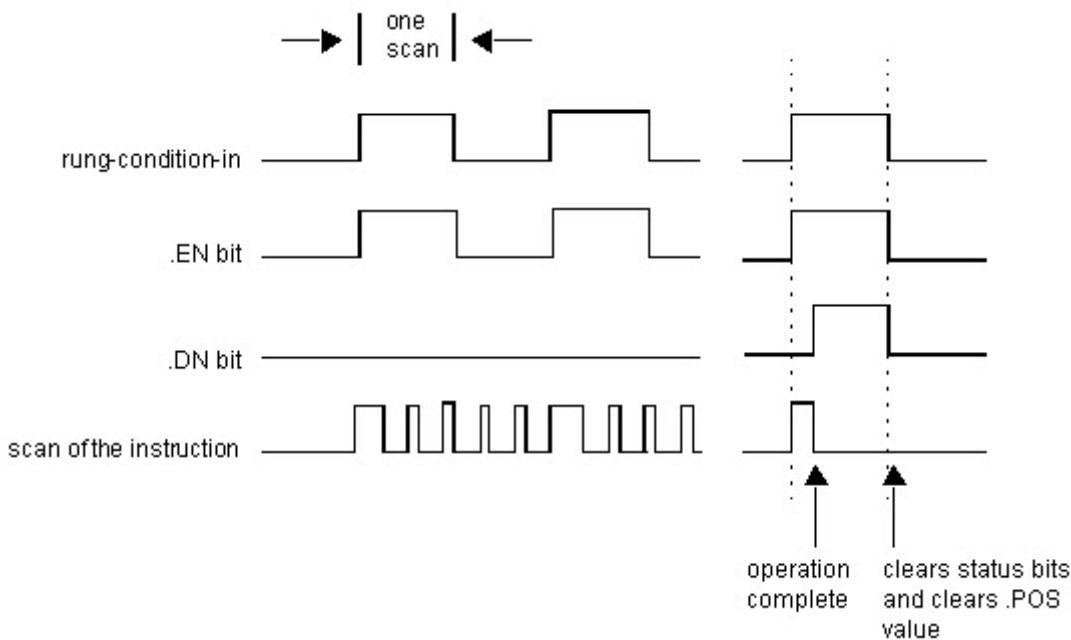


## Modo Incremental

El modo Incremental manipula un elemento de la matriz cada vez que el valor de condición de entrada de peldaño de la instrucción pase de falso a verdadero



El siguiente diagrama de temporización muestra la relación entre los bits de estado y la operación de la instrucción. La ejecución solo se produce en un escaneado en el que el valor de condición de entrada de peldaño pasa de falso a verdadero. Cada vez que esto ocurre, solo se manipula un elemento de la matriz. Si el valor de condición de entrada de peldaño sigue siendo verdadero durante más de un escaneado, la instrucción solo se ejecuta durante el primer escaneado.

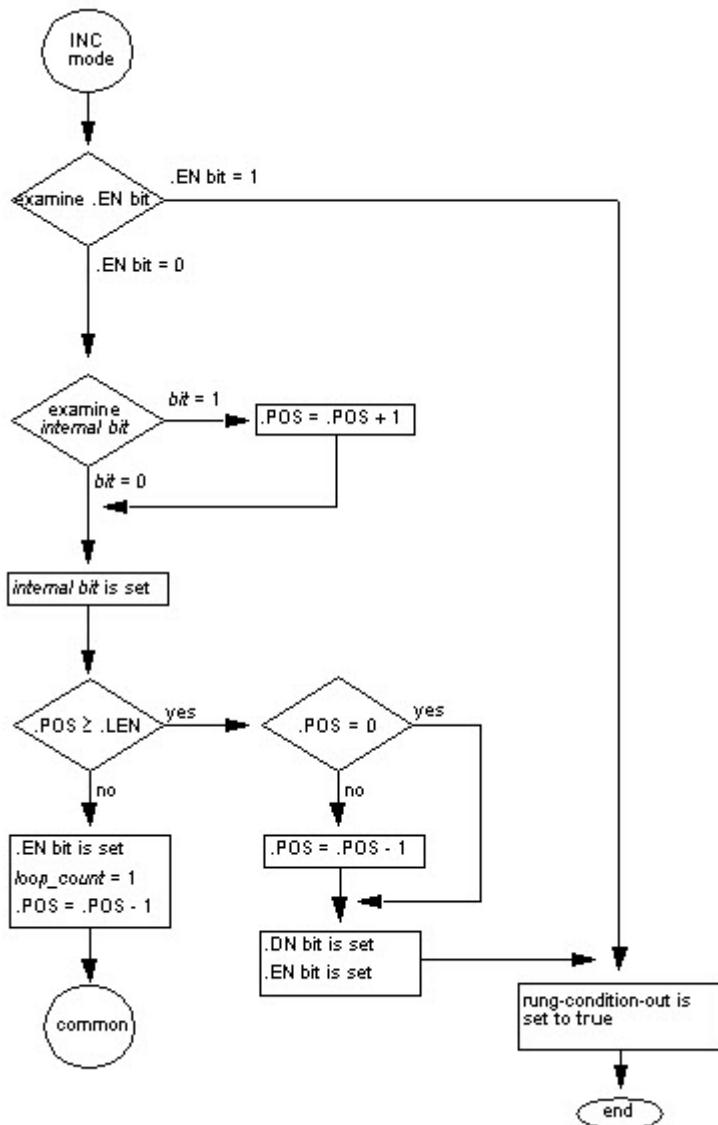


El bit .EN se establece cuando la condición de entrada de reglón es verdadera. El bit .DN se establece cuando se ha manipulado el último elemento de la matriz. Cuando se ha manipulado el último elemento y la condición de entrada de reglón se vuelve falsa, se borran el bit .EN, el bit .DN y el valor .POS.

La diferencia entre el modo Incremental y el modo Numérico a un régimen de un elemento por escaneado es:

- El modo Numérico con cualquier número de elementos por escaneado solo requiere una transición de falso a verdadero de condición de entrada de peldaño para que comience la ejecución. La instrucción sigue ejecutando el número de elementos especificado cada escaneado hasta que se complete, independientemente del estado de condición de entrada de peldaño.
- El modo Incremental requiere que el valor de condición de entrada de peldaño pase de falso a verdadero para manipular un elemento de la matriz.

## Diagrama de flujo del modo Incremental (FSC)



## Etiqueta de matriz

Cuando introduce una etiqueta de matriz, asegúrese de especificar el primer elemento de la matriz que se debe manipular. No utiliza CONTROL.POS para identificar el elemento inicial porque la instrucción modifica el valor de .POS a medida que opera, lo que podría corromper el resultado.

## Desviación estándar

La desviación estándar se calcula conforme a esta fórmula:

$$\text{Standard Deviation} = \sqrt{\frac{\sum_{i=1}^N [x_{(start+i)} - AVE]^2}{(N-1)}}$$

Donde:

- start = subíndice dimensión a variar del operando de matriz
- xi = elemento variable de la matriz
- N = número de elementos especificados en la matriz

$$\bullet \quad AVE = \frac{\sum_{i=1}^N x_{(start+i)}}{N}$$

# Instrucciones de Matriz (Archivo)/Desplazamiento

## Instrucciones de Matriz (Archivo)/Desplaza miento

Se utiliza las instrucciones de matriz (archivo)/desplazamiento para modificar la ubicación de los datos dentro de matrices.

### Instrucciones disponibles

#### Diagrama de escalera

<u>BSL</u>	<u>BS</u>	<u>FFL</u>	<u>FF</u>	<u>LFL</u>	<u>LFU</u>
R			U		

#### Bloque de funciones

No disponible

#### Texto estructurado

No disponible

Si desea:	Utilice esta instrucción:
Cargar bits en el interior de una matriz de bits, desplazar bits por una matriz de bits y descargar bits de una matriz de uno en uno.	BSL BSR
Cargar y descargar valores en el mismo orden.	FFL FFU
Cargar y descargar valores en orden inverso.	LFL LFU

Puede mezclar tipos de datos, pero puede que se produzca una pérdida de precisión y errores de redondeo.

Los tipos de datos en negrita son los óptimos. Una instrucción se ejecutará a la máxima velocidad y la menor cantidad posible de memoria si todos los parámetros emplean el mismo tipo de datos óptimo, normalmente DINT o REAL.

**Consulte también**[Instrucciones de conversión ASCII](#) en la página 892[Instrucciones de puerto serial ASCII](#) en la página 871[Instrucciones de cadena ASCII](#) en la página 871**Desplazamiento de bit a la izquierda (BSL)**

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

La instrucción BSL desplaza una posición a la izquierda los bits especificados dentro de la Array.

**Idiomas disponibles****Diagrama de escalera****Bloque de funciones**

Esta instrucción no está disponible en bloque de funciones.

**Texto estructurado**

Esta instrucción no está disponible en texto estructurado.

**Operandos****Diagrama de escalera**

Operando	Tipo	Formato	Descripción
Array	DINT ARRAY	etiqueta	Matriz a modificar Especifica el primer elemento por el que comenzar el desplazamiento.
Control	CONTROL	etiqueta	Estructura de control para la operación
Source Bit	BOOL	etiqueta	Bit a desplazar en la posición vacante.

Length	DINT	inmediato	Número de bits en la matriz a desplazar
--------	------	-----------	---

### Estructura de CONTROL

Mnemónico	Tipo de datos	Descripción
.EN	BOOL	El bit de habilitación indica que la instrucción BSL está habilitada.
.DN	BOOL	El bit de efectuado se establece para indicar que los bits se desplazaron una posición a la izquierda.
.UL	BOOL	El bit de descarga es la salida de la instrucción. El bit .UL almacena el estado del bit que se desplazó el rango de bits.
.ER	BOOL	El bit de error se establece cuando .LEN < 0.
.LEN	DINT	La longitud especifica el número de bits de la matriz que se van a desplazar.

### Descripción

Cuando está habilitada, la instrucción descarga el bit superior de los bits especificados en el bit .UL, desplaza los bits restantes una posición hacia la izquierda y carga la dirección del Bit en el bit 0 de la Array.

---

**Importante:** Debe comprobar y confirmar que la instrucción no modifica datos que usted no desea que sean modificados.

---

La instrucción BSL opera en posiciones continuas de memoria de datos. La instrucción BSL opera en posiciones continuas de memoria de datos. Únicamente en el caso de los controladores CompactLogix 5370 y ControlLogix 5570, el ámbito de la instrucción se ve limitado por la etiqueta de base. La instrucción BSL no escribirá datos fuera de la etiqueta de base, pero puede pasar los límites de los miembros. Si especifica una matriz que sea miembro de una estructura y la longitud supera el tamaño de dicha matriz, debe probar y confirmar que la instrucción BSL no modifique datos que usted no quiera que sean modificados.

En el caso de Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580., los datos se ven limitados por el miembro especificado.

En esta instrucción de transición, la escalera de relé cambia el valor de condición de entrada de reglón de falso a verdadero para que la instrucción se ejecute.

### Afectar a las marcas de estado matemático

No

**Fallos mayores/menores**

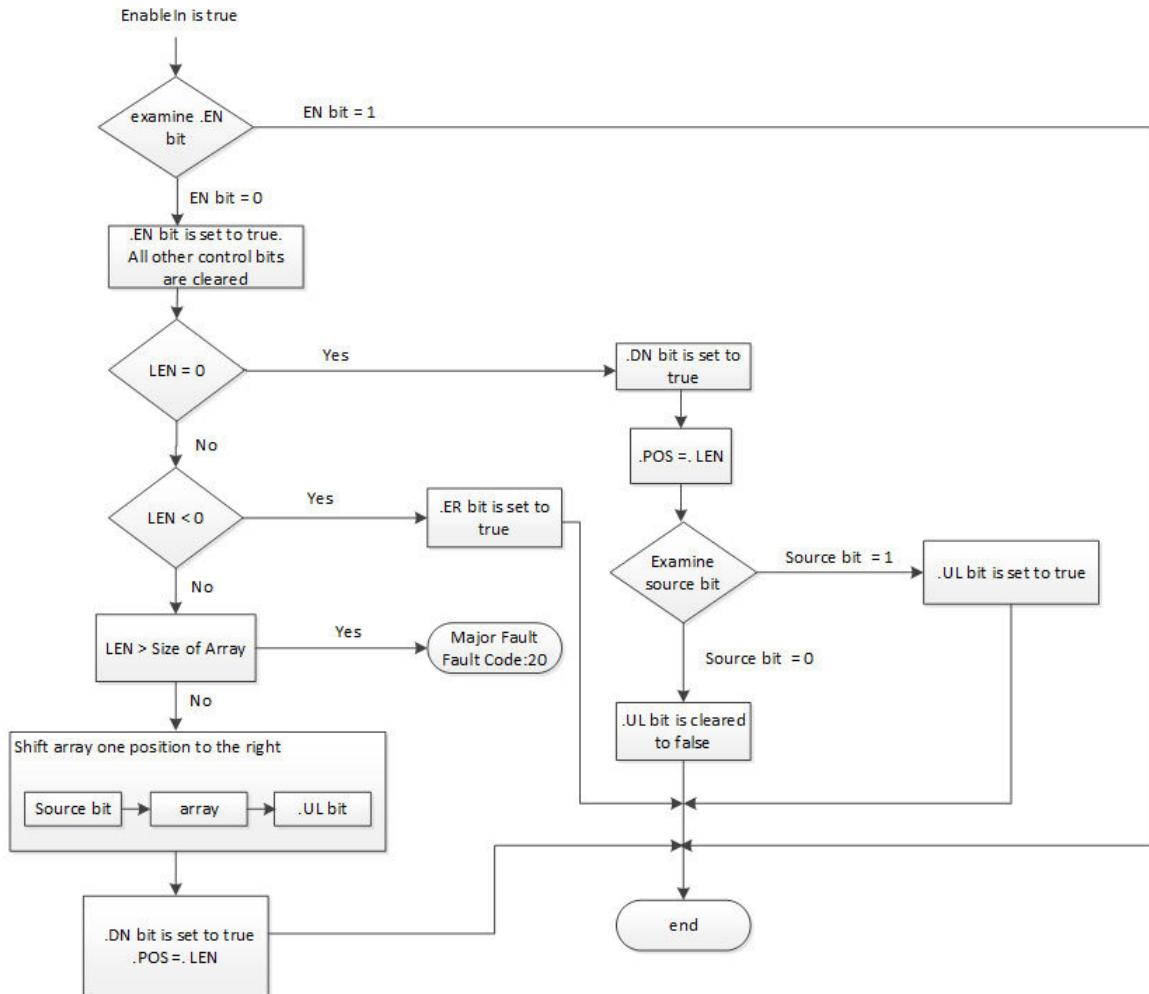
Se produce un fallo mayor si	Tipo de fallo	Código de fallo
El valor de LEN supera el tamaño de la matriz	4	20

Consulte los Atributos comunes para los fallos relacionados con los operandos.

**Ejecución****Diagrama de escalera**

Condición/estado	Acción realizada
Pre-escaneado	El bit .EN se borra a falso. El bit .DN se borra a falso. El bit .ER se borra y se establece en falso Se borra el valor de .POS
La condición de entrada de reglón es falsa	El bit .EN se borra a falso. El bit .DN se borra a falso. El bit .ER se borra y se establece en falso Se borra el valor de .POS.
La condición de entrada de reglón es verdadera	Consultar el Diagrama de flujo de BSL (Verdadero).
Post-escaneado	N/A

### Consulte el Diagrama de flujo de BSL (Verdadero)

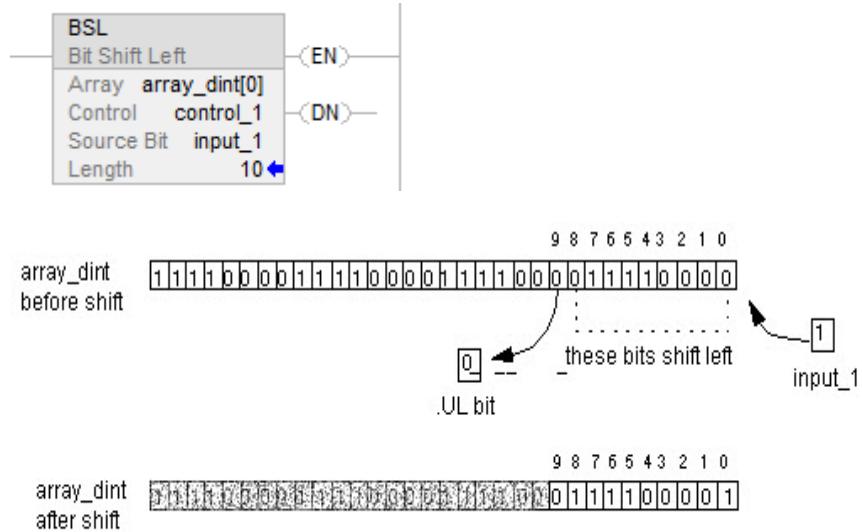


### Ejemplos

#### Ejemplo 1

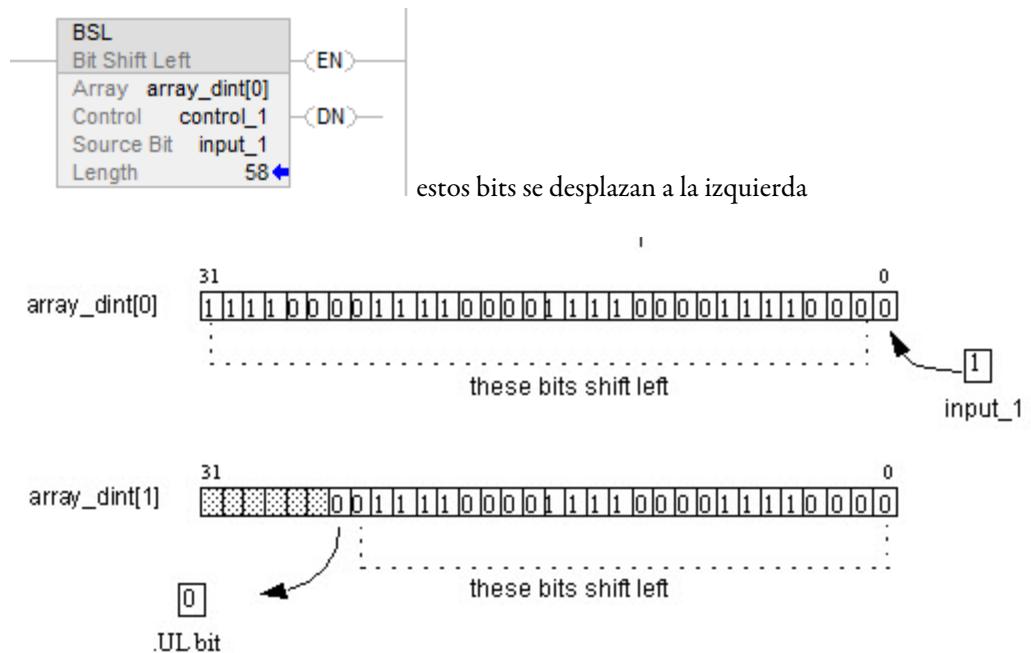
Cuando está habilitada, la instrucción BSL comienza en el bit 0 de array\_dint[0]. La instrucción descarga array\_dint[0].9 dentro del bit .UL, desplaza los bits restantes y carga input\_1 dentro de array\_dint[0].0. Los bits restantes (10-31) no son válidos.

## Diagrama de escalera



## Ejemplo 2:

Cuando está habilitada, la instrucción BSL comienza en el bit 0 de array\_dint[0]. La instrucción descarga array\_dint[1].25 dentro del bit .UL, desplaza los bits restantes y carga input\_1 dentro de array\_dint[0].0. Los bits restantes (31-26 en array\_dint[1]) no son válidos.



### Consulte también

[Atributos comunes](#) en la página 923

[Conversiones de datos](#) en la página 927

## Desplazamiento de bit a la derecha (BSR)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

La instrucción BSR desplaza una posición a la derecha los bits especificados dentro de Array. Cuando está habilitada, la instrucción descarga el valor del bit 0 de la Array en el bit .UL, desplaza los bits restantes una posición hacia la derecha y carga el bit de la dirección del Bit.

---

**Importante:** Pruebe y confirme que la instrucción modificó los datos pertinentes.

La instrucción BSR opera en la memoria continua. Si una Array es una matriz de miembros, la instrucción podría cambiar más allá del límite de la matriz hasta otros miembros que la sigan. Asegúrese de seleccionar una longitud que no haga que se produzca esta situación.

---

La instrucción BSR opera en posiciones continuas de memoria de datos. Solo para Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570, el ámbito de la instrucción se ve limitado por la etiqueta de base. La instrucción BSL no escribirá datos fuera de la etiqueta de base, pero puede pasar los límites de los miembros. Si especifica una matriz que sea miembro de una estructura y la longitud supera el tamaño de dicha matriz, pruebe y confirme que la instrucción BSL modificó los datos pertinentes.

En el caso de Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570, los datos se ven limitados por el miembro especificado.

Si la instrucción intenta leer más allá del final de una matriz (el valor de LEN es demasiado grande), la instrucción establece el bit .ER y genera un fallo grave.

## Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

### Operандos

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción. Consulte Conversión de datos.

### Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción
Array	DINT ARRAY	etiqueta	Matriz a modificar Especifica el primer elemento que se debe desplazar.
Control	CONTROL	etiqueta	Estructura de control para la operación
Source Bit	BOOL	etiqueta	Bit a cargar dentro de la posición vacante.
Length	DINT	inmediato	Número de bits en la matriz a desplazar

### Estructura de CONTROL

Mnemónico	Tipo de datos	Descripción
.EN	BOOL	El bit de habilitación indica que la instrucción BSR está habilitada.
.DN	BOOL	El bit de efectuado se establece para indicar que los bits se desplazaron una posición a la derecha.
.UL	BOOL	El bit de descarga es la salida de la instrucción. El bit .UL almacena el estado del bit que se desplazó el rango de bits.

Mnemónico	Tipo de datos	Descripción
.ER	BOOL	El bit de error se establece cuando .LEN < 0.
.LEN	DINT	La longitud especifica el número de bits de la matriz que se van a desplazar.

### Afectar a las marcas de estado matemático

No

### Fallos mayores/menores

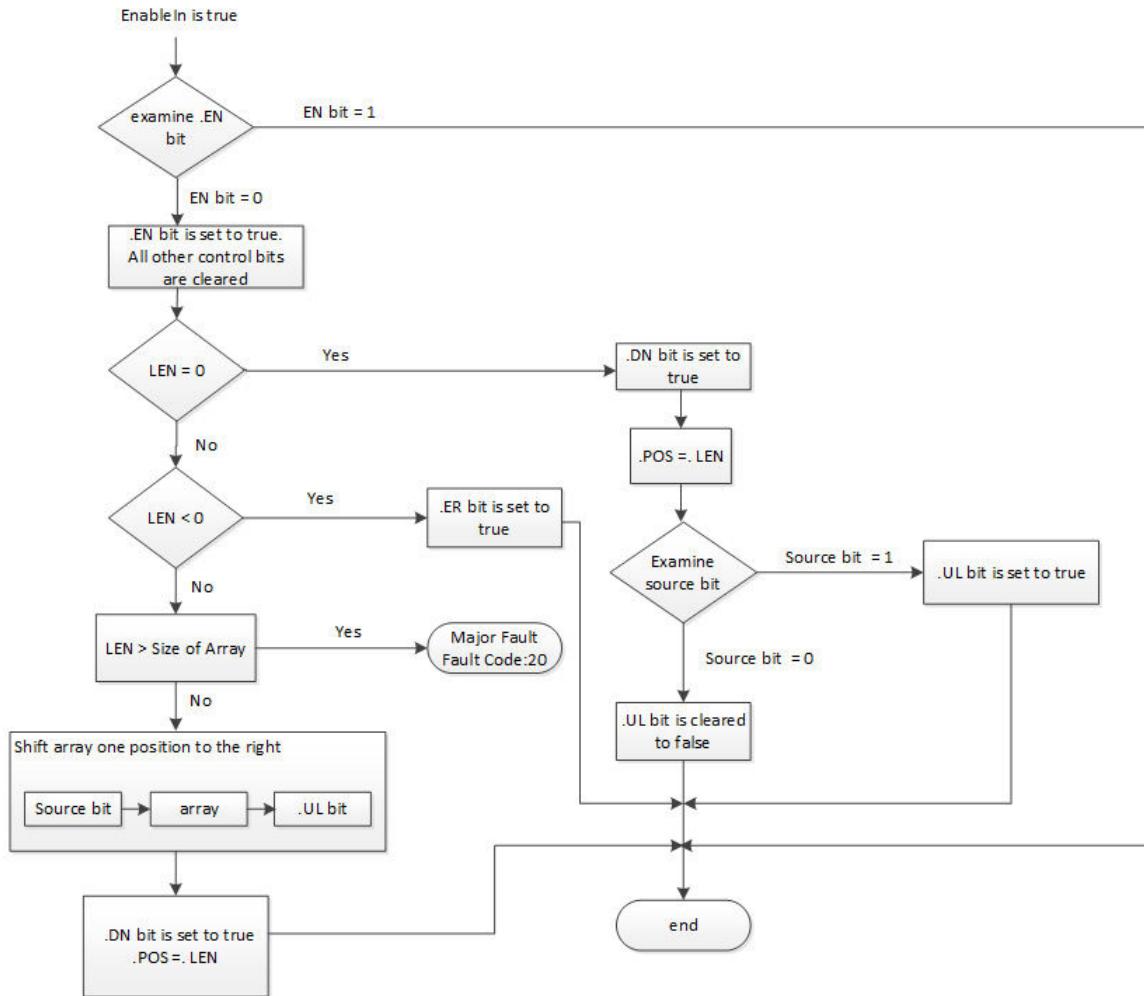
No es específico para esta instrucción. Consulte Índice a través de matrices para ver si hay fallos de indexación de matrices.

### Ejecución

#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	El bit .EN se borra a falso. El bit .DN se borra a falso. El bit .ER se borra y se establece en falso Se borra el valor de .POS.
La condición de entrada de reglón es falsa	El bit .EN se borra a falso. El bit .DN se borra a falso. El bit .ER se borra y se establece en falso Se borra el valor de .POS.
La condición de entrada de reglón es verdadera	Consultar el siguiente Diagrama de flujo de BSR (Verdadero)
Post-escaneado	N/A

## Diagrama de flujo de BSR (Verdadero)

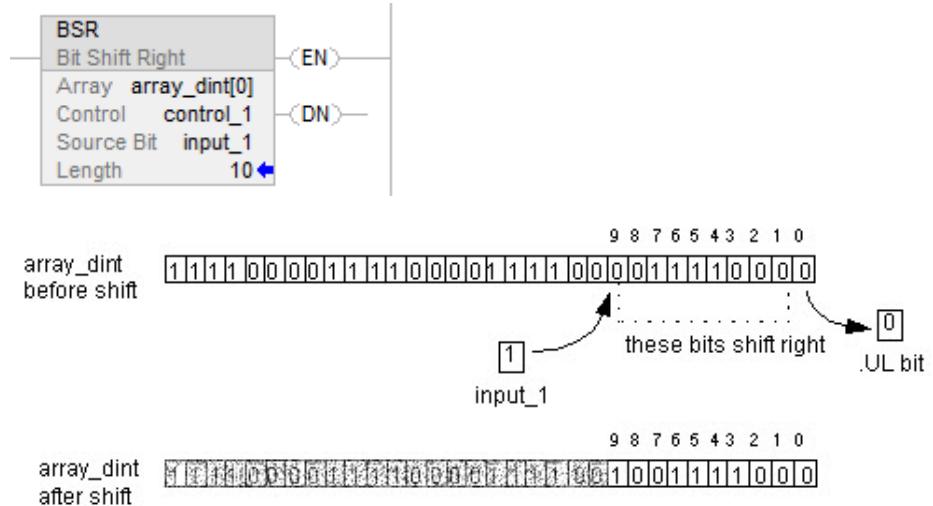


## Ejemplos

## Ejemplo 1

Cuando está habilitada, la instrucción BSR copia array\_dint[0].0 en el bit .UL, desplaza 0-9 a la derecha y carga input\_1 dentro de array\_dint[0].9. Los bits restantes (10-31) no son válidos, lo que indica que no se pueden modificar los bits.

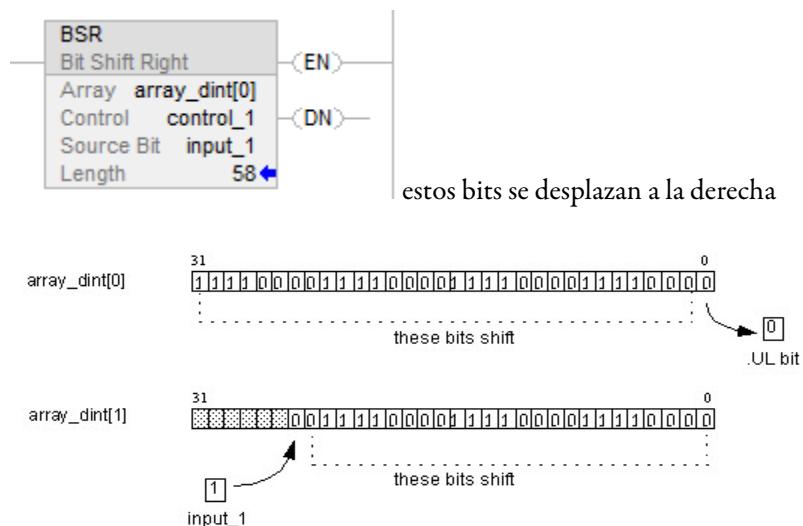
### Diagrama de escalera



### Ejemplo 2

Cuando está habilitada, la instrucción BSR copia `array_dint[0].0` en el bit .UL, desplaza 0-9 a la derecha y carga `input_1` dentro de `array_dint[1].25..` Los bits restantes (31-26 en `dint_array[1]`) no son válidos, lo que indica que los bits no se pueden modificar. Observa como `array_dint[1].0` se desplaza por las palabras de `array_dint[0].31..`

### Diagrama de escalera



### Consulte también

[Índice a través de matrices](#) en la página 936

[Conversiones de datos](#) en la página 927

## Carga FIFO (FFL)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

La instrucción FFL copia el valor de Source en FIFO.

### Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

### Operандos

La conversión solo se produce si el tipo del operando de origen no coincide con el tipo de FIFO.

### Diagrama de escalera

Operando	Tipo	Formato	Descripción
Source	SINT INT DINT REAL Tipo de cadena estructura	inmediato etiqueta	Datos a almacenar en FIFO

FIFO	SINT INT DINT REAL Tipo de cadena estructura	etiqueta de matriz	FIFO a modificar Especifica el primer elemento de FIFO
Control	CONTROL	etiqueta	Estructura de control para la operación Por lo general, se utiliza el mismo CONTROL que la FFU asociada
Length	DINT	inmediato	Número máximo de elementos que puede retener al mismo tiempo el FIFO.
Position	DINT	inmediato	Siguiente ubicación en FIFO en el que la instrucción carga datos. el valor inicial suele ser 0

### Estructura de CONTROL

Mnemónico	Tipo de datos	Descripción
.EN	BOOL	El bit de habilitación indica que la instrucción FFL está habilitada.
.DN	BOOL	El bit de efectuado se establece para indicar que FIFO está lleno. El bit .DN inhibe la carga del FIFO hasta que .POS < .LEN.
.EM	BOOL	El bit vacío indica que FIFO está vacío. Si .LEN es ≤ 0 o .POS < 0, se establece el bit .EM y los bits .DN.
.LEN	DINT	La palabra longitud especifica el número máximo de elementos en FIFO.
.POS	DINT	La palabra posición identifica la ubicación en FIFO en que la instrucción carga el siguiente valor.

### Descripción

Se utiliza la instrucción FFL con la instrucción FFU para almacenar y recuperar datos en un orden de tipo el primero en entrar es el primero en salir. Cuando se utilizan en pareja, las instrucciones FFL y FFU establecen un registro de desplazamiento asíncrono.

Por lo general, Source y FIFO usan el mismo tipo de datos.

Cuando está habilitada, la instrucción FFL carga el valor de Source en la posición de FIFO identificada por el valor de .POS. La instrucción carga un valor cada vez que se habilita la instrucción hasta que FIFO esté lleno.

**Importante:** Comprueba y confirma que la instrucción no modifica datos que usted no desea que sean modificados.

La instrucción FFL opera en posiciones continuas de memoria. La instrucción BSL opera en posiciones continuas de memoria de datos. Solo para Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570, el ámbito de la instrucción se ve limitado por la etiqueta de base. La instrucción BSL no escribirá datos fuera de la etiqueta de base, pero puede pasar los límites de los miembros. Si especifica una matriz que sea miembro de una estructura y la longitud supera el tamaño de dicha matriz, debe probar y confirmar que la instrucción BSL no modifique datos que usted no quiera que sean modificados.

En el caso de Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580., los datos se ven limitados por el miembro especificado.

Si la instrucción intenta leer más allá del final de una matriz, la instrucción genera un fallo grave.

Por lo general, Source y FIFO usan el mismo tipo de datos. Si no coinciden los tipos de Source y FIFO, la instrucción convierte el valor de Source al tipo de datos de la etiqueta de FIFO.

Un entero menor se convierte a un entero mayor mediante extensión de signo.

#### Afectar a las marcas de estado matemático

No

#### Fallos mayores/menores

Se producirá un fallo mayor si:	Tipo de fallo	Código de fallo
El (elemento inicial + .POS) está más allá del final de la matriz de FIFO	4	20

Consulte los Atributos comunes para fallos relacionados con el operando.

#### Ejecución

#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	Consultar el Diagrama de flujo de FFL (Pre-escaneado).

La condición de entrada de reglón es falsa	Consultar el Diagrama de flujo de FFL (Falso)
La condición de entrada de reglón es verdadera	Consultar el Diagrama de flujo de FFL (Verdadero)
Post-escaneado	N/A

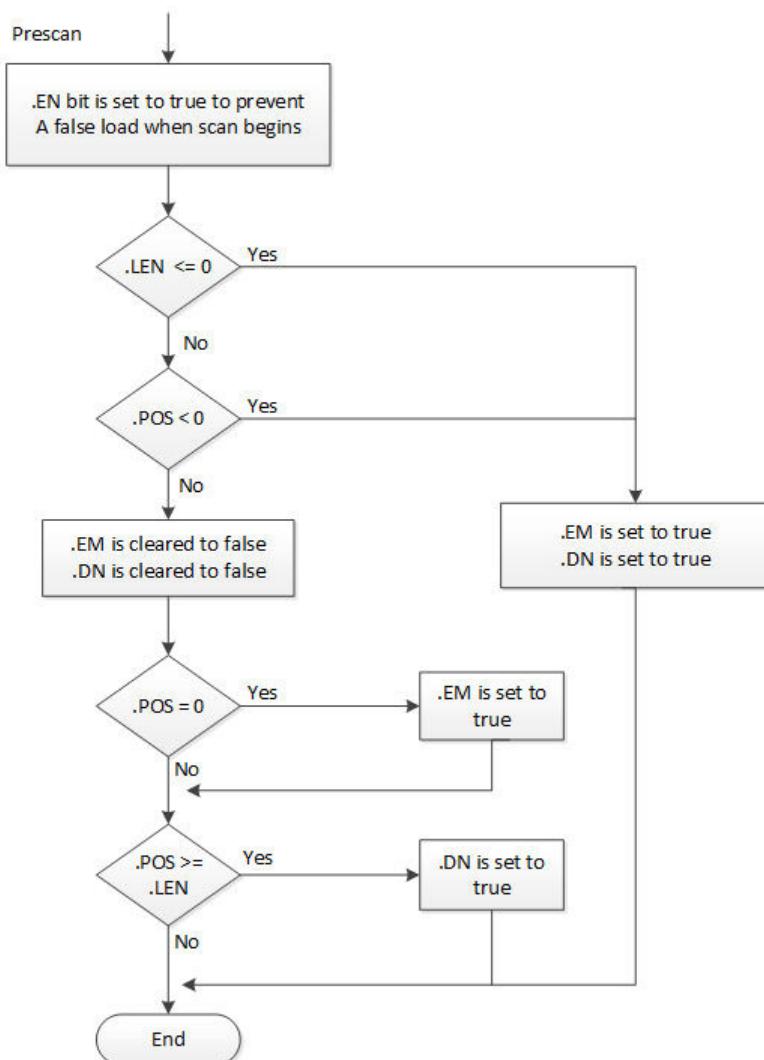
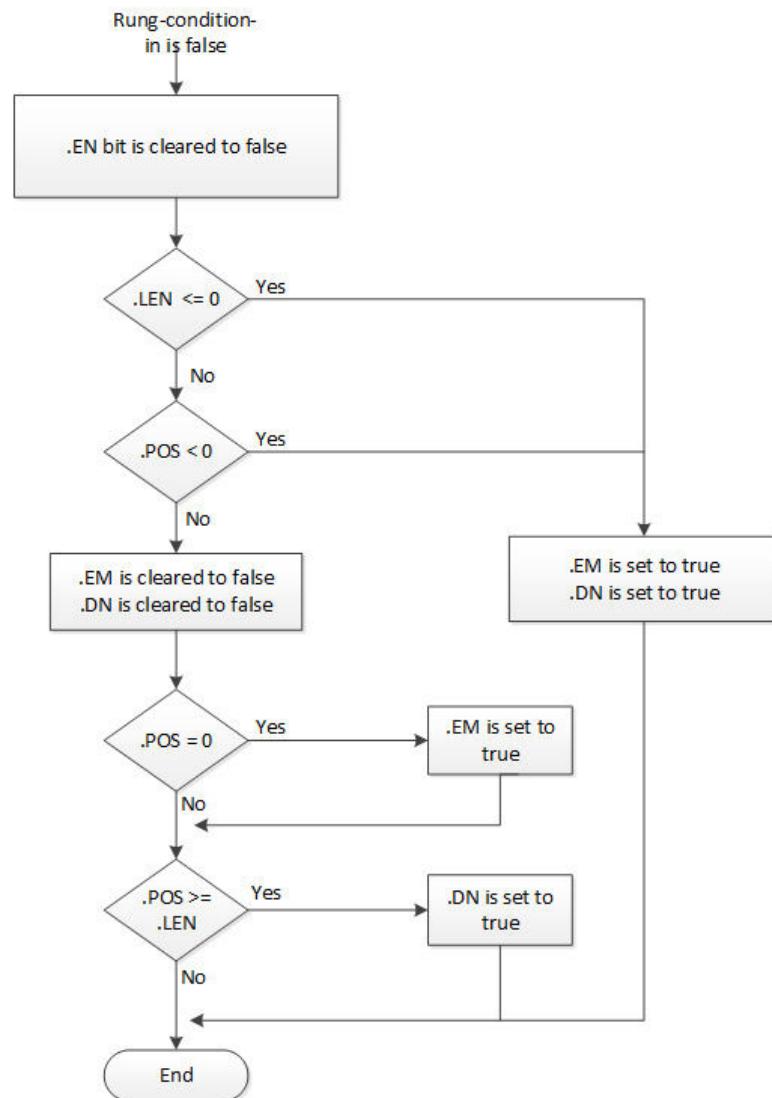
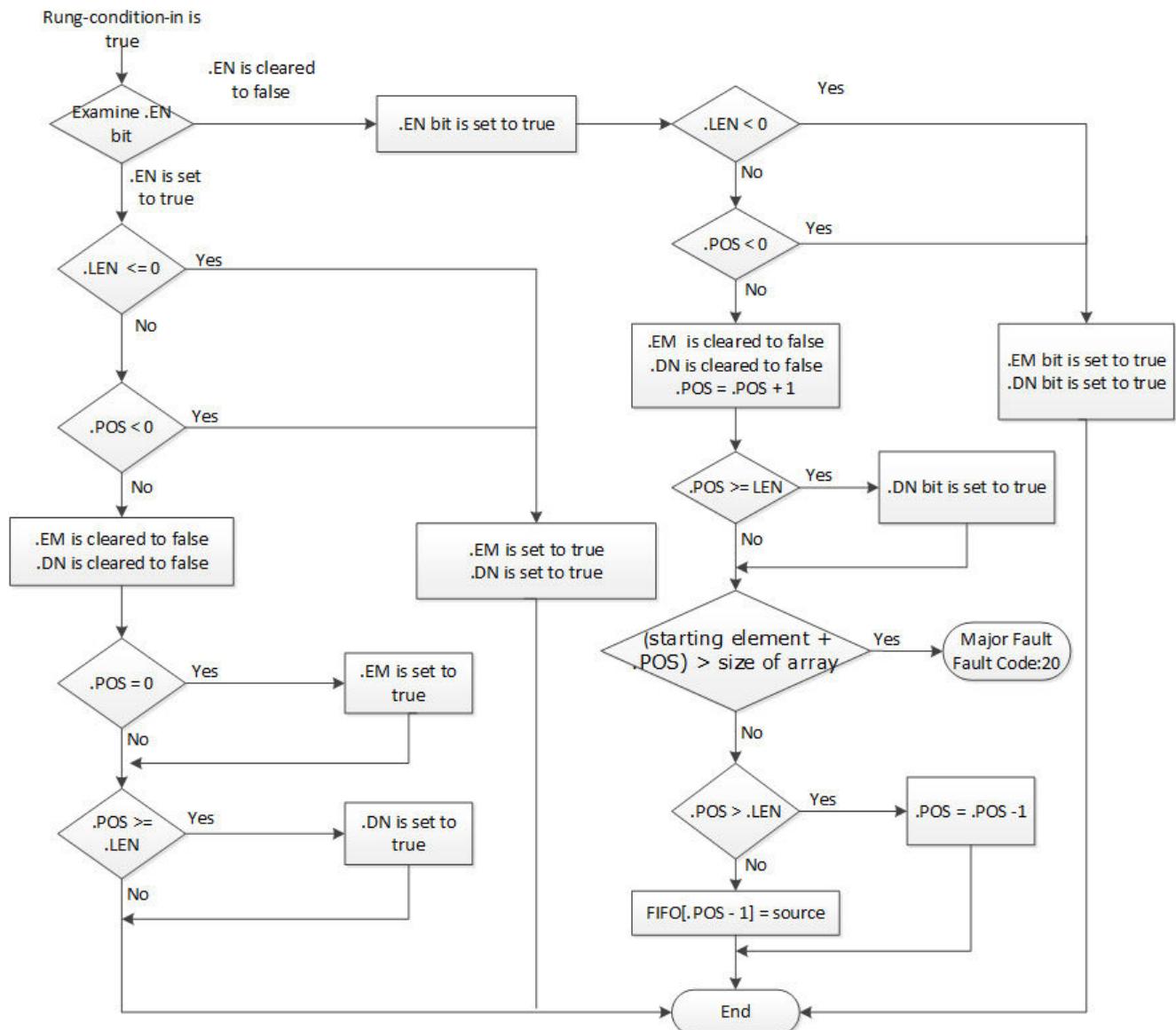
**Diagrama de flujo de FFL (Pre-escaneado).**

Diagrama de flujo de FFL (Falso)



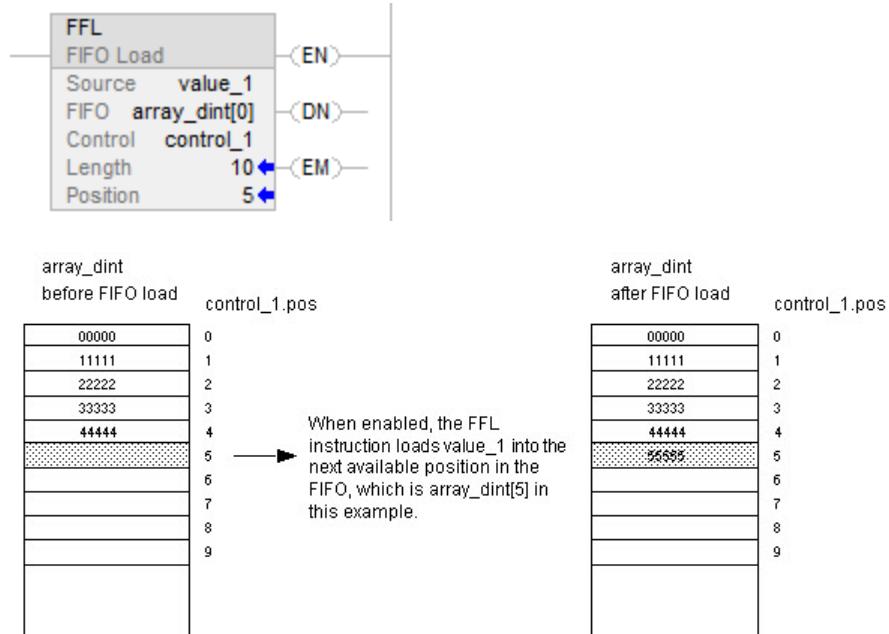
## Diagrama de flujo de FFL (Verdadero)



## Ejemplos

### Ejemplo 1

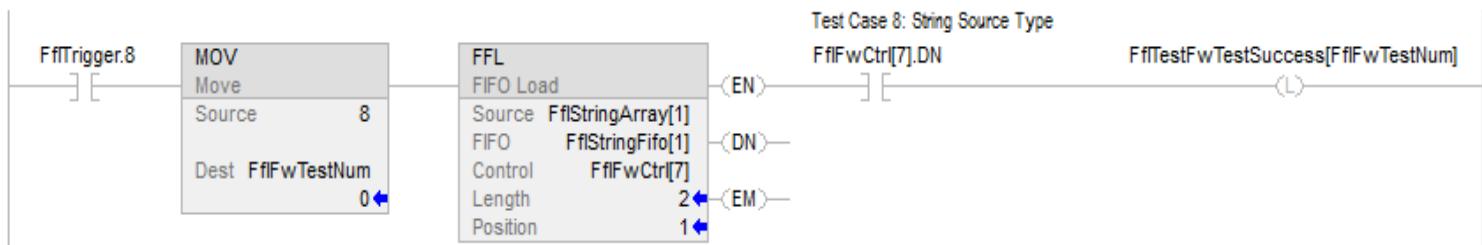
#### Diagrama de escalera



### Ejemplo 2

La matriz Source es una matriz STRING o Structure.

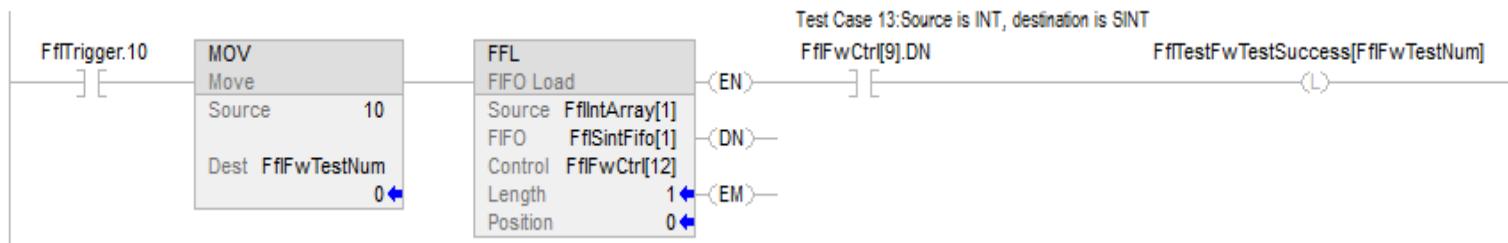
#### Diagrama de escalera



### Ejemplo 3

El tipo de datos de Source no coincide con el tipo de datos de la matriz de FIFO.

### Diagrama de escalera



### Consulte también

[Instrucciones de Matriz \(Archivo\)/Desplazamiento](#) en la página 603

[Descarga FIFO \(FFU\)](#) en la página 621

[Carga LIFO \(LFL\)](#) en la página 628

[Atributos comunes](#) en la página 923

## Descarga FIFO (FFU)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción FFU carga el valor de la posición 0 (primera posición) de FIFO y almacena ese valor en Destination. Los datos restantes de FIFO se desplazan una posición hacia abajo.

### Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

### Operандos

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción.

### Diagrama de escalera

Operando	Tipo	Formato	Descripción
FIFO	SINT INT DINT REAL Tipo de cadena estructura	etiqueta de matriz	FIFO a modificar Especifica el primer elemento de FIFO No se utiliza CONTROL.POS en el subíndice
Destination	SINT INT DINT REAL Tipo de cadena estructura	etiqueta	Valor descargado de FIFO.
Control	CONTROL	etiqueta	Estructura de control para la operación Por lo general, se utiliza el mismo CONTROL que la FFL asociada
Length	DINT	inmediato	Número máximo de elementos que puede retener al mismo tiempo el FIFO.
Position	DINT	inmediato	Siguiente ubicación en FIFO en el que la instrucción carga datos. el valor inicial suele ser 0

### Estructura de CONTROL

Mnemónico	Tipo de datos	Descripción
.EU	BOOL	El bit de habilitación de carga indica que la instrucción FFU está habilitada. El bit .EU se establece para evitar una falsa descarga cuando comienza el pre-escaneado.
.DN	BOOL	El bit de efectuado se establece para indicar que FIFO está lleno (.POS = .LEN).

.EM	BOOL	El bit vacío indica que FIFO está vacío. Si .LEN = 0 o .POS < 0, se establece el bit .EM y los bits .DN.
.LEN	DINT	La longitud especifica el número máximo de elementos en FIFO.
.POS	DINT	La posición identifica el fin de los datos que se han cargado en FIFO.

### Descripción

Se utiliza la instrucción FFU con la instrucción FFL para almacenar y recuperar datos en un orden de tipo el primero en entrar es el primero en salir.

Cuando está habilitada, la instrucción FFU descarga los datos del primer elemento de FIFO y coloca ese valor en Destination. La instrucción descarga un valor cada vez que se habilita la instrucción hasta que FIFO esté vacío. Si FIFO está vacío, FFU devuelve 0 a Destination.

Por lo general, Destination y FIFO usan el mismo tipo de datos. Si difieren los tipos, la instrucción convierte el valor descargado al tipo de la etiqueta de destino.

Un entero menor se convierte a un entero mayor mediante extensión de signo.

### Afectar a las marcas de estado matemático

No

### Fallos mayores/menores

Se producirá un fallo mayor si:	Tipo de fallo	Código de fallo
El valor de Length especificado está más allá del final de la matriz de FIFO	4	20

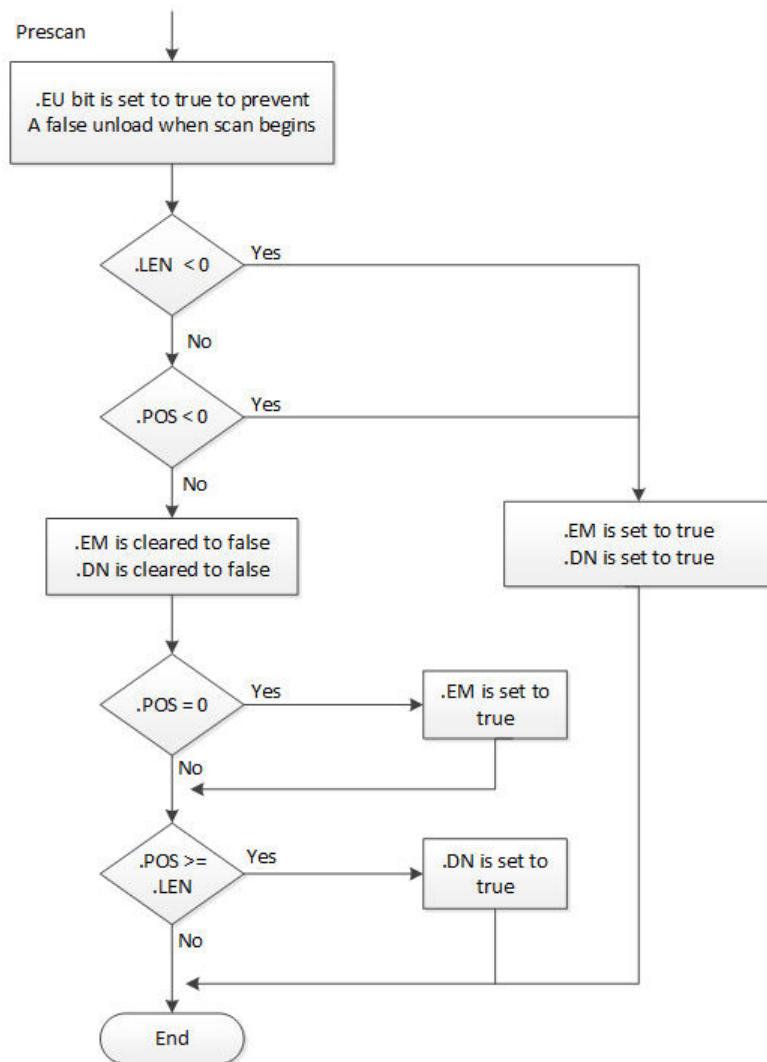
Consulte los Atributos comunes para fallos relacionados con el operando.

### Ejecución

#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	Consultar el Diagrama de flujo de FFU (Pre-escaneado).
La condición de entrada de reglón es falsa	Consultar el Diagrama de flujo de FFL (Falso).
La condición de entrada de reglón es verdadera	Consultar el Diagrama de flujo de FFU (Verdadero)
Post-escaneado	N/A

Diagrama de flujo de FFU (Pre-escaneado)



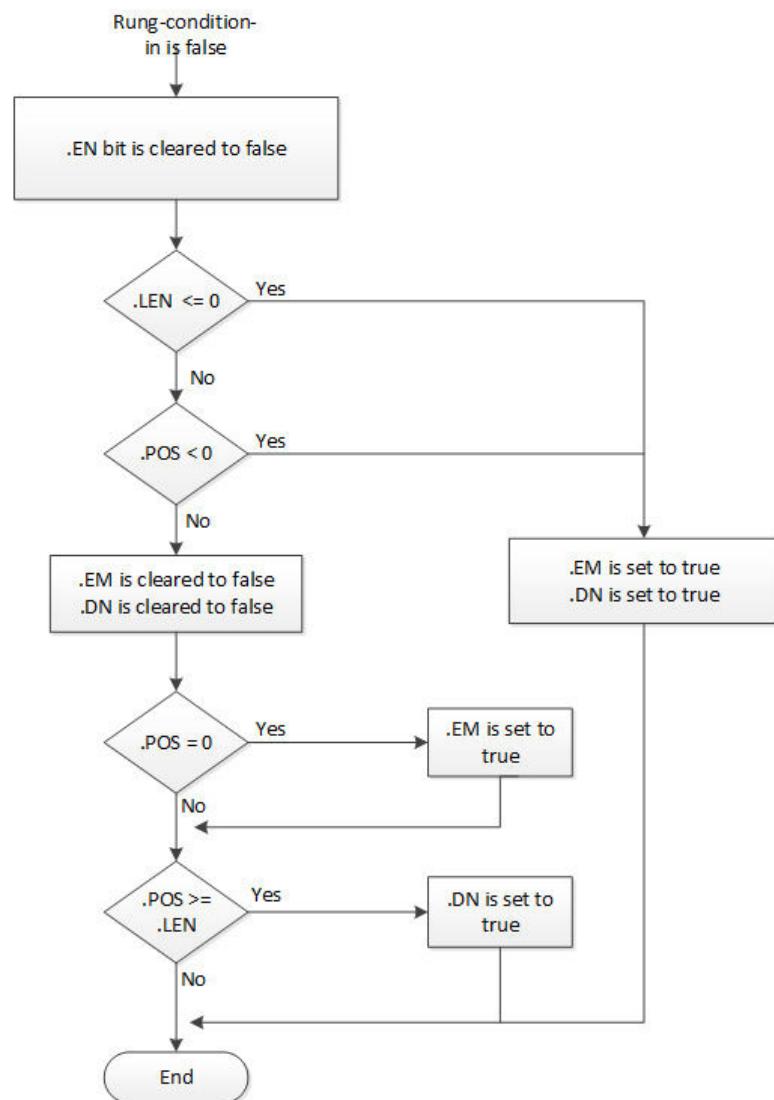
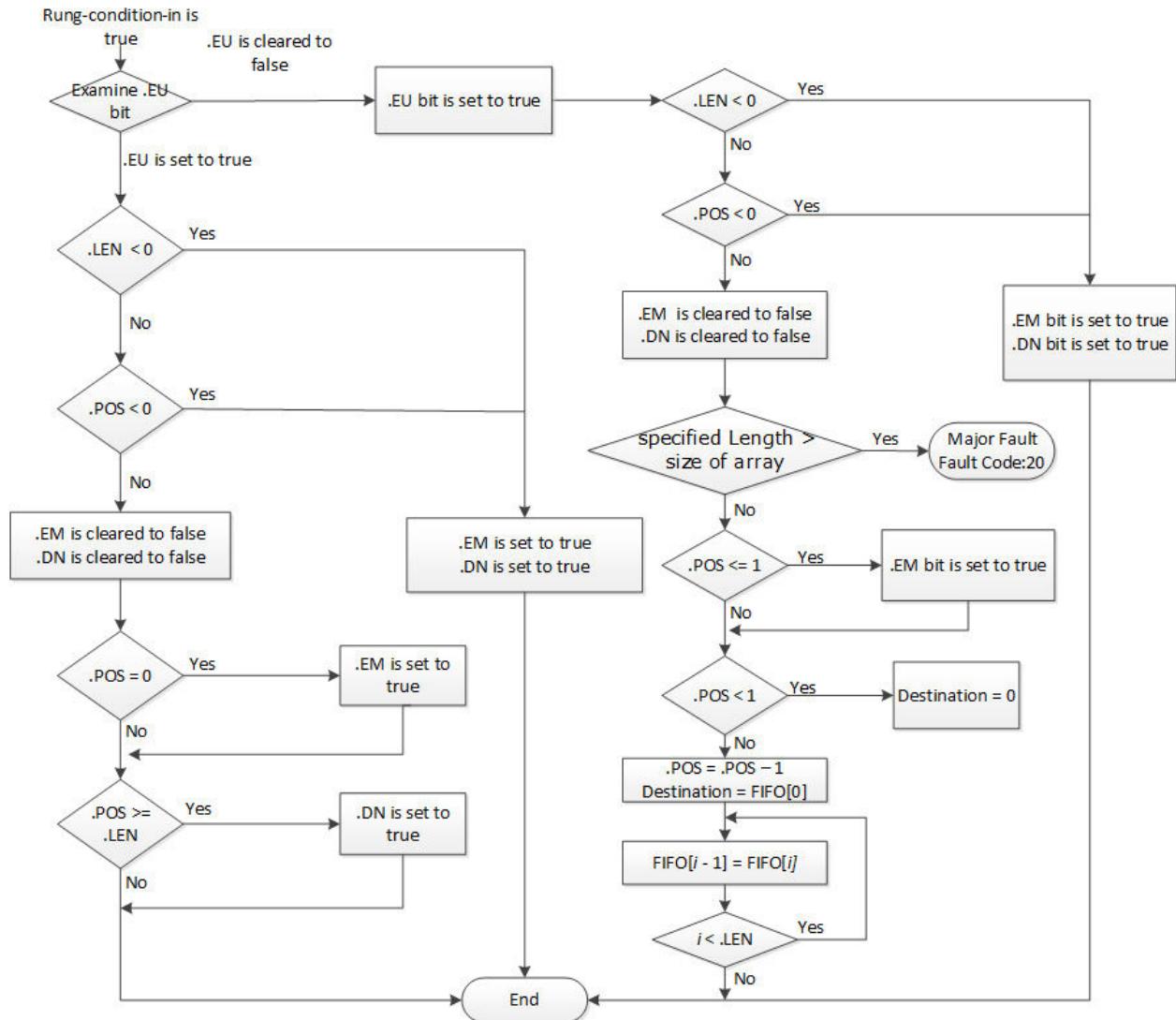
**Diagrama de flujo de FFL (Falso)**

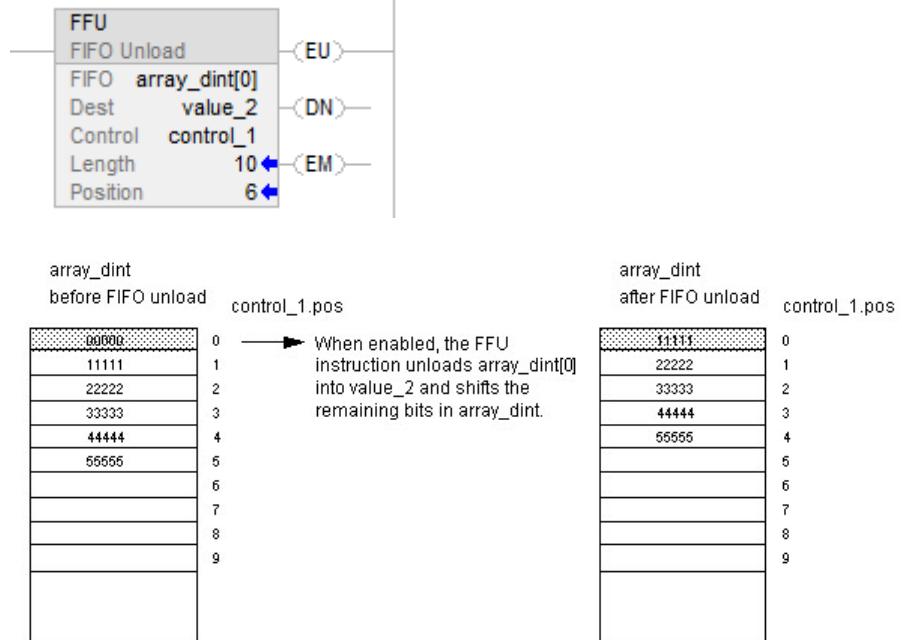
Diagrama de flujo de FFU (Verdadero)



## Ejemplos

### Ejemplo 1

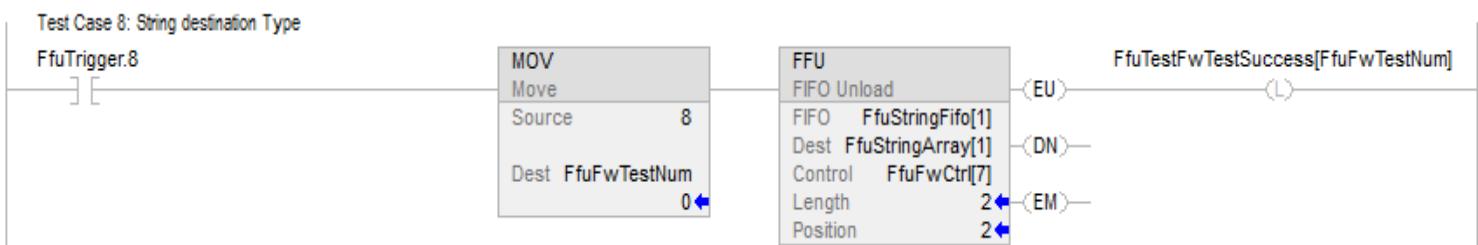
#### Diagrama de escalera



### Ejemplo 2

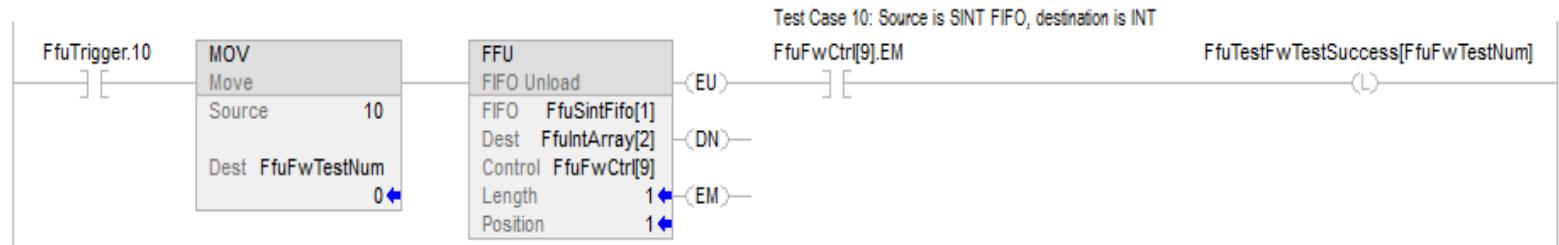
La matriz Destination es una matriz STRING o Structure

#### Diagrama de escalera



### Ejemplo 3

El tipo de datos de la matriz de origen de FIFO no coincide con el tipo de datos de la matriz de destino

**Diagrama de escalera****Consulte también**

[Instrucciones de Matriz \(Archivo\)/Desplazamiento](#) en la página 603

[Atributos comunes](#) en la página 923

[FFL](#) en la página 614

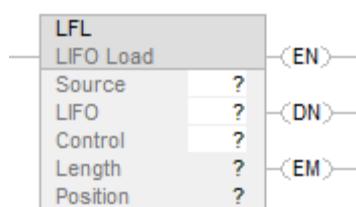
[LFL](#) en la página 628

[LFU](#) en la página 635

**Carga LIFO (LFL)**

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

La instrucción LFL copia el valor de Source en la LIFO.

**Idiomas disponibles****Diagrama de escalera****Bloque de funciones**

Esta instrucción no está disponible en bloque de funciones.

**Texto estructurado**

Esta instrucción no está disponible en texto estructurado.

## Operandos

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción.

### Diagrama de escalera

Operando	Tipo	Formato	Descripción
Source	SINT INT DINT REAL Tipo de cadena estructura	inmediato etiqueta	Datos a almacenar en la LIFO.
LIFO	SINT INT DINT REAL Tipo de cadena estructura	etiqueta de matriz	LIFO a modificar Especifica el primer elemento de la LIFO.
Control	CONTROL	etiqueta	Estructura de control para la operación Por lo general, se utiliza el mismo CONTROL que la LFU asociada.
Length	DINT	inmediato	Número máximo de elementos que puede retener al mismo tiempo la LIFO.
Position	DINT	inmediato	Siguiente ubicación en la LIFO en la que la instrucción carga datos. el valor inicial suele ser 0

### Estructura de CONTROL

Mnemónico	Tipo de datos	Descripción
.EN	BOOL	El bit de habilitación indica que la instrucción LFL está habilitada.
.DN	BOOL	El bit de efectuado se establece para indicar que la LIFO está llena (.POS = .LEN). El bit .DN impide que se cargue la LIFO hasta que .POS < .LEN.

.EM	BOOL	El bit vacío indica que la LIFO está vacía. Si .LEN < 0 = a 0 o .POS < 0, se establece el bit .EM y los bits .DN.
.LEN	DINT	La longitud especifica el número máximo de elementos que puede retener al mismo tiempo la LIFO.
.POS	DINT	La posición identifica la ubicación en la LIFO en que la instrucción cargará el siguiente valor.

### Descripción

Se utiliza la instrucción LFL con la instrucción LFU para almacenar y recuperar datos en un orden de tipo el último en entrar es el primero en salir. Cuando se utilizan en pareja, las instrucciones LFL y LFU establecen un registro de desplazamiento asíncrono.

Por lo general, Source y LIFO usan el mismo tipo de datos.

Cuando está habilitada, la instrucción LFL carga el valor de Source en la posición de la LIFO identificada por el valor de .POS. La instrucción carga un valor cada vez que se habilita la instrucción hasta que la LIFO esté llena.

---

**Importante:** Comprueba y confirma que la instrucción no modifica datos que usted no desea que sean modificados.

---

La instrucción LFL opera en posiciones contiguas de memoria de datos. Para Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570, el ámbito de la instrucción se ve limitado por la etiqueta de base. Por lo general, Source y LIFO usan el mismo tipo de datos. Si no coinciden los tipos de datos de Source y LIFO, la instrucción convierte el valor de Source al tipo de datos de la etiqueta de FIFO. Un entero menor se convierte a un entero mayor mediante extensión de signo.

### Afectar a las marcas de estado matemático

No

### Fallos mayores/menores

Se producirá un fallo mayor si:	Tipo de fallo	Código de fallo
Si (elemento inicial + .POS) está más allá del final de la matriz LIFO	4	20

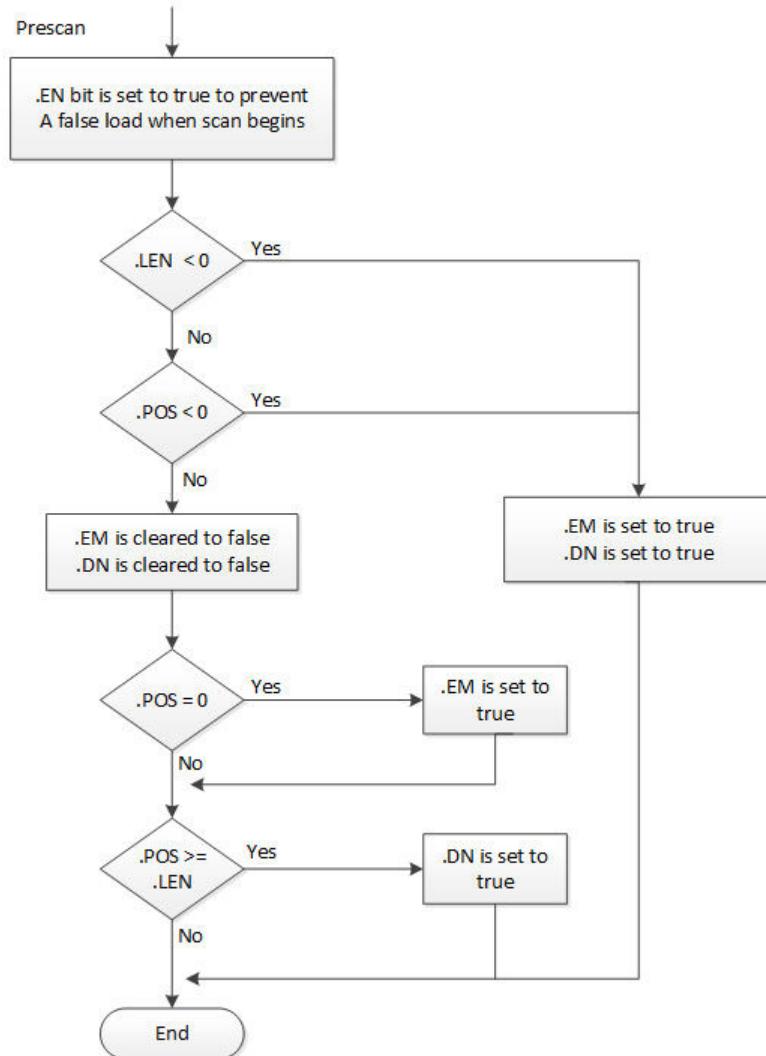
Consulte los Atributos comunes para fallos relacionados con el operando.

## Ejecución

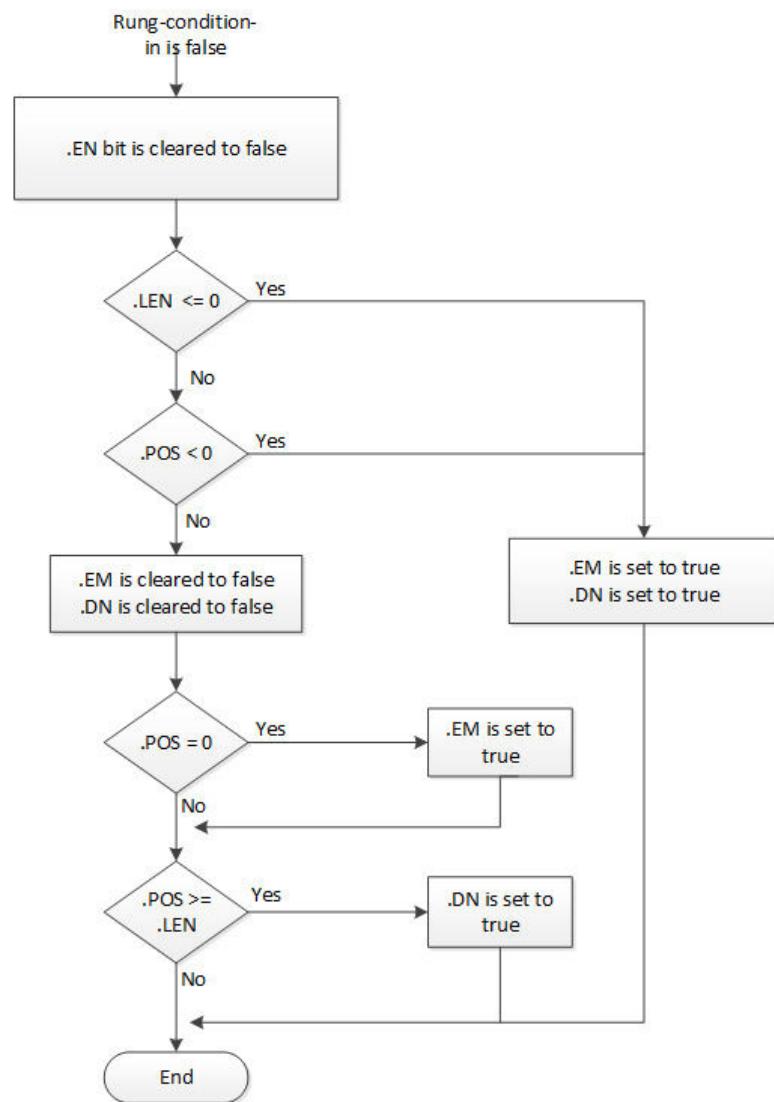
### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	Consultar el Diagrama de flujo de LFL (Pre-escaneado)
La condición de entrada de reglón es falsa	Consultar el Diagrama de flujo de LFL (Falso)
La condición de entrada de reglón es verdadera	Consultar el Diagrama de flujo de LFL (Verdadero)
Post-escaneado	N/A.

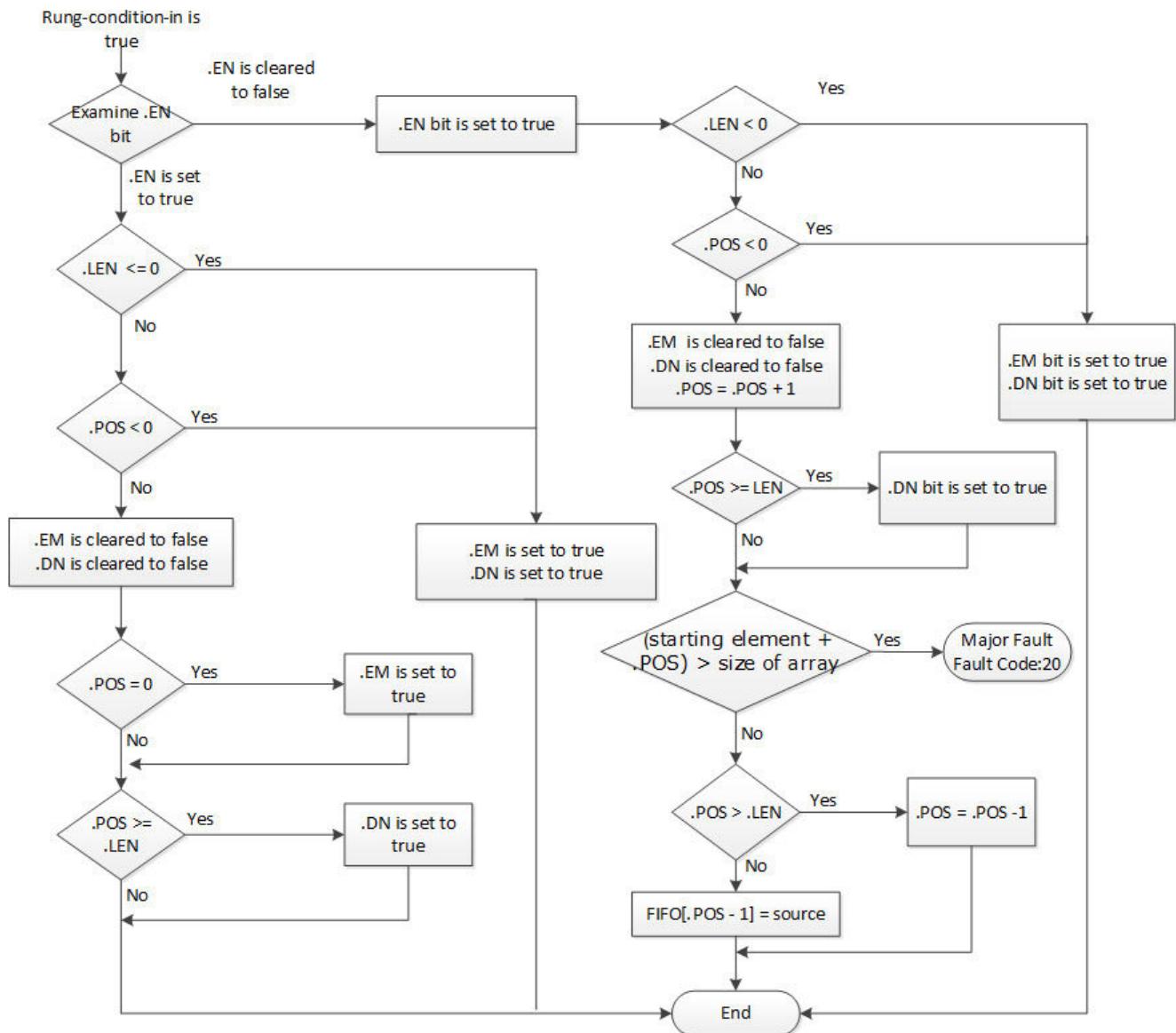
### Diagrama de flujo de LFL (Pre-escaneado)



**Diagrama de flujo de LFL (Falso)**



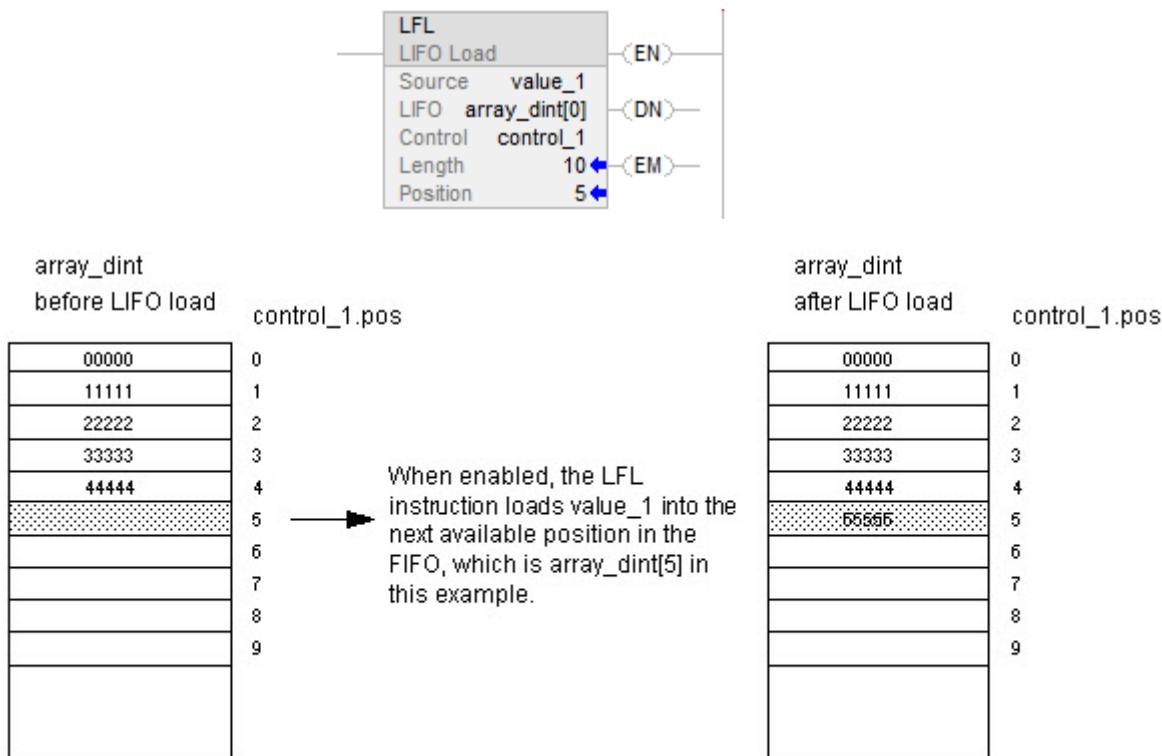
## Diagrama de flujo de LFL (Verdadero)



## Ejemplos

### Ejemplo 1

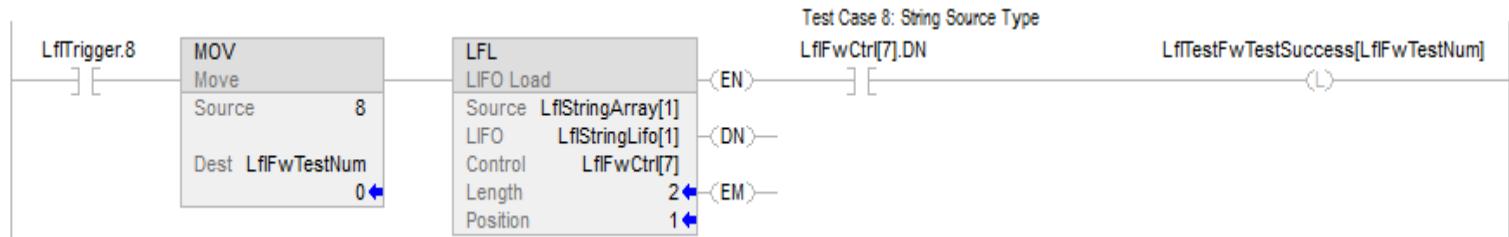
## Diagrama de escalera



## Ejemplo 2

La matriz Source es una matriz STRING o Structure.

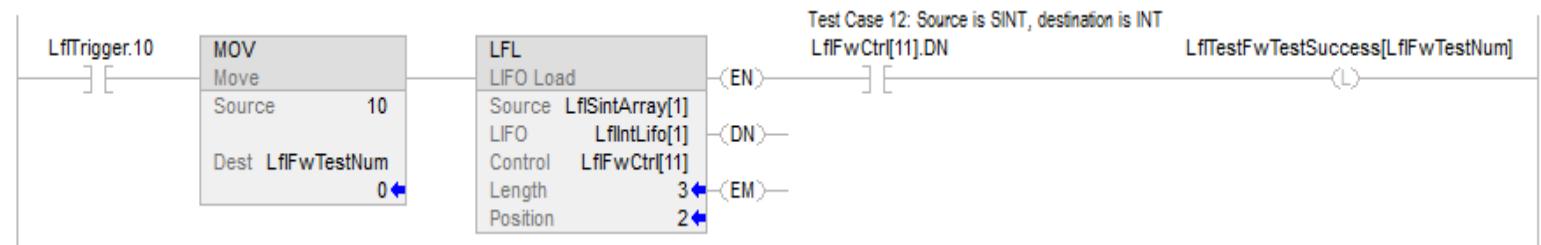
## Diagrama de escalera



### Ejemplo 3

El tipo de datos de Source no coincide con el tipo de datos de la matriz LIFO.

### Diagrama de escalera



### Consulte también

[Instrucciones de Matriz \(Archivo\)/Desplazamiento](#) en la página 603

[Descarga LIFO \(LFU\)](#) en la página 635

[Carga FIFO \(FFL\)](#) en la página 614

[Descarga FIFO \(FFU\)](#) en la página 621

[Atributos comunes](#) en la página 923

## Descarga LIFO (LFU)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

La instrucción LFU descarga el valor en .POS de la LIFO y almacena 0 en esa ubicación.

### Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

### Operандos

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción.

### Diagrama de escalera

Operando	Tipo	Formato	Descripción
LIFO	SINT INT DINT REAL Tipo de cadena estructura	etiqueta de matriz	LIFO a modificar Especifica el primer elemento de la LIFO. No se utiliza CONTROL.POS en el subíndice
Destination	SINT INT DINT REAL Tipo de cadena estructura	etiqueta	Valor descargado de la LIFO.
Control	CONTROL	etiqueta	Estructura de control para la operación Por lo general, se utiliza el mismo CONTROL que la LFL asociada.
Length	DINT	inmediato	Número máximo de elementos que puede retener al mismo tiempo la LIFO.
Position	DINT	inmediato	Siguiente ubicación en la LIFO en la que la instrucción descarga datos. el valor inicial suele ser 0

### Estructura de CONTROL

Mnemónico	Tipo de datos	Descripción
.EU	BOOL	El bit de habilitación indica que la instrucción LFU está habilitada.
.DN	BOOL	El bit de efectuado se establece para indicar que la LIFO está llena (.POS = .LEN).

.EM	BOOL	El bit vacío indica que la LIFO está vacía. Si .LEN es $\leq 0$ o .POS < 0, se establecen los bits .EM y .DN.
.LEN	DINT	La longitud especifica el número máximo de elementos que puede retener al mismo tiempo la LIFO.
.POS	DINT	La posición identifica el fin de los datos que se han cargado en la LIFO.

### Descripción

Utilice la instrucción LFU con la instrucción LFL para almacenar y recuperar datos en un orden de tipo el último en entrar es el primero en salir.

Cuando está habilitada, la instrucción LFU descarga el valor en .POS de la LIFO y coloca ese valor en Destination. La instrucción descarga un valor y lo sustituye por 0 cada vez que la instrucción se habilita hasta que la LIFO esté vacía. Si la LIFO está vacía, la LFU devuelve 0 a Destination.

---

**Importante:** Comprueba y confirma que la instrucción no modifica datos que usted no desea que sean modificados.

---

La instrucción LFU opera en posiciones continuas de memoria. El ámbito de la instrucción se ve limitado por la etiqueta de base. La instrucción LFL no escribirá datos fuera de la etiqueta de base, pero puede pasar los límites de los miembros. Si especifica una matriz que sea miembro de una estructura y la longitud supera el tamaño de dicha matriz, debe probar y confirmar que la instrucción LFL no modifique datos que usted no quiera que sean modificados.

En el caso de Controladores CompactLogix 5380, CompactLogix 5480, ControllLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580., los datos se ven limitados por el miembro especificado.

Si la instrucción intenta leer más allá del final de una matriz, la instrucción establece el bit .ER y genera un fallo mayor.

Por lo general, Source y LIFO usan el mismo tipo de datos. Si no coinciden los tipos de Source y LIFO, la instrucción convierte el valor de Source al tipo de datos de la etiqueta de FIFO.

Un entero menor se convierte a un entero mayor mediante extensión de signo.

### Afectar a las marcas de estado matemático

No

**Fallos mayores/menores**

Se producirá un fallo mayor si:	Tipo de fallo	Código de fallo
Si el valor de Length especificado está más allá del final de la matriz LIFO	4	20

Consulte los Atributos comunes para fallos relacionados con el operando.

**Ejecución**

Todas las condiciones se producen únicamente durante el modo Escaneado normal

**Diagrama de escalera**

Condición/estado	Acción realizada
Pre-escaneado	Consultar el Diagrama de flujo de LFU (Pre-escaneado)
La condición de entrada de reglón es falsa	Consultar el Diagrama de flujo de LFU (Falso)
La condición de entrada de reglón es verdadera	Consultar el Diagrama de flujo de LFU (Verdadero)
Post-escaneado	N/A

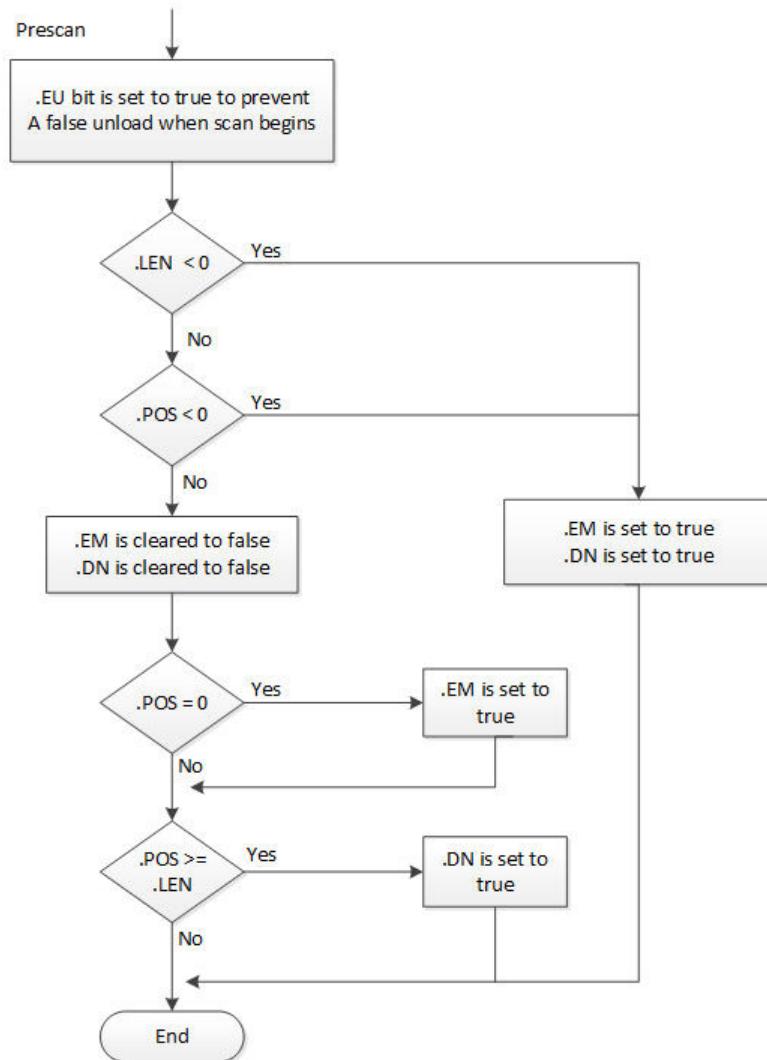
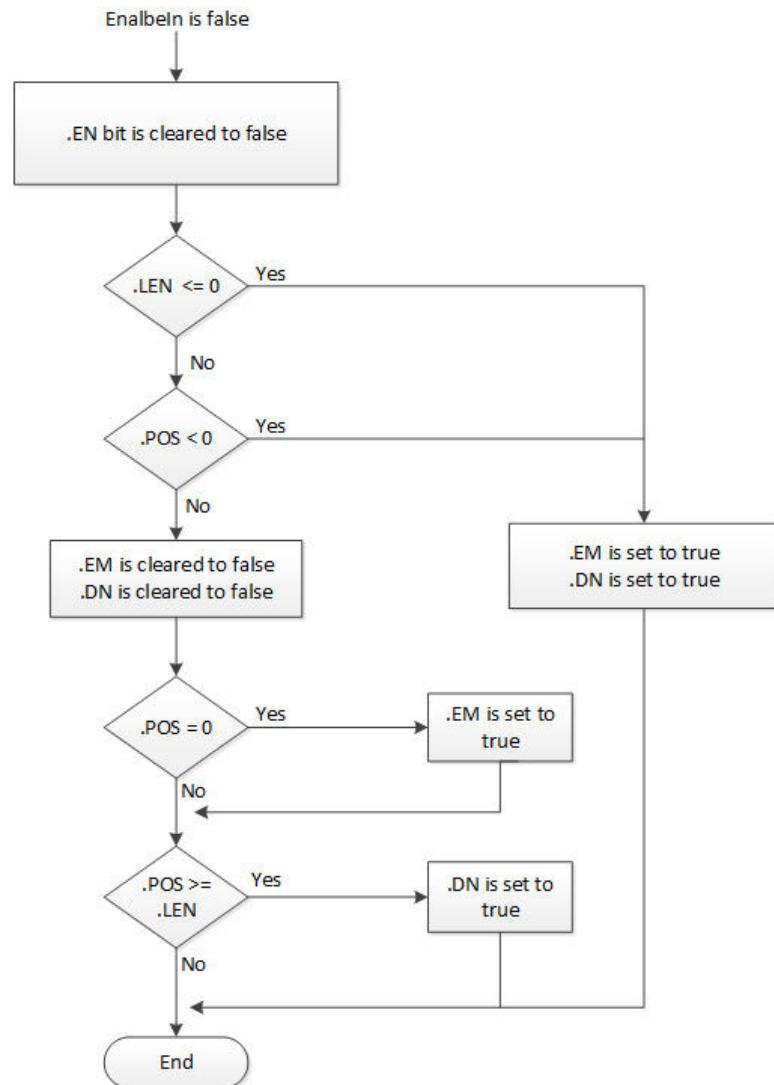
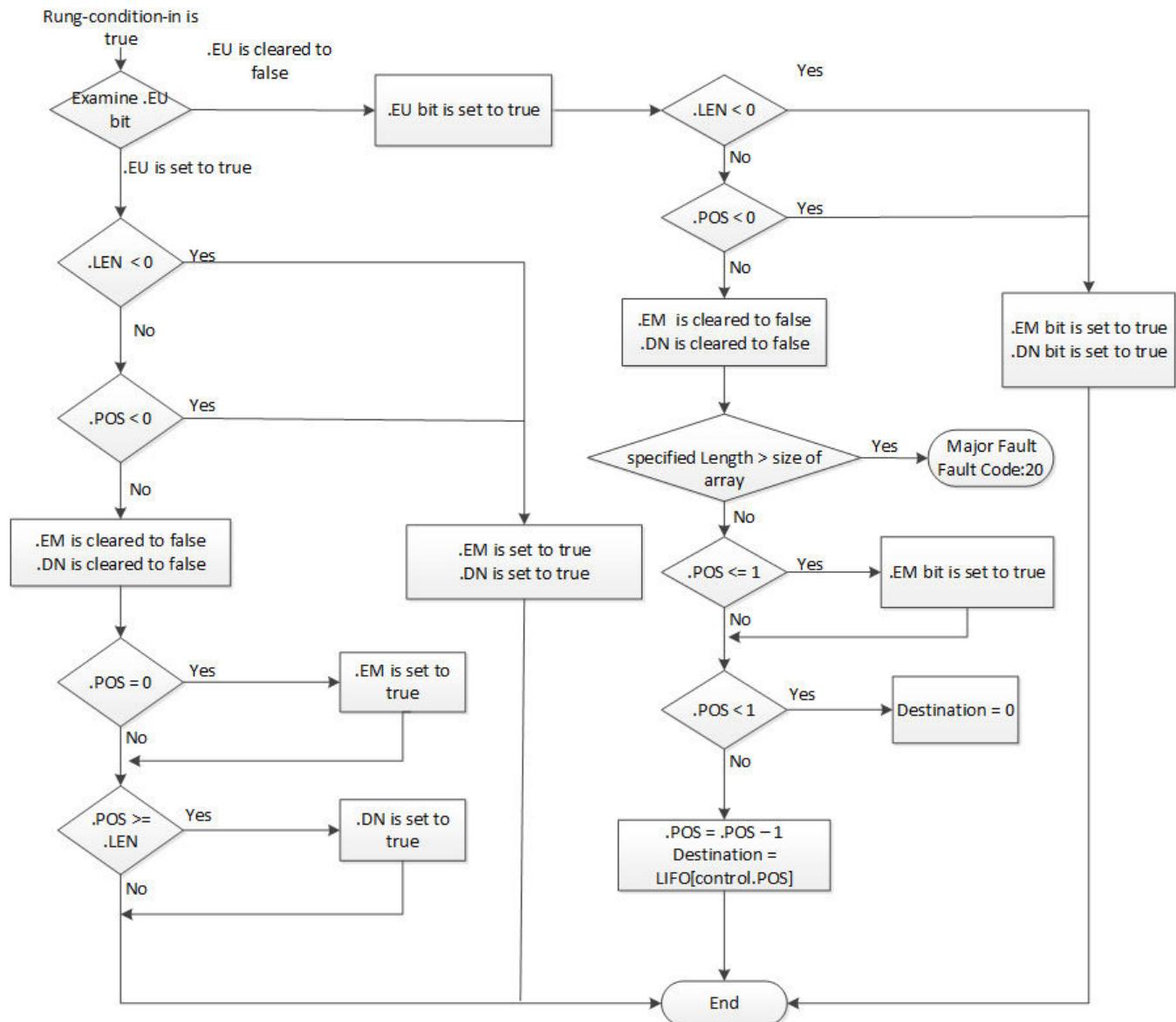
**Diagrama de flujo de LFU (Pre-escaneado)**

Diagrama de flujo de LFU (Falso)



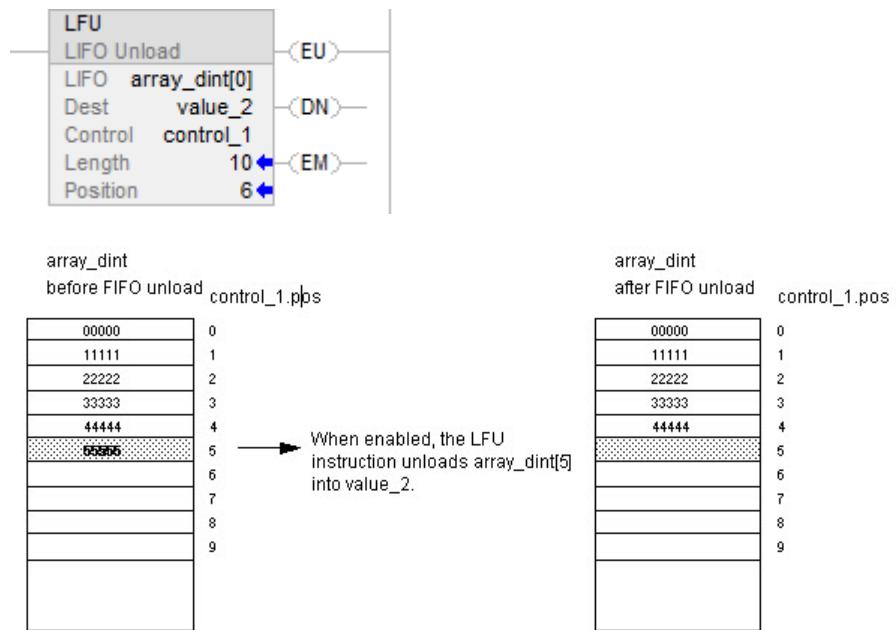
### Diagrama de flujo de LFU (Verdadero)



## Ejemplos

### Ejemplo 1

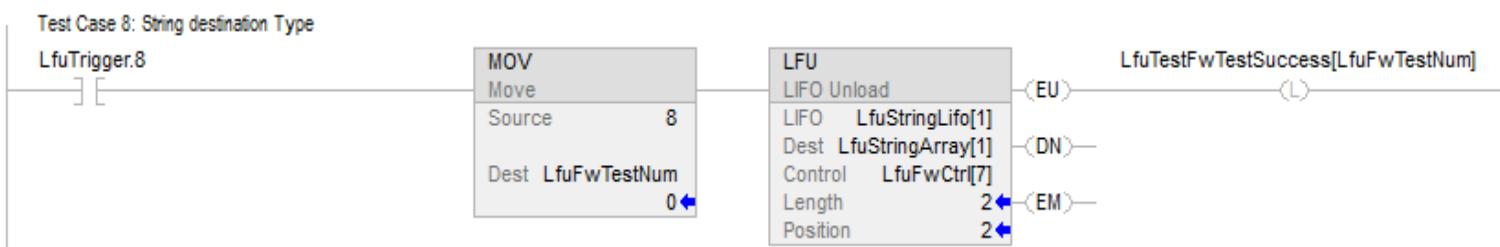
#### Diagrama de escalera



### Ejemplo 2

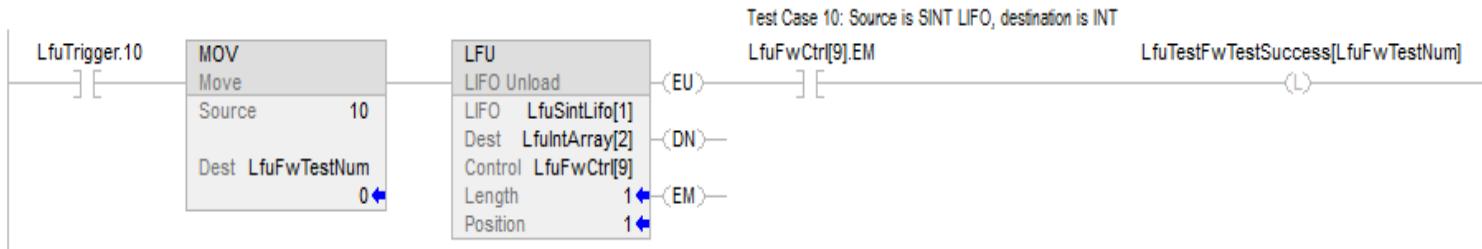
La matriz Destination es una matriz STRING o Structure

#### Diagrama de escalera



### Ejemplo 3

El tipo de datos de la matriz de origen LIFO no coincide con el tipo de datos de la matriz de destino

**Diagrama de escalera****Consulte también**

[Instrucciones de Matriz \(Archivo\)/Desplazamiento](#) en la página 603

[Carga LIFO \(LFL\)](#) en la página 628

[Carga FIFO \(FFL\)](#) en la página 614

[Descarga FIFO \(FFU\)](#) en la página 621

[Atributos comunes](#) en la página 923



# **InSTRUCCIONES DE SECUENCIADOR**

## **InSTRUCCIONES DE SECUENCIADOR**

Las instrucciones de secuenciación supervisan operaciones consistentes y repetibles.

### **InSTRUCCIONES DISPONIBLES**

#### **Diagrama de escalera**

<a href="#">SQI</a>	<a href="#">SQO</a>	<a href="#">SQL</a>
---------------------	---------------------	---------------------

#### **BLOQUE DE FUNCIONES**

No disponible

#### **TEXTO ESTRUCTURADO**

No disponible

<b>SI DESEA</b>	<b>UTILICE ESTA INSTRUCCIÓN</b>
Detectar cuándo se ha completado un paso.	SQI
Establecer condiciones de salida para el siguiente paso.	SQO
Cargar condiciones de referencia en matrices de secuenciador.	SQL

Los tipos de datos que aparecen en **negrita** indican tipos de datos óptimos. Una instrucción se ejecutará a la máxima velocidad y la menor cantidad posible de memoria si todos los parámetros emplean el mismo tipo de datos óptimo, normalmente DINT o REAL.

#### **CONSULTE TAMBIÉN**

[InSTRUCCIONES DE CÁLCULO/MATEMÁTICAS](#) en la página 425

[COMPARAR INSTRUCCIONES](#) en la página 369

[INSTRUCCIONES DE BIT](#) en la página 143

[Instrucciones de cadena ASCII](#) en la página 871

[Instrucciones de conversión ASCII](#) en la página 892

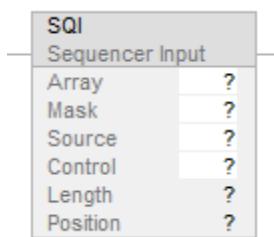
## Secuenciador de entrada (SQI)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción SQI detecta cuando se ha completado un paso en un par de secuencia de instrucciones SQQ/SQI.

### Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

### Operандos

Las reglas de conversión de datos para utilizar diversos tipo de datos dentro de una instrucción. Consulte Conversión de datos.

Operando	Tipo	Formato	Descripción
Array	DINT	etiqueta de matriz	Matriz de secuenciador Especifica el primer elemento de la matriz de secuenciador no se utiliza CONTROL.POS en el subíndice

Mask	SINT INT DINT	etiqueta inmediato	Este operando se utiliza para determinar qué bits bloquear (0) o dejar pasar (1) cuando se aplica al elemento de Array y Source al que hace referencia .POS.  Los tipos INT y SINT se extienden a cero hasta el tamaño de un tipo DINT.
Source	SINT INT DINT	etiqueta inmediato	Los datos de entrada usados para comparar con un elemento de matriz al que hace referencia .POS.
Control	CONTROL	etiqueta	Estructura de control para la operación  Se debería usar la misma etiqueta de control en las instrucciones SQO y SQL.
Length	DINT	inmediato	Esto representa la estructura de CONTROL .LEN.
Position	DINT	inmediato	Esto representa la estructura de CONTROL .POS.

### Estructura de CONTROL

Mnemónico	Tipo de datos	Descripción
.ER (Error)	BOOL	La instrucción detectó un error.
.LEN (Length)	DINT	La longitud especifica el número de pasos de secuenciador en la matriz de secuenciador
.POS (Position)	DINT	La posición identifica el elemento de la Array que la instrucción está comparando con el Source.  El valor inicial suele ser 0

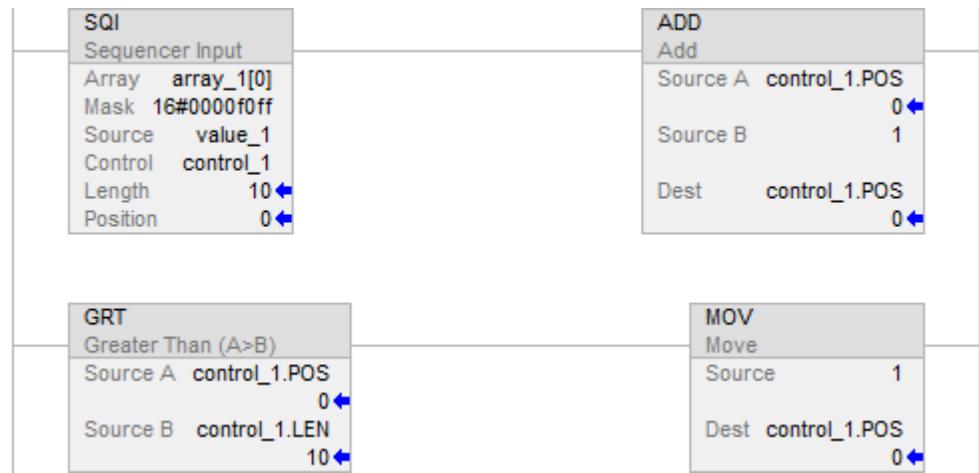
### Descripción

Cuando el valor es Verdadero, la instrucción SQI pasa Source y el elemento de Array actual por Mask. Los resultados de estas operaciones de máscara se comparan y si son iguales, la condición-de salida de reglón se establece en verdadera. De lo contrario, se borra su valor a falso. Por lo general, use la misma estructura de CONTROL que las instrucciones SQO y SQL.

### Usar SQI sin SQO

Cuando la instrucción SQI determina que se ha completado un paso, la instrucción ADD incrementa la matriz de secuenciador. El GRT determina si hay otro valor disponible para comprobarlo en la matriz de secuenciador. La

instrucción MOV restablece el valor de posición tras haber realizado una vez todos los pasos de la matriz de secuenciador.



### Afectar a las marcas de estado matemático

No

### Fallos mayores/menores

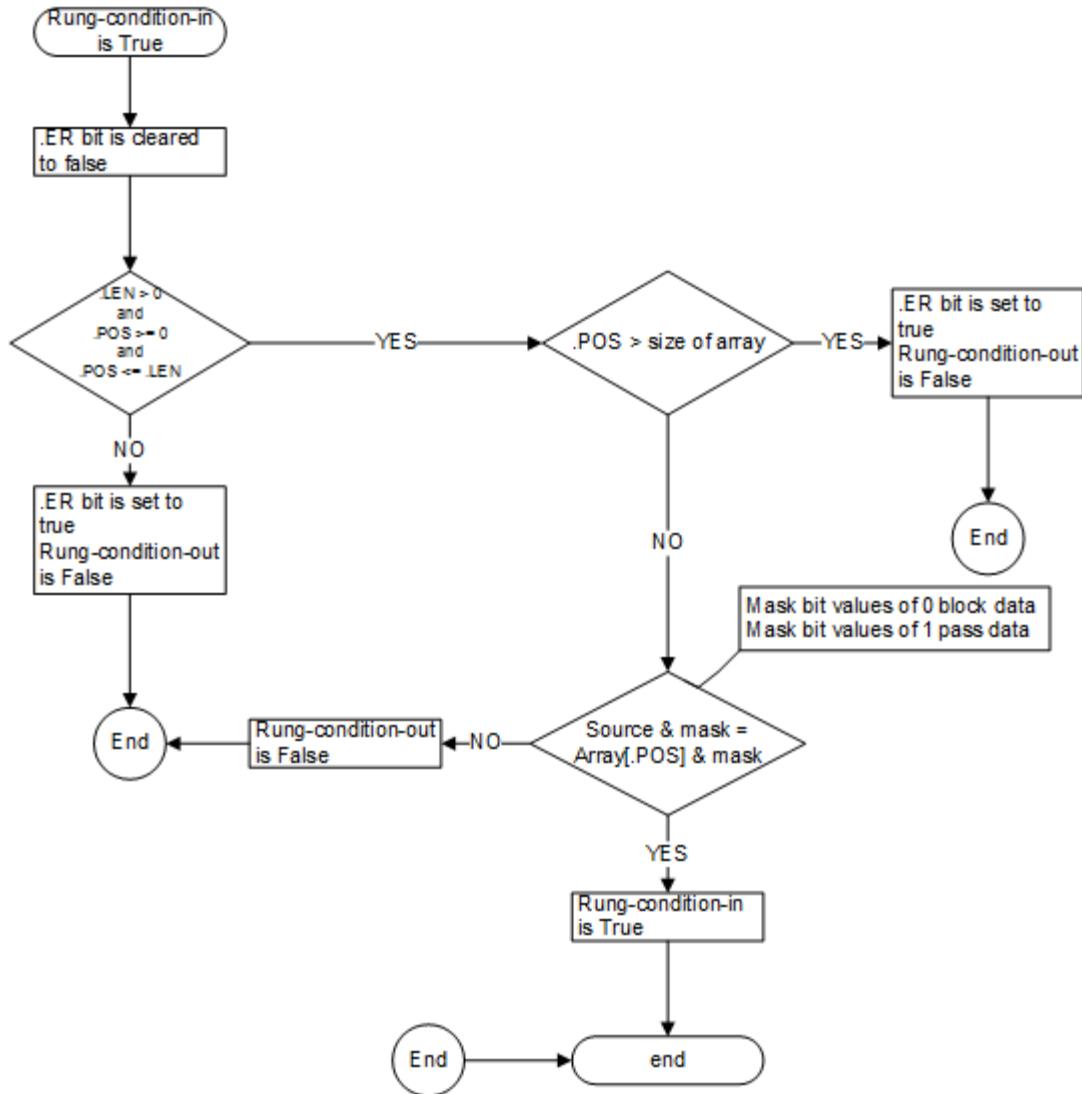
No es específico para esta instrucción. Consulte los Atributos comunes para fallos relacionados con el operando.

### Ejecución

#### Diagrama de escalera

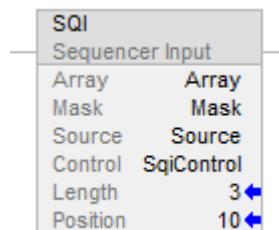
Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	Consultar el Diagrama de flujo (verdadero)
Post-escaneado	N/A

### Diagrama de flujo (verdadero)



### Ejemplo

#### Diagrama de escalera



Si usa la instrucción SQI sin una instrucción SQO emparejada, tiene que incrementar la matriz de secuenciador de manera externa.

El valor de la condición de entrada de reglón se establece en verdadero cuando las instrucciones enableOut sean verdaderas cuando el resultado de realizar una operación AND en el valor de matriz especificado por la Posición, como, p. ej., Array[Position] con el valor de Mask igual al resultado de realizar la operación AND con los valores de Source y Mask. De lo contrario, el valor de condición de salida de pedaño se borrará a falso.

#### Consulte también

[Instrucciones de secuenciador](#) en la página 645

[Atributos comunes](#) en la página 923

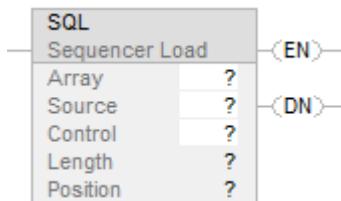
[Conversiones de datos](#) en la página 927

**Carga de secuenciador (SQL)** Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción SQL carga el valor del operando de origen dentro de la matriz de secuenciador.

#### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

#### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

#### Operandos

Las reglas de conversión de datos para utilizar diversos tipo de datos dentro de una instrucción. Consulte Conversión de datos.

Operando	Tipo	Formato	Descripción
Array	DINT	etiqueta de matriz	Matriz de secuenciador especifica el primer elemento de la matriz de secuenciador no se utiliza CONTROL.POS en el subíndice
Source	SINT INT DINT	etiqueta inmediato	Datos a cargar en la matriz de secuenciador en una ubicación especificada por .POS.
Control	CONTROL	etiqueta	estructura de control para la operación Se debería usar la misma etiqueta de control en las instrucciones SQI y SQO.
Length	DINT	inmediato	Esto representa la estructura de CONTROL .LEN.
Position	DINT	inmediato	Esto representa la estructura de CONTROL .POS.

### Estructura de CONTROL

Mnemónico	Tipo de datos	Descripción
.EN (Enable)	BOOL	El bit de habilitación indica que la instrucción SQL está habilitada.
.DN (Done)	BOOL	El bit de efectuado se establece cuando se han cargado todos los elementos especificados en Array.
.ER (Error)	BOOL	El bit de error se establece cuando .LEN < 0 o = a 0, .POS < 0 o .POS > .LEN.
.LEN (Length)	DINT	La longitud especifica el número de pasos de secuenciador en la matriz de secuenciador.
.POS (Position)	DINT	La posición identifica donde se almacenará el valor de Source en Array.

### Descripción

Cuando .EN pasa de falso a verdadero, se incrementa .POS. Se restablece .POS en 1 cuando .POS pasa a ser  $\geq$  .LEN. La instrucción SQL carga el valor de Source en Array en la nueva posición.

Cuando .EN es verdadero, la instrucción SQL carga el valor de Source en Array en la posición actual.

Por lo general, se utiliza la misma estructura de CONTROL que las instrucciones SQI y SQO.

---

**Importante:** Comprueba y confirma que la instrucción no modifica datos que usted no desea que sean modificados.

---

### Afectar a las marcas de estado matemático

No

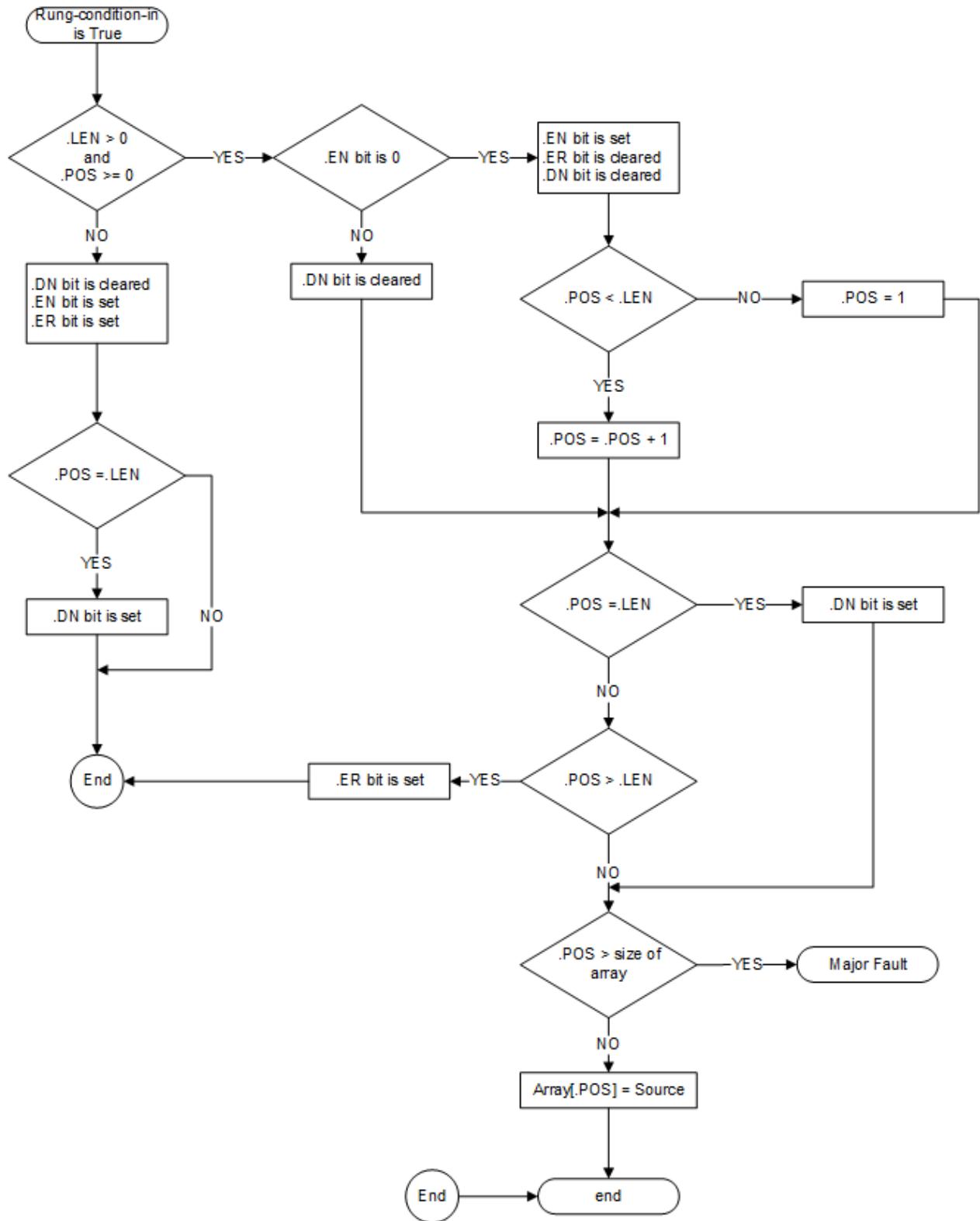
#### Fallos mayores/menores

Se producirá un fallo mayor si:	Tipo de fallo	Código de fallo
posición > tamaño de Array	4	20

### Ejecución

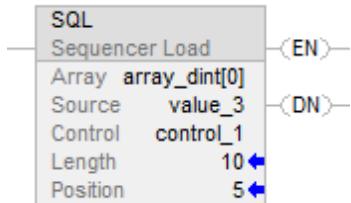
Condición/estado	Acción realizada
Pre-escaneado	Se establece .EN en verdadero.
La condición de entrada de reglón es falsa	Se borra .EN a falso.
La condición de entrada de reglón es verdadera	Consultar el Diagrama de flujo (verdadero)
Post-escaneado	N/A

Diagrama de flujo - verdadero



## Ejemplo

### Diagrama de escalera



Cuando está habilitada, la instrucción SQL carga value\_3 en la siguiente posición de la matriz de secuenciador, que es array\_dint[5] en este ejemplo.

### Consulte también

[Instrucciones de secuenciador](#) en la página 645

[SQO](#) en la página 654

[SQI](#) en la página 646

[Atributos comunes](#) en la página 923

[Conversiones de datos](#) en la página 927

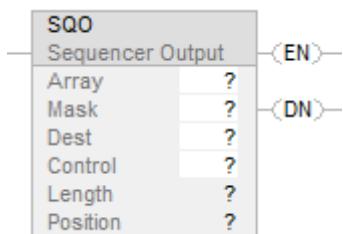
## Secuenciador de salida (SQO)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción SQO establece condiciones de salida para el siguiente paso de un par de secuencia de instrucciones SQO/SQI.

### Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

### Operандos

Las reglas de conversión de datos para utilizar diversos tipo de datos dentro de una instrucción. Consulte Conversión de datos.

Operando	Tipo	Formato	Descripción
Array	DINT	etiqueta de matriz	Matriz de secuenciador especifica el primer elemento de la matriz de secuenciador no se utiliza CONTROL.POS en el subíndice
Mask	SINT INT DINT	etiqueta inmediato	Se utiliza para determinar qué bits bloquear (0) o dejar pasar (1) y se aplica durante la operación de enmascarado de salida.
Destination	DINT	etiqueta	Datos de salida de la matriz de secuenciador. Este valor se utiliza en la operación de enmascarado de salida.
Control	CONTROL	etiqueta	estructura de control para la operación Se debería usar la misma etiqueta de control en las instrucciones SQI y SQL.
Length	DINT	inmediato	Número de elementos en Array (tabla de secuenciador) a la salida
Position	DINT	inmediato	Posición actual de la matriz. El valor inicial suele ser 0.

### Estructura de CONTROL

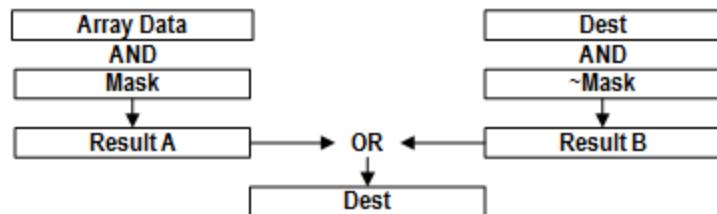
Mnemónico	Tipo de datos	Descripción
.EN (Enable)	BOOL	El bit de habilitación indica que la instrucción SQI está habilitada.
.DN (Done)	BOOL	El bit de efectuado se establece cuando .POS = .LEN.
.ER (Error)	BOOL	Indica que la instrucción detectó un error.
.LEN (Length)	DINT	La longitud especifica el número de pasos de secuenciador en la matriz de secuenciador.

Mnemónico	Tipo de datos	Descripción
.POS (Position)	DINT	La posición identifica el elemento de Array que está usando actualmente la instrucción en la operación de enmascarado de salida.

### Descripción

Cuando .EN pasa de falso a verdadero, se incrementa .POS. El valor de .POS se restablece en 1 cuando .POS pasa a ser mayor o igual a .LEN.

Cuando .EN es verdadero, la instrucción SQO pasa por Mask los datos de Array en .POS y a continuación pasa el valor actual de Destination por Mask complementada. Se aplica una operación OR a todos los resultados de esas operaciones y se almacena el resultado en Destination.



Por lo general, debería usar la misma estructura de CONTROL que las instrucciones SQI y SQL.

### Afectar a las marcas de estado matemático

No

### Fallos mayores/menores

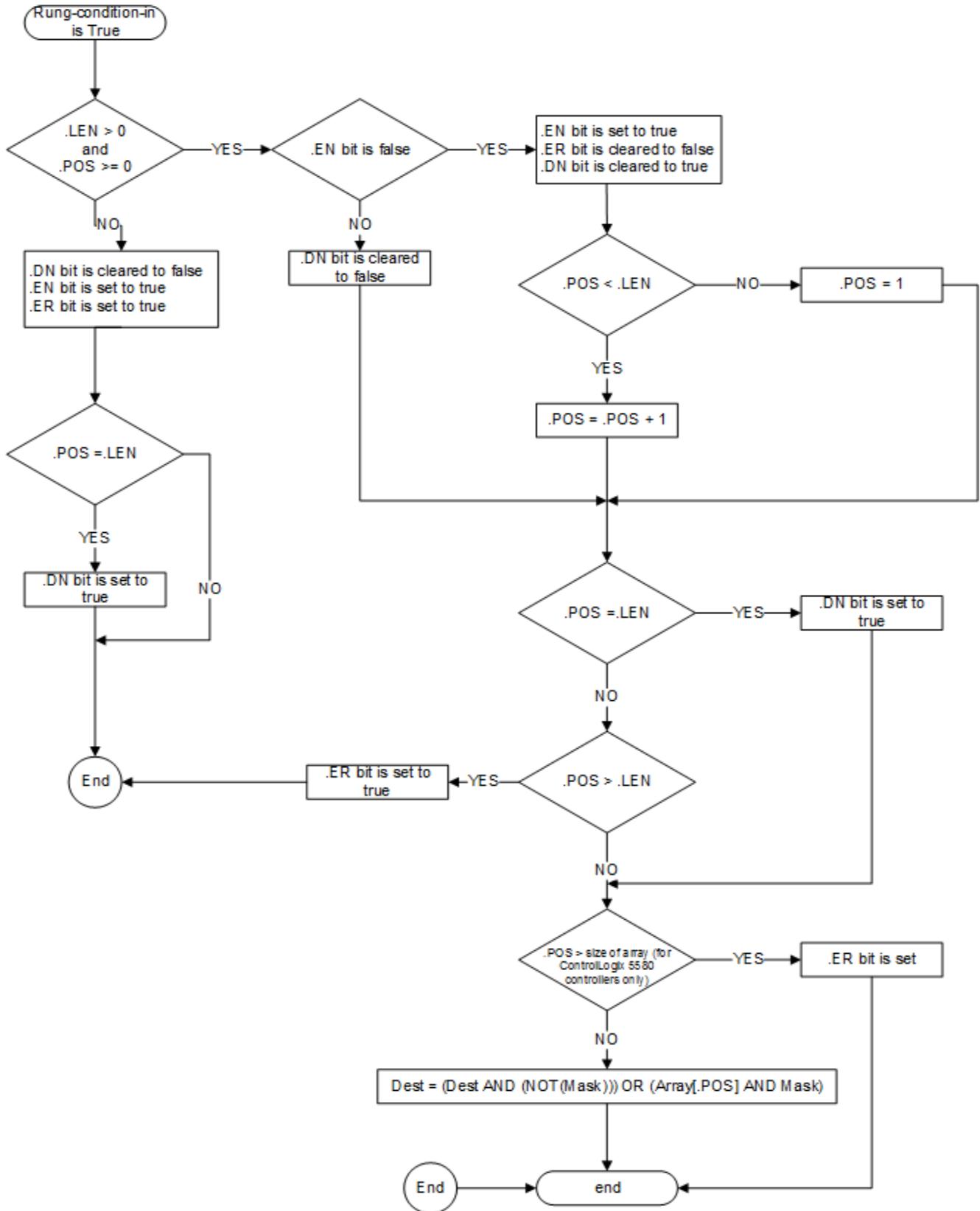
No es específico para esta instrucción. Consulte los Atributos comunes para fallos relacionados con el operando.

### Ejecución

#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	Se establece .EN en verdadero.
La condición de entrada de reglón es falsa	Se borra .EN a falso.
La condición de entrada de reglón es verdadera	Consultar el siguiente Diagrama de flujo (verdadero)
Post-escaneado	N/A

## Diagrama de flujo (verdadero)



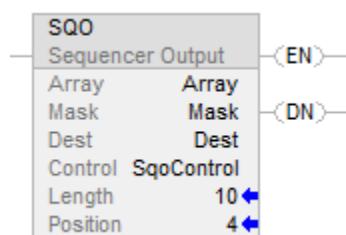
### Ejemplo

El valor de Mask se somete a una operación AND con el valor de la matriz (p. ej., Array[SqoControl.POS]). El complemento del valor de Mask se somete a una operación AND con el valor de Dest actual. Los resultados de estas dos operaciones se someten a una operación OR conjuntamente y se almacena el resultado en Dest.

Para restablecer .POS en el valor inicial (.POS = 0), se utiliza una instrucción RES para borrar la estructura de control. Este ejemplo usa el estado del bit de primer escaneado para borrar el valor de .POS.



**Diagrama de escalera**



### Consulte también

[Instrucciones de secuenciador](#) en la página 645

[SQI](#) en la página 646

[SQL](#) en la página 650

[Atributos comunes](#) en la página 923

[Conversiones de datos](#) en la página 927

# **Instrucciones de control de programa**

Use las instrucciones de control del programa para modificar el flujo de la lógica.

## **Instrucciones disponibles**

### **Diagrama de escalera**

<a href="#">JMP</a>	<a href="#">LBL</a>	<a href="#">JSR</a>	<a href="#">JXR</a>	<a href="#">RET</a>	<a href="#">SBR</a>	<a href="#">TND</a>	<a href="#">MCR</a>
---------------------	---------------------	---------------------	---------------------	---------------------	---------------------	---------------------	---------------------

<a href="#">UID</a>	<a href="#">UIE</a>	<a href="#">SFR</a>	<a href="#">SFP</a>	<a href="#">EVENT</a>	<a href="#">AFI</a>	<a href="#">EOT</a>	<a href="#">NOP</a>
---------------------	---------------------	---------------------	---------------------	-----------------------	---------------------	---------------------	---------------------

### **Bloque de funciones**

<a href="#">JSR</a>	<a href="#">RET</a>	<a href="#">SBR</a>
---------------------	---------------------	---------------------

### **Texto estructurado**

<a href="#">JSR</a>	<a href="#">RET</a>	<a href="#">SBR</a>	<a href="#">TND</a>	<a href="#">EVENT</a>	<a href="#">UID</a>	<a href="#">EOT</a>	<a href="#">SFR</a>
---------------------	---------------------	---------------------	---------------------	-----------------------	---------------------	---------------------	---------------------

<a href="#">UIE</a>	<a href="#">SFP</a>
---------------------	---------------------

<b>Si desea:</b>	<b>Utilice esta instrucción:</b>
Saltar una sección de lógica que no siempre necesita ejecutarse.	JMP LBL
Saltar para separar una rutina, pasar datos a la rutina, ejecutar la rutina y devolver resultados.	JSR SBR RET

Saltar a una rutina externa	JXR
Marcar un fin temporal que detiene la ejecución de la rutina.	TND
Deshabilitar todos los reglones en una sección de lógica	MCR
Deshabilitar las tareas de usuario.	UID
Habilitar las tareas de usuario.	UIE
Pausar un diagrama de funciones secuenciales	SFP
Restablecer un diagrama de funciones secuenciales	SFR
Finalizar una transición para una diagrama de funciones secuenciales	EOT
Desencadenar la ejecución de una tarea de evento	EVENT
Deshabilitar un reglón	AFI
Insertar un indicador de posición en la lógica.	NOP

### Consulte también

[Instrucciones de cálculo/matemáticas](#) en la página 425

[Comparar instrucciones](#) en la página 369

[Instrucciones de bit](#) en la página 143

[Instrucciones de cadena ASCII](#) en la página 871

[Instrucciones de conversión ASCII](#) en la página 892

## Instrucciones de control de programa

### Instrucciones disponibles

#### Diagrama de escalera

<a href="#">JMP</a>	<a href="#">LBL</a>	<a href="#">JSR</a>	<a href="#">JXR</a>	<a href="#">RET</a>	<a href="#">SBR</a>	<a href="#">TND</a>	<a href="#">MCR</a>
---------------------	---------------------	---------------------	---------------------	---------------------	---------------------	---------------------	---------------------

<a href="#">UID</a>	<a href="#">UIE</a>	<a href="#">SFR</a>	<a href="#">SFP</a>	<a href="#">EVENT</a>	<a href="#">AFI</a>	<a href="#">EOT</a>	<a href="#">NOP</a>
---------------------	---------------------	---------------------	---------------------	-----------------------	---------------------	---------------------	---------------------

### Bloque de funciones



### Texto estructurado



Si desea:	Utilice esta instrucción:
Saltar una sección de lógica que no siempre necesita ejecutarse.	JMP LBL
Saltar para separar una rutina, pasar datos a la rutina, ejecutar la rutina y devolver resultados.	JSR SBR RET
Saltar a una rutina externa	JXR
Marcar un fin temporal que detiene la ejecución de la rutina.	TND
Deshabilitar todos los reglones en una sección de lógica	MCR
Deshabilitar las tareas de usuario.	UID
Habilitar las tareas de usuario.	UIE
Pausar un diagrama de funciones secuenciales	SFP
Restablecer un diagrama de funciones secuenciales	SFR
Finalizar una transición para una diagrama de funciones secuenciales	EOT
Desencadenar la ejecución de una tarea de evento	EVENT
Deshabilitar un reglón	AFI
Insertar un indicador de posición en la lógica.	NOP

### Consulte también

[Instrucciones de cálculo/matemáticas](#) en la página 425

[Comparar instrucciones](#) en la página 369

[Instrucciones de bit](#) en la página 143

[Instrucciones de cadena ASCII](#) en la página 871

[Instrucciones de conversión ASCII](#) en la página 892

## Instrucción siempre falso (AFI)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción AFI establece el valor de EnableOut en falso.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

#### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

#### Operандos

#### Diagrama de escalera

Ninguno

#### Descripción

La instrucción AFI establece el valor de EnableOut en falso.

#### Afectar a las marcas de estado matemático

No

## Condiciones de fallo

No es específico para esta instrucción. Consulte *Atributos comunes* para obtener información sobre fallos relacionados con operandos.

## Ejecución

Las condiciones por debajo de la línea continua gruesa solo se pueden dar durante el modo Escaneado normal.

Condición	Acción
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	Borrar EnableOut a falso.
La condición de entrada de reglón es verdadera	Borrar EnableOut a falso.
Post-escaneado	N/A

## Ejemplos

### Diagrama de escalera

Se utiliza la instrucción AFI para deshabilitar de forma temporal un reglón mientras esté depurando un programa. AFI deshabilita todas las instrucciones de este reglón.



### Consulte también

[Instrucciones de control de programa](#) en la página 660

[Restablecimiento de control maestro \(MCR\)](#) en la página 681

[Sin operación \(NOP\)](#) en la página 686

[Fin temporal \(TND\)](#) en la página 693

[Atributos comunes](#) en la página 923

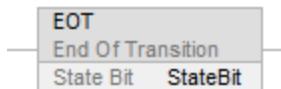
## Fin de transición (EOT)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción EOT se utiliza para establecer el estado de una transición. Por lo general ocurre en una subrutina a la que se llama desde una transición (JSR). El parámetro de bit de estado utilizado en EOT determina el estado de transición. Si el bit de estado se establece en verdadero, SFC pasa al siguiente estado; de lo contrario, EOT actúa como NOP.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

#### Texto estructurado

EOT(StateBit);

#### Operandos

#### Diagrama de escalera

Operando	Tipo	Formato	Descripción
State Bit	BOOL	etiqueta	Estado de la transición (0=ejecutándose, 1=completada)

#### Texto estructurado

Operando	Tipo	Formato	Descripción
State Bit	BOOL	etiqueta	Estado de la transición (0=ejecutándose, 1=completada)

Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

#### Descripción

Dado que la instrucción EOT devuelve un estado booleano, varias rutinas SFC pueden compartir la misma rutina que contenga la instrucción EOT. Si la rutina que originó la llamada no es una transición, la instrucción EOT actúa como una instrucción NOP.

En un controlador Logix, el parámetro de retorno devuelve el estado de la transición, dado que la condición de reglón no está disponible en todos los lenguajes de programación de Logix.

## Afectar a las marcas de estado matemático

No

## **Condiciones de fallo**

No es específico para esta instrucción. Consulte los *Atributos comunes* para fallos relacionados con el operando.

## Ejecución

## Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	La instrucción devuelve el valor del bit de datos a la rutina que originó llamada.
Post-escaneado	N/A

## Texto estructurado

Condición/estado	Acción realizada
Pre-escaneado	N/A
Ejecución normal	La instrucción devuelve el valor del bit de datos a la rutina que originó llamada.
Post-escaneado	N/A

## Ejemplo



### **Consulte también**

Atributos comunes en la página 923

Sintaxis de texto estructurado en la página 955

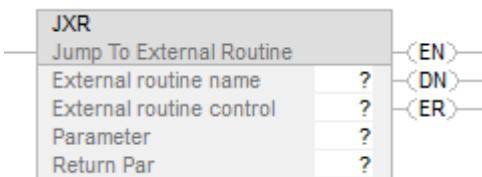
## Saltar a subrutina externa (JXR)

Esta información es válida solo para los controladores SoftLogix 5800.

La instrucción JXR ejecuta una rutina externa.

### Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones

Esta instrucción no está disponible para bloque de funciones.

### Texto estructurado

Esta instrucción no está disponible para texto estructurado.

### Operандos

### Diagrama de escalera

Operando	Tipo	Formato	Descripción
External routine name	ROUTINE	Nombre	Rutina externa a ejecutar
External routine control	EXT_ROUTINE_CONTROL	Etiqueta	Estructura de control
Parameter	BOOL SINT INT DINT REAL estructura	Inmediato Etiqueta Etiqueta de matriz	Datos de esta rutina que quiere copiar en una variable de la rutina externa. Los parámetros son opcionales. Introduce varios parámetros si es necesario. Puede tener hasta 10 parámetros.

Return parameter	BOOL SINT INT DINT REAL	Etiqueta	Etiqueta de esta rutina en la que quiere copiar un resultado de la rutina externa. El parámetro de retorno es opcional. Solo puede tener un parámetro de retorno.
------------------	-------------------------------------	----------	---

### Estructura de EXT\_ROUTINE\_CONTROL

Mnemónico	Tipo de datos	Descripción	Implementación
ErrorCode	SINT	Si se produce un error, este valor identifica el error. Los valores válidos son de 0 a 255.	No hay códigos de error predefinidos. El desarrollador de la rutina externa debe proporcionar los códigos de error.
NumParams	SINT	Este valor indica el número de parámetros asociados a esta instrucción.	Mostrar solo - esta información se deriva a partir de la entrada de la instrucción.
ParameterDefs	EXT_ROUTINE_PARAMETERS[10]	Esta matriz contiene definiciones de los parámetros a pasar a la rutina externa. La instrucción puede pasar hasta un máximo 10 parámetros.	Mostrar solo - esta información se deriva a partir de la entrada de la instrucción.
ReturnParamDef	EXT_ROUTINE_PARAMETERS	Este valor contiene definiciones del parámetro de retorno de la rutina externa. Solo hay un parámetro de retorno.	Mostrar solo - esta información se deriva a partir de la entrada de la instrucción.
EN	BOOL	Cuando está establecido, el bit de habilitación indica que la instrucción JXR está habilitada.	La rutina externa establece este bit.
ReturnsValue	BOOL	Si se establece, este bit indica que se introdujo un parámetro de retorno para la instrucción. Si se borra, este bit indica que no se introdujo ningún parámetro de retorno para la instrucción.	Mostrar solo - esta información se deriva a partir de la entrada de la instrucción.
DN	BOOL	El bit de efectuado se establece cuando la rutina externa se haya ejecutado una vez hasta completarse.	La rutina externa establece este bit.

ER	BOOL	El bit de error se establece si se produce un error. La instrucción deja de ejecutarse hasta que el programa borre el bit de error.	La rutina externa establece este bit.								
FirstScan	BOOL	Este bit identifica si este es o no el primer escaneado tras activar el modo Marcha del controlador. Se utiliza FirstScan para inicializar la rutina externa si fuese necesario.	El controlador establece este bit para reflejar el estado de escaneado.								
EnableOut	BOOL	Habilita la salida.	La rutina externa establece este bit.								
EnableIn	BOOL	Habilita la entrada.	El controlador establece este bit para reflejar el valor de condición de entrada de reglón. La instrucción se ejecuta independientemente del valor de la condición de reglón. El desarrollador de la rutina externa debería supervisar este estado y actuar en consecuencia.								
User1	BOOL	Estos bits están disponibles para el usuario. El controlador no inicializa estos bits.	Tanto la rutina externa como el programa pueden establecer estos bits.								
User0	BOOL										
ScanType1	BOOL	Estos bits identifican el tipo de escaneado actual:  <table border="1"><thead><tr><th>Valores de bit</th><th>Tipo de escaneado</th></tr></thead><tbody><tr><td>00</td><td>Normal</td></tr><tr><td>01</td><td>Pre-escaneado</td></tr><tr><td>10</td><td>Post-escaneado (no aplicable a programas de escalera de relé)</td></tr></tbody></table>	Valores de bit	Tipo de escaneado	00	Normal	01	Pre-escaneado	10	Post-escaneado (no aplicable a programas de escalera de relé)	El controlador establece estos bits para reflejar el estado del escaneado.
Valores de bit	Tipo de escaneado										
00	Normal										
01	Pre-escaneado										
10	Post-escaneado (no aplicable a programas de escalera de relé)										
ScanType0	BOOL										

### Descripción

Se utiliza la instrucción Saltar a rutina externa (JXR) para llamar a la rutina externa desde una rutina de escalera de su proyecto. La instrucción JXR admite varios parámetros para que pueda pasar valores entre la rutina de escalera y la rutina externa.

La instrucción JXR es similar a la instrucción Saltar a subrutina (JSR). La instrucción JXR inicia la ejecución de la rutina externa especificada:

- La rutina externa se ejecuta una vez.
- Después de que se ejecuta la rutina externa, la ejecución de la lógica regresa a la rutina que contiene la instrucción JXR.

### Afectar a las marcas de estado matemático

No

### Fallos mayores/menores

Se producirá un fallo mayor si	Tipo de fallo	Código de fallo:
Se produce una excepción en el DLL de la rutina externa. No se pudo crear el DLL. No se encontró el punto de entrada en el DLL.	4	88

### Ejecución

La instrucción JXR puede ser sincrónica o asincrónica, dependiendo de la implementación del DLL. El código del DLL también determina cómo responder al estado de escaneado, el estado de condición de entrada de reglón y el estado de condición de salida de reglón.

Para obtener más información sobre el uso de la instrucción JXR y la creación de rutinas externas, consulte el Manual de usuario de sistema de SoftLogix5800, publicación 1789-UM002.

### Consulte también

[Atributos comunes](#) en la página 923

## Saltar a etiqueta (JMP) y Etiqueta (LBL)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

Las instrucciones JMP y LBL se saltan partes de la lógica de escalera.

### Idiomas disponibles

#### Diagrama de escalera

—(JMP)—

—[LBL]—

### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

### Operандos

#### Diagrama de escalera

Operando	Tipo	Formato	Descripción
instrucción JMP			
Label name		nombre de etiqueta	Se introduce el nombre de la instrucción LBL asociada
instrucción LBL			
		nombre de etiqueta	La ejecución salta a la instrucción LBL a la que se hace referencia.

### Descripción

Si su valor es verdadero, la instrucción JMP salta hasta la instrucción LBL a la que se hace referencia y el controlador continúa ejecutando a partir de dicho punto. Si su valor es falso, la instrucción JMP no afecta a la ejecución de escalera.

Las instrucciones JMP y LBL a las que hace referencia deben estar en la misma rutina.

La instrucción JMP puede desplazar la ejecución de escalera hacia delante o hacia detrás. Si se salta hacia delante hasta una etiqueta, se ahorra tiempo de escaneado de programa mediante la omisión de un segmento de la lógica hasta que se necesita. Si se salta hacia detrás, se permite al controlador repetir iteraciones de la lógica.

---

**Importante:** Tenga cuidado de no saltar hacia detrás demasiadas veces. El temporizador de vigilancia podría agotar el tiempo de espera porque el escaneado no se completa a tiempo.

---



La lógica que se salta, no se escanea. Coloca las operaciones de lógica críticas fuera de la zona que se salta.

Una instrucción JMP requiere que la etiqueta asociada exista antes de que:

- Descargar al trabajar sin conexión

- Aceptar ediciones al trabajar en línea

La instrucción LBL debe ser la primera instrucción del reglón.

Un nombre de etiqueta debe ser único dentro de una rutina. El nombre puede:

- Tener un máximo de 40 caracteres.
- Contener letras, números y guiones bajos (\_)

#### Afectar a las marcas de estado matemático

No.

#### Fallos mayores/menores

No es específico para esta instrucción. Consulte los *Atributos comunes* para fallos relacionados con el operando.

#### Ejecución

#### Diagrama de escalera

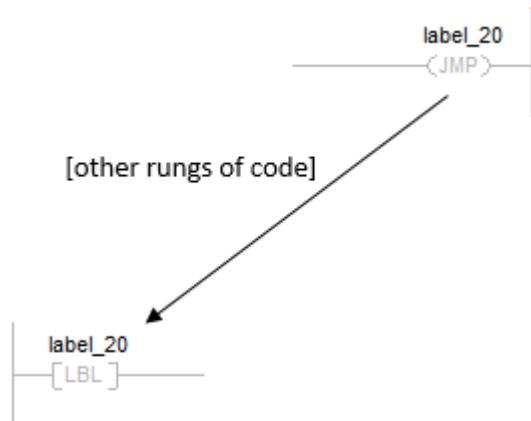
Condición	Acción
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	(Para JMP) La ejecución salta al reglón que contiene la instrucción LBL con el nombre de etiqueta al que se hace referencia. (Para LBL) no se realiza ninguna acción
Post-escaneado	N/A

## Ejemplo

### Diagrama de escalera

#### JMP

Cuando se habilita la instrucción JMP, la ejecución salta por encima de reglones sucesivos de lógica hasta que alcanza el reglón que contiene la instrucción LBL con label\_20.



#### LBL



### Consulte también

[Instrucciones de control de programa](#) en la página 660

[Saltar a subrutina \(JSR\), Subrutina \(SBR\) y Retorno \(RET\)](#) en la página 672

[Instrucción FOR \(FOR\)](#) en la página 707

[Interrupción \(BRK\)](#) en la página 705

[Atributos comunes](#) en la página 923

## Saltar a subrutina (JSR), Subrutina (SBR) y Retorno (RET)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

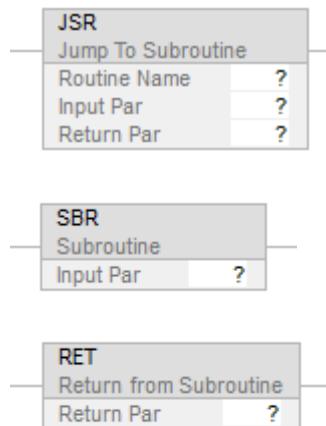
La instrucción JSR invoca otra rutina. Cuando finaliza dicha rutina, la ejecución vuelve a la instrucción JSR.

La instrucción SBR recibe los parámetros de entrada pasados por JSR.

La instrucción RET pasa los parámetros de retorno de vuelva a JSR y termina el escaneado de la subrutina.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones



#### Diagrama de funciones secuenciales



### Texto estructurado

JSR(RoutineName,InputCount,InputPar,ReturnPar);

SBR(InputPar);

RET(ReturnPar);

### Operандos

**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
- Los miembros de un operando de estructura se sobrescriben.
- A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.



Para cada parámetro de una instrucción SBR o RET, se utiliza el mismo tipo de datos (incluyendo cualquier dimensión de matriz) como el parámetro correspondiente en la instrucción JSR. Si utiliza tipos de datos diferentes, podrían producirse resultados imprevistos.

### Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción
<b>Instrucción JSR:</b>			
Routine Name	ROUTINE	nombre	Subrutina a ejecutar.
Input Par	BOOL SINT INT DINT REAL estructura	inmediato etiqueta etiqueta de matriz	Datos de esta rutina a copiar en una etiqueta de la subrutina. <ul style="list-style-type: none"> <li>• Los parámetros de entrada son opcionales</li> <li>• Se introduce un máximo de 40 parámetro de entrada si es necesario</li> </ul>
Return Par	BOOL SINT INT DINT REAL estructura	etiqueta etiqueta de matriz	Etiqueta de esta rutina para copiar el resultado de la subrutina. <ul style="list-style-type: none"> <li>• Los parámetros de retorno son opcionales</li> <li>• Se introduce un máximo de 40 parámetros de retorno si es necesario</li> </ul>
<b>Instrucción SBR:</b>			

Input Par	BOOL SINT INT DINT REAL estructura	etiqueta etiqueta de matriz	Etiqueta de esta rutina en la que copiar al parámetro de entrada correspondiente (máximo 40) en la instrucción JSR.
<b>Instrucción RET:</b>			
Return Par	BOOL SINT INT DINT REAL estructura	inmediato etiqueta etiqueta de matriz	Datos de esta rutina a copiar al parámetro de retorno correspondiente (máximo 40) en la instrucción JSR.

### Afectar a las marcas de estado matemático

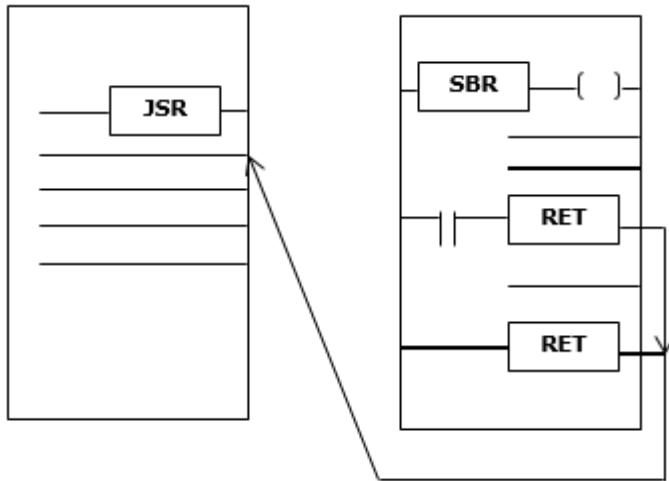
No

### Fallos mayores/menores

Se producirá un fallo mayor si:	Tipo de fallo	Código de fallo
La instrucción JSR tiene menos parámetros de entrada que la instrucción SBR.	4	31
La instrucción JSR pasa a una rutina de fallo.	4 o suministrado por el usuario	990 o suministrado por el usuario
La instrucción RET tiene menos parámetros de retorno que la instrucción JSR.	4	31
La rutina principal contiene una instrucción RET.	4	31

## Descripción

**Importante:** Cualquier rutina podría contener una instrucción JSR pero una instrucción JSR no puede llamar (ejecutar) a la rutina principal.



La instrucción JSR inicia la ejecución de la rutina especificada, a la que se hace referencia como una subrutina:

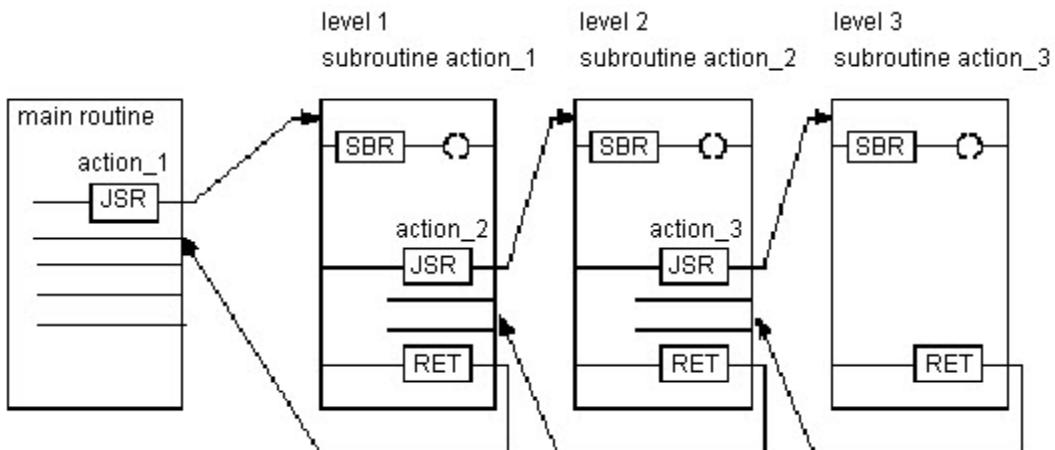
- La subrutina se ejecuta cada vez que se escanea.
- Después de que se ejecute la subrutina, la ejecución de la lógica vuelve a la rutina que contiene la instrucción JSR y continúa con la instrucción que sigue a la JSR.

Para programar un salto a una subrutina, siga estas pautas.

Instrucción	Pautas
JSR	<ul style="list-style-type: none"> <li>• Para copiar datos en una etiqueta en la subrutina, se introduce un parámetro de entrada.</li> <li>• Para copiar un resultado de la subrutina en una etiqueta de esta rutina, se introduce un parámetro de retorno.</li> <li>• Introduce hasta 40 entradas y hasta 40 parámetros de retorno, según sea necesario.</li> </ul>
SBR	<ul style="list-style-type: none"> <li>• Si la instrucción JSR tiene un parámetro de entrada, se introduce una instrucción SBR.</li> <li>• Coloca la instrucción SBR como la primera instrucción de la rutina.</li> <li>• Para cada parámetro de entrada de la instrucción JSR, se introduce la etiqueta en la que quiere copiar los datos.</li> </ul>

Instrucción	Pautas
RET	<ul style="list-style-type: none"> <li>Si la instrucción JSR tiene un parámetro de retorno, se introduce una instrucción RET.</li> <li>Coloca la instrucción RET como la última instrucción de la rutina.</li> <li>Por cada parámetro de retorno de la instrucción JSR, se introduce un parámetro de retorno para enviarlo a la instrucción JSR.</li> <li>En una rutina de escalera, coloca instrucciones RET adicionales para salir de la subrutina basada en diferentes condiciones de entrada, si es necesario (las rutinas del bloque de funciones solo permiten una instrucción RET).</li> </ul>

Invoca hasta 25 subrutinas anidadas con un máximo de 40 parámetros pasados a una subrutina y un máximo de 40 parámetros devueltos de una subrutina.



Cons  
ejo: Seleccione el menú Editar > Editar elemento de escalera (Edit > Edit Ladder Element) para agregar y eliminar operandos de variables. Para las instrucciones JSR y SBR, se añade Parámetro de entrada. Para las instrucciones JSR y RET, se añade Parámetro de salida. Para las tres instrucciones, se elimina Parámetro de instrucción.

## Ejecución

### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	<p>El reglón se establece en falso.</p> <p>El controlador ejecuta todas las subrutinas. Para asegurarse de que se pre-escanean todos los reglones de la subrutina, el controlador ignora las instrucciones RET (es decir, que las instrucciones RET no salen de la subrutina).</p> <p>No se pasan los parámetros de entrada ni los de retorno.</p> <p>Si se invoca varias veces la misma subrutina, solo se pre-escaneará una vez.</p>

Condición/estado	Acción realizada
La condición de entrada de reglón es falsa (para la instrucción JSR)	N/A
La condición de entrada de reglón es verdadera	Los parámetros se pasan y se ejecuta la subrutina.
Post-escaneado	Misma acción que el pre-escaneado

### Bloque de funciones

Condición/estado	Acción realizada
Pre-escaneado	Consultar Pre-escaneado en la tabla Diagrama de escalera.
EnableIn es falso	N/A
EnableIn es verdadero	Consultar EnableIn es verdadero en la tabla Diagrama de escalera.
Primera ejecución de instrucción	N/A
Primer escaneado de instrucción	N/A
Post-escaneado	Consultar Post-escaneado en la tabla Diagrama de escalera.

### Texto estructurado

Condición/estado	Acción realizada
Pre-escaneado	Consultar Pre-escaneado en la tabla Diagrama de escalera.
Ejecución normal	Consultar Tag.EnableIn es verdadero en la tabla Diagrama de escalera.
Post-escaneado	Consultar Post-escaneado en la tabla Diagrama de escalera.

## Ejemplos

### Ejemplo 1

#### Diagrama de escalera

When enabled, the JSR instruction passes value\_1 and value\_2 to routine\_1.

<b>JSR</b>
Jump To Subroutine
Routine Name <b>subroutine_1</b>
Input Par <b>value_1</b>
Input Par <b>value_2</b>
Return Par <b>float_value_1</b>

<b>SBR</b>
Subroutine
Input Par <b>value_a</b>
Input Par <b>value_b</b>

The SBR instruction receives value\_1 and value\_2 from the JSR instruction and copies those values to value\_a and value\_b, respectively. Logic execution continues in this routine.

*[other rungs of code]*

When enabled, the RET instruction sends float\_a to the JSR instruction. The JSR instruction receives float\_a and copies the value to float\_value\_1. Logic execution continues with the next instruction following the JSR instruction.

<b>RET</b>
Return from Subroutine
Return Par <b>float_a</b>

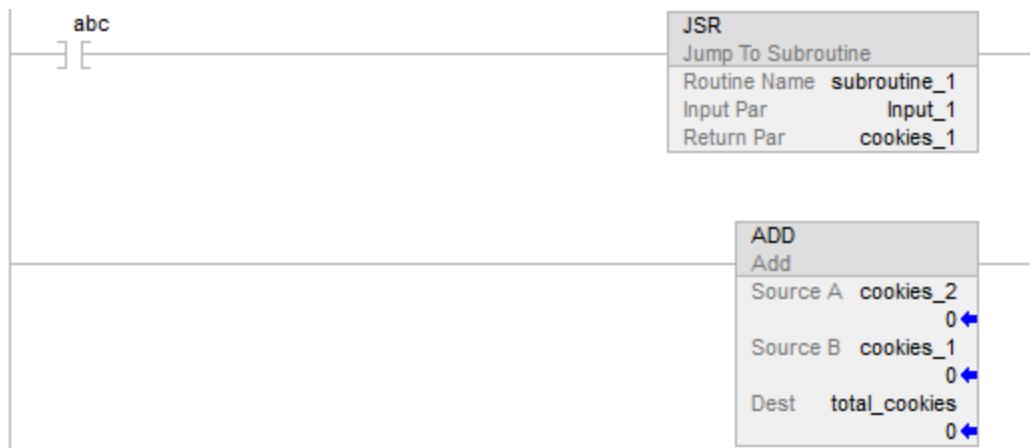
#### Texto estructurado

Rutina	Programa
Rutina principal	JSR(routine_1,2,value_1,value_2,float_value_1);
Subrutina	SBR(value_a,value_b); <statements>; RET(float_a);

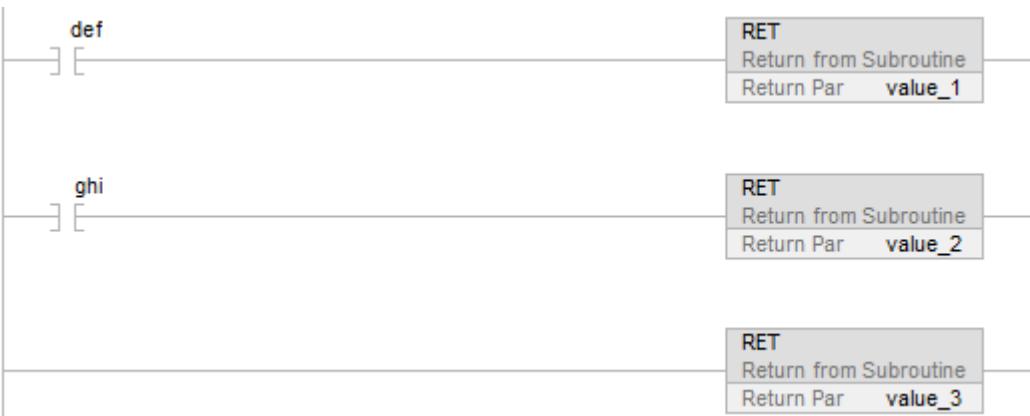
## Ejemplo 2

### Diagrama de escalera

#### Rutina principal



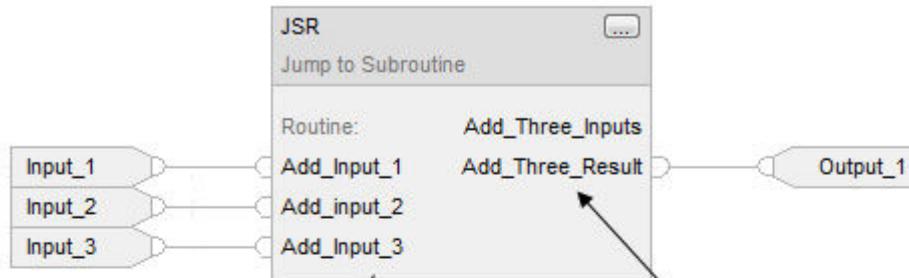
#### subroutine\_1



### Ejemplo 3

#### Bloque de funciones

JSR Instruction in Routine\_A



1. The values in Add\_Input\_1, Add\_Input\_2, and Add\_Input\_3 are copied to Input\_A, Input\_B, and Input\_C, respectively.

3. The value of Sum\_A\_B\_C is copied to Add\_Three\_Result.

Function Blocks of the Add\_Three\_Inputs Routine



2. The ADD instructions add Input\_A, Input\_B, and Input\_C and place the result in Sum\_A\_B\_C.

#### Consulte también

[Instrucciones de control de programa](#) en la página 660

[Índice a través de matrices](#) en la página 936

### Restablecimiento de control maestro (MCR)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix

5580 y GuardLogix 5580.

La instrucción MCR simula un relé de control maestro (un relé cableado obligatorio que se puede desactivar mediante cualquier interruptor de parada de emergencia conectado en serie). Cada vez que el relé se desactiva, sus contactos se abren para desactivar todos los dispositivos de E/S de la aplicación. La instrucción MCR se puede deshabilitar de forma selectiva una sección de reglones.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

#### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

#### Operандos

#### Descripción

La instrucción MCR es capaz de anular el comportamiento normal de los reglones, forzando la ejecución de cada instrucción como si la condición de entrada de reglón es falsa. Por lo general, la ejecución falsa de una instrucción es más rápida que cuando es verdadera, de modo que se deshabiliten de forma selectiva determinadas secciones de código innecesarias podría dar lugar a una mejora generalizada del tiempo de escaneado.

Cada vez que se ejecuta la instrucción MCR cuando la condición de entrada de reglón es falsa, se alterna el comportamiento de anulación. Por consiguiente, generalmente se requieren dos instrucciones MCR: una para comenzar la “zona” y otra para terminarla.

La MCR de inicio está condicionada generalmente por una o varias instrucciones de entrada. Cuando las condiciones de entrada son falsas, la zona se deshabilitará. Cuando las condiciones de entrada son verdaderas, la zona operará con normalidad.

Por lo general, la instrucción MCR de finalización es incondicional. Si la zona está habilitada, la MCR de finalización será verdadera, de modo que no hará nada. Sin

embargo, si la zona está deshabilitada, la MCR de finalización será falsa, de modo que alternará la anulación, volviendo a habilitar los reglones posteriores.

Cuando programe una zona MCR, tenga en cuenta lo siguiente:

La instrucción MCR debe ser la última instrucción de un reglón.

- Debería terminar la zona con una instrucción MCR incondicional. Si la MCR de finalización es falsa y la zona está habilitada, la MCR de finalización deshabilitará todos los reglones posteriores.
- No puede anidar una zona MCR dentro de otra. Solo hay un bit de anulación en cada programa. Cada instrucción MCR tiene la capacidad de alternar esta anulación. Si se intenta anidar zonas MCR, se crearán realmente varias zonas menores.
- No se salta a una zona MCR. Si no se ejecuta la MCR de inicio, no se deshabilitará la zona.
- El bit de anulación se restablece automáticamente al final de la rutina. Si una zona MCR sigue hasta el final de la rutina no tiene que programar una instrucción MCR para finalizar la zona. No obstante, para evitar confusiones a la hora de editar en línea, se recomienda que se utilice siempre la MCR de finalización.

Si la MCR se deshabilita en una subrutina o AOI, el bit de anulación se restablecerá cuando devuelva la subrutina/AOI.

Las AOI tienen su propio bit de anulación, que se inicializa al invocar la AOI. Si se invoca una AOI desde dentro de una zona MCR deshabilitada, la rutina de modo de escaneado falso se ejecutará con normalidad. Una vez que la AOI vuelve, se restaurará el estado de la zona al valor que tenía antes de que se invocase la AOI.

---

**Importante:** La instrucción MCR no puede reemplazar a un relé de control maestro cableado que permite realizar paradas de emergencia. Sigue siendo necesario que instale un relé de control maestro cableado para permitir el apagado de E/S de emergencia.

---

**Importante:** No solape ni anide zonas MCR. Cada zona MCR debe estar separada y completa. Si se solapan o anidan, podría producirse un funcionamiento impredecible de la máquina y podrían dañarse los equipos o producirse lesiones físicas.

Coloque las operaciones críticas fuera de la zona MCR. Si inicia instrucciones como temporizadores en una zona MCR, la ejecución de la instrucción pasa a ser falsa cuando la zona está deshabilitada y el temporizador se borrará.

### Afectar a las marcas de estado matemático

No

### Fallos mayores/menores

No es específico para esta instrucción. Consulte los *Atributos comunes* para fallos relacionados con el operando.

### Ejecución

#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	El comportamiento de anulación se alterna habilitando o deshabilitando los reglones siguientes.
La condición de entrada de reglón es verdadera	N/A
Post-escaneado	N/A

### Ejemplo

#### Diagrama de escalera

Cuando se habilita la primera instrucción MCR (input\_1, input\_2 e input\_3 están establecidas), el controlador ejecuta los reglones en la zona MCR (entre las dos instrucciones MCR) y establece o borra las salidas, dependiendo de las condiciones de entrada.

Cuando se deshabilita la primera instrucción MCR (input\_1, input\_2 e input\_3 no están todas establecidas), el controlador ejecuta los reglones en la zona MCR (entre las dos instrucciones MCR) y el valor de EnableIn pasa a falso para todos los reglones de la zona MCR, independientemente de las condiciones de entrada.



### Consulte también

[Instrucciones de control de programa](#) en la página 660

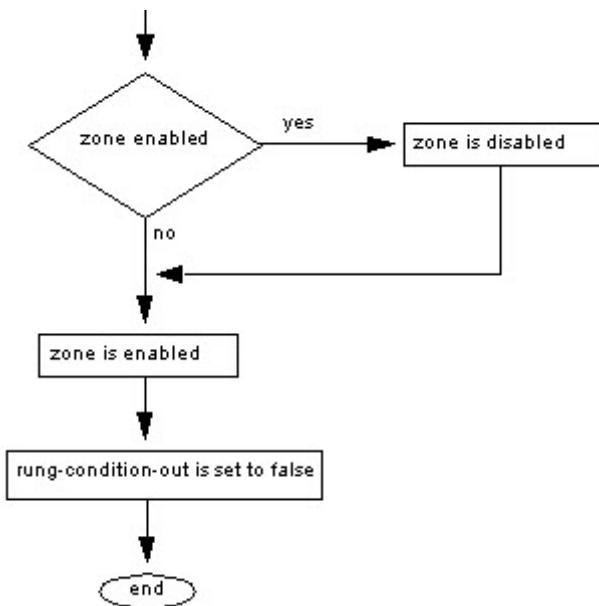
[Instrucción siempre falso \(AFI\)](#) en la página 662

[Sin operación \(NOP\)](#) en la página 686

[Fin temporal \(TND\)](#) en la página 693

[Atributos comunes](#) en la página 923

## Diagrama de flujo de MCR (Falso)



## Sin operación (NOP)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción NOP funciona como un marcador de posición.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

#### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

#### Operandos

#### Diagrama de escalera

Ninguno

## Descripción

Puede colocar la instrucción NOP en cualquier parte de un reglón. Cuando está habilitada, la instrucción NOP no realiza ninguna operación. Cuando está deshabilitada, la instrucción NOP no realiza ninguna operación.

## Afectar a las marcas de estado matemático

No

## Fallos mayores/menores

No es específico para esta instrucción. Consulte los *Atributos comunes* para fallos relacionados con el operando.

## Ejecución

### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	N/A
Post-escaneado	N/A

## Ejemplos

### Diagrama de escalera



## Consulte también

[Instrucciones de control de programa](#) en la página 660

[Instrucción siempre falso \(AFI\)](#) en la página 662

[Restablecimiento de control maestro \(MCR\)](#) en la página 681

[Fin temporal \(TND\)](#) en la página 693

[Atributos comunes](#) en la página 923

## Pausa SFC (SFP)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción SFP pausa una rutina SFC.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

### Texto estructurado

```
SFP(SFCRoutineName,TargetState);
```

### Operandos

#### Diagrama de escalera

Operando	Tipo	Formato	Descripción
SFCRoutineName	ROUTINE	nombre	Rutina SFC a pausar
TargetState	DINT	inmediato	Se selecciona una: <ul style="list-style-type: none"> <li>• En ejecución (o se introduce 0)</li> <li>• Pausada (o se introduce 1)</li> </ul>

### Texto estructurado

Operando	Tipo	Formato	Descripción
SFCRoutineName	ROUTINE	nombre	Rutina SFC a pausar
TargetState	DINT	inmediato	Se selecciona una: <ul style="list-style-type: none"> <li>• En ejecución (o se introduce 0)</li> <li>• Pausada (o se introduce 1)</li> </ul>

Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

### Descripción

La instrucción SFP le permite pausar una rutina SFC que se esté ejecutando.

### Afectar a las marcas de estado matemático

No

### Condiciones de fallo

Se producirá un fallo mayor si:	Tipo de fallo	Código de fallo
El tipo de rutina no es una rutina SFC	4	85

Consulte los *Atributos comunes* para fallos relacionados con el operando.

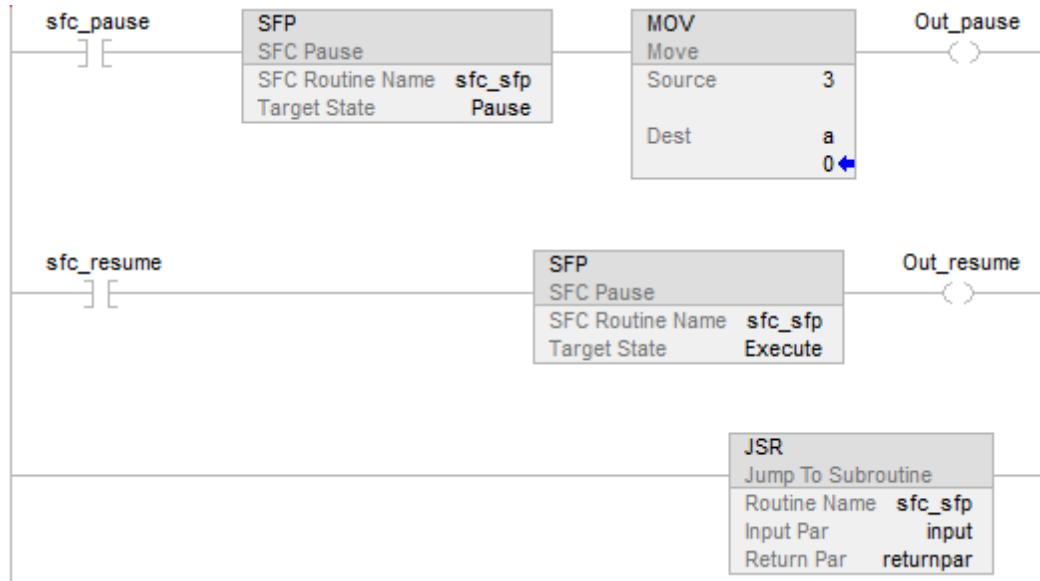
### Ejecución

#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa.	N/A
La condición de entrada de reglón es verdadera	La instrucción pausa o reanuda la ejecución de la rutina SFC especificada.
Post-escaneado	N/A

#### Texto estructurado

Condición/estado	Acción realizada
Pre-escaneado	N/A
Ejecución normal	La instrucción pausa o reanuda la ejecución de la rutina SFC especificada.
Post-escaneado	N/A

**Ejemplo****Diagrama de escalera****Consulte también**

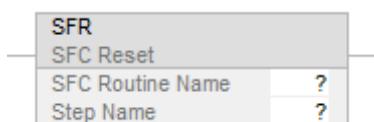
[Atributos comunes](#) en la página 923

[Sintaxis de texto estructurado](#) en la página 955

**Restablecer SFC (SFR)**

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción SFR restablece la ejecución de una rutina SFC en un paso especificado.

**Idiomas disponibles****Diagrama de escalera**

## Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

## Texto estructurado

SFR(SFCRoutineName,StepName);

## Operandos

### Diagrama de escalera

Operando	Tipo	Formato	Descripción
SFCRoutineName	ROUTINE	nombre	Rutina SFC a restablecer
StepName	SFC_STEP	etiqueta	Paso de destino donde reanudar la ejecución

## Texto estructurado

Operando	Tipo	Formato	Descripción
SFCRoutineName	ROUTINE	nombre	Rutina SFC a restablecer
StepName	SFC_STEP	etiqueta	Paso de destino donde reanudar la ejecución

Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

## Descripción

Cuando la instrucción SFR está habilitada:

- En la rutina SFC especificada, todas las acciones almacenadas dejan de ejecutarse (se restablecen).
- La instrucción SFC comienza a ejecutarse en el paso especificado.
- Si el paso de destino es 0, se restablecerá el diagrama a su paso inicial.

La implementación de Logix de la instrucción SFR difiere de la del controlador PLC-5. En el controlador PLC-5, la instrucción SFR se ejecuta cuando la condición de reglón es verdadera. Tras el restablecimiento, la instrucción SFC permanecerá pausada hasta que el reglón que contiene la instrucción SFR pase a ser falso. Esto permitió retardar la ejecución tras un restablecimiento. Esta función de pausa/no pausada de la instrucción SFR de PLC-5 se desacopló de la condición de reglón y se trasladó a la instrucción SFP.

**Afectar a las marcas de estado matemático**

No

**Condiciones de fallo**

Se producirá un fallo mayor si:	Tipo de fallo	Código de fallo
El tipo de rutina no es una rutina SFC	4	85
El paso de destino especificado no existe en la rutina SFC	4	89

Consulte los *Atributos comunes* para fallos relacionados con el operando.

**Ejecución****Diagrama de escalera**

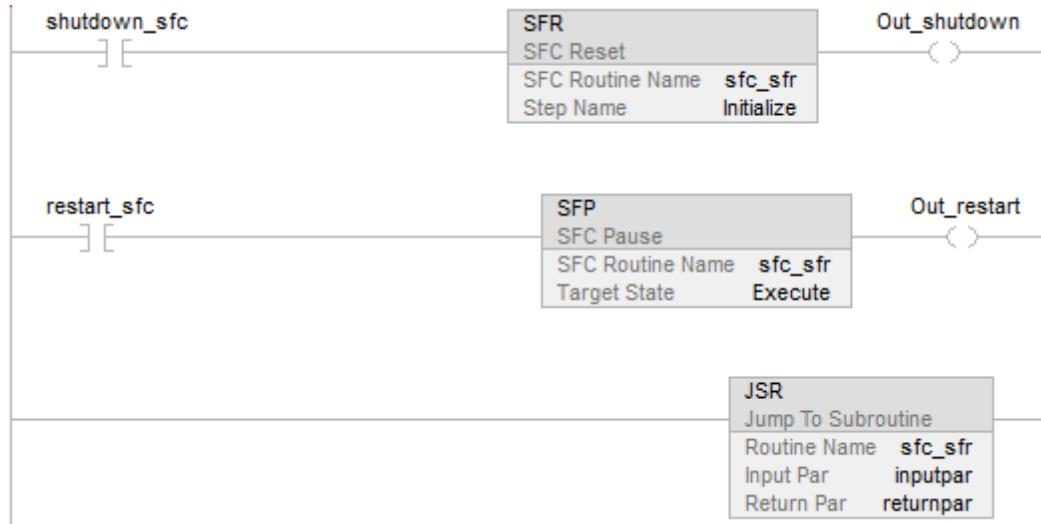
Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	La instrucción restablece la ejecución de la rutina SFC especificada en un paso en concreto.
Post-escaneado	N/A

**Texto estructurado**

Condición/estado	Acción realizada
Pre-escaneado	N/A
Ejecución normal	La instrucción restablece la ejecución de la rutina SFC especificada en un paso en concreto.
Post-escaneado	N/A

## Ejemplo

### Diagrama de escalera



### Consulte también

[Atributos comunes](#) en la página 923

[Sintaxis de texto estructurado](#) en la página 955

## Fin temporal (TND)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción TND termina una rutina de manera condicional.

### Idiomas disponibles

### Diagrama de escalera

—(TND)—

### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

**Texto estructurado**

TND();

**Operandos**

**Diagrama de escalera**

Ninguno

**Texto estructurado**

Ninguno

**Descripción**

Cuando está habilitada, la instrucción TND actúa como el final de la rutina. Si la instrucción TND está en una subrutina, el control regresa a la rutina que llama. Si la instrucción TND está en una rutina principal, el control regresa al siguiente programa dentro de la tarea actual.

**Afectar a las marcas de estado matemático**

No

**Fallos mayores/menores**

No es específico para esta instrucción. Consulte los *Atributos comunes* para fallos relacionados con el operando.

**Ejecución**

**Diagrama de escalera**

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera.	La rutina termina
Post-escaneado	N/A

**Texto estructurado**

Condición/estado	Acción realizada
Pre-escaneado	Consultar Pre-escaneado en la tabla Diagrama de escalera.
Ejecución normal	Consultar La condición de entrada de reglón es verdadera en la tabla Diagrama de escalera.
Post-escaneado	Consultar Post-escaneado en la tabla Diagrama de escalera.

**Texto estructurado**

InputA[:=] OutputB;

IF (InputA) THEN

  TN D();

END\_IF;

InputE [:=] OutputF;

**Consulte también**

[Instrucciones de control de programa](#) en la página 660

[Instrucción siempre falso \(AFI\)](#) en la página 662

[Restablecimiento de control maestro \(MCR\)](#) en la página 681

[Sin operación \(NOP\)](#) en la página 686

[Atributos comunes](#) en la página 923

**Desencadenar tarea de evento (EVENT)**

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción EVENT activa una ejecución de una tarea de evento.

**Idiomas disponibles****Diagrama de escalera****Bloque de funciones**

Esta instrucción no está disponible en bloque de funciones.

**Texto estructurado**

```
EVENT(task_name);
```

**Operandos****Diagrama de escalera**

Operando	Tipo	Formato	Descripción
Task	TASK	nombre	Tarea de evento a ejecutar. Si se especifica una tarea que no es la tarea Evento (Event), dicha tarea no se ejecutará.

**Texto estructurado**

Operando	Tipo	Formato	Descripción
Task	TASK	nombre	Tarea de evento a ejecutar. Si se especifica una tarea que no es la tarea Evento (Event), dicha tarea no se ejecutará.

Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

**Descripción**

Se utiliza la instrucción EVENT para ejecutar mediante programación una tarea de evento.

Cada vez que se ejecute la instrucción, desencadena la tarea de evento especificada.

Asegúrese de que da a la tarea de evento suficiente tiempo para completar su ejecución antes de desencadenarla de nuevo. Si no, se produce un solapamiento.

Si ejecuta una instrucción EVENT cuando se está ejecutando todavía la tarea de evento, el controlador incrementa el contador de solapamiento, pero no desencadena la tarea de evento.

La instrucción EVENT se puede usar para desencadenar Event Task con todos los tipos de desencadenamientos.

### Determinar mediante programación si una instrucción EVENT desencadenó Task

Para determinar si una instrucción EVENT desencadenó una tarea de evento, se utiliza una instrucción Obtener valor del sistema (GSV) para supervisar el atributo Status de la tarea.

Atributo	Tipo de datos	Instrucción	Descripción	
Status	DINT	GSV SSV	Proporciona información de estado sobre la tarea. Una vez que el controlador establece un bit, debe borrar el bit manualmente para determinar si se produjo otro fallo de ese tipo.	
			<b>Para determinar si</b>	<b>Examina este bit</b>
			Una instrucción EVENT desencadenó la tarea (solo tarea de evento)	0
			Un tiempo de espera agotado desencadenó la tarea (solo tarea de evento)	1
			Se produjo un solapamiento para esta tarea	2

El controlador no borra los bits del atributo Status una vez que están establecidos. Para usar un bit para la nueva información de estado, debe borrar el bit manualmente. Utiliza una instrucción Establecer valor del sistema (SSV) para cambiar el valor del atributo.

### Afectar a las marcas de estado matemático

No

### Condiciones de fallo

No es específico para esta instrucción. Consulte los *Atributos comunes* para fallos relacionados con el operando.

## Ejecución

### Diagrama de escalera

Condición	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	La instrucción se ejecuta.
Post-escaneado	N/A

### Texto estructurado

Condición	Acción realizada
Pre-escaneado	N/A
Ejecución normal	La instrucción se ejecuta.
Post-escaneado	N/A

## Ejemplos

### Ejemplo 1

Un controlador usa varios programas, pero un procedimiento de apagado común. Cada programa usa una etiqueta de alcance de programa denominada Shut\_Down\_Line que se enciende si el programa detecta una condición que requiera un apagado. La lógica de cada programa se ejecuta de la siguiente manera.

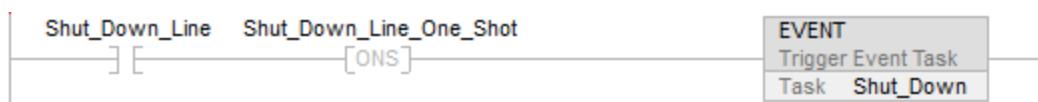
Si Shut\_Down\_Line = activado (las condiciones requieren un apagado), entonces Ejecutar la tarea Shut\_Down una vez

### Diagrama de escalera

#### Programa A



#### Programa B



## Texto estructurado

### Programa A

```
IF Shut_Down_Line AND NOT Shut_Down_Line_One_Shot THEN  
    EVENT (Shut_Down);  
END_IF;
```

```
Shut_Down_Line_One_Shot:=Shut_Down_Line;
```

### Programa B

```
IF Shut_Down_Line AND NOT Shut_Down_Line_One_Shot THEN  
    EVENT (Shut_Down);  
END_IF;
```

```
Shut_Down_Line_One_Shot:=Shut_Down_Line;
```

### Ejemplo 2

El siguiente ejemplo usa una instrucción EVENT para inicializar una tarea de evento. Otro tipo de evento desencadena normalmente la tarea de evento.

### Tarea continua

```
IF Initialize_Task_1 = 1 THEN
```

La instrucción ONS limita la ejecución de la instrucción EVENT a 1 escaneado.

La instrucción EVENT desencadena una ejecución de Task\_1 (tarea de evento).



**Task\_1 (tarea de evento)**

La instrucción GSV establece Task\_Status (etiqueta DINT) = atributo Status para la tarea de evento. En el atributo Instance Name, ESTO significa que el objeto TASK para la tarea en la que está la instrucción (p. ej., Task\_1).



Si Task\_Status.0=1, entonces una instrucción EVENT desencadenó la tarea de evento (esto es, cuando la tarea continua ejecuta su instrucción EVENT para inicializar la tarea de evento).

La instrucción RES restablece un contador que usa la tarea de evento.



El controlador no borra los bits del atributo Status una vez que están establecidos. Para usar un bit para la nueva información de estado, debe borrar el bit manualmente.

Si Task\_Status.0 = 1, borrar ese bit.

La instrucción OTU establece Task\_Status.0 = 0.

La instrucción SSV establece el atributo Status de ESTA tarea (Task\_1) = Task\_Status. Esto incluye el bit borrado.

**Consulte también**

[Atributos comunes](#) en la página 923

[Sintaxis de texto estructurado](#) en la página 955

## Inhabilitación de interrupción de usuario (UID) / Habilitación de interrupción de usuario (UIE)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

Las instrucciones UID y UIE operan conjuntamente para evitar que otras tareas interrumpan un pequeño número de reglones críticos.

### Idiomas disponibles

### Diagramas de escalera



### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

### Texto estructurado

UID();

UIE();

### Operandos

### Diagrama de escalera

Esta instrucción no está disponible en el diagrama de escalera.

### Texto estructurado

Esta instrucción no está disponible en texto estructurado. Debe introducir los paréntesis () después del mnemónico de instrucción, aunque no hay ningún operando.

### Descripción

Cuando la condición de entrada de reglón es verdadera:

- La instrucción UID impide que tareas de mayor prioridad interrumpan la tarea actual, pero no deshabilita la ejecución de una rutina de fallo o del Administrador de fallos del controlador.

- La instrucción UIE permite que otras tareas interrumpan la tarea actual.

Para impedir que se interrumpan una serie de reglones:

1. Limite el número de reglones que no quiere que se interrumpan lo menos posible. Deshabilitar las interrupciones durante un período de tiempo prolongado puede dar lugar a una pérdida de comunicación.
2. Encima del primer reglón que no quiere que se interrumpa, introduzca un reglón y una instrucción UID.
3. Después del último reglón de la serie que no quiere que se interrumpa, introduzca un reglón y una instrucción UIE.
4. Si fuese necesario, puede anidar pares de instrucciones UID/UIE.

Cuando se llama a la instrucción UID por primera vez, aumenta la prioridad, guarda la antigua prioridad e incrementa un contador de anidamiento. Cada una de las siguientes llamadas aumenta el conteo. La instrucción UIE decrementará el contador de anidamiento. Si el nuevo valor es 0, restablecerá la prioridad guardada.

#### Afectar a las marcas de estado matemático

No.

#### Condiciones de fallo

No es específico para esta instrucción. Consulte los *Atributos comunes* para fallos relacionados con el operando.

#### Ejecución

#### Diagrama de escalera

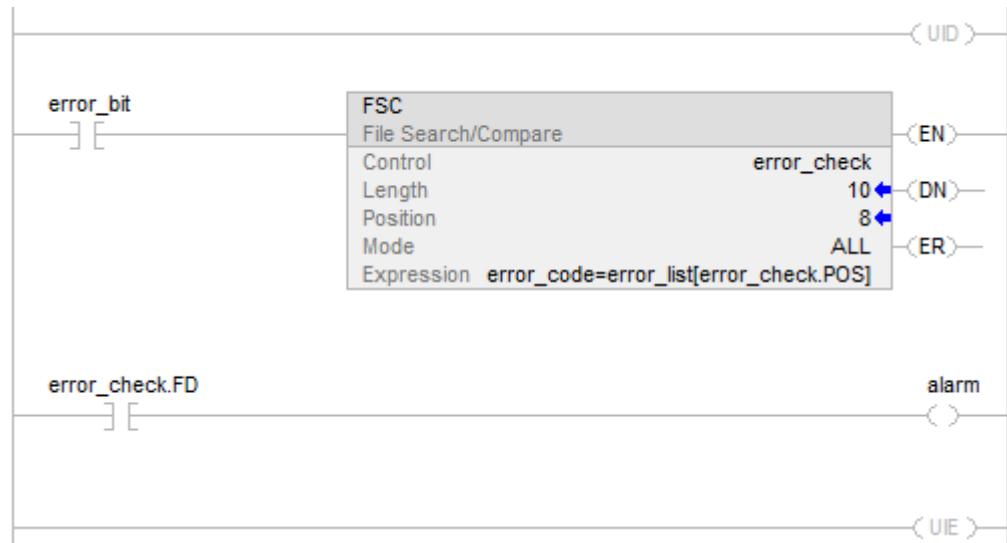
Condición/estado	Acción
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	La instrucción UID impide que se interrumpa la tarea de usuario contenedora. La instrucción UIE permite que se interrumpa la tarea de usuario contenedora, como sucede normalmente en el caso.
Post-escaneado	N/A

### Texto estructurado

Condición/estado	Acción
Pre-escaneado	N/A
Ejecución normal	La instrucción UID impide que se interrumpa la tarea de usuario contenedora. La instrucción UIE permite que se interrumpa la tarea de usuario contenedora, como sucede normalmente en el caso.
Post-escaneado	N/A

### Ejemplo

#### Diagrama de escalera



### Texto estructurado

UID();

<statements>

UIE();

### Consulte también

[Instrucciones de control de programa](#) en la página 660

[Atributos comunes](#) en la página 923

## Instrucción desconocida (UNK)

La instrucción UNK funciona como una indicación de que ha introducido un tipo de instrucción que no está definido dentro del conjunto de instrucciones de Logix Designer.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones

Esta instrucción no está disponible en el bloque de funciones.

#### Texto estructurado

Esta instrucción no está disponible en bloque de funciones.

#### Operandos

#### Diagrama de escalera

Operando	Tipo	Formato	Descripción
Unknown	inmediato	inmediato	

#### Consulte también

[Instrucciones de control de programa](#) en la página 660

# **InSTRUCCIONES PARA/DIVIDIR**

## **InSTRUCCIONES PARA/DIVIDIR**

Use la instrucción FOR para llamar de forma repetida a una subrutina. Use la instrucción BRK para interrumpir la ejecución de una subrutina.

### **InSTRUCCIONES DISPONIBLES**

#### **Diagrama de escalera**



Use la instrucción FOR para llamar de forma repetida a una subrutina. Use la instrucción BRK para interrumpir la ejecución de una subrutina.

<b>Si desea:</b>	<b>Utilice esta instrucción:</b>
Ejecutar una rutina repetidamente.	Instrucción FOR (FOR)
Finalizar la ejecución repetida de una rutina.	Interrupción (BRK)
Volver a la instrucción FOR.	Retorno (RET)

#### **Consulte también**

[InSTRUCCIONES DE CÁLCULO/MATEMÁTICAS](#) en la página 425

[COMPARAR INSTRUCCIONES](#) en la página 369

[INSTRUCCIONES DE BIT](#) en la página 143

[INSTRUCCIONES DE CADENA ASCII](#) en la página 871

[INSTRUCCIONES DE CONVERSIÓN ASCII](#) en la página 892

## **INTERRUPCIÓN (BRK)**

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción BRK interrumpe la ejecución de una rutina que llamó una instrucción FOR.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

#### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

#### Descripción

Cuando está habilitada, la instrucción BRK sale de la rutina y devuelve el control a la rutina que contiene la instrucción FOR que se haya ejecutado más recientemente, reanudando la ejecución tras dicha instrucción. Si ninguna instrucción FOR precedía a esta instrucción BRK en su ejecución durante este escaneado, la instrucción BRK no hace nada.

Si hay instrucciones FOR anidadas, una instrucción BRK devuelve el control a la instrucción FOR situada más adentro.

#### Afectar a las marcas de estado matemático

No

#### Condiciones de fallo

No es específico para esta instrucción. Consulte los *Atributos comunes* para fallos relacionados con el operando.

#### Ejecución

#### Diagrama de escalera

Condición/estado	Acción
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A

La condición de entrada de reglón es verdadera	La instrucción se ejecuta.
Post-escaneado	N/A

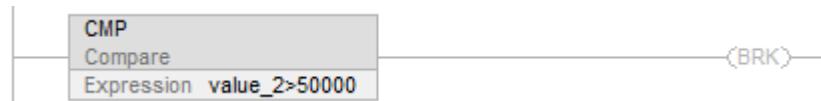
### Ejemplo

Cuando está habilitada, la instrucción BRK deja de ejecutar la rutina actual y regresa a la instrucción que sigue a la llamada de la instrucción FOR.

#### Diagrama de escalera



Esta es routine2:



Consulte también

[Atributos comunes](#) en la página 923

[Instrucciones Para/Dividir](#) en la página 705

[Instrucción FOR \(FOR\)](#) en la página 707

[Saltar a etiqueta \(JMP\) y Etiqueta \(LBL\)](#) en la página 669

[Saltar a subrutina \(JSR\), Subrutina \(SBR\) y Retorno \(RET\)](#) en la página 672

### Instrucción FOR (FOR)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

La instrucción FOR ejecuta una rutina repetidamente.

## Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

### Operandos

### Diagrama de escalera

Operando	Tipo	Formato	Descripción
Routine name	ROUTINE	etiqueta	Subrutina que se invoca cada vez que se ejecuta el lazo FOR.
Index	DINT	etiqueta	Cuenta las veces que se ha ejecutado una rutina
Initial value	SINT INT DINT	inmediato etiqueta	Valor en el que comienza el índice
Terminal value	SINT INT DINT	inmediato etiqueta	Valor en el que se detiene la ejecución de la rutina
Step size	SINT INT DINT	inmediato etiqueta	Cantidad que se añade al índice cada vez que la instrucción FOR ejecuta la rutina

### Descripción

Si se habilita, la instrucción FOR ejecuta la Rutina repetidamente hasta que el valor Index excede el Terminal value. Esta instrucción no pasa parámetros a la rutina.

El valor step puede ser positivo o negativo. Si es negativo, el lazo finaliza cuando el índice es el valor terminal. Si es positivo, el lazo finaliza cuando el índice es mayor que el valor terminal.

Cada vez que la instrucción FOR ejecuta la rutina, añade el Step size al Index.

Procure no hacer demasiados lazos en un solo escaneado. Un número excesivo de repeticiones puede provocar que la vigilancia de control exceda el tiempo de espera, lo que provocaría un fallo mayor.

### Afectar a las marcas de estado matemático

No

### Condiciones de fallo

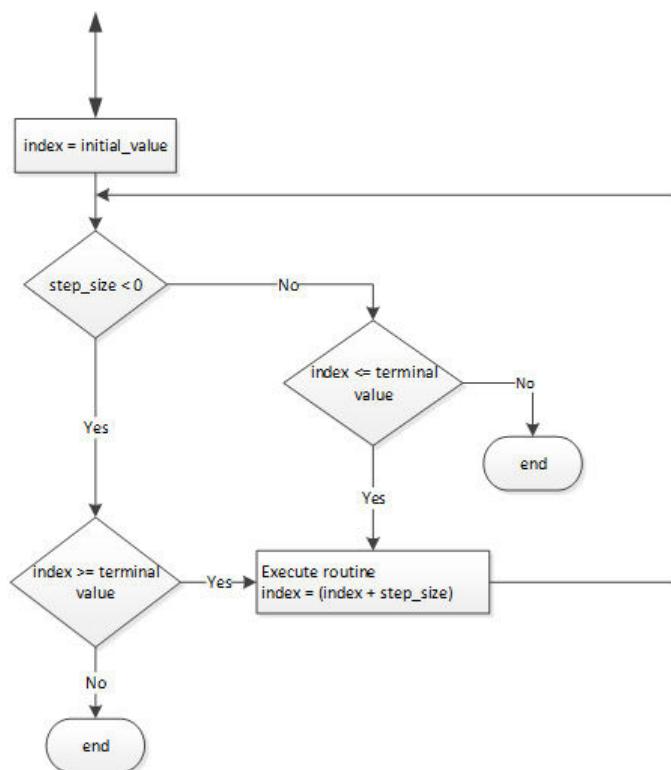
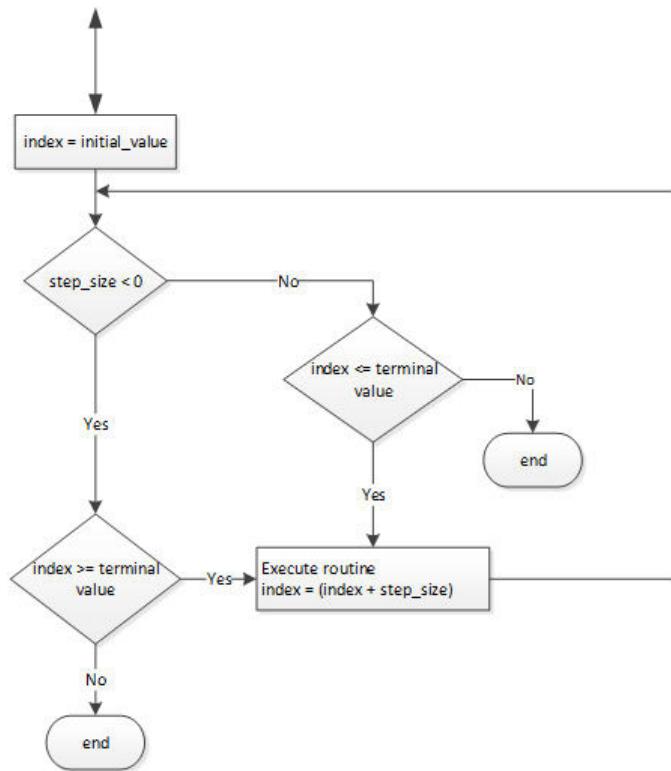
Controladores	Se producirá un fallo mayor si:	Tipo de fallo	Código de fallo
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Límite del nivel de anidamiento > 25	4	94
	la subrutina es un SFC y ya se está ejecutando (llamada recursiva)	4	82
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	N/A	N/A	N/A

Consulte los *Atributos comunes* para fallos relacionados con el operando.

### Ejecución

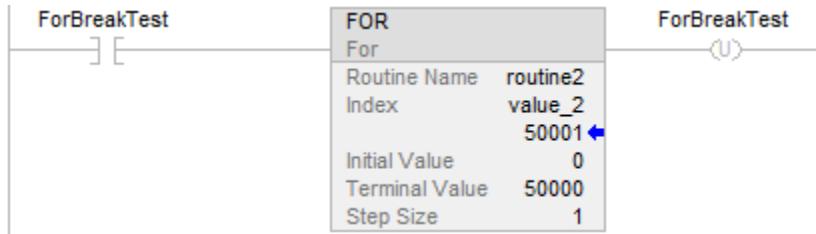
Condición/estado	Acción
Pre-escaneado	La instrucción hará un pre-escaneado de la subrutina nombrada si no ha sido preescaneado antes. <b>Consejo:</b> Si existen instrucciones FOR recursivas o múltiples instrucciones FOR (no recursivas) para la misma subrutina, la subrutina será preescaneada solo una vez. Esto ocurrirá también si la subrutina ha sido preescaneada por JSR.
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	Véase el Diagrama de flujo FOR (Verdadero).
Post-escaneado	La instrucción postescaneará la subrutina nombrada exactamente una vez.

Diagrama de flujo FOR (Verdadero)



## Ejemplos

Si se habilita, la instrucción FOR ejecuta repetidamente routine\_2 e incrementa value\_2 cada vez en uno. Cuando el value\_2 es > 50000 o se habilita una instrucción BRK, la instrucción FOR deja de ejecutar routine\_2.



## Consulte también

[Atributos comunes](#) en la página 923

## Saltar a subrutina (JSR), Subrutina (SBR) y Retorno (RET)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

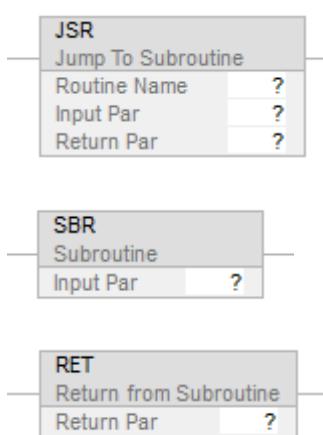
La instrucción JSR invoca otra rutina. Cuando finaliza dicha rutina, la ejecución vuelve a la instrucción JSR.

La instrucción SBR recibe los parámetros de entrada pasados por JSR.

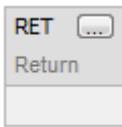
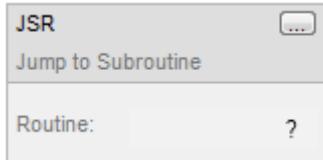
La instrucción RET pasa los parámetros de retorno de vuelva a JSR y termina el escaneado de la subrutina.

### Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones



### Diagrama de funciones secuenciales



### Texto estructurado

```
JSR(RoutineName,InputCount,InputPar,ReturnPar);  
SBR(InputPar);  
RET(ReturnPar);
```

## Operandos

**Importante:** Puede ocurrir un funcionamiento inesperado si:

- Los operandos de etiqueta de salida se sobrescriben.
- Los miembros de un operando de estructura se sobrescriben.
- A no ser que se especifique, los operandos de estructura son compartidos por varias instrucciones.



Para cada parámetro de una instrucción SBR o RET, se utiliza el mismo tipo de datos (incluyendo cualquier dimensión de matriz) como el parámetro correspondiente en la instrucción JSR. Si utiliza tipos de datos diferentes, podrían producirse resultados imprevistos.

## Diagrama de escalera

Operando	Tipo de datos	Formato	Descripción
<b>Instrucción JSR:</b>			
Routine Name	ROUTINE	nombre	Subrutina a ejecutar.
Input Par	BOOL SINT INT DINT REAL estructura	inmediato etiqueta etiqueta de matriz	Datos de esta rutina a copiar en una etiqueta de la subrutina. <ul style="list-style-type: none"> <li>• Los parámetros de entrada son opcionales</li> <li>• Se introduce un máximo de 40 parámetro de entrada si es necesario</li> </ul>
Return Par	BOOL SINT INT DINT REAL estructura	etiqueta etiqueta de matriz	Etiqueta de esta rutina para copiar el resultado de la subrutina. <ul style="list-style-type: none"> <li>• Los parámetros de retorno son opcionales</li> <li>• Se introduce un máximo de 40 parámetros de retorno si es necesario</li> </ul>
<b>Instrucción SBR:</b>			
Input Par	BOOL SINT INT DINT REAL estructura	etiqueta etiqueta de matriz	Etiqueta de esta rutina en la que copiar al parámetro de entrada correspondiente (máximo 40) en la instrucción JSR.
<b>Instrucción RET:</b>			

Return Par	BOOL SINT INT DINT REAL estructura	inmediato etiqueta etiqueta de matriz	Datos de esta rutina a copiar al parámetro de retorno correspondiente (máximo 40) en la instrucción JSR.
------------	---	--	---

**Afectar a las marcas de estado matemático**

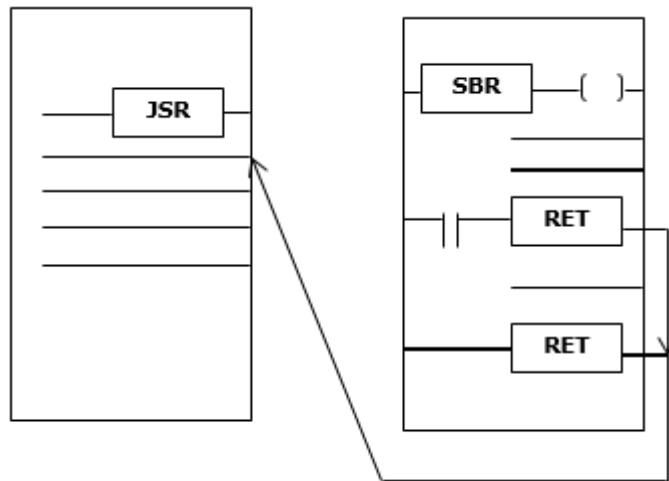
No

**Fallos mayores/menores**

Se producirá un fallo mayor si:	Tipo de fallo	Código de fallo
La instrucción JSR tiene menos parámetros de entrada que la instrucción SBR.	4	31
La instrucción JSR pasa a una rutina de fallo.	4 o suministrado por el usuario	990 o suministrado por el usuario
La instrucción RET tiene menos parámetros de retorno que la instrucción JSR.	4	31
La rutina principal contiene una instrucción RET.	4	31

**Descripción**

**Importante:** Cualquier rutina podría contener una instrucción JSR pero una instrucción JSR no puede llamar (ejecutar) a la rutina principal.



La instrucción JSR inicia la ejecución de la rutina especificada, a la que se hace referencia como una subrutina:

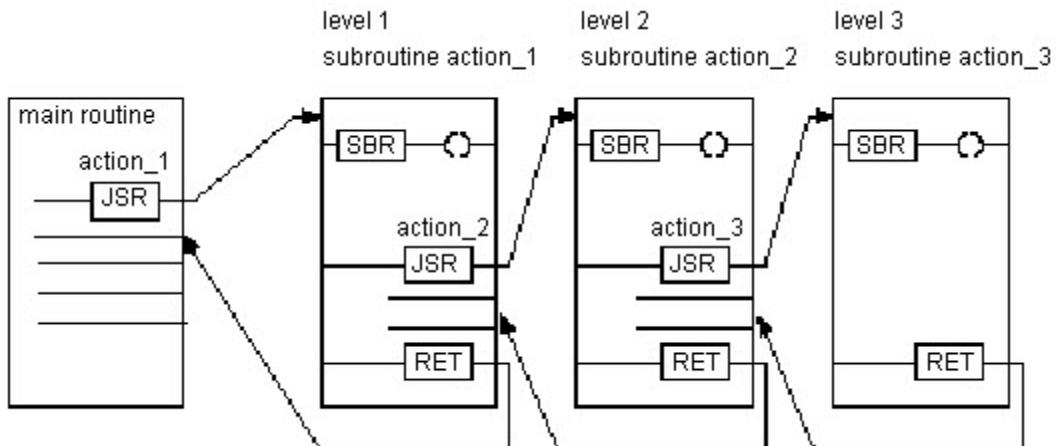
- La subrutina se ejecuta cada vez que se escanea.

- Después de que se ejecute la subrutina, la ejecución de la lógica vuelve a la rutina que contiene la instrucción JSR y continúa con la instrucción que sigue a la JSR.

Para programar un salto a una subrutina, siga estas pautas.

Instrucción	Pautas
JSR	<ul style="list-style-type: none"> <li>Para copiar datos en una etiqueta en la subrutina, se introduce un parámetro de entrada.</li> <li>Para copiar un resultado de la subrutina en una etiqueta de esta rutina, se introduce un parámetro de retorno.</li> <li>Introduce hasta 40 entradas y hasta 40 parámetros de retorno, según sea necesario.</li> </ul>
SBR	<ul style="list-style-type: none"> <li>Si la instrucción JSR tiene un parámetro de entrada, se introduce una instrucción SBR.</li> <li>Coloca la instrucción SBR como la primera instrucción de la rutina.</li> <li>Para cada parámetro de entrada de la instrucción JSR, se introduce la etiqueta en la que quiere copiar los datos.</li> </ul>
RET	<ul style="list-style-type: none"> <li>Si la instrucción JSR tiene un parámetro de retorno, se introduce una instrucción RET.</li> <li>Coloca la instrucción RET como la última instrucción de la rutina.</li> <li>Por cada parámetro de retorno de la instrucción JSR, se introduce un parámetro de retorno para enviarlo a la instrucción JSR.</li> <li>En una rutina de escalera, coloca instrucciones RET adicionales para salir de la subrutina basada en diferentes condiciones de entrada, si es necesario (las rutinas del bloque de funciones solo permiten una instrucción RET).</li> </ul>

Invoca hasta 25 subrutinas anidadas con un máximo de 40 parámetros pasados a una subrutina y un máximo de 40 parámetros devueltos de una subrutina.



Cons  
ejo: Seleccione el menú Editar > Editar elemento de escalera (Edit > Edit Ladder Element) para agregar y eliminar operandos de variables. Para las instrucciones JSR y SBR, se añade Parámetro de entrada. Para las instrucciones JSR y RET, se añade Parámetro de salida. Para las tres instrucciones, se elimina Parámetro de instrucción.

## Ejecución

### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	El reglón se establece en falso. El controlador ejecuta todas las subrutinas. Para asegurarse de que se pre-escaneen todos los reglones de la subrutina, el controlador ignora las instrucciones RET (es decir, que las instrucciones RET no salen de la subrutina). No se pasan los parámetros de entrada ni los de retorno. Si se invoca varias veces la misma subrutina, solo se pre-escaneará una vez.
La condición de entrada de reglón es falsa (para la instrucción JSR)	N/A
La condición de entrada de reglón es verdadera	Los parámetros se pasan y se ejecuta la subrutina.
Post-escaneado	Misma acción que el pre-escaneado

### Bloque de funciones

Condición/estado	Acción realizada
Pre-escaneado	Consultar Pre-escaneado en la tabla Diagrama de escalera.
EnableIn es falso	N/A
EnableIn es verdadero	Consultar EnableIn es verdadero en la tabla Diagrama de escalera.
Primera ejecución de instrucción	N/A
Primer escaneado de instrucción	N/A
Post-escaneado	Consultar Post-escaneado en la tabla Diagrama de escalera.

### Texto estructurado

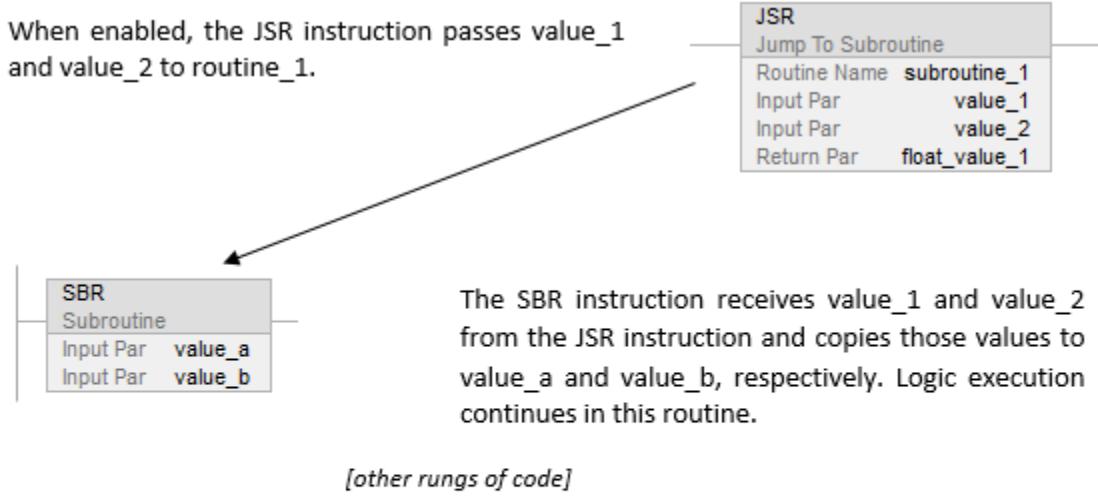
Condición/estado	Acción realizada
Pre-escaneado	Consultar Pre-escaneado en la tabla Diagrama de escalera.
Ejecución normal	Consultar Tag.EnableIn es verdadero en la tabla Diagrama de escalera.

Condición/estado	Acción realizada
Post-escaneado	Consultar Post-escaneado en la tabla Diagrama de escalera.

### Ejemplos

#### Ejemplo 1

##### Diagrama de escalera

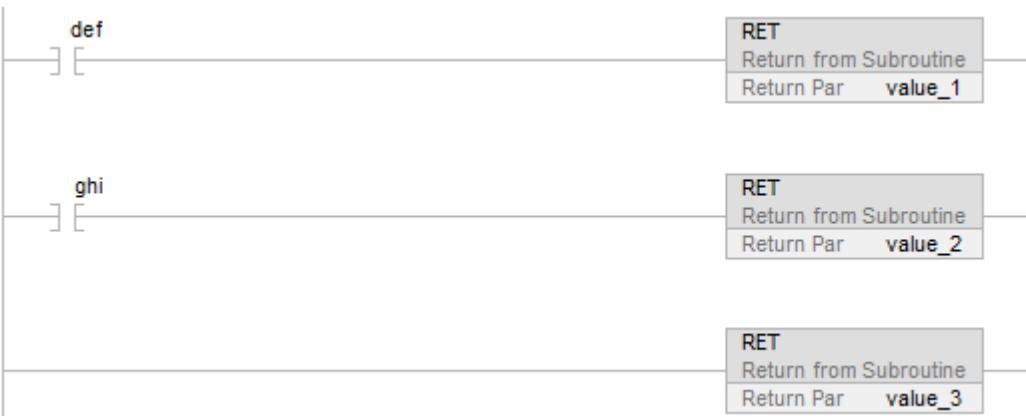


When enabled, the RET instruction sends float\_a to the JSR instruction. The JSR instruction receives float\_a and copies the value to float\_value\_1. Logic execution continues with the next instruction following the JSR instruction.



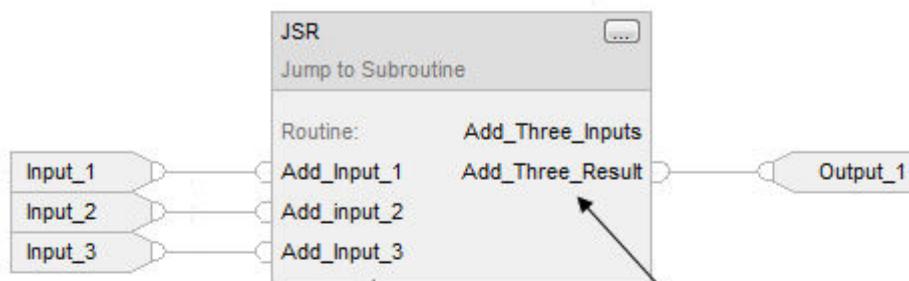
##### Texto estructurado

Rutina	Programa
Rutina principal	JSR(routine_1,2,value_1,value_2,float_value_1);
Subrutina	SBR(value_a,value_b); <statements>; RET(float_a);

**Ejemplo 2****Diagrama de escalera****Rutina principal****subroutine\_1**

**Ejemplo 3****Bloque de funciones**

JSR Instruction in Routine\_A



1. The values in Add\_Input\_1, Add\_Input\_2, and Add\_Input\_3 are copied to Input\_A, Input\_B, and Input\_C, respectively.

3. The value of Sum\_A\_B\_C is copied to Add\_Three\_Result.

Function Blocks of the Add\_Three\_Inputs Routine



2. The ADD instructions add Input\_A, Input\_B, and Input\_C and place the result in Sum\_A\_B\_C.

**Consulte también**

[Instrucciones de control de programa](#) en la página 660

[Índice a través de matrices](#) en la página 936

# **Instrucciones especiales**

## **Instrucciones especiales**

Las instrucciones especiales realizan operaciones específicas de la aplicación.

### **Instrucciones disponibles**

#### **Texto estructurado**

<a href="#">FBC</a>	<a href="#">DDT</a>	<a href="#">DTR</a>	<a href="#">PID</a>
---------------------	---------------------	---------------------	---------------------

#### **Bloque de funciones**

No disponible

#### **Texto estructurado**

No disponible

<b>Si desea:</b>	<b>Utilice esta instrucción:</b>
Comparar los datos con una referencia válida conocida y registrar las incongruencias.	FBC
Comparar los datos con una referencia válida conocida, registrar las incongruencias y actualizar la referencia para que sea igual al origen.	DDT
Pasar los datos de origen por una máscara y comparar los resultados con los datos de referencia y después escribir el origen en la referencia para la próxima comparación.	DTR
Controlar un lazo PID.	PID

#### **Consulte también**

[Usar instrucciones PID](#) en la página 749

[Bloqueo anti-restablecimiento y transferencia sin perturbaciones de manual a automático \(PID\)](#) en la página 753

[Temporización de la instrucción PID](#) en la página 758

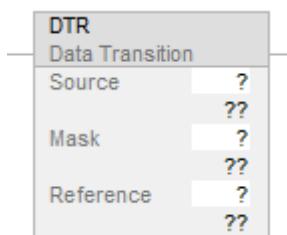
## Transición de datos (DTR)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción DTR pasa el valor de Source por una Mask y compara el resultado con el valor de Reference.

### Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

### Operandos

### Diagrama de escalera

Operando	Tipo	Formato	Descripción
Source	DINT	inmediato etiqueta	Matriz para comparar con la referencia
Mask	DINT	inmediato etiqueta	Qué bits bloquear o dejar pasar
Reference	DINT	etiqueta	Matriz para comparar con el origen

## Descripción

La instrucción DTR pasa el valor de Source por una Mask y compara el resultado con el valor de Reference. La instrucción DTR también escribe el valor de Source enmascarado en el valor de Reference para la próxima comparación. Source permanece sin cambios.

Un "1" en la máscara significa que el bit de datos se ha pasado. Un "0" en la máscara significa que el bit de datos se ha bloqueado.

Cuando está habilitada, Mask pasa los datos cuando están establecidos los bits de Mask, la Máscara bloquea los datos cuando se han borrado los bits de Mask.

Cuando Source enmascarado es diferente de Reference, EnableOut se hace verdadero en un escaneado. Cuando Source enmascarado es el mismo que Reference, EnableOut es falso.

---

**Importante:** Puede ser peligroso programar en línea con esta instrucción. Si el valor de referencia es distinto del valor de Source, EnableOut se hace verdadero. Tenga cuidado si introduce esta instrucción cuando el procesador esté en modo Marcha o en modo Marcha remota.

---

## Introducción de un valor de máscara inmediato

Cuando se introduce una máscara, el software de programación pasa de forma predeterminada a usar valores decimales. Si quiere introducir una máscara usando otro formato, coloque el prefijo adecuado antes del valor.

Prefijo	Descripción
16#	hexadecimal (por ejemplo, 16#0F0F)
8#	octal (por ejemplo, 8#16)
2#	binario (por ejemplo, 2#00110011)

## Afectar a las marcas de estado matemático

No

## Condiciones de fallo

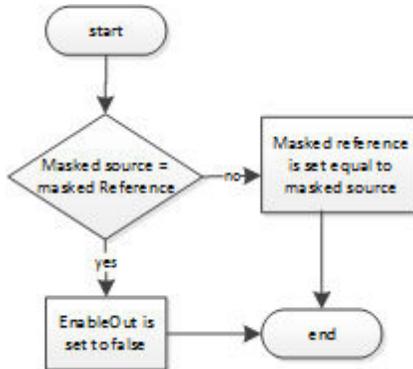
No es específico para esta instrucción. Consulte los Atributos comunes para fallos relacionados con el operando.

## Ejecución

### Diagrama de escalera

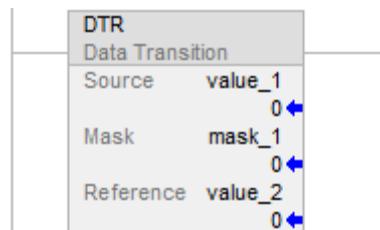
Condición	Acción
Pre-escaneado	Reference = Source AND Mask.
La condición de entrada de reglón es falsa	Reference = Source AND Mask.
La condición de entrada de reglón es verdadera	Consultar el Diagrama de flujo de DTR (verdadero)
Post-escaneado	N/A

### Diagrama de flujo DTR (verdadero)

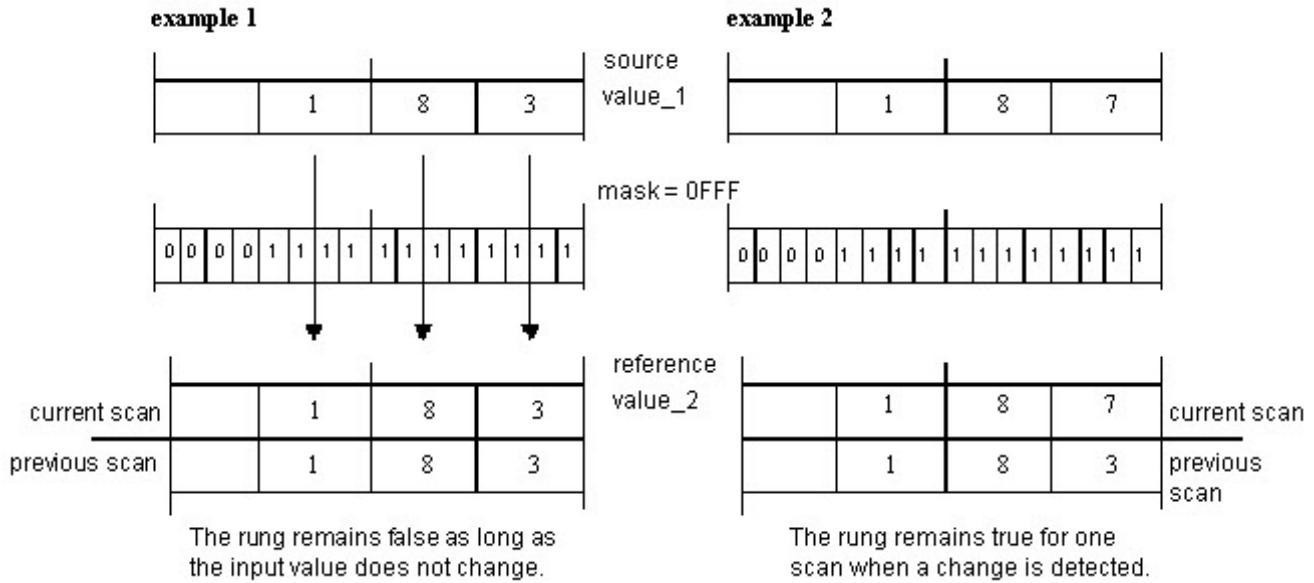


## Ejemplo

### Diagrama de escalera



Si está habilitada, la instrucción DTR enmascara value\_1. Si los dos valores enmascarados son diferentes, EnableOut se establece en verdadero.



En el ejemplo 1, EnableOut se establecerá siempre en falso porque el valor de referencia es igual a sourcevalue\_1 AND máscara. En el ejemplo 2, por alguna razón se ha cambiado el valor de origen, lo que hace que reference\_value no sea igual a source\_value AND máscara. En este caso, EnableOut se establecerá en VERDADERO y referencevalue se actualizará según sourceValue and máscara. Este es el motivo por el que en el escaneado anterior el valor de referencia es 183 pero en el actual el valor es 187. El reglón sigue siendo verdadero solo en un escaneado cuando se detecta un cambio porque en el siguiente escaneado, y siempre que no se modifique el origen, el reglón seguirá en falso porque el valor de referencia volverá a ser igual al valor de origen AND a la máscara.

## Consulte también

Instrucciones especiales en la página 721

FBC en la página 734

DDT en la página 725

Atributos comunes en la página 923

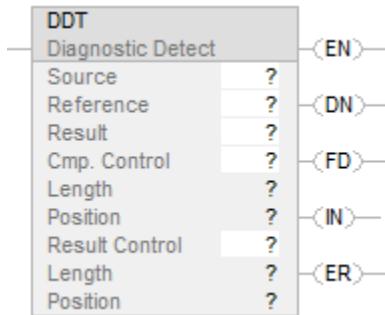
## Detección de diagnóstico (DDT)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción DDT compara los bits de una matriz Source con los de una matriz Reference para encontrar un bit no coincidente. La ubicación del bit no coincidente se registra a continuación y se cambia el Reference bit no coincidente por el valor del Source bit.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

#### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

#### Operандos

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción. Consulte Conversión de datos.

#### Diagrama de escalera

Operando	Tipo	Formato	Descripción
Source	DINT	etiqueta de matriz	Matriz para comparar con la referencia no se utiliza CONTROL.POS en el subíndice
Reference	DINT	etiqueta de matriz	Matriz para comparar con el origen no se utiliza CONTROL.POS en el subíndice
Result	DINT	etiqueta de matriz	Matriz para almacenar los resultados no se utiliza CONTROL.POS en el subíndice
Cmp. Control	CONTROL	estructura	Estructura de control para la comparación.
Length	DINT	inmediato	Número de bits a comparar

Position	DINT	inmediato	Posición actual en el origen el valor inicial suele ser 0
Result control	CONTRO L	estructura	Estructura de control para los resultados.
Length	DINT	inmediato	Número de ubicaciones de almacenamiento en el resultado
Position	DINT	inmediato	Posición actual en el resultado. el valor inicial suele ser 0

---

**Importante:** Se utiliza diferentes etiquetas para la estructura de control de comparación y la estructura de control de resultado. Si utiliza la misma etiqueta para ambos, podría producirse una operación impredecible, que podría dar lugar a daños en el equipamiento y/o lesiones físicas del personal.

---

### Estructura de COMPARE

Mnemónico	Tipo de datos	Descripción
.EN	BOOL	El bit de habilitación indica que la instrucción DDT está habilitada.
.DN	BOOL	El bit de efectuado se establece cuando la instrucción DDT compara el último bit en las matrices Source y Reference.
.FD	BOOL	El bit de encontrado se establece cada vez que la instrucción DDT registra una discrepancia (operación de una en una) o después de registrar todas las incongruencias (operación todas en cada escaneado).
.IN	BOOL	El bit de inhibición indica el modo de búsqueda de DDT. 0 = Modo Todos 1 = modo de discrepancias de una en una
.ER	BOOL	El bit de error se establece si los valores de POS o LEN no son válidos.
.LEN	DINT	El valor de longitud identifica el número de bits a comparar.
.POS	DINT	El valor de posición identifica el bit actual.

### Estructura de RESULT

Mnemónico	Tipo de datos	Descripción
.DN	BOOL	El bit de efectuado se establece cuando la matriz Result está llena.
.LEN	DINT	El valor de longitud identifica el número de ubicaciones de almacenamiento en la matriz Result.
.POS	DINT	El valor de posición identifica la posición actual en la matriz Result.

## Descripción

Si está habilitada , la instrucción DDT compara los bits de la matriz Source con los de la matriz Reference, registra el número de bit de cada discrepancia en la matriz Result y cambia el valor del bit de Reference para que coincida con el valor del bit de Source correspondiente.

---

**Importante:** La instrucción DDT opera en posiciones continuas de memoria. Comprueba y confirma que la instrucción no modifica datos que usted no desea que sean modificados.

---

Las instrucciones DDT y FBC se diferencian en que cada vez que la instrucción DDT encuentra una discrepancia, la instrucción DDT modifica el bit de referencia para que coincida con el bit de origen. La instrucción FBC no cambia el bit de referencia.

Si la instrucción intenta leer más allá del final de una matriz, la instrucción establece el bit .ER y genera un fallo mayor.

### Seleccionar el modo de búsqueda

Si quiere detectar:	Seleccione este modo:
Discrepancias de una en una	<p>Se establece el bit .IN en la estructura de CONTROL de comparación.</p> <p>Cada vez que el valor de EnableIn pasa de falso a verdadero, la instrucción DDT busca la siguiente discrepancia entre las matrices Source y Reference. Tras encontrar una discrepancia, la instrucción se detiene, establece el bit .FD y registra la posición de la discrepancia.</p>
Todas las discrepancias	<p>Se borra el valor del bit .IN en la estructura de CONTROL de comparación.</p> <p>Cada vez que el valor de EnableIn pasa de falso a verdadero, la instrucción DDT busca todas las discrepancias entre las matrices Source y Reference.</p>

### Afectar a las marcas de estado matemático

No

### Condiciones de fallo

Se producirá un fallo mayor si:	Tipo de fallo	Código de fallo
result.POS > tamaño de la matriz Result	4	20

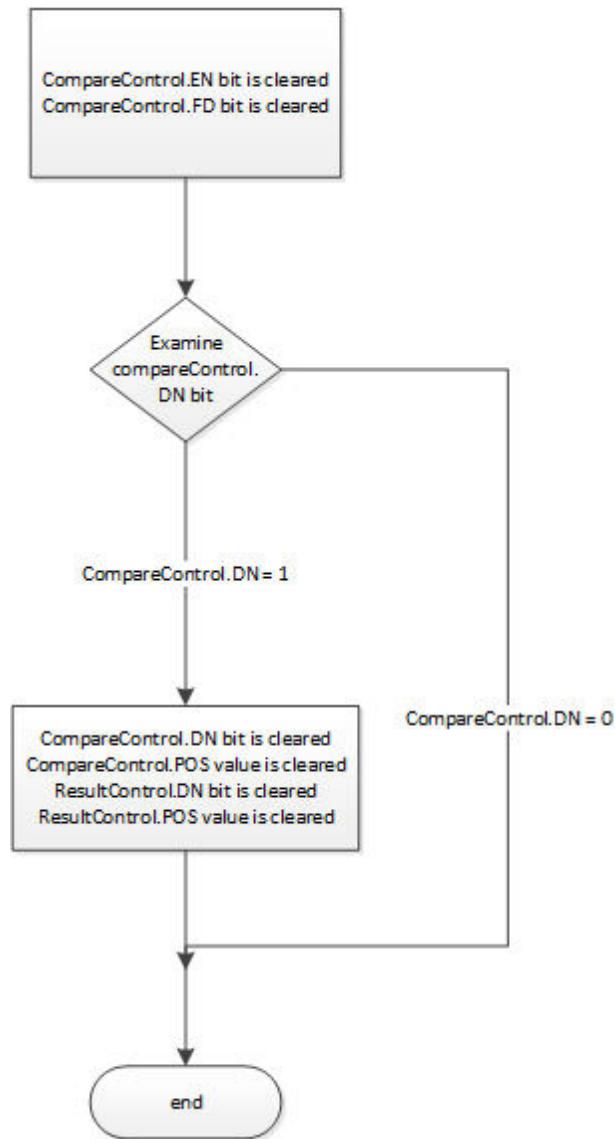
Consulte los Atributos comunes para los fallos relacionados con los operandos.

## Ejecución

### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	Consulte el Diagrama de flujo de DDT (Pre-escaneado)
La condición de entrada de reglón es falsa	Consulte el Diagrama de flujo de DDT (Falso)
La condición de entrada de reglón es verdadera	Consulte el Diagrama de flujo de DDT (Verdadero)
Post-escaneado	N/A

Diagrama de flujo de DDT (Pre-escaneado)



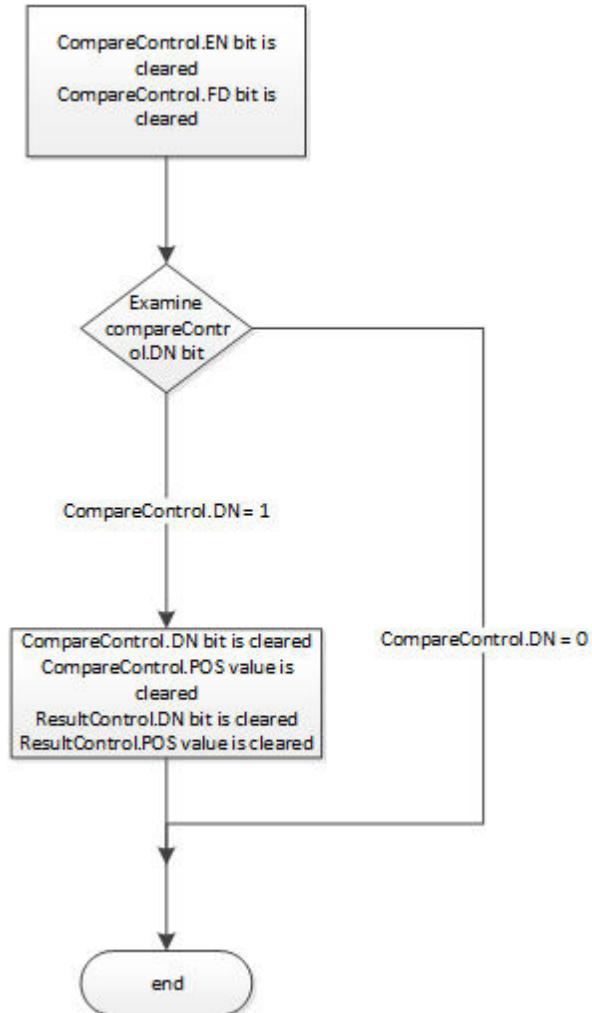
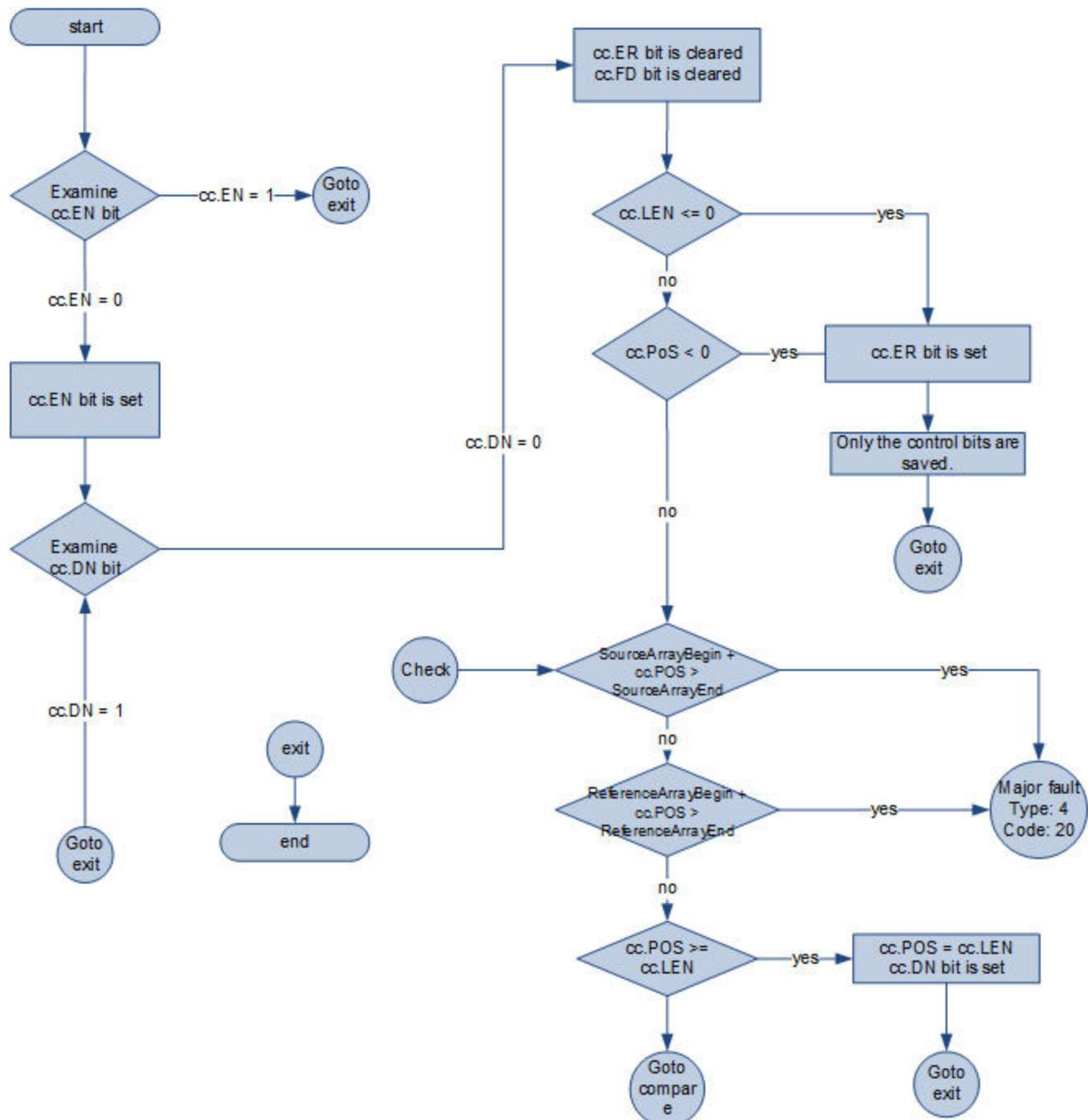
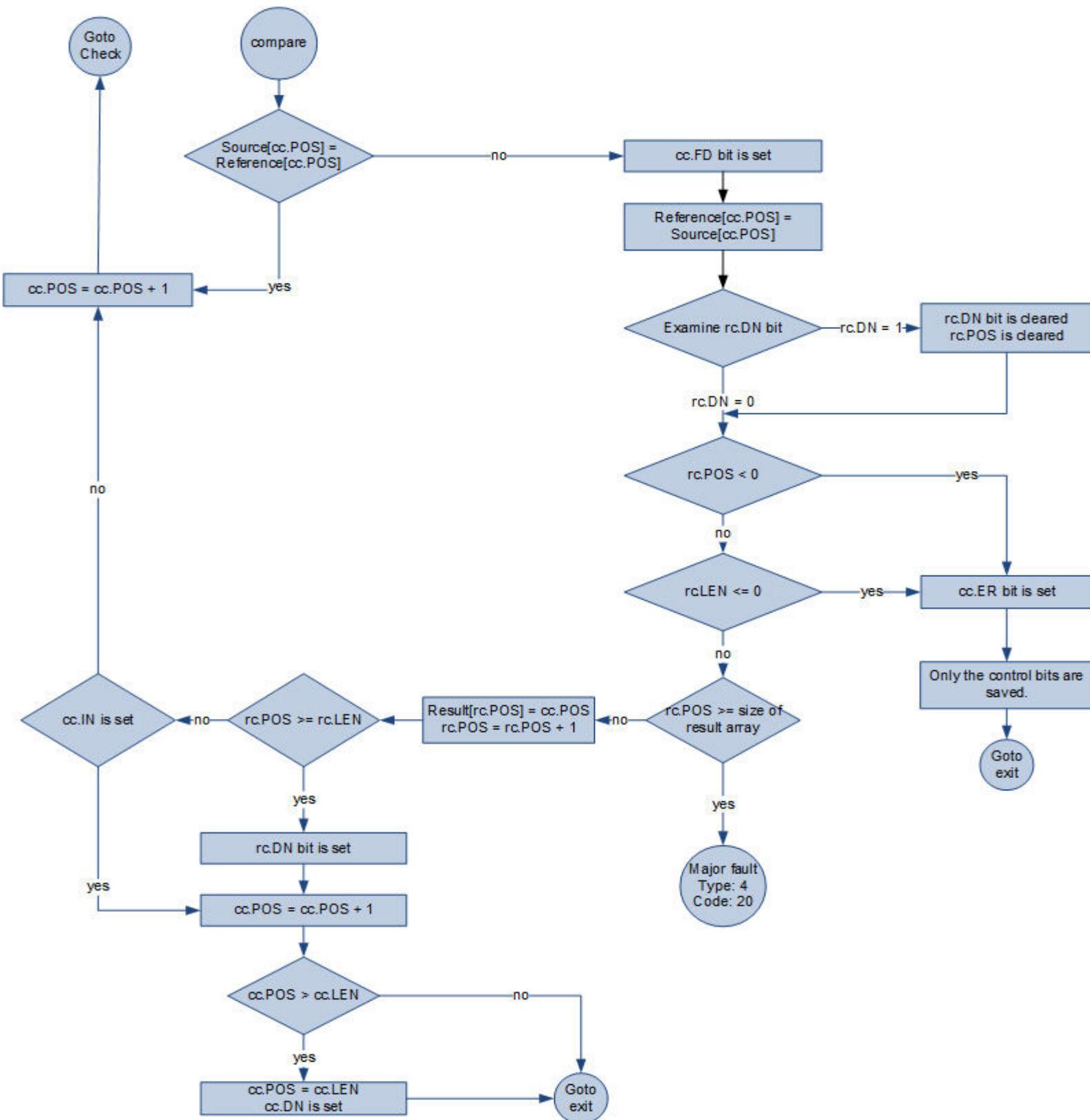
**Diagrama de flujo de DDT (Falso)**

Diagrama de flujo de DDT (Verdadero)

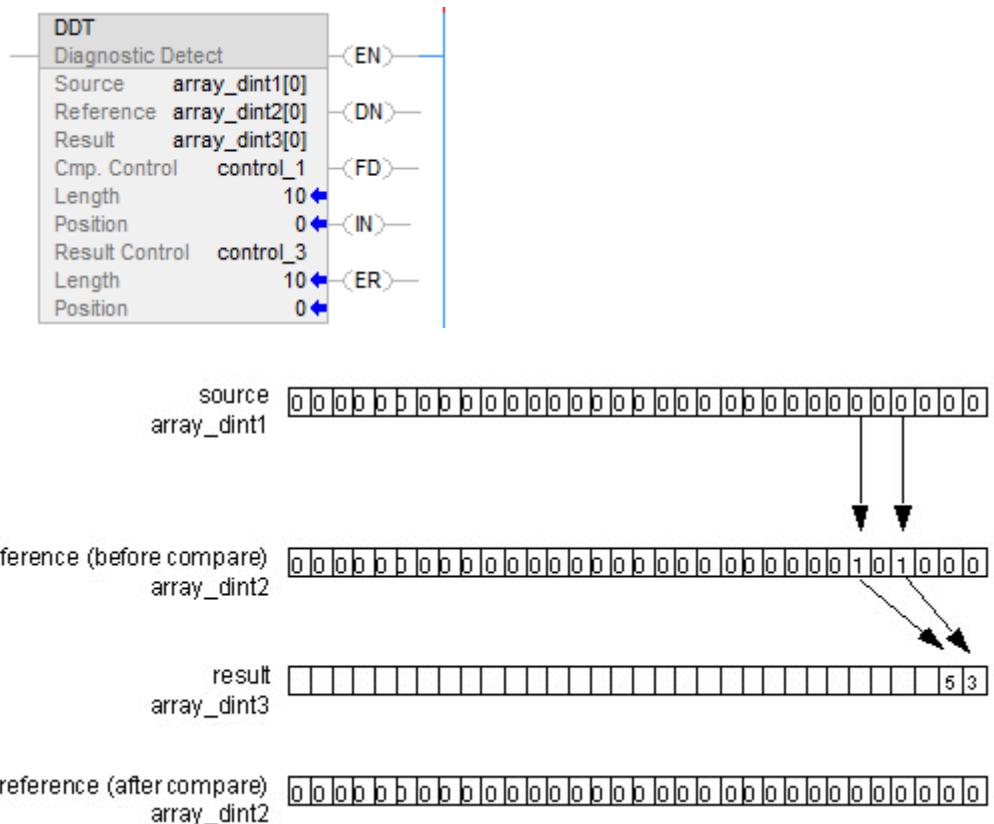


## Diagrama de flujo de DDT (Verdadero) - Continuación



## Ejemplos

## Diagrama de escalera



#### **Consulte también**

[Instrucciones especiales](#) en la página 721

DTR en la página 722

FBC en la página 734

Atributos comunes en la página 923

Conversiones de datos en la página 927

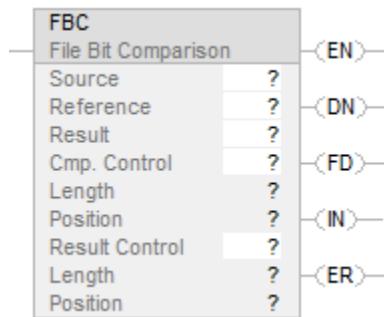
# Comparación de bits de archivo (FBC)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción FBC compara los bits de una matriz Source con los de una matriz Reference.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

#### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

#### Operandos

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción. Consulte Conversión de datos.

#### Diagrama de escalera

Operando	Tipo	Formato	Descripción
Source	DINT	etiqueta de matriz	Matriz para comparar con la referencia no se utiliza CONTROL.POS en el subíndice
Reference	DINT	etiqueta de matriz	Matriz para comparar con el origen no se utiliza CONTROL.POS en el subíndice
Result	DINT	etiqueta de matriz	Matriz para almacenar el resultado no se utiliza CONTROL.POS en los subíndices
Cmp. Control	CONTROL	estructura	Estructura de control para la comparación.
Length	DINT	inmediato	Número de bits a comparar
Position	DINT	inmediato	Posición actual en el origen el valor inicial suele ser 0

Result control	CONTROL	estructura	Estructura de control para los resultados.
Length	DINT	inmediato	número de ubicaciones de almacenamiento en el resultado
Position	DINT	inmediato	Posición actual en el resultado. el valor inicial suele ser 0

**Importante:** Se utiliza diferentes etiquetas para la estructura de control de comparación y la estructura de control de resultado. Si utiliza la misma etiqueta para ambos, podría producirse una operación impredecible, que podría dar lugar a daños en el equipamiento y/o lesiones físicas del personal.

#### Estructura de COMPARE

Mnemónico	Tipo de datos	Descripción
.EN	BOOL	El bit de habilitación indica que la instrucción FBC está habilitada.
.DN	BOOL	El bit de efectuado se establece cuando la instrucción FBC compara el último bit en las matrices Source y Reference.
.FD	BOOL	El bit de encontrado se establece cada vez que la instrucción FBC registra una incongruencia (operación una a la vez) o después de registrar todas las incongruencias (operación todas por escaneado).
.IN	BOOL	El bit de inhibición indica el modo de búsqueda de FBC. 0 = Modo Todos 1 = modo de discrepancias de una en una
.ER	BOOL	El bit de error se establece si POS o LEN no son válidas.
.LEN	DINT	El valor de longitud identifica el número de bits a comparar.
.POS	DINT	El valor de posición identifica el bit actual.

#### Estructura de RESULT

Mnemónico	Tipo de datos	Descripción
.DN	BOOL	El bit de efectuado se establece cuando la matriz Result está llena.
.LEN	DINT	El valor de longitud identifica el número de ubicaciones de almacenamiento en la matriz Result.
.POS	DINT	El valor de posición identifica la posición actual en la matriz Result.

## Descripción

Si está habilitada, la instrucción FBC compara los bits de una matriz Source con los de una matriz Reference y registra el número de bit de cada incongruencia en la matriz Result.

**Importante:** La instrucción FBC opera en la memoria continua. Compruebe y confirme que la instrucción no modifica datos que usted no desea modificar.

Las instrucciones DDT y FBC se diferencian en que cada vez que la instrucción DDT encuentra una incongruencia, modifica el bit de referencia para que coincida con el bit de origen. La instrucción FBC no cambia el bit de referencia.

Si la instrucción intenta leer más allá del final de una matriz, la instrucción establece el bit .ER y genera un fallo mayor.

## Seleccionar el modo de búsqueda

Si quiere detectar:	Seleccione este modo:
Discrepancias de una en una	<p>Se establece el bit .IN en la estructura de CONTROL de comparación.</p> <p>Cada vez que EnableIn pasa de falso a verdadero, la instrucción FBC busca la siguiente incongruencia entre las matrices Source y Reference. Tras encontrar una incongruencia, la instrucción establece el bit .FD, graba la posición de la incongruencia y detiene la ejecución.</p>
Todas las discrepancias	<p>Se borra el valor del bit .IN en la estructura de CONTROL de comparación.</p> <p>Cada vez que EnableIn pasa de falso a verdadero, la instrucción FBC busca todas las incongruencias entre las matrices Source and Reference.</p>

## Afectar a las marcas de estado matemático

No

## Condiciones de fallo

Se producirá un fallo mayor si:	Tipo de fallo	Código de fallo
result.POS > tamaño de la matriz Result	4	20

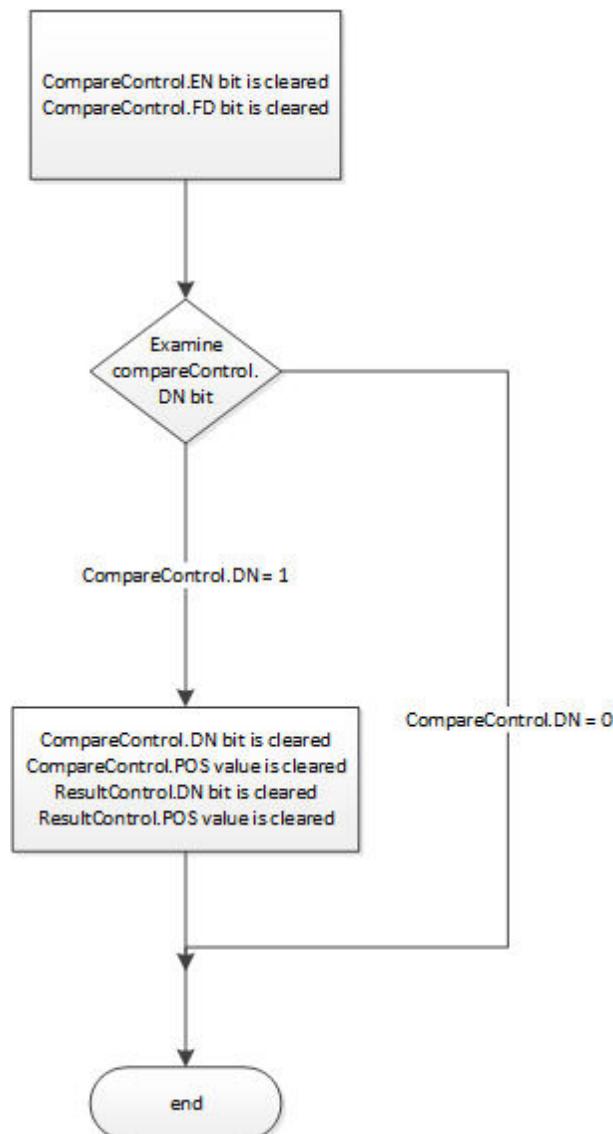
Consulte los Atributos comunes para los fallos relacionados con los operandos.

## Ejecución

### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	Consultar el Diagrama de flujo de FBC (Pre-escaneado)
La condición de entrada de reglón es falsa	Consultar el Diagrama de flujo de FBC (falso)
La condición de entrada de reglón es verdadera	Consultar el diagrama de flujo de FBC (verdadero)
Post-escaneado	N/A

### Diagrama de flujo de FBC (Pre-escaneado)



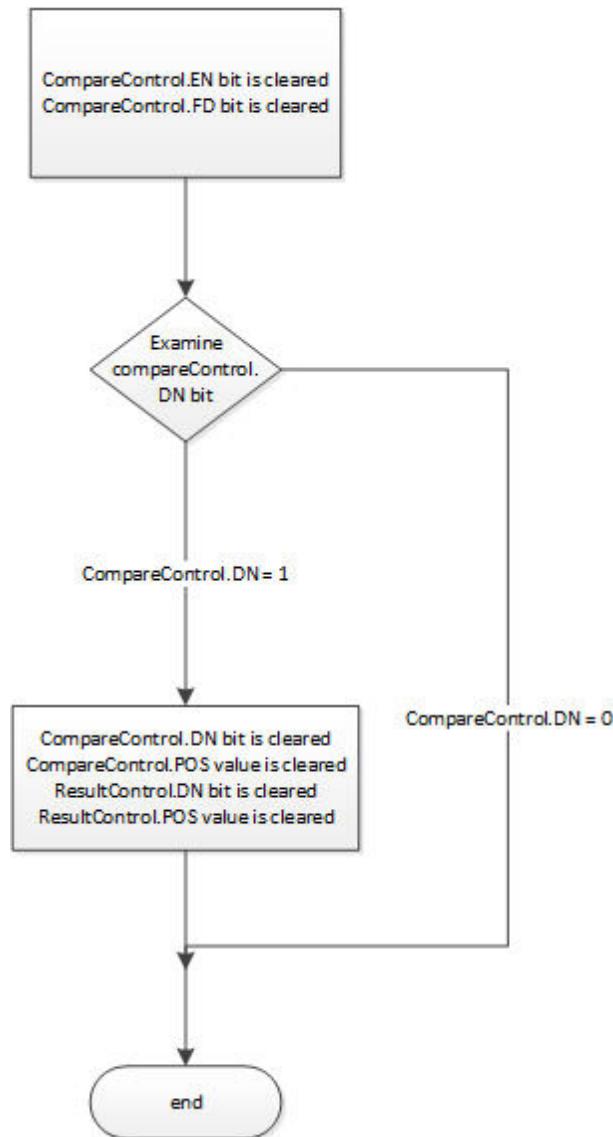
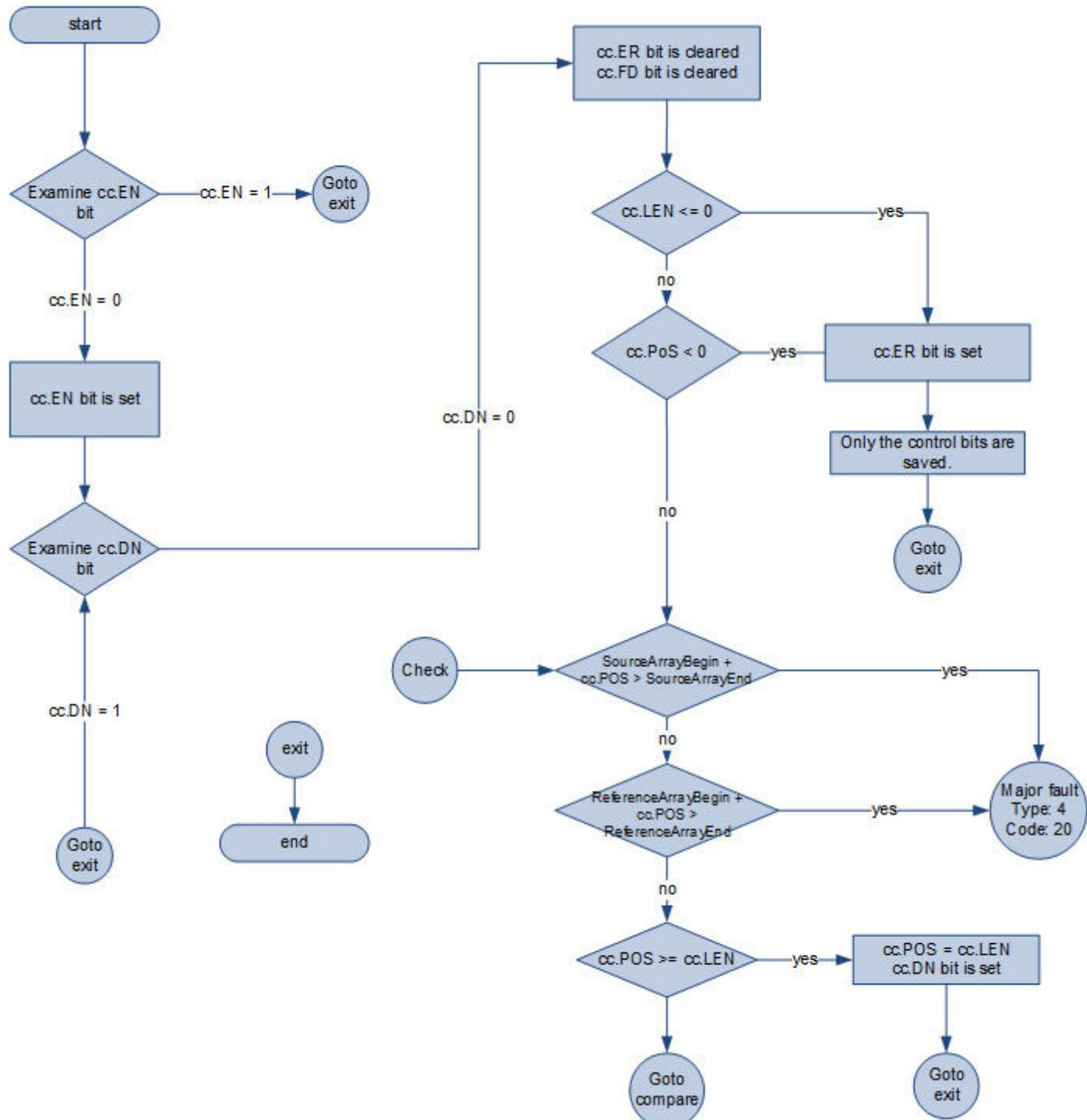
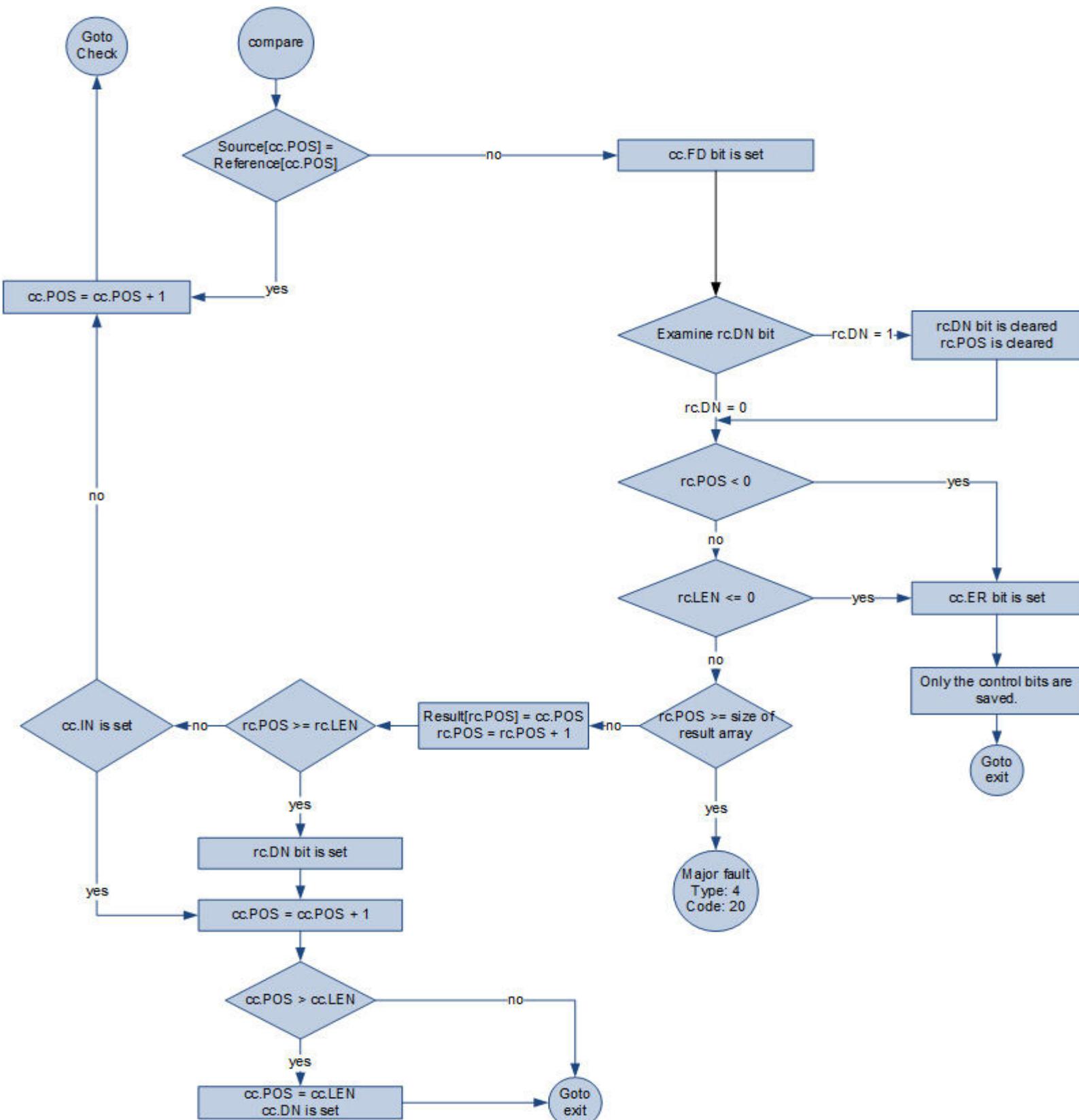
**Diagrama de flujo de FBC (falso)**

Diagrama de flujo de FBC (verdadero)

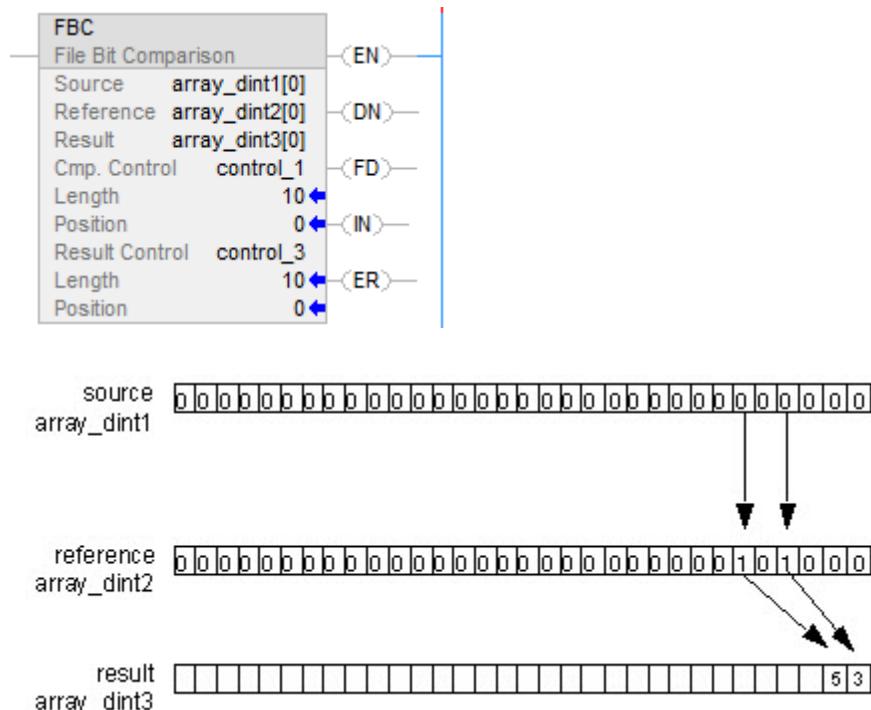


## Diagrama de flujo de FBC (verdadero) - continuación



## Ejemplo

## Diagrama de escalera



#### **Consulte también**

[Instrucciones especiales](#) en la página 721

DDT en la página 725

DTR en la página 722

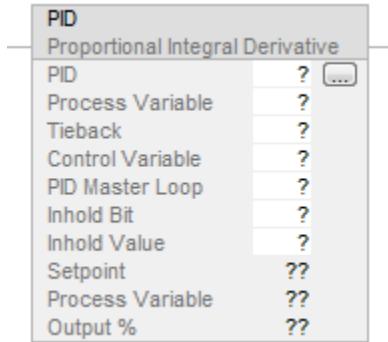
## PID mejorado (PID)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción PID controla una variable de proceso, como por ejemplo el flujo, la presión, la temperatura o el nivel.

## Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

### Texto estructurado

```
PID(PID,ProcessVariable,Tieback,ControlVariable,PIDMasterLoop,InHoldBit,I  
nHoldValue);
```

### Operandos

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción. Consulte Conversión de datos.

### Diagrama de escalera

Operando	Tipo	Formato	Descripción
PID	PID	estructura	Estructura de PID
Process variable	SINT	etiqueta	Valor que desea controlar
	INT		
	DINT		
	REAL		
Tieback	SINT	inmediato	(opcional)
	INT	etiqueta	
	DINT		Salida de una estación automática/de mano que está omitiendo la salida del controlador. Se introduce cero si no desea utilizar este parámetro
	REAL		

Control variable	SINT	etiqueta	Valor que va al dispositivo de control final (válvula, regulador, etc.)
	INT		
	DINT		Si está utilizando la banda muerta, Control variable debe ser REAL o se forzará a 0 si el error está dentro de la banda muerta.
	REAL		
PID master loop	PID	Estructura	Opcional
			Etiqueta PID para el PID maestro
			Sí está ejecutando un control en cascada y este PID es un lazo esclavo, se introduce el nombre del PID maestro
			Se introduce 0 si no desea utilizar este parámetro
Inhold bit	BOOL	etiqueta	Opcional
			Estado actual del bit mantenido de un canal de salida analógica 1756 para
			soportar un reinicio sin problema
Inhold value	SINT	etiqueta	Opcional
	INT		Valores de relectura de un canal de salida analógica 1756
	DINT		para soportar un reinicio sin problema
	REAL		Se introduce cero si no desea utilizar este parámetro
Setpoint			Solo para visualizar en pantalla
			Valor actual del punto de ajuste
Process variable			Solo para visualizar en pantalla
			Valor actual de Process_Variable escalado
Output %			Solo para visualizar en pantalla
			Valor de porcentaje de salida actual

### Texto estructurado

Operando	Tipo	Formato	Descripción
PID	PID	estructura	Estructura de PID
Process variable	SINT	etiqueta	Valor que desea controlar
	INT		
	DINT		
	REAL		
Tieback	SINT	inmediato	(opcional)
	INT	etiqueta	

	DINT REAL		Salida de una estación automática/de mano que está omitiendo la salida del controlador. Se introduce cero si no desea utilizar este parámetro
Control variable	SINT	etiqueta	Valor que va al dispositivo de control final (válvula, regulador, etc.)
	INT		
	DINT		Si está utilizando la banda muerta, Control variable debe ser REAL o se forzará a 0 si el error está dentro de la banda muerta.
	REAL		
PID master loop	PID	Estructura	Opcional Etiqueta PID para el PID maestro Si está ejecutando un control en cascada y este PID es un lazo esclavo, se introduce el nombre del PID maestro Se introduce 0 si no desea utilizar este parámetro
	BOOL		Opcional Estado actual del bit mantenido de un canal de salida analógica 1756 para soportar un reinicio sin problema
Inhold value	SINT INT DINT REAL	etiqueta	Opcional Valores de relectura de un canal de salida analógica 1756 para soportar un reinicio sin problema Se introduce cero si no desea utilizar este parámetro
Setpoint			Solo para visualizar en pantalla Valor actual del punto de ajuste
Process variable			Solo para visualizar en pantalla Valor actual de Process_Variable escalado
Output %			Solo para visualizar en pantalla Valor de porcentaje de salida actual

Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

### Estructura de PID

Especifica una única estructura para cada instrucción PID.

Mnemónico	Tipo de datos	Descripción		
.CTL	DINT	El miembro .CTL da acceso a los miembros de estado (bits) en una palabra de 32 bits. La instrucción PID establece los bits 07-15		
Bit	Número	Descripción		
.EN	31			
.CT	30	tipo cascada (0=esclavo; 1=maestro)		
.CL	29	lazo en cascada (0=no; 1=sí)		
PVT	28	seguimiento de variable de proceso (0=no; 1=sí)		
.DOE	27	derivada de (0=PV; 1=error)		
.SWM	26	modo del software (0=no-auto); 1=sí- sw manual)		
.CA	25	acción de control (0=directo (SP-PV); 1=inverso (PV-SP))		
.MO	24	modo de estación (0=automático; 1=manual)		
.PE	23	ecuación PID (0=independiente; 1=dependiente)		
.NDF	22	uniformidad de derivada (0=no; 1=sí)		
.NOBC	21	cálculo de polarización (0=no; 1=sí)		
.NOZC	20	paso por cero (0=no; 1=para banda muerta)		
.INI	15	PID inicializado (0=no; 1=sí)		
.SPOR	14	punto de ajuste fuera de rango (0=no; 1=sí)		
.OLL	13	CV está por debajo del valor mínimo de salida (0=no; 1=sí)		
.OLH	12	CV está por encima del valor máximo de salida (0=no; 1=sí)		
.EWD	11	error dentro de la banda muerta (0=no; 1=sí)		
.DVNA	10	error con alarma de límite bajo (0=no; 1=sí)		
.DVPA	9	error con alarma de límite alto (0=no; 1=sí)		
.PVLA	8	PV con alarma de límite bajo (0=no; 1=sí)		
.PVHA	7	PV con alarma de límite alto (0=no; 1=sí)		
.SP	REAL	punto de ajuste		

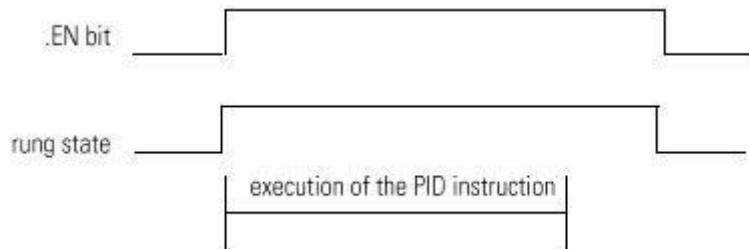
Mnemónico	Tipo de datos	Descripción
.KP	REAL	Independiente - ganancia proporcional (sin unidades)
		Dependiente - ganancia del controlador (sin unidades)
.KI	REAL	Independiente - ganancia integral (1/seg)
		Dependiente - tiempo de restablecimiento (minutos por repetición)
.KD	REAL	Independiente - ganancia derivativa (segundos)
		Dependiente - tiempo de acción derivada (minutos)
.BIAS	REAL	prealimentación o polarización %
.MAXS	REAL	valores máximos de escalado en unidades de ingeniería
.MINS	REAL	valores mínimos de escalado en unidades de ingeniería
.DB	REAL	unidades de ingeniería de banda muerta
.SO	REAL	% salida establecida
.MAXO	REAL	límite máximo de salida (% de salida)
.MINO	REAL	límite mínimo de salida (% de salida)
.UPD	REAL	tiempo de actualización del lazo (segundos)
.PV	REAL	valor PV escalado
.ERR	REAL	valor de error escalado
.OUT	REAL	salida %
.PVH	REAL	límite de alarma alto de variable de proceso
.PVL	REAL	límite de alarma bajo de variable de proceso
.DVP	REAL	límite de alarma de una desviación positiva
.DVN	REAL	límite de alarma de una desviación negativa
.PVDB	REAL	banda muerta de alarma de variable de proceso
.DVDB	REAL	banda muerta de alarma de desviación
.MAXI	REAL	valor PV máximo (entrada no escalada)
.MINI	REAL	valor PV mínimo (entrada no escalada)
.TIE	REAL	valor retenido para control manual
.MAXCV	REAL	valor CV máximo (correspondiente al 100%)
.MINCV	REAL	valor CV mínimo (correspondiente al 0%)
.MINTIE	REAL	valor retenido mínimo (correspondiente al 100%)
.MAXTIE	REAL	valor retenido máximo (correspondiente al 0%)
.DATA[17]	REAL	<b>El miembro .DATA almacena:</b>
		<b>Elemento</b>
		<b>Descripción</b>
		.DATA[0] acumulación integral
		.DATA[1] valor temporal de uniformidad de derivada
		.DATA[2] valor .PV previo
		.DATA[3] valor .ERR previo
		.DATA[4] valor .SP previo válido
		.DATA[5] constante de escalado porcentual
		.DATA[6] constante de escalado .PV

Mnemónico	Tipo de datos	Descripción	
		.DATA[7]	constante de escalado derivativo
		.DATA[8]	valor .KP previo
		.DATA[9]	valor .KI previo
		.DATA[10]	valor .KD previo
		.DATA[11]	.KP de ganancia dependiente
		.DATA[12]	.KI de ganancia dependiente
		.DATA[13]	.KD de ganancia dependiente
		.DATA[14]	valor .CV previo
		.DATA[15]	constante de desactivación de escalado .CV
		.DATA[16]	constante de desactivación de escalado de valor retenido

### Descripción

La instrucción PID recibe típicamente la variable de proceso (PV) desde un módulo de entrada analógica y modula una salida de control variable (CV) en un módulo de salida analógica para mantener la variable de proceso en el punto de ajuste deseado.

El bit .EN muestra el estado de ejecución. El bit .EN se establece cuando EnableIn pasa de falso a verdadero. El bit .EN se borra cuando EnableIn pasa a falso. La instrucción PID no usa el bit .DN. La instrucción PID ejecuta cada escaneado siempre que EnableIn sea verdadero.



### Afectar a las marcas de estado matemático

No

### Condiciones de fallo

Ocurrirá un fallo menor si:	Tipo de fallo	Código de fallo
UPD $\geq 0$	4	35
punto de ajuste fuera de rango	4	36

Consulte los *Atributos comunes* para fallos relacionados con el operando.

## Consulte también

[Instrucciones especiales](#) en la página 721

[Atributos comunes](#) en la página 923

[Sintaxis de texto estructurado](#) en la página 955

[Conversiones de datos](#) en la página 927

## Usar instrucciones PID

Después de introducir la instrucción PID y especificar la estructura PID, se utilizan las pestañas de configuración para especificar cómo debe funcionar.

### Especificar el ajuste

Seleccionar la pestaña de **Ajuste** (Tuning). Los cambios se hacen efectivos al hacer clic en otro campo, al hacer clic en **Aceptar** (OK), al hacer clic en **Aplicar** (Apply), o al presionar **Intro** (Enter).

En este campo:	Haga lo siguiente:
Punto de ajuste (Setpoint, SP)	Introducir un punto de ajuste (.SP).
% salida establecida (Set output %)	Introducir un porcentaje de salida establecido (.SO). Este valor se usa para la salida con el software en modo manual. En el modo auto, este valor muestra el % de salida.
Polarización de salida (Output bias)	Introducir un porcentaje de polarización de salida (.BIAS).
Ganancia proporcional (Proportional gain) (Kp)	Introducir la ganancia proporcional (.KP). En el caso de ganancias independientes, es la ganancia proporcional (sin unidades). En el caso de ganancias dependientes, es la ganancia del controlador (sin unidades).
Ganancia integral (Integral gain) (Ki)	Introducir la ganancia integral (.KI). En el caso de ganancias independientes, es la ganancia integral (1/seg). En el caso de ganancias dependientes, es el tiempo de restablecimiento (minutos por repetición).
Tiempo de derivada (Derivative time) (Kd)	Introducir la ganancia derivativa (.KD). En el caso de ganancias independientes, es la ganancia derivativa (segundos). En el caso de ganancias dependientes, es el régimen de tiempo (en minutos).
Modo manual (Manual mode)	Seleccionar manual (.MO) o manual mediante software (.SWM). El modo manual anula el modo manual mediante software si ambos están seleccionados.

### Especificar configuración

Seleccione la pestaña Configuración (Configuration). Debe hacer clic en Aceptar (OK) o en Aplicar (Apply) para que los cambios surtan efecto.

En este campo:	Haga lo siguiente:
Ecuación PID (PID equation)	Seleccionar ganancias independientes o ganancias dependientes (.PE). Se utilizan las independientes cuando desee que las tres ganancias (P, I y D) funcionen independientemente. Se utilizan las dependientes cuando desee una ganancia general del controlador que afecte a los tres términos de ganancia (P, I y D).
Acción de control (Control action)	Seleccionar E=PV-SP o E=SP-PV para la acción de control (.CA).
Derivada de (Derivative of)	Seleccionar PV o error (.DOE). Se utiliza la derivada de PV para reducir el riesgo de picos de salida causados por los cambios del punto de ajuste. Se utiliza la derivada de error para obtener respuestas rápidas a cambios del punto de ajuste cuando el algoritmo puede tolerar sobreimpulsos.
Tiempo de actualización del lazo (Loop update time)	Introducir el tiempo de actualización (.UPD) para la instrucción.
Límite alto de CV (CV high limit)	Introducir un límite alto para la variable de control (.MAXO). (1)
Límite bajo de CV (CV low limit)	Introducir un límite bajo para la variable de control (.MINO). (1)
Valor de banda muerta (Deadband value)	Introducir un valor de banda muerta (.DB).
Sin uniformidad de derivada (No derivative smoothing)	Habilitar o deshabilitar esta selección (.NDF).
Sin cálculo de polarización (No derivative smoothing)	Habilitar o deshabilitar esta selección (.NOBC).
Sin paso por cero en la banda muerta (No zero crossing in deadband)	Habilitar o deshabilitar esta selección (.NOZC).
Seguimiento de PV (PV tracking)	Habilitar o deshabilitar esta selección (.PVT).
Lazo en cascada (Cascade loop)	Habilitar o deshabilitar esta selección (.CL).
Tipo de cascada (Cascade type)	Si el lazo en cascada está habilitado, seleccione esclavo o maestro (.CT).

(1) Cuando se utiliza una instrucción PID basada en un diagrama de escalera, si establece que MAXO = MINO, la instrucción restablece los valores por defecto. MAXO = 100,0 y MINO = 0,0

### Especificar alarmas

Seleccione la pestaña **Alarmas** (Alarms). Debe hacer clic en **Aceptar** (OK) o en **Aplicar** (Apply) para que los cambios surtan efecto.

En este campo:	Haga lo siguiente:
PV alta (PV high)	Introducir un valor de alarma alta PV (.PVH).
PV baja (PV low)	Introducir un valor de alarma baja PV (.PVL).

Banda muerta de PV (PV deadband)	Introducir un valor de banda muerta de alarma PV (.PVDB).
Desviación positiva (Positive deviation)	Introducir un valor de desviación positiva (.DVP).
Desviación negativa (Negative deviation)	Introducir un valor de desviación negativa (.DVN).
Banda muerta de desviación (Deviation deadband)	Introducir un valor de banda muerta de alarma de desviación (.DVDB).

### Especificar la escala

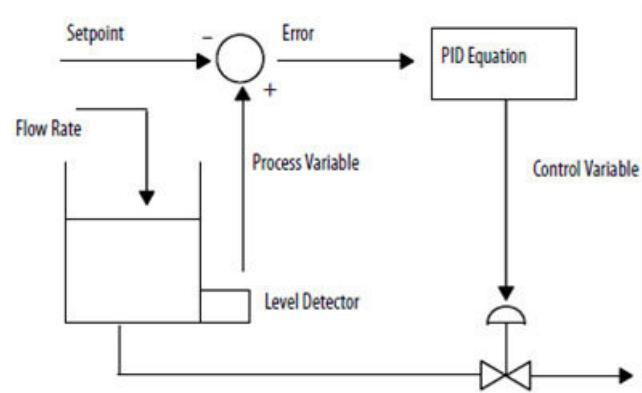
Seleccionar la pestaña Escala (Scaling). Debe hacer clic en Aceptar (OK) o en Aplicar (Apply) para que los cambios surtan efecto.

En este campo:	Haga lo siguiente:
PV máximo sin escala (PV unscaled maximum)	Introducir un valor PV máximo (.MAXI) que sea igual al valor sin escala máximo recibido desde el canal de entrada analógica para el valor PV.
PV mínimo sin escala (PV unscaled minimum)	Introducir un valor PV mínimo (.MINI) que sea igual al valor sin escala mínimo recibido desde el canal de entrada analógica para el valor PV.
PV en unidades de ingeniería máximo (PV engineering units maximum)	Introducir el valor máximo en unidades de ingeniería correspondiente a .MAXI (.MAXS).
PV en unidades de ingeniería mínimo (PV engineering units minimum)	Introducir el valor mínimo en unidades de ingeniería correspondiente a .MINI (.MINS).
CV máximo (CV maximum)	Introducir un valor CV máximo correspondiente al 100% (.MAXCV).
CV mínimo (CV minimum)	Introducir un valor CV mínimo correspondiente al 0% (.MINCV).
Valor retenido máximo (Tieback maximum)	Introducir un valor retenido máximo (.MAXTIE) que sea igual al valor sin escala máximo recibido desde el canal de entrada analógica para el valor retenido.
Valor retenido mínimo (Tieback minimum)	Introducir un valor retenido mínimo (.MINTIE) que sea igual al valor sin escala mínimo recibido desde el canal de entrada analógica para el valor retenido.
PID inicializado (PID Initialized)	Si se cambian las constantes de escala durante el modo Marcha, se deberá desactivar esta opción para reiniciar los valores sin escala internos (.INI).

**Cons ejo:** Cuando se utiliza una instrucción PID basada en un diagrama de escalera, si establece que MAXO = MINO, la instrucción restablece los valores por defecto. MAXO = 100,0 y MINO = 0,0

### Usar instrucciones PID

Un control de lazo cerrado PID retiene una variable del proceso en un punto de ajuste deseado. La siguiente figura muestra un ejemplo de régimen de flujo/nivel de fluido.



En el ejemplo anterior, el nivel del tanque se compara con el punto de ajuste. Si el nivel es más alto que el punto de ajuste, la ecuación PID aumenta la variable de control y hace que la válvula de salida del tanque se abra; de esta manera baja el nivel del tanque.

La ecuación PID que se utiliza en la instrucción PID es una ecuación de forma posicional con la posibilidad de usar ganancias independientes o dependientes. Si se utilizan ganancias independientes, las ganancias proporcionales, integrales y derivativas afectan de manera específica solo sus respectivos términos proporcional, integral y derivativo. Si se utilizan ganancias dependientes, la ganancia proporcional es sustituida por una ganancia de controlador que afecta a los tres términos. Puede usar cualquier forma de ecuación para realizar el mismo tipo de control. Los dos tipos solo se proporcionan para que usted pueda utilizar el tipo de ecuación con el que esté más familiarizado.

Tipos de ganancias	Derivada de
Ganancias dependientes (estándar ISA)	Error (E)
	Process variable (PV)
Ganancias independientes	Error (E)
	Process variable (PV)

Donde:

Variable	Descripción
KP	Ganancia proporcional (sin unidad) Kp = Kc sin unidad

Ki	Ganancia integral (segundos -1) Para convertir entre Ki (ganancia integral) a Ti (tiempo de restablecimiento), se utiliza: $K_i = \frac{K_C}{60 T_i}$
Kd	Ganancia derivativa (segundos) Para convertir entre Kd (ganancia derivativa) y Td (tiempo de acción derivada), se utiliza: Kd = Kc (Td) 60
KC	Ganancia del controlador (sin unidades)
Ti	Tiempo de restablecimiento (minutos/repetición)
Td	Tiempo de acción derivada (minutos)
SP	Punto de ajuste
PV	Process variable
E	Error [(SP-PV) o (PV-SP)]
BIAS	Prealimentación o polarización
CV	Variable de control
dt	Tiempo de actualización del lazo (Loop update time)

Si no desea utilizar un término concreto de la ecuación PID, simplemente establece su ganancia en 0. Por ejemplo, si no desea acción derivativa, establece Kd o Td en cero.

#### Consulte también

[Reinicio sin perturbaciones](#) en la página 754

[Uniformidad de derivada](#) en la página 757

[Establecer la banda muerta](#) en la página 761

[Lazos en cascada](#) en la página 755

[Control de relación](#) en la página 756

#### Bloqueo anti-restablecimiento y transferencia sin perturbaciones de manual a automático (PID)

La instrucción PID evita automáticamente el bloqueo de restablecimiento impidiendo que el término integral se acumule cuando la salida de CV alcance sus valores máximo o mínimo, que se establecen mediante .MAXO y .MINO. El término integral acumulado permanece inmovilizado hasta que la salida de CV esté por debajo de su límite máximo o por encima de su límite mínimo. En ese momento, se reanudará automáticamente la acumulación integral habitual.

La instrucción PID admite dos modos manuales de control.

Modo Manual de control	Descripción
Software manual (.SWM)	<p>Este modo también se conoce como modo de salida establecida y permite al usuario establecer el % de salida desde el software.</p> <p>El valor de salida establecida (.SO) se utiliza como la salida del lazo. El valor de salida establecida por lo general procede de una entrada de operador desde un dispositivo de interfaz de operador.</p>
Manual (.MO)	<p>Este modo coge el valor retenido como una entrada y ajusta sus variables internas para generar el mismo valor en la salida. La entrada retenida a la instrucción PID se escala un 0-100% de acuerdo con los valores de .MINTIE y .MAXTIE y se utiliza como la salida del lazo. La entrada retenida procede por lo general de la salida de una estación de mano/automática de hardware que omite la salida desde el controlador.</p> <p><b>Importante:</b> El modo Manual anula el modo manual mediante software si los dos bits de modo están establecidos como activados.</p>

La instrucción PID proporciona automáticamente transferencias sin perturbaciones del modo manual mediante software al modo automático o desde el modo manual al modo automático. La instrucción PID calcula sobre la base de datos el valor del término de acumulación integral requerido para hacer que el seguimiento de salida de CV sea el valor de salida establecida (.SO) en el manual mediante software o la entrada retenida en el modo manual. De este modo, cuando el lazo cambia al modo automático, la salida de CV empieza desde la salida establecida o el valor retenido y no se produce ninguna perturbación en el valor de salida.

La instrucción PID también puede proporcionar automáticamente una transferencia sin perturbaciones del modo manual al automático aunque no se utiliza el control integral (es decir,  $Ki = 0$ ). En este caso, la instrucción modifica el término .BIAS para hacer que el seguimiento de la salida de CV sea el valor de salida establecida o el valor retenido. Cuando se reanude el control automático, el término .BIAS mantendrá su último valor. Deshabilita el cálculo sobre la base de datos originales del término .BIAS estableciendo el bit .NOBC en la estructura de datos PID. Si establece .NOBC en verdadero, la instrucción PID ya no proporcionará una transferencia sin perturbaciones del modo manual al automático cuando no se use el control integral.

## Reinicio sin perturbaciones (PID)

La instrucción PID puede interactuar con los módulos de salida analógica 1756 para permitir un reinicio sin perturbaciones cuando el controlador pasa del modo Programa al modo Marcha o cuando se enciende el controlador.

Cuando un módulo de salida analógica 1756 pierde la comunicación con el controlador o detecta que el controlador está en el modo Programa, el módulo de salida analógica establece sus salidas en los valores de condición de fallo que especificó cuando configuró el módulo. Cuando, a continuación, el controlador regresa al modo Marcha o vuelve a establecer la comunicación con el módulo de salida analógica, puede hacer que la instrucción PID restablezca de manera automática su salida de variable de control igual a la salida analógica mediante el uso de los parámetros Inhold bit y Inhold Value de la instrucción PID.

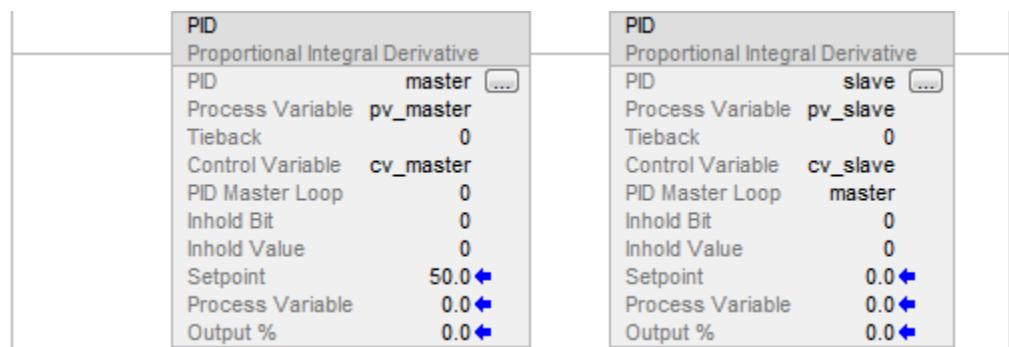
### Instrucciones para configurar un reinicio sin perturbaciones

Haga esto	Detalles
Configura el canal del módulo de salida analógica 1756 que recibe la variable de control desde la instrucción PID.	<p>Selecciona el cuadro <b>Retener para inicialización</b> en la página de propiedades del canal específico del módulo.</p> <p>Esto indica al módulo de salida analógica que cuando el controlador regresa al modo Marcha o reestablece la comunicación con el módulo, el módulo debería retener la salida analógica en su valor actual hasta que el valor enviado desde el controlador coincida (con un margen de 0,1% con respecto al alcance) con el valor actual usado por el canal de salida. La salida del canal es controlada por rampa hasta que alcanza el valor de salida retenido actualmente mediante el uso del término .BIAS. Este control por rampa es similar a la transferencia sin perturbaciones automática.</p>
Introduce la etiqueta Inhold bit y la etiqueta Inhold Value en la instrucción PID	<p>El módulo de salida analógica 1756 devuelve dos valores para cada canal en su estructura de datos de entrada. Cuando el valor del bit de estado InHold (.Ch2InHold, por ejemplo) cuando verdadero, indica que el canal de salida analógica está reteniendo su valor. El valor de relectura de datos (.Ch2Data, por ejemplo), muestra el valor de salida actual en unidades de ingeniería.</p> <p>Introduce la etiqueta del bit de estado InHold como el parámetro de bit InHold de la instrucción PID. Introduce la etiqueta del valor de relectura de datos como el parámetro de Inhold Value.</p> <p>Si el Inhold bit es verdadero, la instrucción PID mueve el Inhold Value a la salida de Control variable y reinicializa para permitir un reinicio sin perturbaciones en ese valor. Cuando el módulo de salida analógica recibe este valor de vuelta del controlador, desactiva el bit de estado de InHold, que permite a la instrucción PID comenzar a controlar con normalidad.</p>

### Lazos en cascada (PID)

El PID conecta dos lazos en cascada asignando la salida en porcentaje del lazo maestro al punto de ajuste del lazo esclavo. El lazo esclavo transforma automáticamente la salida del lazo maestro a unidades de ingeniería correctas para el punto de ajuste del lazo esclavo, basándose en los valores .MAXS y .MINS del lazo esclavo.

### Escalera de relés



### Texto estructurado

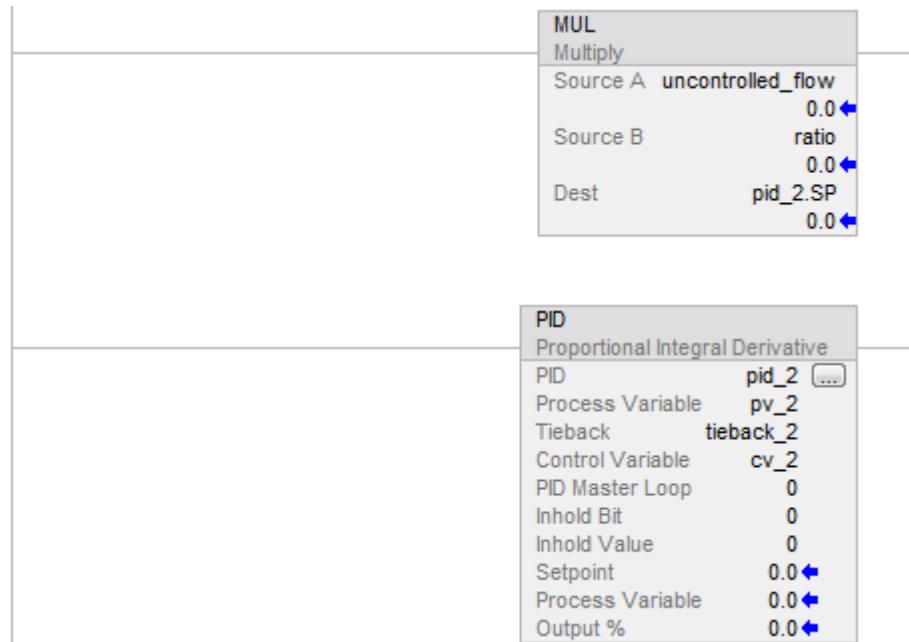
```
PID(master,pv_master,0,cv_master,0,0,0); PID  
(slave,pv_slave,0,cv_slave,master,0,0);
```

## Control de relación (PID)

Puede mantener la relación entre dos valores utilizando los siguientes parámetros:

- Valor no controlado
- Valor controlado (el punto de ajuste resultante que se usará para la instrucción PID)
- Relación entre estos dos valores

### Escalera de relés



**Cons** Para evitar el bloqueo de PID con valores internos  
**ejo:** inválidos de punto flotante, asegúrese de que PV no sea  
 INF o NAN antes de invocar instrucciones como:  
 XIC (PC\_timer.DN)  
 MOV(Local:0:1.Ch0Data, Local:0:1.Ch0Data)  
 XIO(S:V)  
 PID(...)

### Texto estructurado

```
pid_2.sp := uncontrolled_flow * ratio
```

PID(pid\_2,pv\_2,tieback\_2,cv\_2,0,0,0);

**Cons** Para evitar el bloqueo de PID con valores internos  
**ejo:** inválidos de punto flotante, asegúrese de que PV no sea  
 INF o NAN antes de invocar instrucciones como:  
 XIC (PC\_timer.DN)  
 MOV(Local:0:1.Ch0Data, Local:0:1.Ch0Data)  
 XIO(S:V)  
 PID(...)

Para esta multiplicación	Se introduce estos valores
Destination	Valor controlado
Source A	Valor no controlado
Source B	Ratio

## Uniformidad de la derivada (PID)

Un filtro de uniformidad de la derivada mejora el cálculo de la misma. Este filtro digital de primer orden y paso minimiza los picos altos del término de la derivada causados por ruido en la PV. Esta uniformidad se hace más intensa con valores mayores de ganancia derivativa. Se puede deshabilitar la uniformidad derivativa si el proceso requiere valores muy altos de ganancia derivativa ( $K_d > 10$ , por ejemplo).

**Para deshabilitar la uniformidad:**

- Seleccione la opción **Sin uniformidad derivativa** (No derivative smoothing) en la pestaña **Configuración** (Configuration) o establezca el bit .NDF en la estructura de PID.

## Prealimentación o polarización de salida (PID)

Prealimente una perturbación desde el sistema introduciendo el valor de .BIAS en el valor de prealimentación/polarización de la instrucción PID.

El valor de prealimentación representa una perturbación introducida en la instrucción PID antes de que la perturbación tenga la oportunidad de cambiar la variable de proceso. A menudo la prealimentación se utiliza para controlar procesos con un retardo de transporte. Por ejemplo, un valor de prealimentación que representa el agua fría que se vierte en una mezcla caliente podría disparar el valor de salida más rápidamente que si se esperase a que la variable de proceso cambiase como resultado de la mezcla.

Por lo general, un valor de polarización se utiliza cuando no se utiliza ningún control integral. En este caso, el valor de polarización se ajusta para que mantenga la salida dentro del rango requerido para mantener el PV cerca del punto de ajuste.

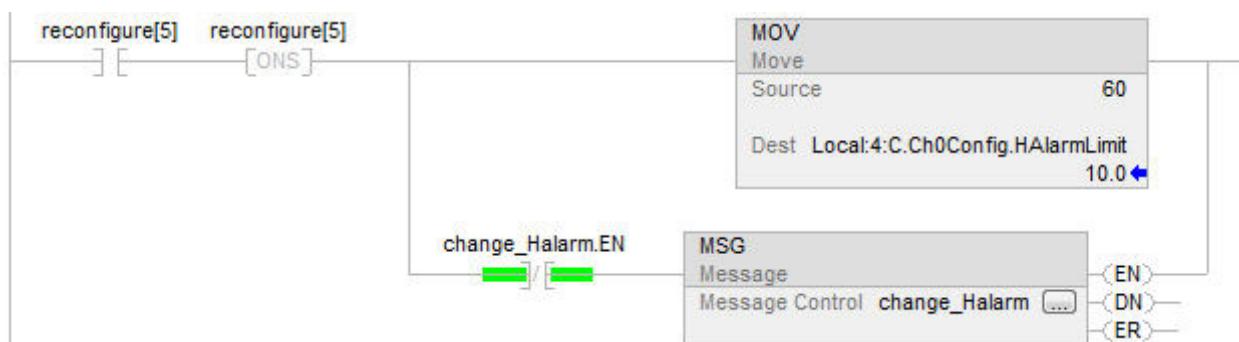
## Temporización de la instrucción PID

Es necesario actualizar periódicamente la instrucción PID y el muestreo de la variable de proceso. Este tiempo de actualización está relacionado con el proceso físico que usted está controlando. En el caso de lazos muy lentos, como por ejemplo los lazos de temperatura, para tener un buen control suele ser suficiente un periodo de una actualización por segundo o incluso un periodo mayor. En lazos más rápidos, como por ejemplo los lazos de presión o de flujo, puede que sean necesarios unos periodos de actualización como de 250 milisegundos. Solo en casos raros, como por ejemplo en el control de la tensión de un carrete de desbobinador, se requieren periodos de actualización de 10 milisegundos o de menor tiempo.

Debido a que la instrucción PID utilizan una base de tiempo para su cálculo, necesitará sincronizar la ejecución de la instrucción con el muestreo de la variable de proceso (PV).

La forma más sencilla de ejecutar una instrucción PID es introducirla en una tarea periódica. Establezca que el tiempo de actualización del lazo (.UPD) sea el mismo que el régimen de la tarea periódica y asegúrese de que la instrucción PID se ejecute en cada escaneado de la tarea.

## Escalera de relés



<b>Cons ejo:</b>	Para evitar el bloqueo de PID con valores internos inválidos de punto flotante, asegúrese de que PV no sea INF o NAN antes de invocar instrucciones como:  XIC (PC_timer.DN) MOV(Local:0:1.Ch0Data, Local:0:1.Ch0Data) XIO(S:V) PID(...)
----------------------	---

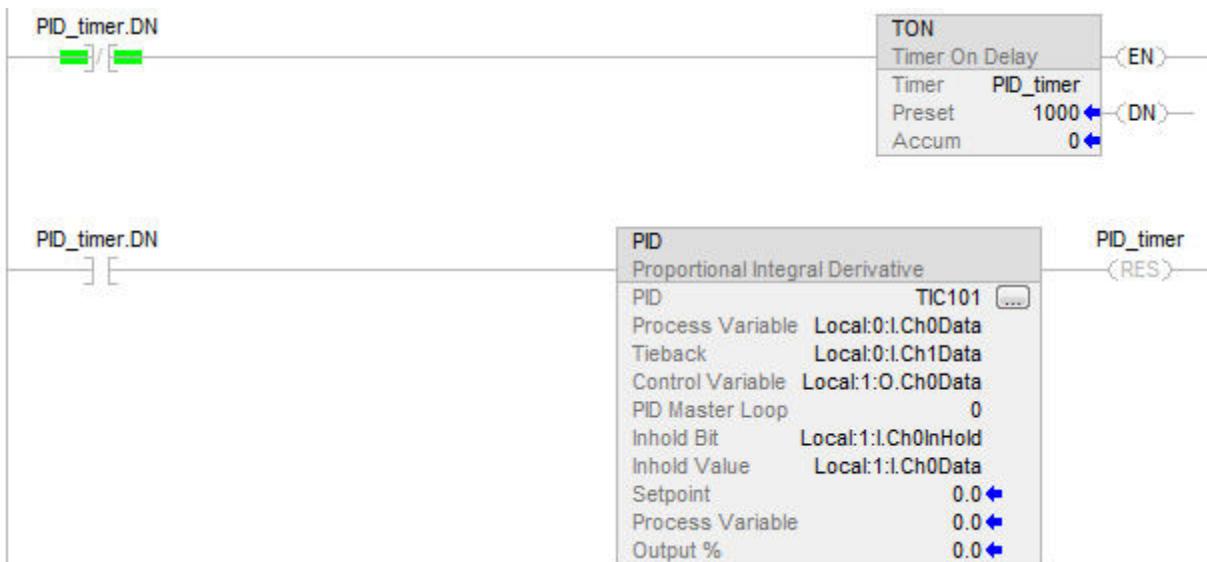
## Texto estructurado

PID(TIC101,Local:0:I.Ch0Data,Local:0:I.Ch1Data,  
Local:1:O.Ch4Data,0.Local:1:I.Ch4InHold, Local:1:I.Ch4Data);

Cuando use una tarea periódica, asegúrese de que la entrada analógica empleada para la variable de proceso se actualice con el procesador a un régimen que sea considerablemente más rápido que el régimen de la tarea periódica. Idealmente, la variable de proceso debería enviarse al procesador al menos de 5 a 10 veces más rápido que el régimen de la tarea periódica. De esta manera se minimiza la diferencia de tiempo entre las muestras actuales de la variable de proceso y la ejecución del lazo PID. Por ejemplo, si el lazo PID está en una tarea periódica de 250 ms, se utiliza un tiempo de actualización de 250 ms (.UPD = .25), y se configura el módulo de entrada analógica para que produzca datos al menos cada 25 o 50 ms.

Otro método, aunque menos exacto, para ejecutar una instrucción consiste en introducir la instrucción en una tarea continua y utilizar un bit de efectuado de temporizador para desencadenar la ejecución de la instrucción PID.

### Escalera de relés



**Cons** Para evitar el bloqueo de PID con valores internos  
**ejo:** inválidos de punto flotante, asegúrese de que PV no sea  
 INF o NAN antes de invocar instrucciones como:  
 XIC (PC\_timer.DN)  
 MOV(Local:0:1.Ch0Data, Local:0:1.Ch0Data)  
 XIO(S:V)  
 PID(...)

### Texto estructurado

```
PID_timer.pre := 1000
```

```
TONR(PID_timer);
```

```
IF PID_timer.DN THEN PID(TIC101,Local:0:I.Ch0Data,Local:0:I.Ch1Data,  
Local:1:O.Ch0Data,0,Local:1:I.Ch0InHold,  
Local:1:I.Ch0Data);  
END_IF;
```

**Cons ejo:** Para evitar el bloqueo de PID con valores internos inválidos de punto flotante, asegúrese de que PV no sea INF o NAN antes de invocar instrucciones como:  
XIC (PC\_timer.DN)  
MOV(Local:0:1.Ch0Data, Local:0:1.Ch0Data)  
XIO(S:V)  
PID(...)

Con este método, el tiempo de actualización de lazo de la instrucción PID debe ser el mismo que el valor preestablecido en el temporizador. Igual que cuando se utiliza una tarea periódica, debería establecer que el módulo de entrada analógica produzca la variable de proceso a un régimen considerablemente más rápido que el régimen de actualización de lazo. Solo debería utilizar el método de temporización de la ejecución PID en lazos cuyos tiempos de actualización sean al menos varias veces mayor que el tiempo de ejecución más largo posible de la tarea continua.

La forma más exacta de ejecutar una instrucción PID es usar la función de muestreo en tiempo real (RTS) de los módulos de entrada analógicas 1756. El módulo de entrada analógica muestrea sus entradas en tiempo real del régimen de muestreo que usted establezca al configurar el módulo. Cuando expira el periodo de muestreo en tiempo real, el módulo actualiza sus entradas y actualiza también una marca de tiempo continua que produce el módulo (representada por el miembro .RollingTimestamp de la estructura de entradas analógicas).

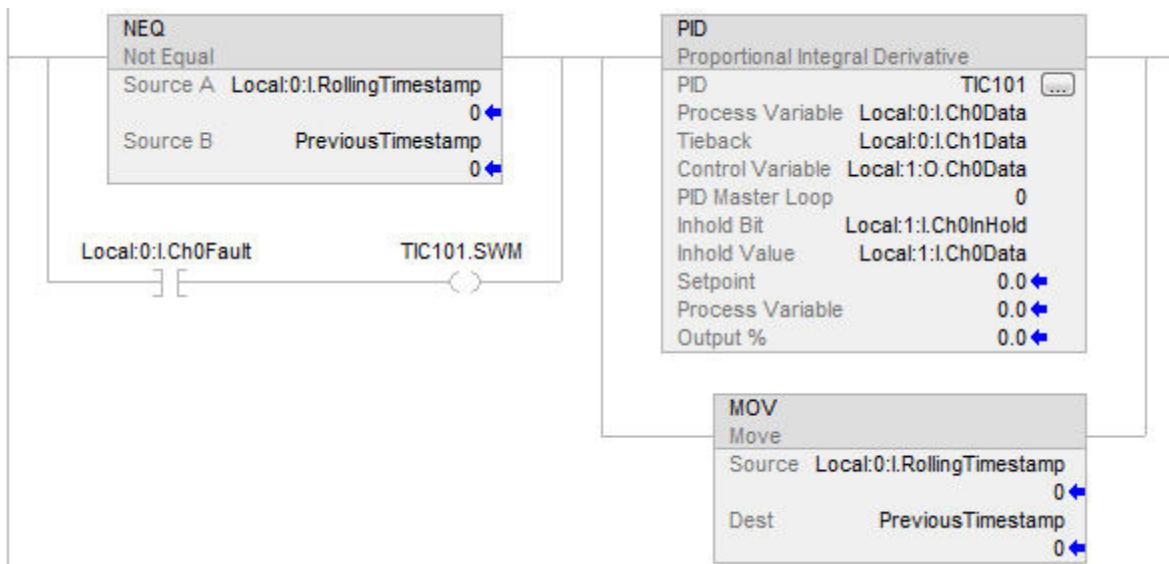
El rango de la marca de tiempo oscila entre 0 y 32,767 milisegundos. Se monitorea la marca de tiempo. Si cambia, significa que se ha recibido una muestra nueva de la variable de proceso. Cada vez que cambia una marca de tiempo, se ejecuta una instrucción PID. Puesto que el muestreo de la variable de proceso lo activa un módulo de entrada analógica, el tiempo de muestreo de entrada es muy exacto y el tiempo de actualización de lazo que utiliza la instrucción PID debe ser el mismo que el tiempo RTS del módulo de entrada analógica.

Para asegurarse de no estar omitiendo muestreos de la variable de proceso, ejecute la lógica a un régimen más rápido que el tiempo RTS. Por ejemplo, si el tiempo RTS es de 250 ms, puede colocar la lógica PID en una tarea periódica que se ejecute cada

100 ms y así se asegurará de que no pierde ningún muestra nunca. Incluso podría colocar la lógica PID en una tarea continua siempre que se asegure de que se actualizará con una frecuencia mayor de 250 ms.

A continuación se muestra un ejemplo del método de ejecución RTS. La ejecución de la instrucción PID depende de la recepción de datos de entrada analógica nuevos. Si falla o se desmonta el módulo de entrada analógica, el controlador deja de recibir las marcas de tiempo continuas y el lazo PID deja de ejecutarse. Debe monitorear el bit de estado de la entrada analógica PV y, si muestra un estado incorrecto, se fuerza el lazo al modo manual de software y se ejecuta el lazo en cada escaneado. De esta manera el operador podría cambiar de forma manual la salida del lazo PID.

### Escalera de relés



### Texto estructurado

```

IF (Local:0:I.Ch0Fault) THEN TIC101.SWM [:=] 1;
ELSE TIC101.SWM := 0; END_IF;

IF (Local:0:I.RollingTimestamp<>PreviousTimestamp) OR
(Local:0:I.Ch0Fault) THEN

    PreviousTimestamp := Local:0:I.RollingTimestamp;
    PID(TIC101,Local:0:I.Ch0Data,Local:0:I.Ch1Data,
        Local:1:O.Ch0Data,0,Local:1:I.Ch0InHold,
        Local:1:I.Ch0Data);

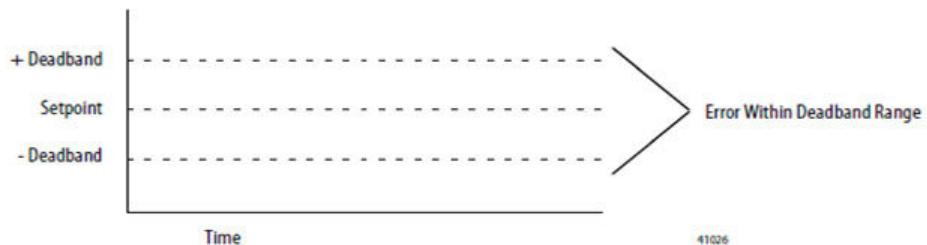
END_IF;

```

### Establecer la banda muerta (PID)

La banda muerta ajustable permite seleccionar un rango de error por encima y por debajo del punto de ajuste donde la salida no cambia, siempre que el error permanece dentro de este rango. Esta banda muerta permite controlar la precisión

de la variable del proceso con el punto de ajuste sin cambiar la salida. La banda muerta también ayuda a minimizar el desgaste del dispositivo de control final.



El cruce por cero es un control de banda muerta que permite que la instrucción utiliza el error con fines de cálculo desde que la variable del proceso cruza hacia la banda muerta hasta que cruza el punto de ajuste. Cuando la variable del proceso haya cruzado el punto de ajuste (el error cruza por cero y cambia de signo) y mientras que permanece en la banda muerta, la salida no cambiará.

La banda muerta se extiende por arriba y por abajo del punto de ajuste según el valor que usted especifica. Introduzca "0" para inhibir la banda muerta. La banda muerta tiene las mismas unidades de escala que el punto de ajuste. Para utilizar la banda muerta sin la función de cruce por cero seleccionando **Sin cruce por cero para banda muerta** (No zero crossing for deadband) en la pestaña **Configuración** (Configuration) o estableciendo el bit .NOZC en la estructura PID.

Si está utilizando la banda muerta, Control variable debe ser REAL o se forzará a 0 si el error está dentro de la banda muerta.

#### Para inhibir la banda muerta:

- Introduzca cero (0).

La banda muerta tiene las mismas unidades de escala que el punto de ajuste.

#### Para utilizar la banda muerta sin la función de cruce por cero:

- Seleccione **Sin cruce por cero para banda muerta** (No zero crossing for deadband) en la pestaña **Configuración** (Configuration) o establezca el bit .NOZC en la estructura PID.

Si está utilizando la banda muerta, Control variable debe ser REAL o se forzará a 0 si el error está dentro de la banda muerta.

## Usar el límite de salida (PID)

Establece un límite de salida (porcentaje de salida) en la salida de control. Cuando la instrucción detecta que la salida ha alcanzado un límite, establece un bit de alarma e impide que la salida supere tanto el límite inferior como el superior.

# Instrucciones trigonométricas

Las instrucciones trigonométricas evalúan las operaciones aritméticas utilizando operaciones trigonométricas.

## Instrucciones disponibles

### Diagrama de escalera, Bloque de funciones y Texto estructurado

<u>SIN</u>	<u>ATN</u>	<u>COS</u>	<u>TAN</u>	<u>ASN</u>	<u>ACS/ASO S</u>
------------	------------	------------	------------	------------	----------------------

Si desea:	Utilice esta instrucción:
Hallar el seno de un valor.	SIN
Hallar el coseno de un valor.	COS
Hallar la tangente de un valor.	TAN
Hallar el arcoseno de un valor.	ASN
Hallar el arcocoseno de un valor.	ACS
Hallar la arcotangente de un valor.	ATN

Puede mezclar tipos de datos, pero puede que se produzca una pérdida de precisión y errores de redondeo y la instrucción tarda más en ejecutarse. Compruebe el bit S:V para ver si el resultado se ha truncado.

Los tipos de datos que aparecen en **negrita** indican tipos de datos óptimos. Una instrucción se ejecutará a la máxima velocidad y la menor cantidad posible de memoria si todos los parámetros emplean el mismo tipo de datos óptimo, normalmente DINT o REAL.

Una instrucción trigonométrica ejecuta la instrucción una vez cada vez que se escanea la instrucción y durante tanto tiempo como sea verdadera la condición de entrada de reglón. Si desea que se evalúe la instrucción solo una vez, se utiliza la instrucción ONS para desencadenar la instrucción trigonométrica.

**Consulte también**[Instrucciones de temporizador y contador](#) en la página 171[Instrucciones especiales](#) en la página 721[Instrucciones de secuenciador](#) en la página 645[Instrucciones de control de programa](#) en la página 660[Instrucciones de movimiento/lógicas](#) en la página 467

## Instrucciones trigonométricas

Las instrucciones trigonométricas evalúan las operaciones aritméticas utilizando operaciones trigonométricas.

**Instrucciones disponibles****Diagrama de escalera, Bloque de funciones y Texto estructurado**

<a href="#">SIN</a>	<a href="#">ATN</a>	<a href="#">COS</a>	<a href="#">TAN</a>	<a href="#">ASN</a>	<a href="#">ACS/ASO S</a>
---------------------	---------------------	---------------------	---------------------	---------------------	-------------------------------

Si desea:	Utilice esta instrucción:
Hallar el seno de un valor.	SIN
Hallar el coseno de un valor.	COS
Hallar la tangente de un valor.	TAN
Hallar el arcoseno de un valor.	ASN
Hallar el arcocoseno de un valor.	ACS
Hallar la arcotangente de un valor.	ATN

Puede mezclar tipos de datos, pero puede que se produzca una pérdida de precisión y errores de redondeo y la instrucción tarda más en ejecutarse. Compruebe el bit S:V para ver si el resultado se ha truncado.

Los tipos de datos que aparecen en **negrita** indican tipos de datos óptimos. Una instrucción se ejecutará a la máxima velocidad y la menor cantidad posible de memoria si todos los parámetros emplean el mismo tipo de datos óptimo, normalmente DINT o REAL.

Una instrucción trigonométrica ejecuta la instrucción una vez cada vez que se escanea la instrucción y durante tanto tiempo como sea verdadera la condición de

entrada de reglón. Si desea que se evalúe la instrucción solo una vez, se utiliza la instrucción ONS para desencadenar la instrucción trigonométrica.

### Consulte también

[Instrucciones de temporizador y contador](#) en la página 171

[Instrucciones especiales](#) en la página 721

[Instrucciones de secuenciador](#) en la página 645

[Instrucciones de control de programa](#) en la página 660

[Instrucciones de movimiento/lógicas](#) en la página 467

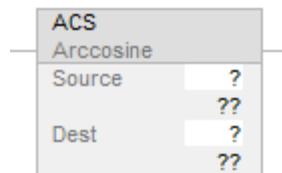
## Arcocoseno (ACS, ACOS)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

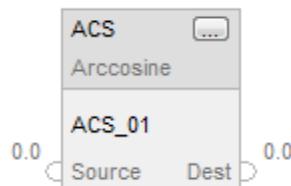
La instrucción ACS coge el arcocoseno del valor de Source y almacena el resultado en Destination (en radianes).

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones



#### Texto estructurado

```
dest := ACOS(source);
```

## Operandos

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción. Consulte Conversión de datos.

### Diagrama de escalera

Operando	Tipo	Formato	Descripción
Source	SINT INT DINT REAL	Inmediato etiqueta	averiguar el coseno de este valor
Destination	SINT INT DINT REAL	etiqueta	etiqueta para almacenar el resultado

### Texto estructurado

Operando	Tipo	Formato	Descripción
Source	SINT INT DINT REAL	inmediato etiqueta	averiguar el coseno de este valor

Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

Se utiliza ACOS como una función. Esta función calcula el arcocoseno de Source y devuelve el resultado REAL.

### Bloque de funciones

Operando	Tipo	Formato	Descripción
ACS tag	FBD_MATH_ADVANCE D	Estructura	Estructura de ACS

### Estructura de FBD\_MATH\_ADVANCED

Parámetro de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si se borra, la instrucción no se ejecuta y las salidas no se actualizan. Está establecido de forma predeterminada.
Source	REAL	Entrada a la instrucción matemática.

Parámetro de salida	Tipo de datos	Descripción
EnableOut	BOOL	Indica si la instrucción está habilitada.
Dest	REAL	Resultado de la instrucción matemática.

### Descripción

La instrucción ACS coge el arcocoseno del valor de Source y almacena y devuelve el resultado REAL en Destination (en radianes). El valor de Source debe ser mayor o igual a -1 y menor o igual a 1. El valor resultante en Destination es mayor o igual a 0 o menor o igual a pi. Si Source es inferior a -1 o superior a 1, Destination se establece en NAN.

Puede utilizar ACS como operador en expresiones de escalera; puede usar ACOS como operador en declaraciones de Texto estructurado.

### Afectar a las marcas de estado matemático

Controladores	Afectar a la marca de estado matemático
ControlLogix 5580	Condicional, consulte Marcas de estado matemático.
CompactLogix 5370, ControlLogix 5570	Sí

### Fallos mayores/menores

Si el destino se establece en NAN, se generará un desbordamiento, con su fallo menor condicional.

### Ejecución

#### Diagrama de escalera

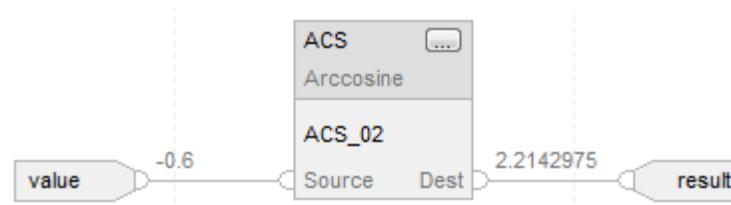
Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	El controlador calcula el arcocoseno de Source y coloca el resultado en Destination.
Post-escaneado	N/A

**Bloque de funciones**

Condición/estado	Acción realizada
Pre-escaneado	N/A
Tag.EnableIn es falso	El valor de EnableOut se borra a falso.
Tag.EnableIn es verdadero	El valor de EnableOut se establece en verdadero Si el bloque genera un desbordamiento, el valor de EnableOut se borra a falso.
Primer escaneado de instrucción	N/A
Primera ejecución de instrucción	N/A
Post-escaneado	N/A

**Texto estructurado**

Condición/estado	Acción realizada
Pre-escaneado	N/A.
Ejecución normal	El controlador calcula el arcocoseno de Source y coloca el resultado en Destination.
Post-escaneado	N/A

**Ejemplo****Diagrama de escalera****Bloque de funciones****Texto estructurado**

```
result := ACOS(value);
```

**Consulte también**

[Instrucciones de trigonometría](#) en la página 764

[Atributos comunes](#) en la página 923

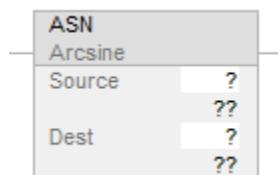
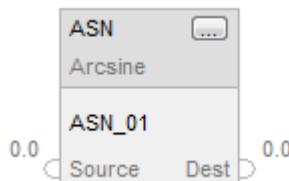
[Marcas de estado matemático](#) en la página 923

[Sintaxis de texto estructurado](#) en la página 955

**Arcoseno (ASN, ASIN)**

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

La instrucción ASN coge el arcoseno del valor de Source y almacena el resultado en Destination (en radianes).

**Idiomas disponibles****Diagrama de escalera****Bloque de funciones****Texto estructurado**

```
dest :=ASIN(source);
```

**Operандos**

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción. Consulte *Conversión de datos*.

**Diagrama de escalera**

<b>Operando</b>	<b>Tipo</b>	<b>Formato</b>	<b>Descripción</b>
Source	SINT INT DINT <b>REAL</b>	Inmediato etiqueta	averiguar el arcoseno de este valor
Destination	SINT INT DINT REAL	etiqueta	etiqueta para almacenar el resultado

**Texto estructurado**

<b>Operando</b>	<b>Tipo</b>	<b>Formato</b>	<b>Descripción</b>
Source	SINT INT DINT REAL	inmediato etiqueta	averiguar el arcoseno de este valor

Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

Se utiliza ASIN como una función. Esta función calcula el arcoseno de Source y devuelve el resultado REAL.

**Bloque de funciones**

<b>Operando</b>	<b>Tipo</b>	<b>Formato</b>	<b>Descripción</b>
ASN tag	FBD_MATH_ADVANCED	Estructura	Estructura de ASN

**Estructura de FBD\_MATH\_ADVANCED**

<b>Parámetro de entrada</b>	<b>Tipo de datos</b>	<b>Descripción</b>
EnableIn	BOOL	Habilita la entrada. Si el valor es falso, la instrucción no se ejecuta y las salidas no se actualizan. El valor predeterminado es verdadero.
Source	REAL	Entrada a la instrucción matemática. Válido = cualquier punto flotante

Parámetro de salida	Tipo de datos	Descripción
EnableOut	BOOL	Indica si la instrucción está habilitada.
Dest	REAL	Resultado de la instrucción.

### Descripción

La instrucción ASN calcula el arcoseno del valor de Source y almacena y devuelve el resultado REAL en Destination (en radianes). El valor de Source debe ser mayor o igual a -1 y menor o igual a 1. El valor resultante en Destination es mayor o igual a -pi/2 y menor o igual a pi/2. Si Source es inferior a -1 o superior a 1, Destination se establece en NAN.

Puede utilizar ASN como operador en expresiones de escalera; puede usar ASIN como operador en declaraciones de Texto estructurado.

La instrucción proporciona una mayor precisión con respecto a los controladores antiguos para proporcionar mejores resultados.

### Afectar a las marcas de estado matemático

Controladores	Afectar a la marca de estado matemático
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Condicional, consulte <i>Marcas de estado matemático</i> .
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí

### Fallos mayores/menores

Si el destino se establece en NAN, se generará un desbordamiento, con su fallo menor condicional.

### Ejecución

#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A

La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	La instrucción se ejecuta.
Post-escaneado	N/A

### Bloque de funciones

Condición/estado	Acción realizada
Pre-escaneado	N/A
Tag.EnableIn es falso	El valor de EnableOut se borra a falso.
Tag.EnableIn es verdadero	El valor de EnableOut se establece en verdadero Si el bloque genera un desbordamiento, el valor de EnableOut se borra a falso.
Primer escaneado de instrucción	N/A
Primera ejecución de instrucción	N/A
Post-escaneado	N/A

### Texto estructurado

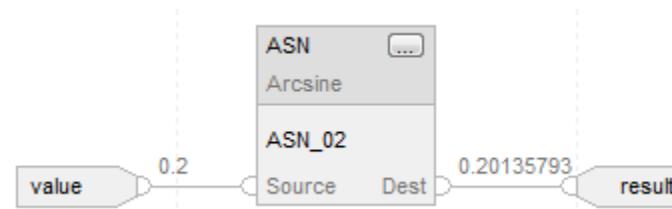
Condición/estado	Acción realizada
Pre-escaneado	N/A.
Ejecución normal	El controlador calcula el arcoseno de Source y coloca el resultado en Destination.
Post-escaneado	N/A

### Ejemplo

#### Diagrama de escalera



### Bloque de funciones



### Texto estructurado

```
result := ASIN(value);
```

### Consulte también

[Instrucciones de trigonometría](#) en la página 764

[Atributos comunes](#) en la página 923

[Marcas de estado matemático](#) en la página 923

[Sintaxis de texto estructurado](#) en la página 955

## Arcotangente (ATN, ATAN)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

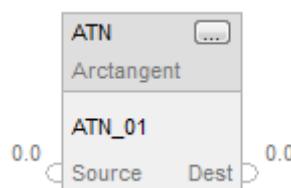
La instrucción ATN calcula la arcotangente del valor de Source y almacena el resultado en Destination (en radianes).

### Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones



### Texto estructurado

```
dest := ATAN(source);
```

## Operandos

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción. Consulte *Conversión de datos*.

## Diagrama de escalera

Operando	Tipo	Formato	Descripción
Source	SINT INT DINT REAL	Inmediato etiqueta	Averiguar la arcotangente de este valor
Destination	SINT INT DINT REAL	etiqueta	Etiqueta para almacenar el resultado

## Texto estructurado

Operando	Tipo	Formato	Descripción
Source	SINT INT DINT REAL	Inmediato etiqueta	Averiguar la arcotangente de este valor

Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

Se utiliza ATAN como una función. Esta función calcula la arcotangente de Source y devuelve el resultado REAL.

## Bloque de funciones

Operando	Tipo	Formato	Descripción
ATN tag	FBD_MATH_ADVANCED	Estructura	Estructura de ATN

## Estructura de FBD\_MATH\_ADVANCED

Parámetro de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si el valor es falso, la instrucción no se ejecuta y las salidas no se actualizan. El valor predeterminado es verdadero.
Source	REAL	Entrada a la instrucción matemática. Válido = cualquier punto flotante

Parámetro de salida	Tipo de datos	Descripción
EnableOut	BOOL	Indica si la instrucción está habilitada.
Dest	REAL	Resultado de la instrucción.

### Descripción

La instrucción ATN calcula la arcotangente del valor de Source y almacena el resultado en Destination (en radianes). El valor resultante en Destination es mayor o igual a -pi/2 y menor o igual a pi/2.

Puede utilizar ATN como operador en expresiones de escalera; puede usar ATAN como operador en declaraciones de Texto estructurado.

### Afectar a las marcas de estado matemático

Controladores	Afectar a la marca de estado matemático
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Condicional, consulte Marcas de estado matemático.
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí

### Fallos mayores/menores

No es específico para esta instrucción. Consulte *Atributos comunes* para los fallos relacionados con los operandos

### Ejecución

#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	El controlador calcula la arcotangente de Source y coloca el resultado en Destination.

Post-escaneado	N/A
----------------	-----

### Bloque de funciones

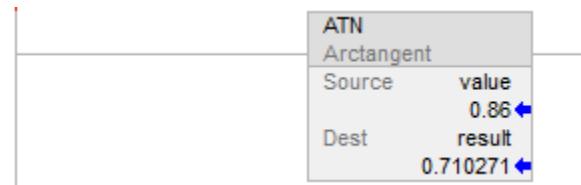
Condición/estado	Acción realizada
Pre-escaneado	N/A
Tag.EnableIn es falso	El valor de EnableOut se borra a falso.
Tag.EnableIn es verdadero	El valor de EnableOut se establece en verdadero Si el bloque genera un desbordamiento, el valor de EnableOut se borra a falso.
Primer escaneado de instrucción	N/A
Primera ejecución de instrucción	N/A
Post-escaneado	N/A

### Texto estructurado

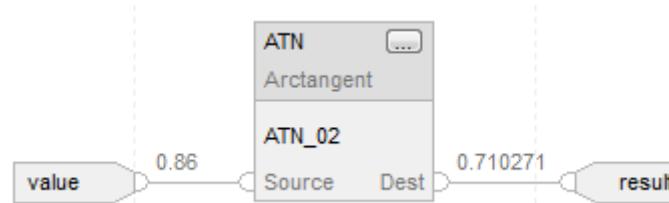
Condición/estado	Acción realizada
Pre-escaneado	N/A
Ejecución normal	El controlador calcula la arcotangente de Source y coloca el resultado en Destination.
Post-escaneado	N/A

### Ejemplo

#### Diagrama de escalera



### Bloque de funciones



### Texto estructurado

```
result := ATAN(value);
```

### Consulte también

[Instrucciones de trigonometría](#) en la página 764

[Atributos comunes](#) en la página 923

[Conversiones de datos](#) en la página 927

[Marcas de estado matemático](#) en la página 923

[Sintaxis de texto estructurado](#) en la página 955

## Coseno (COS)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

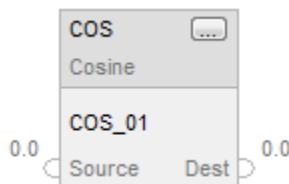
La instrucción COS coge el coseno del valor de Source (en radianes) y almacena el resultado en Destination.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones



### Texto estructurado

```
dest := COS(source);
```

### Operandos

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción. Consulte *Conversión de datos*.

**Diagrama de escalera**

<b>Operando</b>	<b>Tipo</b>	<b>Formato</b>	<b>Descripción</b>
Source	SINT INT DINT REAL	Inmediato etiqueta	averiguar el coseno de este valor
Destination	SINT INT DINT REAL	etiqueta	etiqueta para almacenar el resultado

**Texto estructurado**

<b>Operando</b>	<b>Tipo</b>	<b>Formato</b>	<b>Descripción</b>
Source	SINT INT DINT REAL	inmediato etiqueta	averiguar el coseno de este valor

Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

**Bloque de funciones**

<b>Operando</b>	<b>Tipo</b>	<b>Formato</b>	<b>Descripción</b>
COS tag	FBD_MATH_ADVANCED	Estructura	Estructura de COS

**Estructura de FBD\_MATH\_ADVANCED**

<b>Parámetro de entrada</b>	<b>Tipo de datos</b>	<b>Descripción</b>
EnableIn	BOOL	Habilita la entrada. Si se borra, la instrucción no se ejecuta y las salidas no se actualizan. Está establecido de forma predeterminada.
Source	REAL	Entrada a la instrucción matemática.

<b>Parámetro de salida</b>	<b>Tipo de datos</b>	<b>Descripción</b>
EnableOut	BOOL	Indica si la instrucción está habilitada.
Dest	REAL	Resultado de la instrucción matemática.

### Descripción

La instrucción COS calcula el coseno del valor Source (en radianes) y almacena el resultado en Destination.

La instrucción calcula el coseno de Source y devuelve el resultado REAL. El valor resultante siempre es mayor o igual a -1 y menor o igual a 1.

puede usar COS como función en expresiones de escalera y en declaraciones de Texto estructurado.

La instrucción proporciona una mayor precisión con respecto a los controladores antiguos para proporcionar mejores resultados.

### Afectar a las marcas de estado matemático

controladores	Afectar a la marca de estado matemático
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Condicional, consulte Marcas de estado matemático.
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí

### Fallos mayores/menores

Ninguno. Consulte *Atributos comunes* para los fallos relacionados con los operandos

### Ejecución

#### Diagrama de escalera

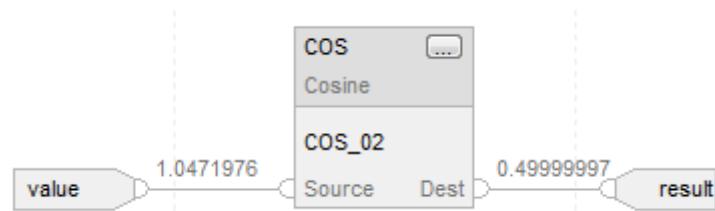
Condición/estado	Acción realizada
Pre-escaneado	N/A.
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	El controlador calcula el coseno de Source y coloca el resultado en Destination.
Post-escaneado	N/A

**Bloque de funciones**

Condición/estado	Acción realizada
Pre-escaneado	N/A
Tag.EnableIn es falso	El valor de EnableOut se borra a falso.
Tag.EnableIn es verdadero	El valor de EnableOut se establece en verdadero Si el bloque genera un desbordamiento, el valor de EnableOut se borra a falso.
Primer escaneado de instrucción	N/A
Primera ejecución de instrucción	N/A
Post-escaneado	N/A

**Texto estructurado**

Condición/estado	Acción realizada
Pre-escaneado	N/A.
Ejecución normal	El controlador calcula el coseno de Source y coloca el resultado en Destination.
Post-escaneado	N/A

**Ejemplo****Diagrama de escalera****Bloque de funciones****Texto estructurado**

```
result := COS(value);
```

### Consulte también

[Instrucciones de trigonometría](#) en la página 764

[Radianes \(RAD\)](#) en la página 817

[Atributos comunes](#) en la página 923

[Marcas de estado matemático](#) en la página 923

[Sintaxis de texto estructurado](#) en la página 955

## Seno (SIN)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

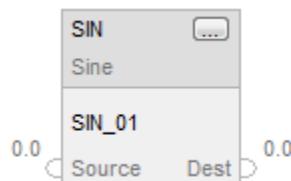
La instrucción SIN coge el seno del valor de Source (en radianes) y almacena el resultado en Destination.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones



### Texto estructurado

dest := SIN(source);

### Operando

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción. Consulte *Conversión de datos*.

**Diagrama de escalera**

<b>Operando</b>	<b>Tipo</b>	<b>Formato</b>	<b>Descripción</b>
Source	SINT INT DINT REAL	Inmediato etiqueta	averiguar el seno de este valor
Destination	SINT INT DINT REAL	etiqueta	etiqueta para almacenar el resultado

**Texto estructurado**

<b>Operando</b>	<b>Tipo</b>	<b>Formato</b>	<b>Descripción</b>
Source	SINT INT DINT REAL	Inmediato etiqueta	averiguar el seno de este valor

Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

**Bloque de funciones**

<b>Operando</b>	<b>Tipo</b>	<b>Formato</b>	<b>Descripción</b>
SIN tag	FBD_MATH_ADVANCED	Estructura	Estructura de SIN

**Estructura de FBD\_MATH\_ADVANCED**

<b>Parámetro de entrada</b>	<b>Tipo de datos</b>	<b>Descripción</b>
EnableIn	BOOL	Habilita la entrada. Si se borra, la instrucción no se ejecuta y las salidas no se actualizan. Está establecido de forma predeterminada.
Source	REAL	Entrada a la instrucción matemática.

<b>Parámetro de salida</b>	<b>Tipo de datos</b>	<b>Descripción</b>
EnableOut	BOOL	Indica si la instrucción está habilitada.
Dest	REAL	Resultado de la instrucción matemática.

## Aspectos del operador

El operador SIN se puede utilizar en varias expresiones. De manera similar, la función SIN se invoca en declaraciones de Texto estructurado. Ambas aplicaciones de SIN devuelven un resultado de tipo REAL que contiene el seno del valor de Source. Dependiendo del contexto, este valor podría convertirse a continuación si correspondiese.

### Descripción

La instrucción SIN coge el seno del valor de Source (en radianes) y almacena el resultado en Destination.

La instrucción calcula el seno de Source y devuelve el resultado REAL. El valor resultante siempre es mayor o igual a -1 y menor o igual a 1.

Puede utilizar SIN como operador en expresiones de escalera y como función en declaraciones de texto estructurado.

### Afectar a las marcas de estado matemático

Controladores	Afectar a las marcas de estado matemático
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Condicional, consulte <i>Marcas de estado matemático</i> .
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí

### Fallos mayores/menores

No es específico para esta instrucción. Consulte *Atributos comunes* para los fallos relacionados con los operandos

### Ejecución

#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A.
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	El controlador calcula el seno de Source y coloca el resultado en Destination.

Post-escaneado	N/A
----------------	-----

### Bloque de funciones

Condición/estado	Acción realizada
Pre-escaneado	N/A
Tag.EnableIn es falso	El valor de EnableOut se borra a falso.
Tag.EnableIn es verdadero	El valor de EnableOut se establece en verdadero Si el bloque genera un desbordamiento, el valor de EnableOut se borra a falso.
Primer escaneado de instrucción	N/A
Primera ejecución de instrucción	N/A
Post-escaneado	N/A

### Texto estructurado

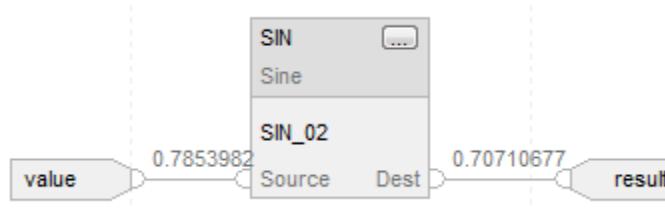
Condición/estado	Acción realizada
Pre-escaneado	N/A.
Ejecución normal	El controlador calcula el seno de Source y coloca el resultado en Destination.
Post-escaneado	N/A

### Ejemplo

#### Diagrama de escalera



### Bloque de funciones



### Texto estructurado

```
result := SIN(value);
```

### Consulte también

[Instrucciones de trigonometría](#) en la página 764

[Atributos comunes](#) en la página 923

[Marcas de estado matemático](#) en la página 923

[Sintaxis de texto estructurado](#) en la página 955

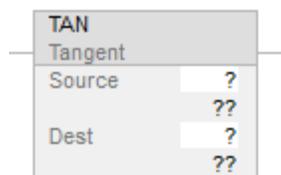
## Tangente (TAN)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

La instrucción TAN calcula la tangente de Source (en radianes) y almacena el resultado en Destination.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones



### Texto estructurado

dest := TAN(source);

### Operандos

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción. Consulte *Conversión de datos*.

**Diagrama de escalera**

<b>Operando</b>	<b>Tipo</b>	<b>Formato</b>	<b>Descripción</b>
Source	SINT INT DINT REAL	Inmediato etiqueta	averiguar el coseno de este valor
Destination	SINT INT DINT REAL	etiqueta	etiqueta para almacenar el resultado

**Texto estructurado**

<b>Operando</b>	<b>Tipo</b>	<b>Formato</b>	<b>Descripción</b>
Source	SINT INT DINT REAL	inmediato etiqueta	Averigua la tangente de este valor.

Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

**Bloque de funciones**

<b>Operando</b>	<b>Tipo</b>	<b>Formato</b>	<b>Descripción</b>
TAN tag	FBD_MATH_ADVANCE D	Estructura	Estructura de TAN

**Estructura de FBD\_MATH\_ADVANCED**

<b>Parámetro de entrada</b>	<b>Tipo de datos</b>	<b>Descripción</b>
EnableIn	BOOL	Habilita la entrada. Si se borra, la instrucción no se ejecuta y las salidas no se actualizan. Está establecido de forma predeterminada.
Source	REAL	Entrada a la instrucción matemática.

<b>Parámetro de salida</b>	<b>Tipo de datos</b>	<b>Descripción</b>
EnableOut	BOOL	Indica si la instrucción está habilitada.
Dest	REAL	Resultado de la instrucción matemática.

### Descripción

La instrucción TAN calcula la tangente de Source (en radianes) y almacena el resultado en Destination.

La instrucción calcula la tangente de Source y devuelve el resultado REAL.

Puede utilizar TAN como operador en expresiones de escalera y en declaraciones de Texto estructurado.

La instrucción proporciona una mayor precisión con respecto a los controladores antiguos para proporcionar mejores resultados.

### Afectar a las marcas de estado matemático

Controladores	Afectar a la marca de estado matemático
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Condicional, consulte Marcas de estado matemático
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí

### Fallos mayores/menores

No es específico para esta instrucción. Consulte los Atributos comunes para los fallos relacionados con los operandos.

### Ejecución

#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	El controlador calcula la tangente de Source y coloca el resultado en Destination.
Post-escaneado	N/A

**Bloque de funciones**

Condición/estado	Acción realizada
Pre-escaneado	N/A
Tag.EnableIn es falso	El valor de EnableOut se borra a falso.
Tag.EnableIn es verdadero	El valor de EnableOut se establece en verdadero Si el bloque genera un desbordamiento, el valor de EnableOut se borra a falso.
Primer escaneado de instrucción	N/A
Primera ejecución de instrucción	N/A
Post-escaneado	N/A

**Texto estructurado**

Condición/estado	Acción realizada
Pre-escaneado	N/A.
Ejecución normal	El controlador calcula la tangente de Source y coloca el resultado en Destination.
Post-escaneado	N/A

**Ejemplo****Diagrama de escalera****Bloque de funciones****Texto estructurado**

```
result := TAN(value);
```

**Consulte también**

[Instrucciones de trigonometría](#) en la página 764

[Atributos comunes](#) en la página 923

[Marcas de estado matemático](#) en la página 923

[Sintaxis de texto estructurado](#) en la página 955



# Matemática avanzada

## Instrucciones matemáticas avanzadas

Las instrucciones matemáticas avanzadas incluyen las siguientes instrucciones:

### Diagrama de escalera y bloque de funciones



### Texto estructurado



Si desea:	Utilice esta instrucción:
Calcular el logaritmo natural de un valor	LN
Calcular el logaritmo de base 10 de un valor	LOG
Elevar un valor a la potencia de otro valor	XPY

La mezcla de diferentes tipos de datos puede ocasionar pérdidas de exactitud y errores de redondeo y puede hacer que la instrucción necesite más tiempo para ejecutarse. Compruebe el bit S:V para ver si el resultado se ha truncado.

Los tipos de datos que aparecen **negrita** indican tipos de datos óptimos. Una instrucción se ejecutará a la máxima velocidad y la menor cantidad posible de memoria si todos los parámetros emplean el mismo tipo de datos óptimo, normalmente DINT o REAL.

Una instrucción de matemática avanzada ejecuta la instrucción una vez cada vez que se escanee la instrucción y tanto tiempo como sea verdadera la condición de entrada de reglón. Si desea que se evalúe la instrucción solo una vez, utilice la instrucción ONS para desencadenar la instrucción matemática.

### Consulte también

[Matriz \(Archivo\)/Instrucciones misceláneas](#) en la página 527

[Instrucciones de conversión ASCII](#) en la página 892

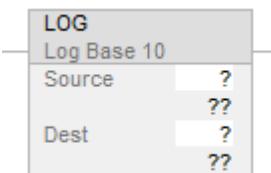
## Logaritmo de base 10 (LOG)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

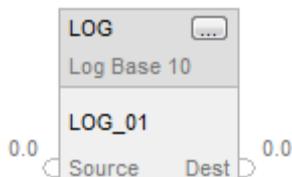
La instrucción LOG coge el logaritmo de base 10 de Source y almacena el resultado en Destination.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones



#### Texto estructurado

```
dest := LOG(source);
```

#### Operando

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción. Consulte *Conversión de datos*.

#### Diagrama de escalera

Operando	Tipo	Formato	Descripción
Source	SINT INT DINT REAL	Inmediato etiqueta	Averiguar el logaritmo de este valor.

Destination	SINT INT DINT REAL	etiqueta	Etiqueta para almacenar el resultado
-------------	-----------------------------	----------	--------------------------------------

### Texto estructurado

Se utiliza LOG como una función. Esta función calcula el logaritmo de Source y almacena el resultado en dest.

Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

### Bloque de funciones

Operando	Tipo	Formato	Descripción
LOG tag	FBD_MATH_ADVANCE D	Estructura	Estructura de LOG

### Estructura de FBD\_MATH\_ADVANCED

Parámetro de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si se borra, la instrucción no se ejecuta y las salidas no se actualizan. Está establecido de forma predeterminada.
Source	REAL	Entrada a la instrucción matemática.

Parámetro de salida	Tipo de datos	Descripción
EnableOut	BOOL	Habilita la salida.
Dest	REAL	Resultado de la instrucción matemática. Se establecen marcas de estado matemático para esta salida.

### Descripción

La instrucción LOG coge el logaritmo de base 10 de Source y almacena el resultado en Destination. El valor de Source debe ser mayor que cero o se generará un fallo menor.

Source	Destination
No es un número Número negativo Infinito negativo,	No es un número, se produce un fallo menor de desbordamiento
Cero Número negativo Número positivo	Infinito negativo, se produce un fallo menor de desbordamiento
Número positivo	Resultados normales
Infinito positivo	Infinito positivo, se produce un fallo menor de desbordamiento

### Afectar a las marcas de estado matemático

Controladores	Afectar a la marca de estado matemático
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Condicional, consulte Marcas de estado matemático.
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí

### Fallos mayores/menores

No es específico para esta instrucción. Consulte los *Atributos comunes* para fallos relacionados con el operando.

### Ejecución

#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A.
La condición de entrada de reglón es falsa	N/A.
La condición de entrada de reglón es verdadera	El controlador calcula el logaritmo natural de Source y coloca el resultado en Destination.
Post-escaneado	N/A.

### Bloque de funciones

Condición/estado	Acción realizada
Pre-escaneado	N/A
Tag.EnableIn es falso	El valor de EnableOut se borra a falso.

Tag.EnableIn es verdadero	El valor de EnableOut se establece en verdadero Si el bloque genera un desbordamiento, el valor de EnableOut se borra a falso.
Primer escaneado de instrucción	N/A
Primera ejecución de instrucción	N/A
Post-escaneado	N/A

### Texto estructurado

Condición/estado	Acción realizada
Pre-escaneado	N/A.
Ejecución normal	Consultar La condición de entrada de reglón es verdadera en la tabla Diagrama de escalera.
Post-escaneado	N/A.

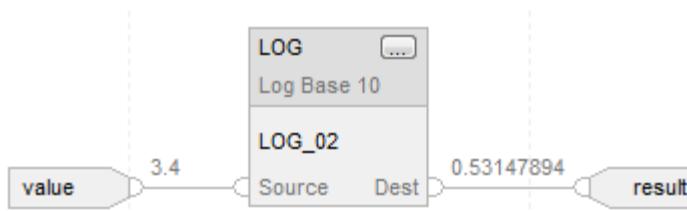
### Ejemplo

Calcular el logaritmo del valor y colocar el resultado en result.

### Diagrama de escalera



### Bloque de funciones



### Texto estructurado

result := LOG(value);

### Consulte también

[Atributos comunes](#) en la página 923

[Instrucciones matemáticas avanzadas](#) en la página 791

[Marcas de estado matemático](#) en la página 923

[Conversiones de datos](#) en la página 927

[Sintaxis de texto estructurado](#) en la página 955

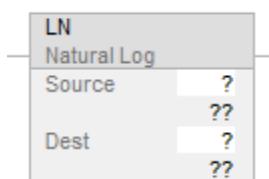
## Logaritmo natural (LN)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

La instrucción LN coge el logaritmo natural del valor de Source y almacena el resultado en Destination.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones



### Texto estructurado

```
dest := LN(source);
```

### Operandos

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción. Consulte *Conversión de datos*.

### Diagrama de escalera

Operando	Tipo	Formato	Descripción
Source	SINT INT DINT REAL	Inmediato etiqueta	Averiguar el logaritmo natural de este valor
Destination	SINT INT DINT REAL	etiqueta	Etiqueta para almacenar el resultado

### Texto estructurado

Se utiliza LN como una función. Esta función calcula el logaritmo natural de Source y almacena el resultado en dest.

Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

### Bloque de funciones

Parámetro de salida	Tipo de datos	Descripción
EnableOut	BOOL	Habilita la salida.
Dest	REAL	Resultado de la instrucción matemática. Se establecen marcas de estado matemático para esta salida.

### Estructura de FBD\_MATH\_ADVANCED

Parámetro de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si se borra, la instrucción no se ejecuta y las salidas no se actualizan. Está establecido de forma predeterminada.
Source	REAL	Entrada a la instrucción matemática.

### Descripción

La instrucción LN coge el logaritmo natural del valor de Source y almacena el resultado en Destination. El valor de Source debe ser mayor que cero o se generará un fallo menor.

En la siguiente tabla se ilustran los casos especiales para valores de origen de punto flotante.

Source	Destination
No es un número Número negativo Infinito negativo,	No es un número, se produce un fallo menor de desbordamiento
Cero Número negativo Número positivo	Infinito negativo, se produce un fallo menor de desbordamiento
Infinito positivo	Infinito positivo, se produce un fallo menor de desbordamiento

### Afectar a las marcas de estado matemático

Controladores	Afectar a las marcas de estado matemático
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Condicional, consulte <i>Marcas de estado matemático</i> .
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí

### Fallos mayores/menores

Consulte los *Atributos comunes* para fallos relacionados con el operando.

### Ejecución

#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	El controlador calcula el logaritmo natural de Source y coloca el resultado en Destination.
Post-escaneado	N/A

#### Bloque de funciones

Condición/estado	Acción realizada
Pre-escaneado	N/A
Tag.EnableIn es falso	El valor de EnableOut se borra a falso.

Tag.EnableIn es verdadero	El valor de EnableOut se establece en verdadero Si el bloque genera un desbordamiento, el valor de EnableOut se borra a falso.
Primer escaneado de instrucción	N/A
Primera ejecución de instrucción	N/A
Post-escaneado	N/A

### Texto estructurado

Condición/estado	Acción realizada
Pre-escaneado	N/A.
Ejecución normal	Consultar La condición de entrada de reglón es verdadera en la tabla Diagrama de escalera.
Post-escaneado	N/A.

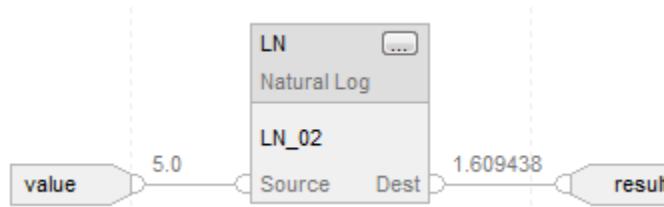
### Ejemplo

Calcular el logaritmo natural del valor y colocar el resultado en result.

### Diagrama de escalera



### Bloque de funciones



### Texto estructurado

result := LN(value);

### Consulte también

[Instrucciones matemáticas avanzadas](#) en la página 791

[Atributos comunes](#) en la página 923

[Marcas de estado matemático](#) en la página 923

[Conversiones de datos](#) en la página 927

[Sintaxis de texto estructurado](#) en la página 955

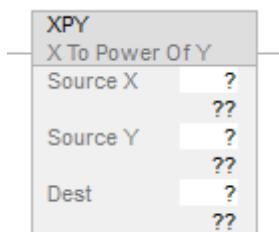
## X a la potencia de Y (XPY)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

La instrucción XPY eleva Source A (X) a la potencia de Source B (Y) y almacena el resultado en Destination.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones



### Texto estructurado

```
dest := sourceX ** sourceY;
```

### Operando

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción. Consulte *Conversión de datos*.

### Diagrama de escalera

Operando	Tipo	Formato	Descripción
Source X	SINT INT DINT REAL	inmediato etiqueta	valor que se va a elevar
Source Y	SINT INT DINT REAL	inmediato etiqueta	exponente
Dest	SINT INT DINT REAL	etiqueta	etiqueta para almacenar el resultado

### Texto estructurado

Se utiliza dos signos de multiplicación adyacentes “\*\*” como operador dentro de una expresión.

Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

### Bloque de funciones

Operando	Tipo	Formato	Descripción
XPY tag	FBD_MATH	Estructura	Estructura de XPY

### Estructura de FBD\_MATH

Parámetro de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si se borra, la instrucción no se ejecuta y las salidas no se actualizan. Está establecido de forma predeterminada.
SourceA	REAL	Valor de base.
SourceB	REAL	Exponente.
Parámetro de salida	Tipo de datos	Descripción
EnableOut	BOOL	Habilita la salida.
Dest	REAL	Resultado de la instrucción matemática. Se establecen marcas de estado matemático para esta salida.

### Descripción

La instrucción XPY eleva Source A (X) a la potencia de Source B (Y) y almacena el resultado en Destination.

Si Source A (X) es un valor negativo, Source B (Y) debe ser un valor no fraccionario o se generará un fallo menor.

En el caso de los controladores CompactLogix 5370 y ControlLogix 5570, si la base es negativa y el exponente es real, se utiliza el valor absoluto de la base.

### Afectar a las marcas de estado matemático

Controladores	Afectar a las marcas de estado matemático
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Condicional, consulte <i>Marcas de estado matemático</i> .
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí

### Condiciones de fallo

Controladores	Se producirá un fallo mayor si:	Tipo de fallo	Código del fallo
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	N/A	N/A	N/A
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Source X es negativo y Source Y no es un valor entero	4	4

No es específico para esta instrucción. Consulte los *Atributos comunes* para fallos relacionados con el operando.

### Ejecución

#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A.

La condición de entrada de reglón es falsa.	N/A.
La condición de entrada de reglón es verdadera.	El controlador eleva Source X a la potencia de Source Y y coloca el resultado en Destination.
Post-escaneado	N/A.

### Bloque de funciones

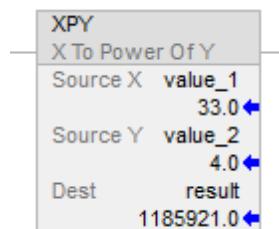
Condición/estado	Acción realizada
Pre-escaneado	N/A
Tag.EnableIn es falso	El valor de EnableOut se borra a falso.
Tag.EnableIn es verdadero	El valor de EnableOut se establece en verdadero Si el bloque genera un desbordamiento, el valor de EnableOut se borra a falso.
Primer escaneado de instrucción	N/A
Primera ejecución de instrucción	N/A
Post-escaneado	N/A

### Texto estructurado

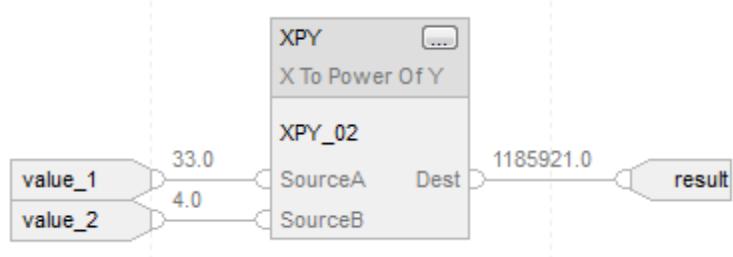
Condición/estado	Acción realizada
Pre-escaneado	N/A.
Ejecución normal	Consultar La condición de entrada de reglón es verdadera.
Post-escaneado	N/A.

### Ejemplo

#### Diagrama de escalera



## Bloque de funciones



### Texto estructurado

```
result := (value_1 ** value_2);
```

### Consulte también

[Sintaxis de texto estructurado](#) en la página 955

[Instrucciones matemáticas avanzadas](#) en la página 791

[Marcas de estado matemático](#) en la página 923

[Atributos comunes](#) en la página 923

# Instrucciones de conversión matemática

## Instrucciones de conversión matemática

Las instrucciones de conversión matemática convierte los valores.

### Instrucciones disponibles

#### Diagrama de escalera y bloque de funciones



#### Texto estructurado



Si desea	Utilice esta instrucción
Convertir radianes en grados.	DEG
Convertir grados en radianes.	RAD
Convertir un valor entero en un valor BCD.	TOD
Convertir un valor BCD en un valor entero.	FRD
Quitar la parte fraccional de un valor.	TRN

Se pueden mezclar data tipos de datos, pero es posible que se pierda precisión, que haya errores de redondeo y que la instrucción tarde más tiempo en ejecutarse. Compruebe el bit S:V para ver si el resultado se ha truncado.

Los tipos de datos que aparecen en **negrita** indican tipos de datos óptimos. Una instrucción se ejecutará a la máxima velocidad y la menor cantidad posible de memoria si todos los parámetros emplean el mismo tipo de datos óptimo, normalmente DINT o REAL.

Una instrucción de conversión matemática ejecuta la instrucción una vez cada vez que se escanee la instrucción y tanto tiempo como sea verdadera la condición de

entrada de peldaño. Si desea que se evalúe la instrucción solo una vez, utilice la instrucción ONS para desencadenar la instrucción de conversión.

### Consulte también

[Instrucciones de cálculo/matemáticas](#) en la página 425

[Comparar instrucciones](#) en la página 369

[Instrucciones de bit](#) en la página 143

[Instrucciones de cadena ASCII](#) en la página 871

[Instrucciones de conversión ASCII](#) en la página 892

## Convertir en BCD (TOD)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

La instrucción TOD convierte un valor decimal ( $0 \leq \text{Source} \leq 99,999,999$ ) en un valor de BCD y almacena el resultado en Destination.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones



#### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

## Operando

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción. Consulte *Conversión de datos*.

## Diagrama de escalera

Operando	Tipo	Formato	Descripción
Source	SINT INT DINT	Inmediato etiqueta	Valor a convertir en BCD $0 \leq \text{Source} \leq 99,999,999$
Destination	SINT INT DINT	etiqueta	etiqueta para almacenar el resultado

## Bloque de funciones

Operando	Tipo	Formato	Descripción
TOD tag	FBD_CONVERT	Estructura	Estructura de TOD

## Estructura de FBD\_CONVERT

Parámetro de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si se borra, la instrucción no se ejecuta y las salidas no se actualizan. Está establecido de forma predeterminada.
Source	DINT	Entrada a la instrucción de conversión. Válido = cualquier entero

Parámetro de salida	Tipo de datos	Descripción
EnableOut	BOOL	Habilita la salida.
Dest	DINT	Resultado de la instrucción de conversión. Se establecen marcas de estado matemático para esta salida.

## Descripción

BCD es el sistema de números decimales codificados binariamente que expresa dígitos decimales individuales (0-9) en una anotación binaria de 4 bits.

Source	Destination	Tipo de destino
Source negativo < 0	0	
Source > 99,999,999	16#9999_9999	DINT
Source > 99,999,999	16#9999	INT
Source > 99,999,999	16#99	SINT

### Afectar a las marcas de estado matemático

Controladores	Afectar a las marcas de estado matemático
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Condicional, consulte <i>Marcas de estado matemático</i> .
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí

### Fallos mayores/menores

No es específico para esta instrucción. Consulte *Atributos comunes* para los fallos relacionados con los operandos

### Ejecución

#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A.
La condición de entrada de reglón es falsa	N/A.
La condición de entrada de reglón es verdadera	El controlador convierte Source en BCD y coloca el resultado en Destination.
Post-escaneado	N/A.

#### Bloque de funciones

Condición/estado	Acción realizada
Pre-escaneado	N/A
Tag.EnableIn es falso	El valor de EnableOut se borra a falso.
Tag.EnableIn es verdadero	El valor de EnableOut se establece en verdadero Si el bloque genera un desbordamiento, el valor de EnableOut se borra a falso.

Primer escaneado de instrucción	N/A
Primera ejecución de instrucción	N/A
Post-escaneado	N/A

### Ejemplo

#### Ejemplo 1

La instrucción TOD convierte value\_1 en un valor de BCD y coloca el resultado en result\_a.

#### Diagrama de escalera



#### Bloque de funciones

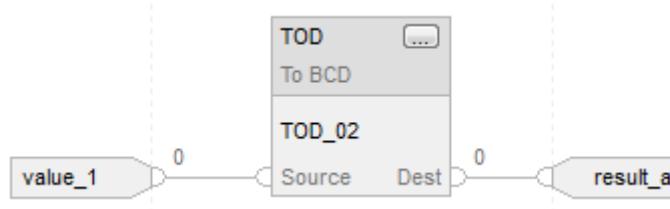
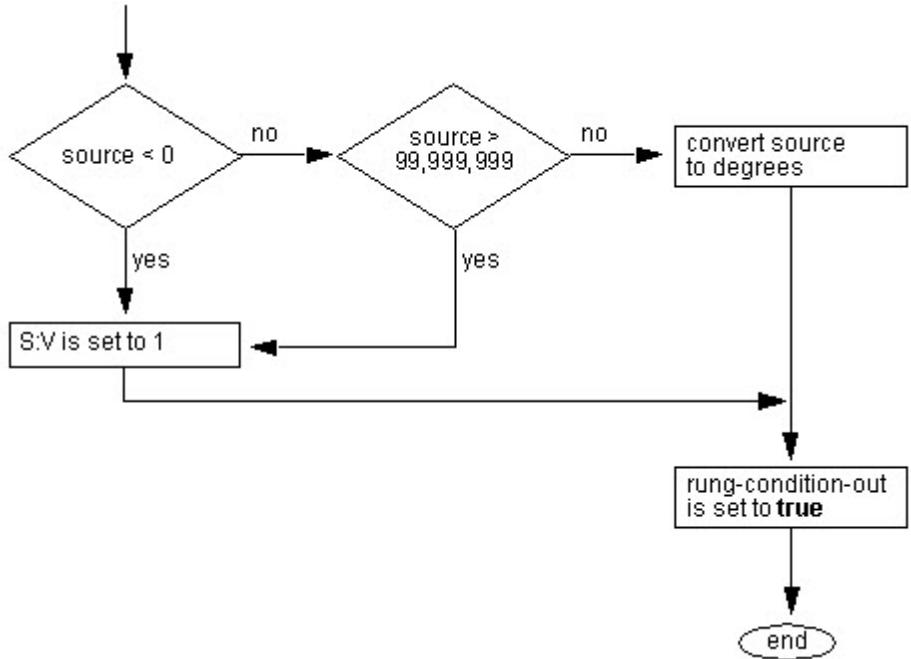


Diagrama de flujo de TOD (verdadero)

**Consulte también**[Instrucciones de cálculo](#) en la página 425[Atributos comunes](#) en la página 923[Marcas de estado matemático](#) en la página 923[Conversiones de datos](#) en la página 927**Convertir en entero (FRD)**

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

La instrucción FRD convierte un valor de BCD (Source) en un valor decimal y almacena el resultado en Destination.

### Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones



### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

### Operандos

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción. Consulte *Conversión de datos*.

### Diagrama de escalera

Operando	Tipo	Formato	Descripción
Source	SINT INT DINT	Inmediato etiqueta	Valor a convertir en decimal
Destination	SINT INT DINT	etiqueta	etiqueta para almacenar el resultado

### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

### Bloque de funciones

Operando	Tipo	Formato	Descripción
FRD tag	FBD_CONVERT	Estructura	Estructura de FRD

**Estructura de FBD\_CONVERT**

Parámetro de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si el valor es falso, la instrucción no se ejecuta y las salidas no se actualizan. El valor predeterminado es verdadero.
Source	DINT	Entrada a la instrucción de conversión. Válido = cualquier entero

Parámetros de salida	Tipo de datos	Descripción
EnableOut	BOOL	Indica si la instrucción está habilitada.
Dest	DINT	Resultado de la instrucción de conversión.

**Descripción**

La instrucción FRD convierte un valor de BCD (Source) en un valor decimal y almacena el resultado en Destination

**Afectar a las marcas de estado matemático**

Controladores	Afectar a las marcas de estado matemático
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Condicional, consulte <i>Marcas de estado matemático</i> .
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí

**Fallos mayores/menores**

No es específico para esta instrucción. Consulte *Atributos comunes* para los fallos relacionados con los operandos

## Ejecución

### Diagrama de escalera

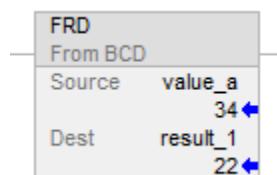
Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	El controlador convierte Source en un valor decimal y coloca el resultado en Destination.
Post-escaneado	N/A

### Bloque de funciones

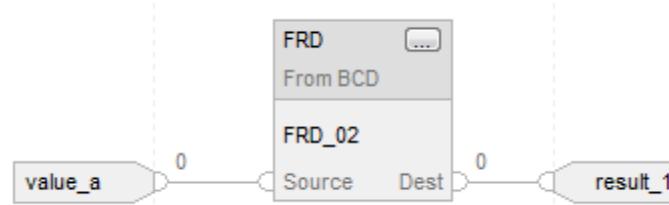
Condición/estado	Acción realizada
Pre-escaneado	N/A
Tag.EnableIn es falso	El valor de EnableOut se borra a falso.
Tag.EnableIn es verdadero	El valor de EnableOut se establece en verdadero Si el bloque genera un desbordamiento, el valor de EnableOut se borra a falso.
Primer escaneado de instrucción	N/A
Primera ejecución de instrucción	N/A
Post-escaneado	N/A

## Ejemplos

### Diagrama de escalera



### Bloque de funciones



### Consulte también

[Instrucciones de cálculo](#) en la página 425

[Atributos comunes](#) en la página 923

[Marcas de estado matemático](#) en la página 923

[Conversiones de datos](#) en la página 927

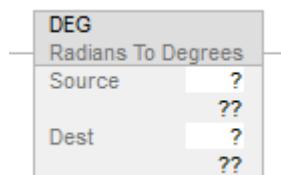
## Grados (DEG)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

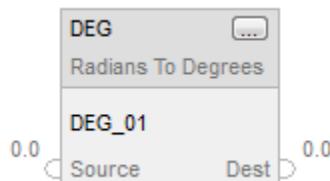
La instrucción DEG convierte Source (en radianes) en grados y almacena el resultado en Destination.

### Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones



### Texto estructurado

```
dest := DEG(source);
```

### Operandos

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción. Consulte *Conversión de datos*.

### Diagrama de escalera

Operando	Tipo	Formato	Descripción
Source	SINT INT DINT REAL	Inmediato etiqueta	Valor a convertir en grados
Destination	SINT INT DINT REAL	etiqueta	etiqueta para almacenar el resultado

### Texto estructurado

Se utiliza DEG como una función. Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

### Bloque de funciones

Operando	Tipo	Formato	Descripción
DEG tag	FBD_MATH_ADVANCED	Estructura	Estructura de DEG

### Estructura de FBD\_MATH\_ADVANCED

Parámetro de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si el valor es falso, la instrucción no se ejecuta y las salidas no se actualizan. El valor predeterminado es verdadero.
Source	REAL	Entrada a la instrucción de conversión.

Parámetro de salida	Tipo de datos	Descripción
EnableOut	BOOL	Indica si la instrucción está habilitada.
Dest	REAL	Resultado de la instrucción de conversión.

### Descripción

La instrucción DEG usa este algoritmo:

$$\text{Source}^*180/\pi = \text{Source}^*57.29578$$

**Afectar a las marcas de estado matemático**

Controladores	Marcas de estado matemático afectadas
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Condicional, consulte <i>Marcas de estado matemático</i> .
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí

**Fallos mayores/menores**

Ocurrirá un fallo menor si:	Tipo de fallo	Código de fallo
Se detecta un desbordamiento	4	4

Consulte *Atributos comunes* para los fallos relacionados con los operandos

**Ejecución****Diagrama de escalera**

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	El controlador convierte Source en radianes y coloca el resultado en Destination.
Post-escaneado	N/A

**Bloque de funciones**

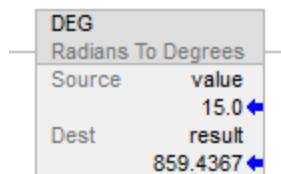
Condición/estado	Acción realizada
Pre-escaneado	N/A
Tag.EnableIn es falso	El valor de EnableOut se borra a falso.
Tag.EnableIn es verdadero	El valor de EnableOut se establece en verdadero Si el bloque genera un desbordamiento, el valor de EnableOut se borra a falso.
Primer escaneado de instrucción	N/A
Primera ejecución de instrucción	N/A
Post-escaneado	N/A

### Texto estructurado

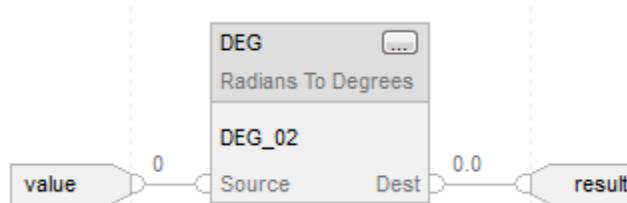
Condición/estado	Acción realizada
Pre-escaneado	N/A
Ejecución normal	Consultar La condición de entrada de reglón es verdadera en la tabla Diagrama de escalera.
Post-escaneado	N/A

### Ejemplo

#### Diagrama de escalera



#### Bloque de funciones



### Texto estructurado

```
result := DEG(value);
```

### Consulte también

[Instrucciones matemáticas avanzadas](#) en la página 791

[Atributos comunes](#) en la página 923

[Marcas de estado matemático](#) en la página 923

[Sintaxis de texto estructurado](#) en la página 955

[Conversiones de datos](#) en la página 927

## Radianes (RAD)

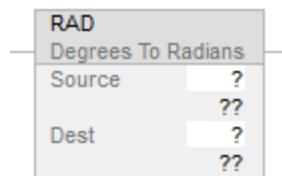
Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact

GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

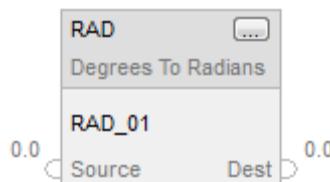
La instrucción RAD convierte Source (en grados) en radianes y almacena el resultado en Destination.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones



#### Texto estructurado

```
dest := RAD(source);
```

#### Operandos

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción. Consulte *Conversion de datos*.

#### Diagrama de escalera

Operando	Tipo	Formato	Descripción
Source	SINT INT DINT REAL	Inmediato etiqueta	Valor a convertir en radianes
Destination	SINT INT DINT REAL	etiqueta	etiqueta para almacenar el resultado

### Texto estructurado

Se utiliza RAD como una función. Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

### Bloque de funciones

Operando	Tipo	Formato	Descripción
RAD tag	FBD_MATH_ADVANCED	estructura	Estructura de FRD

### Estructura de FBD\_MATH\_ADVANCED

Parámetro de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si el valor es falso, la instrucción no se ejecuta y las salidas no se actualizan. El valor predeterminado es verdadero.
Source	REAL	Entrada a la instrucción de conversión.

Parámetro de salida	Tipo de datos	Descripción
EnableOut	BOOL	Habilita la salida.
Dest	REAL	Resultado de la instrucción de conversión.

### Afectar a las marcas de estado matemático

Controladores	Marcas de estado matemático afectadas
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Condicional, consulte <i>Marcas de estado matemático</i> .
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí

### Condiciones de fallo

No es específico para esta instrucción. Consulte *Atributos comunes* para los fallos relacionados con los operandos

## Ejecución

### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	El controlador convierte Source en radianes y coloca el resultado en Destination.
Post-escaneado	N/A

### Bloque de funciones

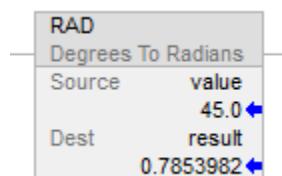
Condición/estado	Acción realizada
Pre-escaneado	N/A
Tag.EnableIn es falso	El valor de EnableOut se borra a falso.
Tag.EnableIn es verdadero	El valor de EnableOut se establece en verdadero Si el bloque genera un desbordamiento, el valor de EnableOut se borra a falso.
Primer escaneado de instrucción	N/A
Primera ejecución de instrucción	N/A
Post-escaneado	N/A

### Texto estructurado

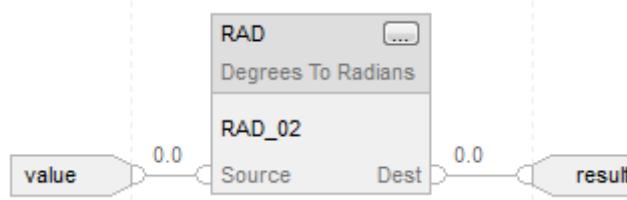
Condición/estado	Acción realizada
Pre-escaneado	N/A
Ejecución normal	Consultar La condición de entrada de reglón es verdadera en la tabla Diagrama de escalera.
Post-escaneado	N/A

## Ejemplo

### Diagrama de escalera



## Bloque de funciones



### Texto estructurado

```
result := RAD(value);
```

### Consulte también

[Sintaxis de texto estructurado](#) en la página 955

[Atributos comunes](#) en la página 923

[Marcas de estado matemático](#) en la página 923

[Conversiones de datos](#) en la página 927

[Instrucciones matemáticas avanzadas](#) en la página 791

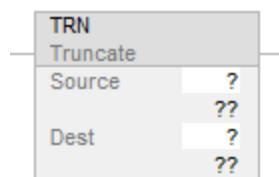
## Truncar (TRN)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580. Las diferencias de controladores se indican cuando corresponda.

La instrucción TRN elimina (trunca) la parte fraccionaria de Source y almacena el resultado en Destination.

### Idiomas disponibles

#### Diagrama de escalera



## Bloque de funciones



### Texto estructurado

```
dest := TRUNC(source);
```

### Operандos

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción. Consulte *Conversión de datos*.

El diagrama de escalera y el Bloque de funciones utilizan TRN como una instrucción. Cuando se utiliza la instrucción TRN en el Diagrama de escalera, el operando Source solo acepta la etiqueta REAL o valores inmediatos, mientras que el destino puede ser REAL, DINT, SINT o INT. Pero en el caso del Bloque de funciones, el destino solo puede ser DINT.

El texto estructurado utiliza TRUNC como operador. Para el operador TRUNC, el operando Source puede aceptar REAL, SINT, INT y DINT. Pero el destino solo pueden ser DINT.

Cuando se utiliza TRUNC dentro de una instrucción de expresión, como CPT, utiliza TRUNC como operador. El operando Source puede ser de cualquiera de los tipos de enteros, como SINT, INT, DINT, así como REAL.

### Diagrama de escalera

Operando	Tipo	Formato	Descripción
Source*	REAL	inmediato etiqueta	Valor a truncar
Destination	SINT INT DINT REAL	etiqueta	etiqueta para almacenar el resultado
Conversión de datos: Las etiquetas SINT e INT tienen extensión de signo.			

### Bloque de funciones

Operando	Tipo	Formato	Descripción
TRN tag	FBD_TRUNCATE	Estructura	Estructura de TRN

### Estructura FBD\_TRUNCATE

Parámetro de entrada	Tipo de datos	Descripción
EnableIn	BOOL	Habilita la entrada. Si el valor es falso, la instrucción no se ejecuta y las salidas no se actualizan. El valor predeterminado es verdadero.
Source	REAL	Entrada a la instrucción de conversión. La entrada también coge DINT, SINT e INT. No obstante, el tipo de entero se convertirá primero en el tipo REAL. Al convertir SINT o INT en REAL, no se produce pérdida de precisión de datos. Sin embargo, al convertir DINT en REAL, podría perderse precisión de datos. Ambos tipos de datos almacenan datos en 32 bits, pero el tipo REAL utiliza algunos de sus 32 bits para almacenar el valor del exponente. Si se pierde precisión, el controlador la toma de la porción menos significativa del DINT.

Parámetro de salida	Tipo de datos	Descripción
EnableOut	BOOL	Habilita la salida. Se borra a falso, si Dest produce un desbordamiento. De lo contrario, se establece en verdadero.
Dest	DINT	Resultado de la instrucción de conversión.

### Texto estructurado

Se utiliza TRUNC como una función. Esta función trunca el origen y devuelve un resultado entero.

Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

Operando	Tipo	Formato	Descripción
Source	REAL DINT SINT INT	inmediato etiqueta	Entrada a la instrucción de conversión.

### Descripción

Truncar un valor, no lo redondea, sino que, en su lugar, la parte no fraccionaria permanece igual, independientemente del valor de la parte fraccionaria.

Truncar un número real grande que podría desbordar las matemáticas internas devuelve un valor en lugar de un valor cero.

Puede utilizar TRN como operador en expresiones del diagrama de escalera; puede utilizar TRUNC como operador en declaraciones de Texto estructurado.

### Afectar a las marcas de estado matemático

Controladores	Marcas de estado matemático afectadas
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Condicional, consulte <i>Marcas de estado matemático</i> .
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Sí

### Fallos mayores/menores

No es específico para esta instrucción. Consulte *Atributos comunes* para los fallos relacionados con los operandos

### Ejecución

#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	El reglón se establece en falso.
La condición de entrada de reglón es falsa	N/A.
La condición de entrada de reglón es verdadera	El controlador elimina la parte fraccionaria de Source y coloca el resultado en Destination. La condición de entrada de reglón se establece en verdadera.
Post-escaneado	El reglón se establece en falso.

#### Bloque de funciones

Condición/estado	Acción realizada
Pre-escaneado	N/A
Tag.EnableIn es falso	El valor de EnableOut se borra a falso.
Tag.EnableIn es verdadero	El valor de EnableOut se establece en verdadero Si el bloque genera un desbordamiento, el valor de EnableOut se borra a falso.

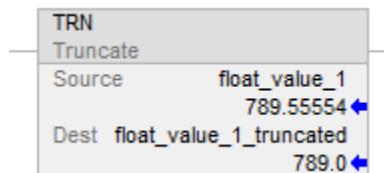
Primer escaneado de instrucción	N/A
Primera ejecución de instrucción	N/A
Post-escaneado	N/A

### Texto estructurado

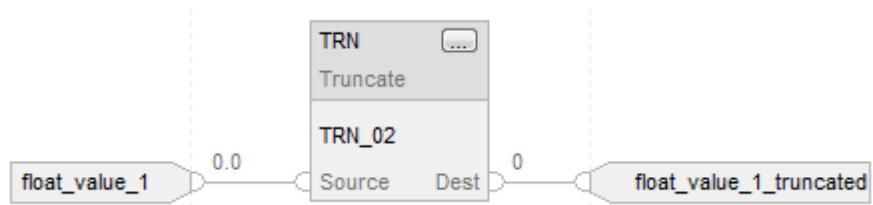
Condición/estado	Acción realizada
Pre-escaneado	Consultar Pre-escaneado en Diagrama de escalera.
Ejecución normal	Consultar La condición de entrada de reglón se establece en verdadera en el Diagrama de escalera.
Post-escaneado	Consultar Post-escaneado en la tabla Diagrama de escalera.

### Ejemplo

#### Diagrama de escalera



#### Bloque de funciones



### Texto estructurado

```
float_value_1_truncated := TRUNC(float_value_1);
```

### Consulte también

[Sintaxis de texto estructurado](#) en la página 955

[Instrucciones matemáticas avanzadas](#) en la página 791

[Atributos comunes](#) en la página 923

[Marcas de estado matemático](#) en la página 923

[Conversiones de datos](#) en la página 927

# Instrucciones de puerto serial ASCII

## Instrucciones de puerto serial ASCII

Utilice las instrucciones ASCII de puerto serial para leer y escribir los caracteres ASCII.

**Importante:** Para utilizar las instrucciones ASCII de puerto serial tiene que configurar el puerto serie del controlador. Para obtener más información, consulte LOGIX 5000 Controller Common Procedures manual (publicación 1756-PM001) para obtener más información

**Cons  
ejo:** Las instrucciones del puerto serial ASCII (AWT, AWA, ARD, ARL, ABL, ACB, AHL, ACL) no están disponibles para los proyectos que utilicen controladores sin puertos seriales.

### Instrucciones disponibles

#### Diagrama de escalera y Texto estructurado

<a href="#">ABL</a>	<a href="#">ACB</a>	<a href="#">ACL</a>	<a href="#">AHL</a>	<a href="#">ARD</a>	<a href="#">ARL</a>	<a href="#">AWA</a>	<a href="#">AWT</a>
---------------------	---------------------	---------------------	---------------------	---------------------	---------------------	---------------------	---------------------

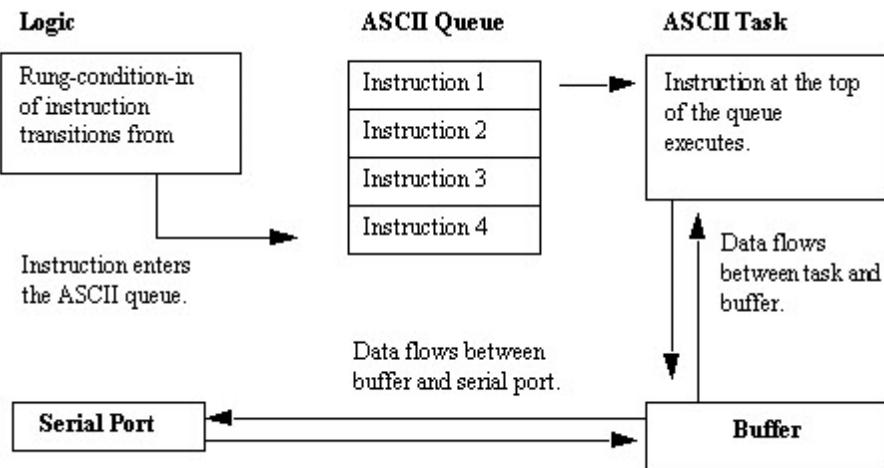
#### Bloque de funciones

No disponible

Si desea:	Utilice esta instrucción:
Verificar si hay datos que contengan caracteres de terminación	ABL
Verificar el número requerido de caracteres antes de leer el búfer	ACB
Borrar el búfer. Por ejemplo, borrar datos antiguos del búfer al momento de la puesta en marcha o sincronizar el búfer con un dispositivo. Limpiar las instrucciones de puerto serial ASCII que se están ejecutando o que se encuentran en cola.	ACL

Obtener el estado de las líneas de control del puerto serial. Por ejemplo, hacer que el módem se cuelgue. Activar o desactivar la señal DTR Activar o desactivar la señal RTS	AHL
Leer un número fijo de caracteres. Por ejemplo, leer los datos de un dispositivo que envía el mismo número de caracteres en cada transmisión	ARD
Leer un número variable de caracteres incluyendo el primer conjunto de caracteres de terminación. Por ejemplo, leer los datos de un dispositivo que envía un número variable de caracteres en cada transmisión.	ARL
Enviar caracteres y añadir automáticamente uno o dos caracteres adicionales para marcar el final de los datos. Por ejemplo, enviar un mensaje que siempre usen los mismos caracteres de terminación.	AWA
Enviar caracteres. Por ejemplo, enviar mensajes que usan una variedad de caracteres de terminación.	AWT

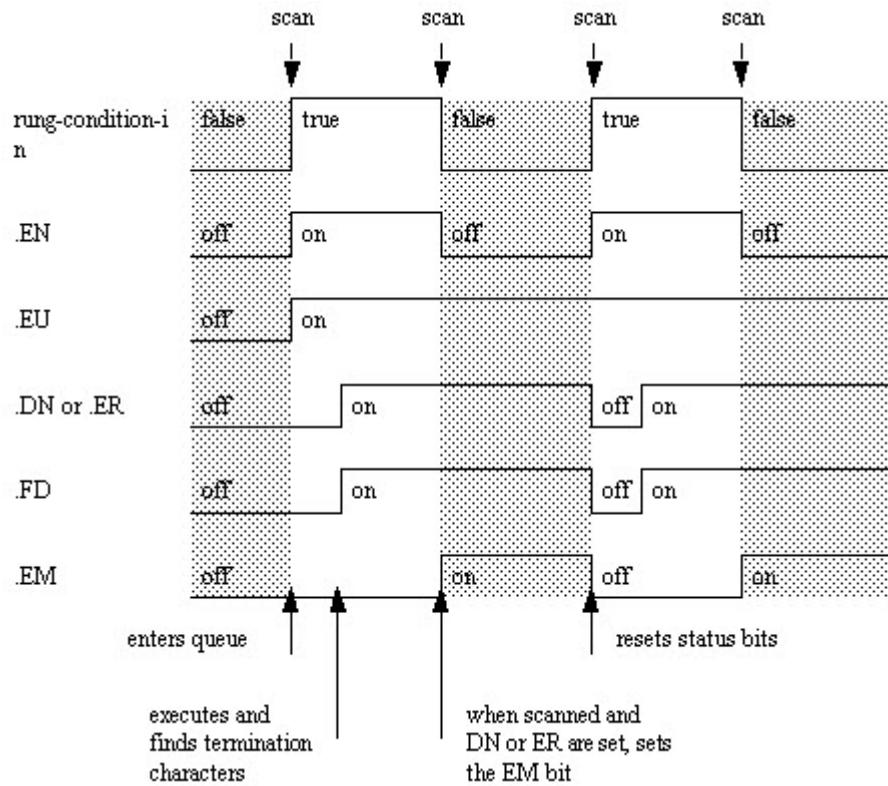
Las instrucciones de puerto serial ASCII se ejecutan de manera asincrónica con el escaneado de lógica:



Cada instrucción ASCII, excepto la instrucción ACL, utilizan una estructura SERIAL\_PORT\_CONTROL. El operando SerialPort Control:

- controla la ejecución de la instrucción
  - proporciona información de estado sobre la instrucción
- Las instrucciones ASCII se ejecutan de manera asincrónica con el escaneado de lógica:

Los bits del operando SerialPort Control ofrecen información del estado:



### Consulte también

[Tipos de cadena](#) en la página 867

[Códigos de error ASCII](#) en la página 868

## Caracteres ASCII en el búfer (ACB)

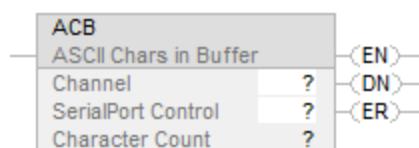
Esta instrucción es compatible con los controladores Studio 5000 Logix Emulate solo.

La instrucción ACB cuenta los caracteres en el búfer.

**Cons** Las instrucciones de puerto serial ASCII (AWT, AWA, ejo: ARD, ARL, ABL, ACB, AHL y ACL) no están disponibles para controladores que no tengan puertos seriales.

### Idiomas disponibles

#### Diagrama de escalera



### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

### Texto estructurado

ACB(Channel,SerialPortControl);

### Operandos

#### Diagrama de escalera

Operando	Tipo	Formato	Descripción
Channel	DINT	inmediato etiqueta	0
SerialPort Control	SERIAL_PORT_CONTROL	etiqueta	Etiqueta que controla la operación
Character Count	DINT	inmediato	0 Durante la ejecución, muestra el número de caracteres en el búfer, incluyendo el primer conjunto de caracteres de terminación.

### Texto estructurado

Operando	Tipo	Formato	Descripción
Channel	DINT	inmediato etiqueta	0
SerialPort Control	SERIAL_PORT_CONTROL	etiqueta	Etiqueta que controla la operación
Character Count	DINT	inmediato	0 Durante la ejecución, muestra el número de caracteres en el búfer, incluyendo el primer conjunto de caracteres de terminación.

Puede especificar el valor de Character Count accediendo al miembro .POS de la estructura SERIAL\_PORT\_CONTROL en lugar de incluir el valor en la lista de operandos.

Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

### Estructura de SERIAL\_PORT\_CONTROL

Mnemónico	Tipo de datos	Descripción
.EN	BOOL	El bit de habilitación indica que la instrucción está habilitada.
.EU	BOOL	La cola indica que la instrucción entró en la cola ASCII.
.DN	BOOL	El bit de efectuado indica cuándo se ha completado la instrucción, pero es asíncrona con respecto al escaneado de la lógica.
.RN	BOOL	El bit de ejecución indica que la instrucción se está ejecutando.
.EM	BOOL	El bit de vacío indica que se ha completado la instrucción, pero es síncrona con respecto al escaneado de la lógica.
.ER	BOOL	El bit de error indica cuándo falla la instrucción (errores).
.FD	BOOL	El bit de encontrado indica que la instrucción encontró un carácter.
.POS	DINT	La posición determina el número de caracteres en el búfer hasta e incluso el primer conjunto de caracteres de terminación.
.ERROR	DINT	El error contiene un valor hexadecimal que identifica la causa de un error.

#### Descripción

La instrucción ACB cuenta los caracteres en el búfer.

Para programar la instrucción ACB, siga estas pautas:

- Configure el puerto serial del controlador para el modo Usuario.

Se trata de una instrucción de transición:

- En el diagrama de escalera, cambie EnableIn de borrado a establecido cada vez que deba ejecutarse la instrucción.
- En el texto estructurado, condicione la instrucción para que solo se ejecute en una transición.

#### Marcas de estado matemático

No

#### Condiciones de fallo

No es específico para esta instrucción. Consulte los *Atributos comunes* para fallos relacionados con el operando.

## Ejecución

### Diagrama de escalera

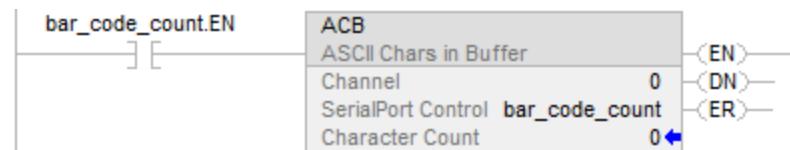
Condición	Acción del Diagrama de escalera
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	La instrucción se ejecuta cuando el valor de EnableIn pasa de borrado a establecido.
Post-escaneado	N/A

### Texto estructurado

Condición	Acción de texto estructurado
Pre-escaneado	N/A
Ejecución normal	La instrucción se ejecuta cuando el valor de EnableIn pasa de borrado a establecido.
Post-escaneado	N/A

## Ejemplo

### Diagrama de escalera



### Texto estructurado

```
ACB(0,bar_code_count);
```

### Consulte también

[Instrucciones de puerto serial ASCII](#) en la página 827

[Sintaxis de texto estructurado](#) en la página 955

[Atributos comunes](#) en la página 923

## Borrar ASCII búfer (ACL)

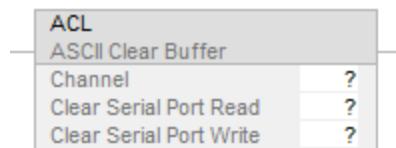
Esta instrucción es compatible con los controladores Studio 5000 Logix Emulate solo.

La instrucción ACL borra inmediatamente el búfer y la cola ASCII.

**Cons** Las instrucciones de puerto serial ASCII (AWT, AWA, ejo: ARD, ARL, ABL, ACB, AHL y ACL) no están disponibles para controladores que no tengan puertos seriales.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

#### Texto estructurado

```
ACL(Channel,ClearSerialPortRead,ClearSerialPortWrite);
```

#### Operandos

#### Diagrama de escalera

Operando	Tipo	Formato	Descripción
Channel	DINT	inmediato etiqueta	0
Clear Serial Port Read	BOOL	inmediato etiqueta	Para borrar el búfer y eliminar las instrucciones ARD y ARL de la cola, introduzca 1.
Clear Serial Port Write	BOOL	inmediato etiqueta	Para borrar las instrucciones AWA y AWT de la cola, introduzca 1.

#### Texto estructurado

Operando	Tipo	Formato	Descripción
Channel	DINT	inmediato etiqueta	0
Clear Serial Port Read	BOOL	inmediato etiqueta	Para borrar el búfer y eliminar las instrucciones ARD y ARL de la cola, introduzca 1.
Clear Serial Port Write	BOOL	inmediato etiqueta	Para borrar las instrucciones AWA y AWT de la cola, introduzca 1.

Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

## Descripción

La instrucción ACL realiza inmediatamente una de las dos acciones siguientes o ambas:

- Borra el búfer de caracteres y borra la cola ASCII de instrucciones de lectura
- Borra la cola ASCII de instrucciones de escritura. Para programar la instrucción ACL, siga estas pautas:

Configure el puerto serial del controlador

Si su aplicación:	Entonces:
Usa instrucciones ARD o ARL	Seleccione el modo Usuario
No usa instrucciones ARD o ARL	Seleccione el modo Sistema o el modo Usuario

Para determinar si una instrucción fue quitada de la cola o se canceló, examine lo siguiente de la instrucción apropiada:

- .ER bit se establece
- el miembro .ERROR es 16#E

## Afectar a las marcas de estado matemático

No

## Condiciones de fallo

No es específico para esta instrucción. Consulte los *Atributos comunes* para fallos relacionados con el operando.

## Ejecución

### Diagrama de escalera

Condición	Acción del Diagrama de escalera
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	La instrucción se ejecuta.
Post-escaneado	N/A

### Texto estructurado

Condición	Acción de texto estructurado
Pre-escaneado	N/A
Ejecución normal	La instrucción borra la instrucción especificada y los búferes.
Post-escaneado	N/A

### Ejemplo

#### Diagrama de escalera



### Texto estructurado

```

IF (osri_1.OutputBit THEN
    ACL(0,0,1);
END_IF;

```

### Consulte también

[Instrucciones de puerto serial ASCII](#) en la página 827

[Prueba ASCII para línea de búfer \(ABL\)](#) en la página 851

[Caracteres ASCII en el búfer \(ACB\)](#) en la página 829

[Líneas de handshake ASCII \(AHL\)](#) en la página 836

[Lectura ASCII \(ARD\)](#) en la página 840

[Lectura ASCII de línea \(ARL\)](#) en la página 845

[Escritura ASCII con anexo \(AWA\)](#) en la página 861

[Escritura ASCII \(AWT\)](#) en la página 855

[Sintaxis de texto estructurado](#) en la página 955

[Atributos comunes](#) en la página 923

[Conversiones de datos](#) en la página 927

## Líneas de handshake ASCII (AHL)

Esta instrucción es compatible con los controladores Studio 5000 Logix Emulate solo.

La instrucción AHL obtiene el estado de las líneas de control y activa o desactiva las señales DTR y RTS.

**Cons ejo:** Las instrucciones de puerto serial ASCII (AWT, AWA, ARD, ARL, ABL, ACB, AHL y ACL) no están disponibles para controladores que no tengan puertos seriales.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

#### Texto estructurado

```
AHL(Channel,ANDMask,ORMask,SerialPortControl);
```

#### Operандos

#### Diagrama de escalera

Operando	Tipo	Formato	Descripción
Channel	DINT	inmediato etiqueta	0
ANDMask	DINT	inmediato etiqueta	Consulte las descripciones
ORMask	DINT	inmediato etiqueta	
SerialPort Control	SERIAL_PORT_CONTROL	etiqueta	Etiqueta que controla la operación
Channel Status (Decimal)	DINT	inmediato	0 Durante la ejecución, muestra el estado de las líneas de control. <b>Para el estado de esta línea de control:</b> <span style="border: 1px solid black; padding: 2px;">Examine este bit:</span>

			CTS	0
			RTS	1
			DSR	2
			DCD	3
			DTR	4
			Recibió el carácter XOFF	5

### Texto estructurado

Operando	Tipo	Formato	Descripción	
Channel	DINT	inmediato etiqueta	0	
ANDMask	DINT	inmediato etiqueta	Consulte las descripciones	
ORMask	DINT	inmediato etiqueta		
SerialPort Control	SERIAL_PORT_CONTROL	etiqueta	Etiqueta que controla la operación	
Channel Status (Decimal)	DINT	inmediato	0 Durante la ejecución, muestra el estado de las líneas de control. <b>Para el estado de esta línea de control:</b>	<b>Examine este bit:</b>
			CTS	0
			RTS	1
			DSR	2
			DCD	3
			DTR	4
			Recibió el carácter XOFF	5

Puede especificar el Channel Status accediendo al miembro .POS de la estructura SERIAL\_PORT\_CONTROL en lugar de incluir el valor en la lista operandos.

Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

### Estructura de SERIAL\_PORT\_CONTROL

Mnemónico	Tipo de datos	Descripción
.EN	BOOL	El bit de habilitación indica que la instrucción está habilitada.
.EU	BOOL	El bit de cola indica que la instrucción entró en la cola ASCII.
.DN	BOOL	El bit de efectuado indica que ha finalizado la instrucción pero que es asíncrono con respecto al escaneado de lógico.

.RN	BOOL	El bit de ejecución indica que la instrucción se está ejecutando.
.EM	BOOL	El bit de vacío indica que se ha completado la instrucción, pero es sincrónica con respecto al escaneado de la lógica.
.ER	BOOL	El bit de error indica cuándo falla la instrucción (errores).
.FD	BOOL	El bit de encontrado no se aplica a esta instrucción.
.POS	DINT	La posición determina el número de caracteres en el búfer hasta e incluso el primer conjunto de caracteres de terminación.
.ERROR	DINT	El error contiene un valor hexadecimal que identifica la causa de un error.

### Descripción

La instrucción AHL puede:

- Obtener el estado de las líneas de control del puerto serial
- Activar o desactivar la señal de Terminal de Datos Listo (DTR)
- Activar o desactivar la señal de Solicitud de Envío (RTS)

Para programar la instrucción AHL, siga estas pautas:

Configure el puerto serial del controlador

Si su aplicación:	Entonces:
Usa instrucciones ARD o ARL	Seleccione el modo Usuario
No usa instrucciones ARD o ARL	Seleccione el modo Sistema o el modo Usuario

Utilice la siguiente tabla para seleccionar los valores correctos para los operandos ANDMask y ORMask:

Para cambiar DTR al siguiente estado:	Para cambiar RTS al siguiente estado:	Introduzca este valor ANDMask:	E introduzca este valor ORMask:
desactivado	desactivado	3	0
	activado	1	
	inalterado	1	
Activado	desactivado	2	1
	activado	0	
	inalterado	0	
Inalterado	desactivado	2	0
	activado	0	
	inalterado	0	

Se trata de una instrucción de transición:

- En el diagrama de escalera, cambie EnableIn de borrado a establecido cada vez que deba ejecutarse la instrucción.
- En el texto estructurado, condicione la instrucción para que solo se ejecute en una transición.

#### Afectar a las marcas de estado matemático

No

#### Condiciones de fallo

Tipo	Código	Causa	Método de recuperación
4	57	La instrucción AHL no pudo ejecutarse porque el puerto serial está establecido en no handshaking.	Cambie el parámetro Línea de control del puerto serial. o Elimine la instrucción AHL

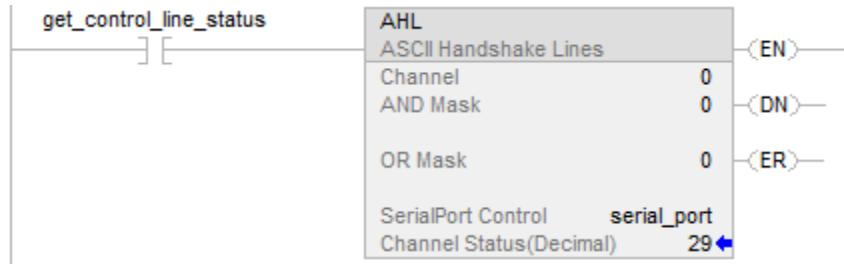
#### Ejecución

##### Diagrama de escalera

Condición	Acción del Diagrama de escalera
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	La instrucción se ejecuta cuando la condición de entrada de reglón cambia de borrado a establecido.
Post-escaneado	N/A

##### Texto estructurado

Condición	Acción de texto estructurado
Pre-escaneado	N/A
Ejecución normal	La instrucción se ejecuta cuando la condición de entrada de reglón cambia de borrado a establecido.
Post-escaneado	N/A

**Ejemplo****Diagrama de escalera****Texto estructurado**

```

osri_1.InputBit := get_control_line_status;
OSRI(osri_1);
IF (osri_1.OutputBit) THEN
  AHL(0,0,0,serial_port);
END_IF;

```

**Consulte también**

[Instrucciones de puerto serial ASCII](#) en la página 827

[Prueba ASCII para línea de búfer \(ABL\)](#) en la página 851

[Carácteres ASCII en el búfer \(ACB\)](#) en la página 829

[Borrar ASCII búfer \(ACL\)](#) en la página 832

[Lectura ASCII \(ARD\)](#) en la página 840

[Lectura ASCII de línea \(ARL\)](#) en la página 845

[Escritura ASCII con anexo \(AWA\)](#) en la página 861

[Escritura ASCII \(AWT\)](#) en la página 855

[Atributos comunes](#) en la página 923

**Lectura ASCII (ARD)**

Esta instrucción es compatible con los controladores Studio 5000 Logix Emulate solo.

La instrucción ARD quita caracteres del búfer y los almacena en Destination.

**Cons** Las instrucciones de puerto serial ASCII (AWT, AWA, ejo: ARD, ARL, ABL, ACB, AHL y ACL) no están disponibles para controladores que no tengan puertos seriales.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

#### Texto estructurado

ARD(Channel,Destination,SerialPortControl);

#### Operandos

#### Diagrama de escalera

Operando	Tipo	Formato	Descripción	Notas
Channel	DINT	inmediato etiqueta	0	
Destination	Cadena SINT INT DINT	etiqueta	etiqueta a la que se transfieren los caracteres (i.e., lectura): Para un tipo de datos de cadena, introduzca el nombre de la etiqueta. Para un matriz SINT, INT o DINT, introduzca el primer elemento del matriz.	Si desea comparar, convertir o manipular los caracteres, introduzca una etiqueta de cadena. Los tipos de datos de cadena son: tipo de datos STRING predeterminado cualquier tipo de cadena nuevo que usted cree

Serial Port Control	SERIAL_PORT_CONTROL	etiqueta	Etiqueta que controla la operación	
Serial Port Control Length	DINT	inmediato	número de caracteres a transferir al destino (lectura)  Si desea establecer el Serial Port Control Length igual que el tamaño de Destination, introduzca 0.	El Serial Port Control Length debe ser menor o igual que el tamaño de Destination.
Characters Read	DINT	inmediato	0	Durante la ejecución, muestra el número de caracteres en el búfer, incluyendo el primer conjunto de caracteres de terminación.

### Texto estructurado

Operando	Tipo	Formato	Descripción	Notas
Channel	DINT	inmediato etiqueta	0	
Destination	Cadena SINT INT DINT	etiqueta	etiqueta a la que se transfieren los caracteres (i.e., lectura): Para un tipo de datos de cadena, introduzca el nombre de la etiqueta. Para un matriz SINT, INT o DINT, introduzca el primer elemento del matriz.	Si desea comparar, convertir o manipular los caracteres, introduzca una etiqueta de cadena. Los tipos de datos de cadena son: tipo de datos STRING predeterminado cualquier tipo de cadena nuevo que usted cree
Serial Port Control	SERIAL_PORT_CONTROL	etiqueta	Etiqueta que controla la operación	
Serial Port Control Length	DINT	inmediato	número de caracteres a transferir al destino (lectura)	El Serial Port Control Length debe ser menor o igual que el tamaño de Destination. Si desea establecer el Serial Port Control Length igual que el tamaño de Destination, introduzca 0.
Characters Read	DINT	inmediato	0	Durante la ejecución, muestra el número de caracteres en el búfer, incluyendo el primer conjunto de caracteres de terminación.

Puede especificar los valores del Serial Port Control Length y de los Characters Read accediendo a los miembros .LEN y .POS de la estructura SERIAL\_PORT\_CONTROL en lugar de incluir valores en la lista operandos.

Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

### Estructura de SERIAL\_PORT\_CONTROL

Mnemónico	Tipo de datos	Descripción
.EN	BOOL	El bit de habilitación indica que la instrucción está habilitada.
.EU	BOOL	El bit de cola indica que la instrucción entró en la cola ASCII.
.DN	BOOL	El bit de efectuado indica que ha finalizado la instrucción pero que es asíncrono con respecto al escaneado de lógico.
.RN	BOOL	El bit de ejecución indica que la instrucción se está ejecutando.
.EM	BOOL	El bit de vacío indica que se ha completado la instrucción, pero es sincrónica con respecto al escaneado de la lógica.
.ER	BOOL	El bit de error indica cuándo falla la instrucción (errores).
.FD	BOOL	El bit de encontrado no se aplica a esta instrucción.
.LEN	DINT	La longitud indica el número de caracteres a transferir al destino (lectura).
.POS	DINT	La posición muestra el número de caracteres leídos.
.ERROR	DINT	El error contiene un valor hexadecimal que identifica la causa de un error.

### Descripción

La instrucción ARD quita el número de caracteres especificados del búfer y los almacena en Destination.

- La instrucción ARD sigue ejecutándose hasta que elimina el número de caracteres especificado (operando Serial Port Control Length).
- Mientras que la instrucción ARD se está ejecutando, no se ejecuta ninguna otra instrucción de puerto serial ASCII.

Para programar la instrucción ARD, siga estas pautas:

1. Configure el puerto serial del controlador para el modo Usuario.
2. Use los resultados de una instrucción ACB para desencadenar la instrucción ARD.  
Así evitará que la instrucción ARD retenga la cola mientras espera el

número requerido de caracteres. Consulte el ejemplo de ARD que se muestra a continuación para obtener más información.

3. Se trata de una instrucción de transición:  
En el diagrama de escalera, cambie EnableIn de borrado a establecido cada vez que deba ejecutarse la instrucción.  
En texto estructurado, condicione la instrucción para que solo se ejecute en una transición.
4. Para desencadenar una acción subsiguiente cuando la instrucción está hecho, examine el bit .EM.

#### Afectar a las marcas de estado matemático

No

#### Condiciones de fallo

No es específico para esta instrucción. Consulte los *Atributos comunes* para fallos relacionados con el operando.

#### Ejecución

##### Diagrama de escalera

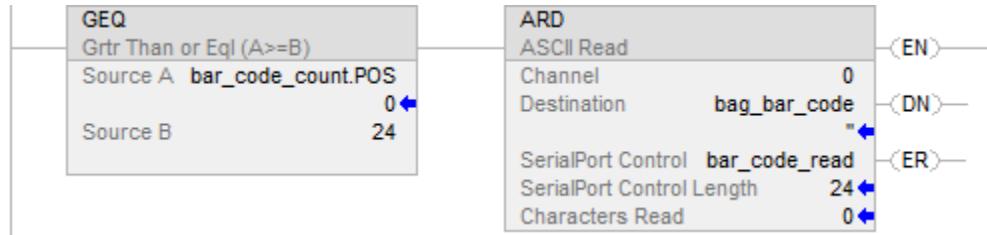
Condición	Acción del Diagrama de escalera
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	La instrucción se ejecuta. EnableIn cambia de borrado a establecido.
Post-escaneado	N/A

##### Texto estructurado

Condición	Acción de texto estructurado
Pre-escaneado	N/A
Ejecución normal	La instrucción se ejecuta. EnableIn cambia de borrado a establecido.
Post-escaneado	N/A

## Ejemplos

### Diagrama de escalera



### Texto estructurado

```

ACB(o,bar_code_count);

IF bar_code_count.POS >= 24 THEN

    bar_code_read.LEN := 24;

    ARD(0,bag_bar_code,bar_code_read);

END_IF;

```

### Consulte también

[Instrucciones de puerto serial ASCII](#) en la página 827

[Atributos comunes](#) en la página 923

[Sintaxis de texto estructurado](#) en la página 955

## Lectura ASCII de línea (ARL)

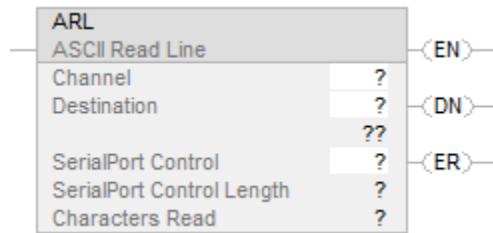
Esta instrucción es compatible con los controladores Studio 5000 Logix Emulate solo.

La instrucción ARL elimina caracteres del búfer y los almacena en Destination.

**Cons  
ejo:** Las instrucciones de puerto serial ASCII (AWT, AWA, ARD, ARL, ABL, ACB, AHL y ACL) no están disponibles para controladores que no tengan puertos seriales.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

#### Texto estructurado

```
ARL(Channel, Destination, SerialPortControl);
```

#### Operandos

#### Diagrama de escalera

Operando	Tipo	Formato	Descripción	Notas
Channel	DINT	inmediato etiqueta	0	
Destination	Cadena SINT INT DINT	etiqueta	etiqueta a la que se transfieren los caracteres (lectura) Para un tipo de datos de cadena, introduzca el nombre de la etiqueta. Para un matriz SINT, INT o DINT, introduzca el primer elemento del matriz.	Si desea comparar, convertir o manipular los caracteres, introduzca una etiqueta de cadena. Los tipos de datos de cadena son: tipo de datos STRING predeterminado cualquier tipo de cadena nuevo que usted cree
SerialPort Control	SERIAL_POR T_CONTROL	etiqueta	Etiqueta que controla la operación	

Serial Port Control Length	DINT	inmediato	número máximo de caracteres que se deben leer si no se encuentra ningún carácter de terminación.	Introduzca el número máximo de caracteres que tendrá cualquier mensaje (es decir, cuándo parar de leer si no se encuentra ningún carácter de terminación). Por ejemplo, si los mensajes oscilan entre 3 y 6 caracteres, introduzca 6. El Serial Port Control Length debe ser menor o igual que el tamaño de Destination. Si desea establecer el Serial Port Control Length igual que el tamaño de Destination, introduzca 0.
Characters Read	DINT	inmediato	0	Durante la ejecución, muestra el número de caracteres leídos.

### Texto estructurado

Operando	Tipo	Formato	Descripción	Notas
Channel	DINT	inmediato etiqueta	0	
Destination	Cadena SINT INT DINT	etiqueta	etiqueta a la que se transfieren los caracteres (lectura) Para un tipo de datos de cadena, introduzca el nombre de la etiqueta. Para un matriz SINT, INT o DINT, introduzca el primer elemento del matriz.	Si desea comparar, convertir o manipular los caracteres, introduzca una etiqueta de cadena. Los tipos de datos de cadena son: tipo de datos STRING predeterminado cualquier tipo de cadena nuevo que usted cree
SerialPort Control	SERIAL_PORT_CONTROL	etiqueta	Etiqueta que controla la operación	

Serial Port Control Length	DINT	inmediato	número máximo de caracteres que se deben leer si no se encuentra ningún carácter de terminación.	Introduzca el número máximo de caracteres que tendrá cualquier mensaje (es decir, cuándo parar de leer si no se encuentra ningún carácter de terminación). Por ejemplo, si los mensajes oscilan entre 3 y 6 caracteres, introduzca 6. El Serial Port Control Length debe ser menor o igual que el tamaño de Destination. Si desea establecer el Serial Port Control Length igual que el tamaño de Destination, introduzca 0.
Characters Read	DINT	inmediato	0	Durante la ejecución, muestra el número de caracteres leídos.

Sin embargo, puede especificar los valores de Serial Port Control Length y de Characters Read accediendo a los miembros .LEN y .POS de la estructura SERIAL\_PORT\_CONTROL en lugar de incluir el valor en la lista operando.

Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

#### Estructura de SERIAL\_PORT\_CONTROL

Mnemónico	Tipo de datos	Descripción
.EN	BOOL	El bit de habilitación indica que la instrucción está habilitada.
.EU	BOOL	El bit de cola indica que la instrucción entró en la cola ASCII.
.DN	BOOL	El bit de efectuado indica que ha finalizado la instrucción pero que es asíncrono con respecto al escaneado de lógico.
.RN	BOOL	El bit de ejecución indica que la instrucción se está ejecutando.
.EM	BOOL	El bit de vacío indica que se ha completado la instrucción, pero es sincrónica con respecto al escaneado de la lógica.
.ER	BOOL	El bit de error indica cuándo falla la instrucción (errores).
.FD	BOOL	El bit de encontrado no se aplica a esta instrucción.
.LEN	DINT	La longitud indica el número máximo de caracteres que se tendrá que pasar al destino (es decir, cuándo parar de leer si no se encuentra ningún carácter de terminación).
.POS	DINT	La posición muestra el número de caracteres leídos.
.ERROR	DINT	El error contiene un valor hexadecimal que identifica la causa de un error.

## Descripción

La instrucción ARL quita caracteres del búfer y los almacena en Destination, tal y como se muestra a continuación:

- La instrucción ARL se seguirá ejecutando hasta eliminar:
  - bien el primer conjunto de caracteres de terminación
  - o bien el número especificado de caracteres (operando String Length)

Mientras que la instrucción ARL se está ejecutando, no se ejecuta ninguna otra instrucción ASCII. Para programar la instrucción ARL, siga estas pautas:

1. Configure el puerto serial del controlador para el modo Usuario y defina los caracteres que sirven como caracteres de terminación.
2. Use los resultados de una instrucción ABL para desencadenar la instrucción ARL.  
Así evitará que la instrucción ARL retenga la cola mientras espera los caracteres terminales. Consulte el ejemplo de ARL que se muestra a continuación para obtener más información.
3. Se trata de una instrucción de transición:  
En el diagrama de escalera, cambie EnableIn de borrado a establecido cada vez que deba ejecutarse la instrucción. En texto estructurado, condicione la instrucción para que solo se ejecute en una transición.
4. Para desencadenar una acción subsiguiente cuando la instrucción está hecho, examine el bit .EM.

## Afectar a las marcas de estado matemático

No

## Condiciones de fallo

No es específico para esta instrucción. Consulte los *Atributos comunes* para fallos relacionados con el operando.

## Ejecución

### Diagrama de escalera

Condición	Acción del Diagrama de escalera
Pre-escaneado	N/A

La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	La instrucción se ejecuta. EnableIn cambia de borrado a establecido.
Post-escaneado	N/A

### Texto estructurado

Condición	Acción de texto estructurado
Pre-escaneado	N/A
Ejecución normal	La instrucción se ejecuta. EnableIn cambia de borrado a establecido.
Post-escaneado	N/A

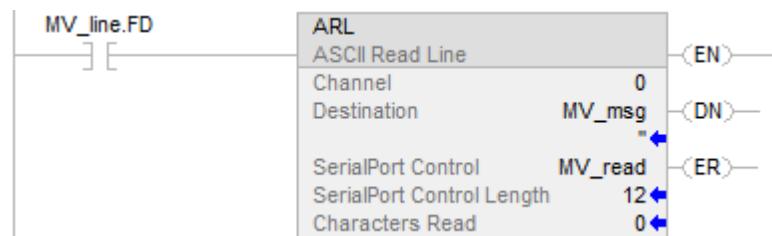
### Ejemplo

Comprueba continuamente el búfer buscando un mensaje del terminal MessageView. Puesto que cada mensaje termina en un retorno de carro (\$r), el retorno de carro se configura como carácter de terminación en la pestaña Protocolo de usuario (User Protocol) del cuadro de diálogo Propiedades del controlador (Controller Properties).

Cuando ABL encuentra un retorno de carro, establece el bit .FD. Cuando la instrucción ABL encuentra un retorno de carro (MV\_line.FD es establece), el controlador ha recibido un mensaje completo.

La instrucción ARL elimina los caracteres del búfer, incluido el retorno de carro, y los coloca en el miembro DATA de la etiqueta MV\_msg, que es un tipo de cadena.

### Diagrama de escalera



### Texto estructurado

ABL(0,MV\_line);

```
osri_1.InputBit :=MVLine.FD  
OSRI(osri_1);  
IF osri_1.OutputBit) THEN  
mv_read.LEN := 12;  
ARL(0,MV_msg,MV_read);  
END_IF;
```

### Consulte también

[Instrucciones de puerto serial ASCII](#) en la página 827

[Prueba ASCII para línea de búfer \(ABL\)](#) en la página 851

[Caracteres ASCII en el búfer \(ACB\)](#) en la página 829

[Borrar ASCII búfer \(ACL\)](#) en la página 832

[Líneas de handshake ASCII \(AHL\)](#) en la página 836

[Lectura ASCII \(ARD\)](#) en la página 840

[Escritura ASCII con anexo \(AWA\)](#) en la página 861

[Escritura ASCII \(AWT\)](#) en la página 855

[Atributos comunes](#) en la página 923

[Sintaxis de texto estructurado](#) en la página 955

## Prueba ASCII para línea de búfer (ABL)

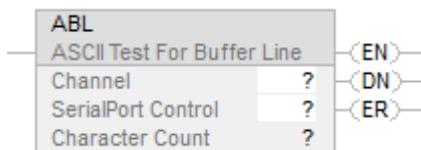
Esta instrucción es compatible con los controladores Studio 5000 Logix Emulate solo.

La instrucción ABL cuenta los caracteres en el búfer hasta e incuido el primer carácter de terminación.

**Cons ejo:** Las instrucciones de puerto serial ASCII (AWT, AWA, ARD, ARL, ABL, ACB, AHL y ACL) no están disponibles para controladores que no tengan puertos seriales.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

#### Texto estructurado

```
ABL(Channel,SerialPortControl);
```

#### Operandos

#### Diagrama de escalera

Operando	Tipo	Formato	Descripción
Channel	DINT	inmediato	0
SerialPort Control	SIGNAL_PORT_CONTROL	etiqueta	Etiqueta que controla la operación
Character Count	DINT	inmediato	0 Durante la ejecución, muestra el número de caracteres en el búfer, incluyendo el primer conjunto de caracteres de terminación.

#### Texto estructurado

Operando	Tipo	Formato	Descripción
Channel	DINT	inmediato	0
SerialPort Control	SIGNAL_PORT_CONTROL	etiqueta	Etiqueta que controla la operación
Character Count	DINT	inmediato	0 Durante la ejecución, muestra el número de caracteres en el búfer, incluyendo el primer conjunto de caracteres de terminación.

Puede acceder al valor de Character Count a través del miembro .POS de la estructura SERIAL\_PORT\_CONTROL.

Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

### Estructura de SERIAL\_PORT\_CONTROL

Mnemónico	Tipo de datos	Descripción
.EN	BOOL	El bit de habilitación indica que la instrucción está habilitada.
.EU	BOOL	El bit de cola indica que la instrucción entró en la cola ASCII.
.DN	BOOL	El bit de efectuado indica cuándo se ha completado la instrucción, pero es asincrónica con respecto al escaneado de la lógica.
.RN	BOOL	El bit de ejecución indica que la instrucción se está ejecutando.
.EM	BOOL	El bit de vacío indica que se ha completado la instrucción, pero es sincrónica con respecto al escaneado de la lógica.
.ER	BOOL	El bit de error indica cuándo falla la instrucción (errores).
.FD	BOOL	El bit de encontrado indica que la instrucción encontró el/los carácter/es de terminación.
.POS	DINT	La posición determina el número de caracteres en el búfer hasta e incluido el primer conjunto de caracteres de terminación. La instrucción solo devuelve este número después de encontrar el/los carácter/es de terminación.
.ERROR	DINT	El error contiene un valor hexadecimal que identifica la causa de un error.

### Descripción

La instrucción ABL busca en el búfer el primer grupo de caracteres de terminación. Si la instrucción encuentra los caracteres de terminación, entonces:

- establece el bit .FD
- cuenta los caracteres en el búfer hasta e incluido el primer grupo de caracteres de terminación

En la pestaña **Protocolo de usuario** (User Protocol) del cuadro de diálogo **Propiedades del controlador** (Controller properties) se definen los caracteres ASCII que la instrucción considera que son los caracteres de terminación.

Para programar la instrucción ABL, siga estas pautas:

- Configure el puerto serial del controlador para el modo Usuario y defina los caracteres que sirven como caracteres de terminación.

Se trata de una instrucción de transición:

- En el diagrama de escalera, cambie EnableIn de borrado a establecido cada vez que deba ejecutarse la instrucción.
- En el texto estructurado, condicione la instrucción para que solo se ejecute en una transición.

#### Afectar a las marcas de estado matemático

No

#### Condiciones de fallo

No es específico para esta instrucción. Consulte los *Atributos comunes* para fallos relacionados con el operando.

#### Ejecución

##### Diagrama de escalera

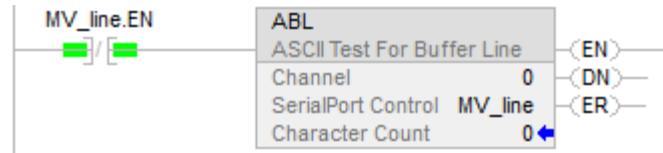
Condición	Acción del Diagrama de escalera
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	La instrucción se ejecuta. EnableIn cambia de borrado a establecido.
Post-escaneado	N/A

##### Texto estructurado

Condición	Acción de texto estructurado
Pre-escaneado	N/A
Ejecución normal	La instrucción se ejecuta. EnableIn cambia de borrado a establecido.
Post-escaneado	N/A

## Ejemplo

### Diagrama de escalera



### Texto estructurado

```
ABL(0,MV_line);
```

### Consulte también

[Instrucciones de puerto serial ASCII](#) en la página 827

[Caracteres ASCII en el búfer \(ACB\)](#) en la página 829

[Borrar ASCII búfer \(ACL\)](#) en la página 832

[Líneas de handshake ASCII \(AHL\)](#) en la página 836

[Lectura ASCII \(ARD\)](#) en la página 840

[Lectura ASCII de línea \(ARL\)](#) en la página 845

[Escritura ASCII con anexo \(AWA\)](#) en la página 861

[Escritura ASCII \(AWT\)](#) en la página 855

[Atributos comunes](#) en la página 923

[Sintaxis de texto estructurado](#) en la página 955

## Escritura ASCII (AWT)

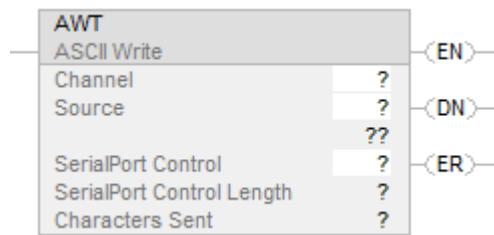
Esta instrucción es compatible con los controladores Studio 5000 Logix Emulate solo.

La instrucción AWT envía los caracteres de la matriz Origen al dispositivo en serie.

**Cons  
ejo:** Las instrucciones de puerto serial ASCII (AWT, AWA,  
ARD, ARL, ABL, ACB, AHL y ACL) no están disponibles  
para controladores que no tengan puertos seriales.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

#### Texto estructurado

```
AWT(Channel,Source,SerialPortControl);
```

#### Operandos

#### Diagrama de escalera

Operando	Tipo	Formato	Descripción	Notas
Channel	DINT	inmediato etiqueta	0	
Source	Cadena SINT INT DINT	etiqueta	Etiqueta que contiene los caracteres se deben enviar. Para un tipo de datos de cadena, introduzca el nombre de la etiqueta. Para un matriz SINT, INT o DINT, introduzca el primer elemento del matriz.	Si desea comparar, convertir o manipular los caracteres, introduzca una etiqueta de cadena. Los tipos de datos de cadena son: tipo de datos STRING predeterminado cualquier tipo de cadena nuevo que usted cree
Serial Port Control	SERIAL_PORT CONTROL	etiqueta	Etiqueta que controla la operación	
Serial Port Control Length	DINT	inmediato	Número de caracteres que deben enviarse	El Serial Port Control Length debe ser menor o igual que el tamaño de Source. Si desea establecer el Serial Port Control Length igual que el número de caracteres de Source, introduzca 0.

Characters Sent	DINT	inmediato	0	Durante la ejecución, muestra el número de caracteres que se han enviado
-----------------	------	-----------	---	--

### Texto estructurado

Operando	Tipo	Formato	Descripción	Notas
Channel	DINT	inmediato etiqueta	0	
Source	Cadena SINT INT DINT	etiqueta	Etiqueta que contiene los caracteres se deben enviar. Para un tipo de datos de cadena, introduzca el nombre de la etiqueta. Para un matriz SINT, INT o DINT, introduzca el primer elemento del matriz.	Si desea comparar, convertir o manipular los caracteres, introduzca una etiqueta de cadena. Los tipos de datos de cadena son: tipo de datos STRING predeterminado cualquier tipo de cadena nuevo que usted cree
Serial Port Control	SERIAL_PORT_CONTROL	etiqueta	Etiqueta que controla la operación	
Serial Port Control Length	DINT	inmediato	Número de caracteres que deben enviarse	El Serial Port Control Length debe ser menor o igual que el tamaño de Source. Si desea establecer el Serial Port Control Length igual que el número de caracteres de Source, introduzca 0.
Characters Sent	DINT	inmediato	0	Durante la ejecución, muestra el número de caracteres que se han enviado

Puede especificar los valores de Serial Port Control Length y de Characters Sent accediendo a los miembros .LEN y .POS de la estructura

SERIAL\_PORT\_CONTROL en lugar de incluir el valor en la lista operandos.

Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

### Estructura de SERIAL\_PORT\_CONTROL

Mnemónico	Tipo de datos	Descripción
.EN	BOOL	El bit de habilitación indica que la instrucción está habilitada.
.EU	BOOL	El bit de cola indica que la instrucción entró en la cola ASCII.
.DN	BOOL	El bit de efectuado indica que ha finalizado la instrucción pero que es asíncrono con respecto al escaneado de lógico.

.RN	BOOL	El bit de ejecución indica que la instrucción se está ejecutando.
.EM	BOOL	El bit de vacío indica que se ha completado la instrucción, pero es sincrónica con respecto al escaneado de la lógica.
.ER	BOOL	El bit de error indica cuándo falla la instrucción (errores).
.FD	BOOL	El bit de encontrado no se aplica a esta instrucción.
.LEN	DINT	La longitud indica el número de caracteres que deben enviarse.
.POS	DINT	La posición muestra el número de caracteres enviados.
.ERROR	DINT	El error contiene un valor hexadecimal que identifica la causa de un error.

### Descripción

La instrucción AWT envía el número de caracteres especificado (longitud de control del puerto serial) de la etiqueta de Source al dispositivo conectado al puerto serial del controlador.

Para programar la instrucción AWT, siga estas pautas:

- Configure el puerto serial del controlador

Si su aplicación:	Entonces:
Usa instrucciones ARD o ARL	Seleccione el modo Usuario
No usa instrucciones ARD o ARL	Selecione el modo sistema o el modo usuario

- Se trata de una instrucción de transición: En el diagrama de escalera, cambie EnableIn de borrado a establecido cada vez que deba ejecutarse la instrucción. En el texto estructurado, condicione la instrucción para que solo se ejecute en una transición.
- Cada vez que se ejecuta la instrucción, ¿envía siempre el mismo número de caracteres?

Si:	Entonces:
Sí	Introduzca el número de caracteres que deseé enviar en el Serial Port Control Length.
No	Antes de que se ejecute la instrucción, mueva el miembro LEN de la etiqueta de Source al miembro LEN de la etiqueta de Serial Port Control. Consulte el ejemplo 2 más abajo.

**Afectar a las marcas de estado matemático**

No

**Condiciones de fallo**

No es específico para esta instrucción. Consulte los *Atributos comunes* para fallos relacionados con el operando.

**Ejecución****Diagrama de escalera**

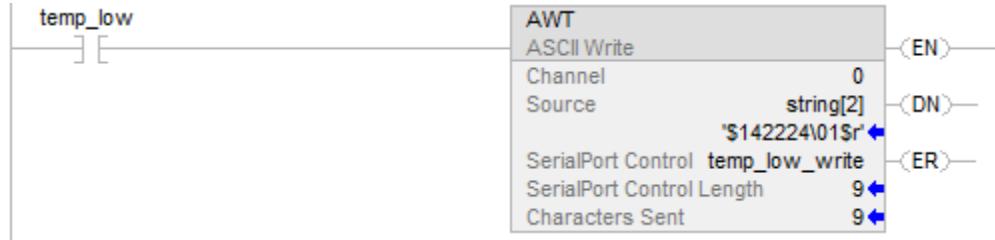
Condición	Acción del Diagrama de escalera
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	La instrucción se ejecuta. EnableIn cambia de borrado a establecido.
Post-escaneado	N/A

**Texto estructurado**

Condición	Acción de texto estructurado
Pre-escaneado	N/A
Ejecución normal	La instrucción se ejecuta. EnableIn cambia de borrado a establecido.
Post-escaneado	N/A

**Ejemplos****Ejemplo 1**

Cuando la temperatura llega al límite bajo (temp\_low se establece), la instrucción AWT envía un mensaje al terminal MessageView conectado al puerto serial del controlador. El mensaje contiene nueve caracteres del miembro DATA de la etiqueta string[2], que es un tipo de cadena. (El carácter \$14 se cuenta como un carácter; es el código hexadecimal del carácter Ctrl-T). El último carácter es un retorno de carro (\$r), que marca el final del mensaje.

**Diagrama de escalera****Texto estructurado**

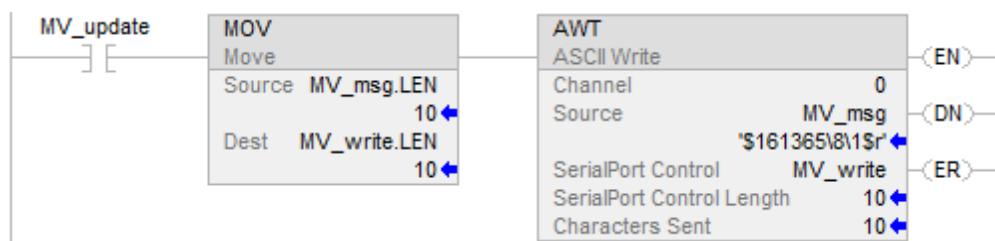
```

osri_1.InputBit := temp_low;
OSRI(osri_1);
IF (osri_1.OutputBit) THEN
    temp_low_write.LEN := 9;
    AWT(0.string[2],temp_low_write);
END_IF;

```

**Ejemplo 2**

Cuando `MV_update` se establece, la instrucción `AWT` envía el número de caracteres especificado en `MV_msg`. Puesto que el número de caracteres de `MV_msg` varía, el reglón mueve primero la longitud de la cadena (`MV_msg.LEN`) a la instrucción `AWT` de Serial Port Control Length (`MV_write.LEN`). (En `MV_msg`, el carácter \$16 se cuenta como un carácter; es el código hexadecimal del carácter Ctrl-V.)

**Diagrama de escalera****Texto estructurado**

```

osri_1.InputBit := MV_update;
OSRI(osri_1);

```

```
IF (osri_1.OutputBit) THEN  
    MV_write.LEN := Mv_msg.LEN;  
    AWT(0.MV_msg,MV_write);  
END_IF;
```

### Consulte también

[Instrucciones de puerto serial ASCII](#) en la página 827

[Prueba ASCII para línea de búfer \(ABL\)](#) en la página 851

[Caracteres ASCII en el búfer \(ACB\)](#) en la página 829

[Borrar ASCII búfer \(ACL\)](#) en la página 832

[Líneas de handshake ASCII \(AHL\)](#) en la página 836

[Lectura ASCII \(ARD\)](#) en la página 840

[Lectura ASCII de línea \(ARL\)](#) en la página 845

[Escritura ASCII con anexo \(AWA\)](#) en la página 861

[Atributos comunes](#) en la página 923

[Sintaxis de texto estructurado](#) en la página 955

## Escritura ASCII con anexo (AWA)

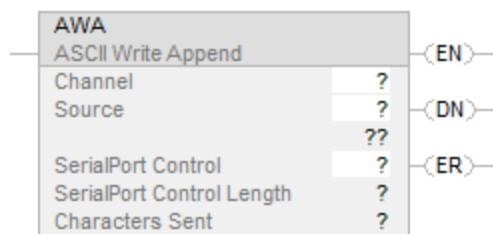
Esta instrucción es compatible con los controladores Studio 5000 Logix Emulate solo.

La instrucción AWA envía los caracteres de la matriz de Source a un dispositivo en serie y añade uno o dos caracteres predefinidos.

**Cons ejo:** Las instrucciones de puerto serial ASCII (AWT, AWA, ARD, ARL, ABL, ACB, AHL y ACL) no están disponibles para controladores que no tengan puertos seriales.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

#### Texto estructurado

AWA(Channel,Source,SerialPortControl);

#### Operandos

#### Diagrama de escalera

Operando	Tipo	Formato	Descripción	Notas
Channel	DINT	inmediato	0	
Source	Cadena SINT INT DINT	etiqueta	etiqueta que contiene los caracteres se deben enviar. Para un tipo de datos de cadena, introduzca el nombre de la etiqueta Para un matriz SINT, INT o DINT, introduzca el primer elemento del matriz.	Si desea comparar, convertir o manipular los caracteres, introduzca una etiqueta de cadena. Los tipos de datos de cadena son: tipo de datos STRING predeterminado cualquier tipo de cadena nuevo que usted cree
Serial Port Control	SERIAL_PORT_CONTROL	etiqueta	Etiqueta que controla la operación	
Serial Port Control Length	DINT	inmediato	número de caracteres que deben enviarse	El Serial Port Control Length debe ser menor o igual que el tamaño de Source. Si desea establecer el Serial Port Control Length igual que el número de caracteres de Source, introduzca 0.

Characters Sent	DINT	inmediato	0	Durante la ejecución, muestra el número de caracteres que se han enviado.
-----------------	------	-----------	---	---

### Texto estructurado

Operando	Tipo	Formato	Descripción	Notas
Channel	DINT	inmediato etiqueta	0	
Source	Cadena SINT INT DINT	etiqueta	etiqueta que contiene los caracteres se deben enviar. Para un tipo de datos de cadena, introduzca el nombre de la etiqueta Para un matriz SINT, INT o DINT, introduzca el primer elemento del matriz.	Si desea comparar, convertir o manipular los caracteres, introduzca una etiqueta de cadena. Los tipos de datos de cadena son: tipo de datos STRING predeterminado cualquier tipo de cadena nuevo que usted cree
Serial Port Control	SERIAL_PORT_CONTROL	etiqueta	Etiqueta que controla la operación	
Serial Port Control Length	DINT	inmediato	número de caracteres que deben enviarse	El Serial Port Control Length debe ser menor o igual que el tamaño de Source. Si desea establecer el Serial Port Control Length igual que el número de caracteres de Source, introduzca 0.
Characters Sent	DINT	inmediato	0	Durante la ejecución, muestra el número de caracteres que se han enviado.

Puede especificar los valores de Serial Port Control Length y de Characters Sent accediendo a los miembros .LEN y .POS de la estructura SERIAL\_PORT\_CONTROL en lugar de incluir el valor en la lista operandos.

Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

### Estructura de SERIAL\_PORT\_CONTROL

Mnemónico	Tipo de datos	Descripción
.EN	BOOL	El bit de habilitación indica que la instrucción está habilitada.
.EU	BOOL	El bit de cola indica que la instrucción entró en la cola ASCII.
.DN	BOOL	El bit de efectuado indica que ha finalizado la instrucción pero que es asíncrono con respecto al escaneado de lógico.
.RN	BOOL	El bit de ejecución indica que la instrucción se está ejecutando.
.EM	BOOL	El bit de vacío indica que se ha completado la instrucción, pero es sincrónica con respecto al escaneado de la lógica.
.ER	BOOL	El bit de error indica cuándo falla la instrucción (errores).

.FD	BOOL	El bit de encontrado no se aplica a esta instrucción.
.LEN	DINT	La longitud indica el número de caracteres que deben enviarse.
.POS	DINT	La posición muestra el número de caracteres enviados.
.ERROR	DINT	El error contiene un valor hexadecimal que identifica la causa de un error.

### Descripción

La instrucción AWA:

- Envía el número de caracteres especificado (control de longitud del puerto serial) de la etiqueta de Source al dispositivo conectado al puerto serial del controlador.
- Añade al final de los caracteres (agrega) uno o dos caracteres definidos en la pestaña Protocolo de usuario (User Protocol) del diálogo Propiedades del controlador (Controller Properties).

Para programar la instrucción AWA, siga estas pautas:

- Configure el puerto serial del controlador

Si su aplicación:	Entonces:
Usa instrucciones ARD o ARL	Seleccione el modo Usuario
No usa instrucciones ARD o ARL	Seleccione el modo Sistema o el modo Usuario

- Se trata de una instrucción de transición: En el diagrama de escalera, cambie EnableIn de borrado a establecido cada vez que deba ejecutarse la instrucción.  
En el texto estructurado, condicione la instrucción para que solo se ejecute en una transición.
- Cada vez que se ejecuta la instrucción, ¿envía siempre el mismo número de caracteres?

Si:	Entonces:
Sí	Introduzca el número de caracteres que deseé enviar en el Serial Port Control Length.
No	Antes de que se ejecute la instrucción, mueva el miembro LEN de la etiqueta de Source al miembro LEN de la etiqueta de Serial Port Control. (Consulte el ejemplo 2 más abajo.)

### Afectar a las marcas de estado matemático

No

### Condiciones de fallo

No es específico para esta instrucción. Consulte los *Atributos comunes* para fallos relacionados con el operando.

### Ejecución

#### Diagrama de escalera

Condición	Acción del Diagrama de escalera
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	La instrucción se ejecuta. EnableIn cambia de borrado a establecido.
Post-escaneado	N/A

#### Texto estructurado

Condición	Acción de texto estructurado
Pre-escaneado	N/A
Ejecución normal	La instrucción se ejecuta. EnableIn cambia de borrado a establecido.
Post-escaneado	N/A

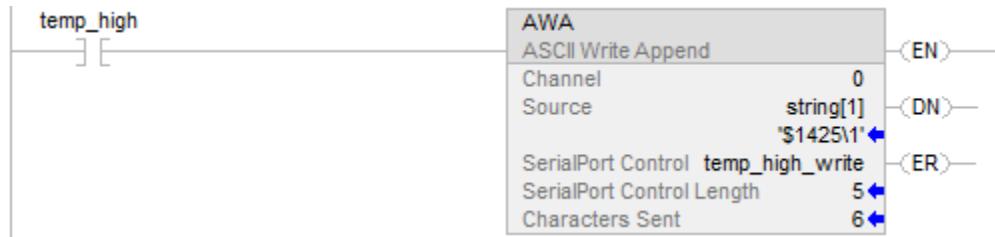
### Ejemplos

#### Ejemplo 1

Cuando la temperatura supera el límite alto (temp\_high se establece), la instrucción AWA envía un mensaje al terminal MessageView conectado al puerto serial del controlador.

El mensaje contiene cinco caracteres del miembro DATA de la etiqueta string[1], que es un tipo de cadena. (El carácter \$14 se cuenta como un carácter; es el código hexadecimal del carácter Ctrl-T).

La instrucción también envía (agrega) los caracteres definidos en las propiedades del controlador. En este ejemplo, la instrucción AWA envía un retorno de carro (\$0D), que marca el final del mensaje.

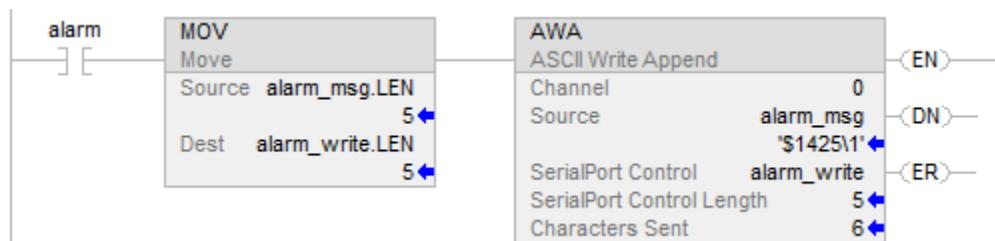
**Diagrama de escalera****Texto estructurado**

```
IF temp_high THEN
    temp_high_write.LEN := 5;
    AWA(o,string[1],temp_high_write);
    temp_high := 0;
END_IF;
```

**Ejemplo 2**

Cuando la alarma se establece, la instrucción AWA envía el número de caracteres especificado en `alarm_msg` y agrega uno o varios caracteres de terminación. Puesto que el número de caracteres en `alarm_msg` varía, el reglón mueve primero la longitud de la cadena (`alarm_msg.LEN`)

a `Serial Port Control Length` de la instrucción AWA (`alarm_write.LEN`). En `alarm_msg`, el carácter \$14 se cuenta como un carácter; es el código hexadecimal del carácter Ctrl-T.

**Diagrama de escalera****Texto estructurado**

```
osri_1.InputBit := alarm;
OSRI(osri_1);
```

```
IF(osri_1.OutputBit) THEN  
    alarm_write.LEN := alarm_msg.LEN;  
    AWA(0,alarm_msg.alarm_write);  
END_IF;
```

### Consulte también

[Instrucciones de puerto serial ASCII](#) en la página 827

[Prueba ASCII para línea de búfer \(ABL\)](#) en la página 851

[Caracteres ASCII en el búfer \(ACB\)](#) en la página 829

[Borrar ASCII búfer \(ACL\)](#) en la página 832

[Líneas de handshake ASCII \(AHL\)](#) en la página 836

[Lectura ASCII \(ARD\)](#) en la página 840

[Lectura ASCII de línea \(ARL\)](#) en la página 845

[Escritura ASCII \(AWT\)](#) en la página 855

[Atributos comunes](#) en la página 923

[Sintaxis de texto estructurado](#) en la página 955

## Tipos de cadena

Almacene caracteres ASCII en etiquetas que usan datos de tipo cadena para:

- Usar el tipo de datos STRING predeterminado, que puede almacenar hasta 82 caracteres.
- Crear un nuevo tipo de cadena que almacene menos o más caracteres.

Para crear un nuevo tipo de cadena, consulte LOGIX 5000 Controllers ASCII Strings Programming Manual 1756-PM013.

Cada tipo de cadena contiene los siguientes miembros:

Nombre	Tipo de datos	Descripción	Notas
LEN	DINT	número de caracteres de la cadena	<p>El miembro LEN se actualiza automáticamente al nuevo conteo de caracteres cada vez que lo usa:</p> <ul style="list-style-type: none"> <li>• El Explorador de cadenas para introducir caracteres.</li> <li>• Instrucciones que lean, conviertan o manipulen una cadena.</li> </ul> <p>El miembro LEN muestra la longitud de la cadena actual. El miembro DATA puede contener caracteres adicionales y antiguos que no estén incluidos en el conteo de LEN.</p>
DATA	Matriz SINT	Caracteres ASCII de la cadena	<p>Para acceder a los caracteres de la cadena, dirige al nombre de la etiqueta. Por ejemplo, para acceder a los caracteres de la etiqueta string_1, introduzca string_1.</p> <p>Cada uno de los elementos de la matriz DATA contiene un carácter.</p> <p>Crear nuevos tipos de cadena que almacenen menos o más caracteres..</p>

### Consulte también

[Literales de cadena de caracteres](#) en la página 968

**Códigos de error ASCII** Si falla la ejecución de una instrucción del puerto serial ASCII, el miembro ERROR de su estructura SERIAL\_PORT\_CONTROL contendrá algunos de los siguientes códigos de error hexadecimales:

Hex code	Indica que:
16#2	El módem se desconectó de la red.
16#3	La señal CTS se perdió durante la comunicación.
16#4	El puerto serial estaba en modo Sistema.
16#5	Las instrucciones no se han podido enviar o recibir porque la configuración del canal se ha desconectado a través del menú de configuración del canal.
16#6	Se han transmitido parámetros incorrectos al driver ASCII.
16#7	Las instrucciones no se han podido enviar o recibir porque la configuración del canal se ha desconectado a través del menú de configuración del canal.
16#8	La transmisión ya está ejecutando. Esto provocará que falle la instrucción que se está ejecutando.
16#9	La configuración actual del canal no soporta la Comunicación ASCII solicitada.

16#10	Intento de ejecutar una instrucción AHL mientras el Channel estaba en modo sistema.
16#A	Se ha establecido el bit UL antes de que se ejecutara la instrucción. Esto detendrá la ejecución de la instrucción
16#B	El puerto sobre el que se pedía que operara la instrucción no existe.
16#C	Cambia el modo del controlador de Marcha a Programa. Esto detiene la ejecución de una instrucción de puerto serial ASCII y borra la cola.
16#D	En la pestaña Protocolo de usuario (User Protocol) del cuadro de diálogo Propiedades del controlador (Controller Properties), se han cambiado y aplicado los parámetros de tamaño del búfer y de modo eco. Esto detiene la ejecución de una instrucción de puerto serial ASCII y borra la cola.
16 # E	La instrucción ACL ejecutó y detuvo o eliminó este tipo de instrucción.
16#F	La configuración del puerto serial cambió de modo Usuario a modo Sistema. Esto detiene la ejecución de una instrucción de puerto serial ASCII y borra la cola.
16#51	El valor LEN de la etiqueta de cadena es negativo o mayor que el tamaño de DATA de la etiqueta de cadena.
16#54	Serial Port Control length es mayor que el tamaño del búfer.
16#55	Serial Port Control length es negativo o mayor que el tamaño de Source o Destination.



# Instrucciones de cadena ASCII

## Instrucciones de cadena ASCII

Se utiliza las instrucciones de cadena ASCII para modificar y crear cadenas de caracteres ASCII.

### Instrucciones disponibles

#### Diagrama de escalera y Texto estructurado

<a href="#">FIND</a>	<a href="#">INSERT</a>	<a href="#">MID</a>	<a href="#">CONCAT</a>	<a href="#">DELETE</a>
----------------------	------------------------	---------------------	------------------------	------------------------

### Bloque de funciones

No disponible

Si desea:	Utilice esta instrucción:
Añadir caracteres de terminación o delimitadores a una cadena	CONCAT
Borrar caracteres de una cadena (por ejemplo, eliminar encabezados o caracteres de control de una cadena)	DELETE
Determinar el carácter de inicio de una cadena secundaria	FIND
Insertar caracteres en una cadena	INSERT
Extraer caracteres de una cadena	MID

También puede utilizar las siguientes instrucciones para comparar o convertir caracteres ASCII:

Si desea:	Utilice esta instrucción:
Comparar una cadena con otra	CMP
Determinar si unos caracteres son iguales que otros caracteres concretos	EQU
Determinar si unos caracteres son diferentes de otros caracteres concretos	NEQ
Determinar si unos caracteres son iguales o mayores que otros caracteres concretos	GEQ
Determinar si unos caracteres son mayores que otros caracteres concretos	GRT
Determinar si unos caracteres son iguales o menores que otros caracteres concretos	LEQ

Determinar si unos caracteres son menores que otros caracteres concretos	LES
Reordenar los bytes de una etiqueta INT, DINT o REAL	SWPB
Encontrar una cadena en una matriz de cadenas	FSC
Convertir los caracteres en valores SINT, INT, DINT o REAL	STOD
Convertir los caracteres en un valor REAL	STOR
Convertir un valor SINT, INT, DINT o REAL en una cadena de caracteres ASCII	DTOS
Convertir un valor REAL en una cadena de caracteres ASCII	RTOS

### Consulte también

[Códigos de error ASCII](#) en la página 868

[Tipos de cadena](#) en la página 867

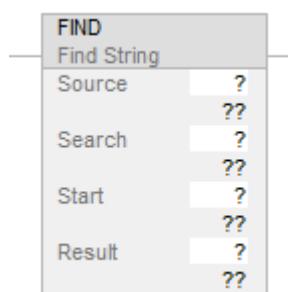
## Encontrar cadena (FIND)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción FIND encuentra la posición de inicio de una cadena especificada dentro de otra cadena.

### Idiomas disponibles

#### Diagrama de escalera



### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

## Texto estructurado

`FIND(Source,Search,Start,Result);`

### Operандos

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción. Consulte Conversión de datos.

### Diagrama de escalera y Texto estructurado

Operando	Tipo	Formato	Descripción	Notas
Source	ANY_STRING	Etiqueta	La cadena en la que buscar	Los tipos de datos de cadena son: tipo de datos STRING predeterminado con una longitud máxima de 82 caracteres.
Search	ANY_STRING	Etiqueta	La cadena que se debe encontrar	cualquier tipo de cadena que usted cree con una longitud de caracteres configurable.
Start	SINT INT DINT	Immediato etiqueta	La posición en Source a partir de la que hay que empezar la búsqueda	Introduzca un número entre 1 y el tamaño de DATA del Source.
Result	DINT SINT INT	Etiqueta	La posición en Source donde se ha encontrado la cadena de búsqueda	

Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

### Descripción

La instrucción FIND busca la cadena de Source dentro de la cadena de Search. Si la instrucción encuentra la cadena de Search, Result muestra la posición de inicio de la cadena de Search dentro de la cadena de Source. En caso contrario, Result será cero.

### Afectar a las marcas de estado matemático

No

**Fallos mayores/menores**

Ocurrirá un fallo menor si:	Tipo de fallo	Código de fallo
El valor LEN de la etiqueta de cadena es mayor que el tamaño de DATA de la etiqueta de cadena.	4	51
El valor de inicio no es válido o la cadena de Source está vacía.	4	56

No es específico para esta instrucción. Consulte *Atributos comunes* para los fallos relacionados con los operandos

**Ejecución****Diagrama de escalera**

Condición	Acción
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	La instrucción se ejecuta.
Post-escaneado	N/A

**Texto estructurado**

Condición	Acción
Pre-escaneado	Consultar Pre-escaneado en la tabla de diagrama de escalera
Ejecución normal	Consultar La condición de entrada de reglón es verdadera en la tabla Diagrama de escalera.
Post-escaneado	Consultar Post-escaneado en la tabla de Diagramas de escalera

**Ejemplo**

Un mensaje del terminal MessageView contiene diferentes componentes de información. Los diferentes componentes de información están separados por una contrabarra (\). Para localizar un componente de información, la instrucción FIND busca la contrabarra y guarda la posición en find\_pos.

### Diagrama de escalera



### Texto estructurado

```
IF MV_read.EM THEN
    FI ND(MV_msg,find,1,find_pos);
    MV_read.EM := 0;
END_IF;
```

### Consulte también

[Atributos comunes](#) en la página 923

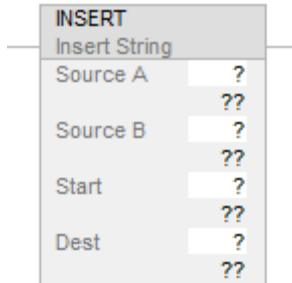
[Sintaxis de texto estructurado](#) en la página 955

[Conversiones de datos](#) en la página 927

## Insertar cadena (INSERT)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

Utiliza la instrucción INSERT para añadir caracteres ASCII a una ubicación concreta dentro de una cadena.

**Idiomas disponibles****Diagrama de escalera****Bloque de funciones****Texto estructurado**

```
INSERT (SourceA,SourceB,Start,Dest);
```

**Operandos**

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción. Consulte Conversión de datos. La instrucción INSERT utiliza los siguientes operandos.

**Diagrama de escalera y Texto estructurado**

Operando	Tipo	Formato	Descripción	Notas
Source A	Tipo de cadena	Etiqueta	Cadena a la que añadir los caracteres	Los tipos de cadena son STRING por defecto o cualquier otro tipo de cadena que usted cree
Source B	Tipo de cadena	Etiqueta	Cadena que contiene los caracteres a añadir	
Start	SINT DINT	Inmediato etiqueta	Posición en la que añadir los caracteres en Source A	Introduzca un número entre 1 y el tamaño de DATA del Source.
Destination	Tipo de cadena	Etiqueta	Cadena para almacenar el resultado	

Consulte Sintaxis de texto estructurado para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

## Descripción

La instrucción INSERT añade los caracteres al Source B en una posición designada dentro del Source A y coloca el resultado en el Destination.

- Inicio define el lugar de Source A en el que se añade ese Source B.
- A menos que Source A y Destination sean la misma etiqueta, Source A no cambia.

## Afectar a las marcas de estado matemático

No

## Fallos mayores/menores

Tipo	Código	Causa	Método de recuperación
4	51	El valor LEN de la etiqueta de cadena es mayor que el tamaño de DATA de la etiqueta de cadena.	<ol style="list-style-type: none"> <li>1. Comprueba que ninguna instrucción esté escribiendo en el miembro LEN de la etiqueta de cadena.</li> <li>2. En el valor LEN, introduzca el número de caracteres que contiene la cadena.</li> </ol>
4	56	El valor de inicio o cantidad no es válido.	Comprueba que el valor de inicio se encuentre entre 1 y el tamaño de DATA del Source.

## Ejecución

### Diagrama de escalera

Condición	Acción del Diagrama de escalera
Pre-escaneado	La condición de salida de reglón se establece en falsa.
La condición de entrada de reglón es falsa	La condición de salida de reglón se establece en falsa.
La condición de entrada de reglón es verdadera	La instrucción ejecuta. La condición de entrada de reglón se establece en verdadera.
Post-escaneado	La condición de salida de reglón se establece en falsa.

## Ejecución

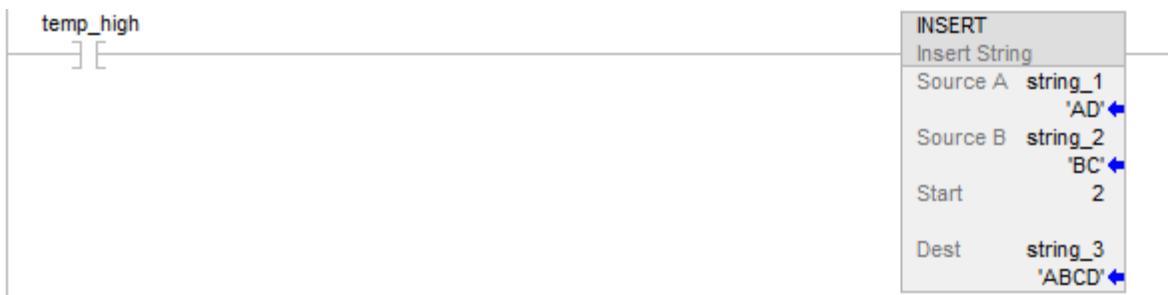
## Texto estructurado

Condición	Acción
Pre-escaneado	Consultar Pre-escaneado en la tabla de diagrama de escalera
Ejecución normal	Consultar La condición de entrada de reglón es verdadera en la tabla Diagrama de escalera.
Post-escaneado	Consultar Post-escaneado en la tabla de Diagramas de escalera

## Ejemplo

Cuando se establece *temp\_high* la instrucción INSERT añade los caracteres de *string\_2* en la posición 2 dentro de *string\_1* y coloca el resultado en *string\_3*.

## Diagrama de escalera



## **Texto estructurado**

```
IF temp_high THEN  
    INSER T(string_1,string_2,2,string_3);  
  
    temp_high := 0;  
  
END IF;
```

**Consulte también**

Instrucciones de cadena ASCII en la página 871

[Atributos comunes](#) en la página 923

Sintaxis de texto estructurado en la página 955

Conversiones de datos en la página 927

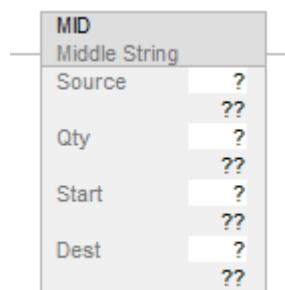
## Cadena central (MID)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción MID copia un número de caracteres ASCII especificados de una cadena y los almacena en otra cadena.

### Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

### Texto estructurado

MID(Source,Qty,Start,Dest);

### Operandos

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción. Consulte Conversión de datos.

**Diagrama de escalera y Texto estructurado**

Operando	Tipo	Formato	Descripción	Notas
Source	ANY_STRING	Etiqueta	La cadena de la que se debe copiar los caracteres	Los tipos de datos de cadena son: tipo de datos STRING predeterminado con una longitud máxima de 82 caracteres. cualquier tipo de cadena que usted cree con una longitud de caracteres configurable.
Quantity	SINT INT DINT	Immediato etiqueta	Número de caracteres que deben ser copiados	La suma de Start y Quantity debe ser menor o igual que la longitud del Source más 1.
Start	SINT INT DINT	Immediato etiqueta	La posición del primer carácter que se debe copiar	Introduzca un número entre 1 y el tamaño de DATA del Source.
Destination	ANY_STRING	Etiqueta	La cadena a la que se debe copiar los caracteres	

Consulte Sintaxis de texto estructurado para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

**Descripción**

La instrucción MID copia un grupo de caracteres de Source y coloca el resultado en Destination

- La posición Start y Quantity definen los caracteres que deben copiarse.
- A menos que Source y Destination sean la misma etiqueta, Source no cambia.

**Afectar a las marcas de estado matemático**

No

**Fallos mayores/menores**

Ocurrirá un fallo menor si:	Tipo de fallo	Código de fallo
El valor de LEN de la etiqueta de cadena de Source es mayor que el tamaño de DATA de la etiqueta de cadena de Source.	4	51

La longitud de la cadena de salida es mayor que el tamaño de DATA de la etiqueta de cadena de destino.	4	52
El valor de inicio o cantidad no es válido.	4	56

## Ejecución

### Diagrama de escalera

Condición	Acción del Diagrama de escalera
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	La instrucción se ejecuta.
Post-escaneado	N/A

### Texto estructurado

Condición	Acción
Pre-escaneado	Consultar Pre-escaneado en la tabla de diagrama de escalera
Ejecución normal	Consultar La condición de entrada de reglón es verdadera en la tabla Diagrama de escalera.
Post-escaneado	Consultar Post-escaneado en la tabla de Diagramas de escalera

## Ejemplo

En el transportador de equipajes de un aeropuerto, cada maleta tiene un código de barras. Los caracteres que van del 9 al 17 de ese código son el número de vuelo y el aeropuerto de destino de la maleta. Después de leer el código de barras (bag\_read.EM activo), la instrucción MID copia el número de vuelo y el aeropuerto de destino en la cadena bag\_flt\_and\_dest. Los siguientes reglones utilizarán bag\_flt\_and\_dest para determinar hacia dónde llevar la maleta.

### Diagrama de escalera



### Texto estructurado

```
IF bag_read.EM THEN  
    MID(bag_barcode,9,9, bag_flt_and_dest);  
    bag_read.EM := 0;  
END_IF;
```

### Consulte también

[Atributos comunes](#) en la página 923

[Sintaxis de texto estructurado](#) en la página 955

[Conversiones de datos](#) en la página 927

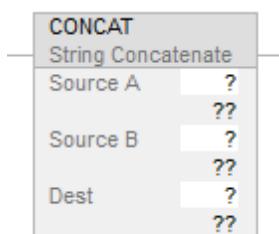
## Concatenar cadenas (CONCAT)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción CONCAT añade caracteres ASCII al final de una cadena.

### Idiomas disponibles

#### Diagrama de escalera



### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

### Texto estructurado

```
CONCAT(SourceA,SourceB,Dest);
```

## Operandos

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción. Consulte Atributos comunes para obtener más información sobre Conversión de datos.

### Diagrama de escalera y Texto estructurado

Operando	Tipo	Formato	Descripción	Notas
Source A	ANY_STRING	etiqueta	Etiqueta que contiene los caracteres iniciales	Los tipos de datos de cadena son: <ul style="list-style-type: none"><li>• Tipo de datos STRING predeterminado con una longitud máxima de 82 caracteres.</li></ul>
Source B	ANY_STRING	etiqueta	Etiqueta que contiene los caracteres finales	
Destination	ANY_STRING	etiqueta	Etiqueta para almacenar el resultado	<ul style="list-style-type: none"><li>• Cualquier tipo de cadena que usted cree con una longitud de caracteres configurable.</li></ul>

Consulte Atributos de texto estructurado para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

### Descripción

La instrucción CONCAT combina los caracteres de Source A con los caracteres de Source B y coloca el resultado en el Destination.

Los caracteres de Source A van en primer lugar seguidos de los caracteres de Source B.

A menos que Source A y Destination sean la misma etiqueta, Source A no cambia.

### Afectar a las marcas de estado matemático

No

### Fallos mayores/menores

Ocurrirá un fallo menor si:	Tipo de fallo	Código de fallo
El valor LEN de la etiqueta de cadena es mayor que el tamaño de DATA de la etiqueta de cadena.	4	51

La suma de la longitud de Source A y Source B es mayor que el tamaño de DATA de la etiqueta de cadena.	4	51
--	---	----

Consulte Índice a través de matrices para ver si hay fallos de indexación de matrices.

### Ejecución

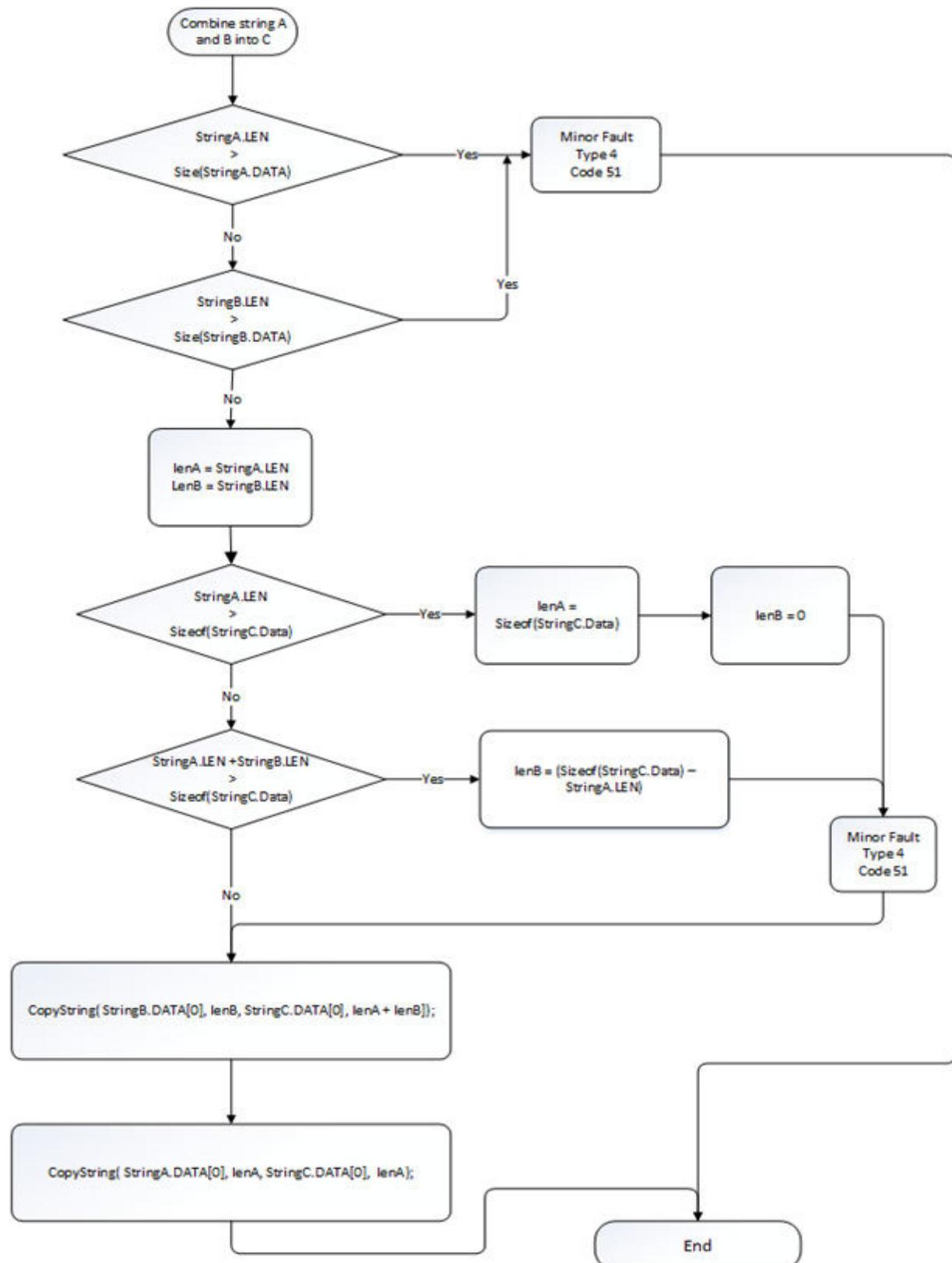
#### Diagrama de escalera

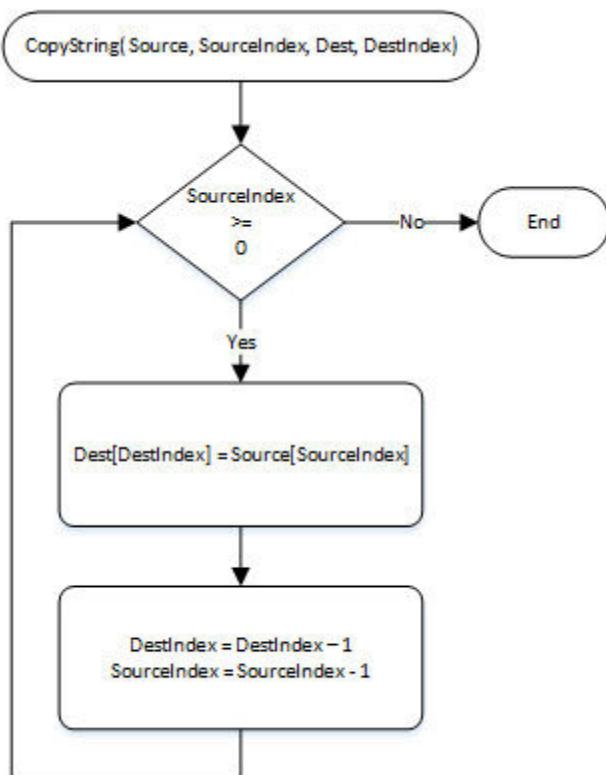
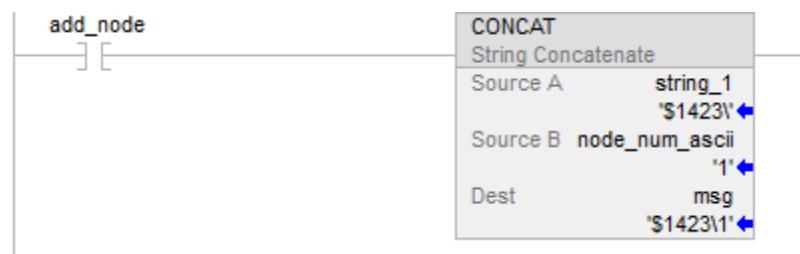
Condición	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	La instrucción se ejecuta.
Post-escaneado	N/A

#### Texto estructurado

Condición	Acción realizada
Pre-escaneado	Consultar Pre-escaneado en la tabla Diagrama de escalera.
Ejecución normal	Consultar La condición de entrada de reglón es verdadera en la tabla Diagrama de escalera.
Post-escaneado	Consultar Post-escaneado en la tabla Diagrama de escalera.

### Diagrama de flujo de cadena Concat



**Ejemplo****Diagrama de escalera****Texto estructurado**

```
CONCAT(string_1,string_2,msg);
```

**Consulte también**

[Atributos comunes](#) en la página 923

[Atributos de texto estructurado](#) en la página 983

[Conversiones de datos](#) en la página 927

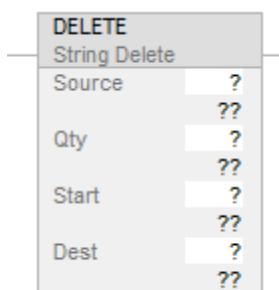
## Eliminar cadena (DELETE)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción DELETE elimina caracteres ASCII de una cadena.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

#### Texto estructurado

```
DELETE(Source,Qty,Start,Dest);
```

#### Operandos

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción. Consulte Conversión de datos.

#### Diagrama de escalera y Texto estructurado

Operando	Tipo	Formato	Descripción	Notas
Source	ANY_STRING	etiqueta	La etiqueta que contiene la cadena de la que desea borrar caracteres.	Los tipos de datos de cadena son: tipo de datos STRING predeterminado con una longitud máxima de 82 caracteres. cualquier tipo de cadena que usted cree con una longitud de caracteres configurable.

Quantity	SINT INT DINT	inmediato etiqueta	Número de caracteres que deben ser borrados	Start más Quantity deben ser menor o igual que la longitud del Source más 1.
Start	SINT INT DINT	inmediato etiqueta	La posición del primer carácter que debe ser borrado	Introduzca un número entre 1 y el tamaño de DATA del Source.
Destination	Tipo de cadena	etiqueta	La etiqueta para almacenar el resultado	

Consulte Sintaxis de texto estructurado para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

### Descripción

La instrucción DELETE borra (elimina) uno o más caracteres de Source y coloca los caracteres restantes en Destination.

- La posición Start y Quantity definen los caracteres que deben eliminarse.
- A menos que Source A y Destination sean la misma etiqueta, Source A no cambia.

### Afectar a las marcas de estado matemático

No

### Fallos mayores/menores

Ocurrirá un fallo menor si:	Tipo de fallo	Código de fallo
El valor de LEN de la etiqueta de cadena de Source es mayor que el tamaño de DATA de la etiqueta de cadena de Source.	4	51
La longitud de la cadena de salida es mayor que el tamaño de DATA de la etiqueta de cadena de destino.	4	52
El valor de inicio o cantidad no es válido.	4	56

Consulte los Atributos comunes para fallos relacionados con el operando.

## Ejecución

### Diagrama de escalera

Condición/estado	Acción
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	La instrucción se ejecuta.
Post-escaneado	N/A

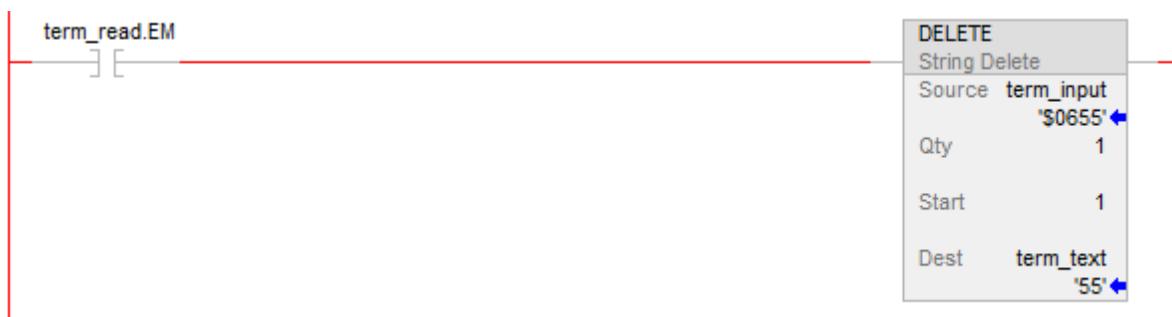
### Texto estructurado

Condición/estado	Acción
Pre-escaneado	Consultar Pre-escaneado en la tabla Diagrama de escalera.
Ejecución normal	Consultar La condición de entrada de reglón es verdadera en la tabla Diagrama de escalera.
Post-escaneado	Consultar Post-escaneado en la tabla Diagrama de escalera.

## Ejemplos

La información ASCII de un terminal contiene un carácter encabezado. Después de que el controlador haya leído los datos (term\_read.EM activo), la instrucción DELETE elimina el carácter encabezado. El controlador podrá entonces utilizar el texto del mensaje o pasarlo a otro dispositivo.

### Diagrama de escalera



### Texto estructurado

```

IF term_read.EM THEN
  DEL ETE(term_input,1,1,term_text);
  term_read.EM := 0;
  
```

END\_IF;

**Consulte también**

[Atributos comunes](#) en la página 923

[Sintaxis de texto estructurado](#) en la página 955

[Conversiones de datos](#) en la página 927

# Instrucciones de conversión ASCII

Utilice las instrucciones de conversión ASCII para convertir los datos en caracteres ASCII o transformar los caracteres ASCII.

## Instrucciones disponibles

### Diagrama de escalera y Texto estructurado

<u>STOD</u>	<u>STOR</u>	<u>RTO S</u>	<u>DTOS</u>	<u>LOWER</u>	<u>UPPE R</u>
-------------	-------------	------------------	-------------	--------------	-------------------

### Bloque de funciones

No disponible

Si desea convertir:	Utilice esta instrucción:
las representaciones ASCII de valores enteros en valores SINT, INT, DINT o REAL (p. ej. convertir el valor de una báscula o de otro dispositivo ASCII en un número entero que pueda usar en su lógica).	STOD
las representaciones ASCII de un valor de punto flotante en un valor REAL (p. ej. convertir el valor de una báscula o de otro dispositivo ASCII en un número entero que pueda usar en su lógica).	STOR
valores SINT, INT, DINT o REAL en una cadena de caracteres ASCII (p. ej. convertir una variable en una cadena ASCII para enviarla a un terminal MessageView™).	DTOS
valores REAL en una cadena de caracteres ASCII (p. ej. convertir una variable en una cadena ASCII para enviarla a un terminal MessageView).	RTOS
las letras de una cadena de caracteres ASCII en mayúsculas (p. ej. convertir en mayúsculas las entradas de un operador para poder buscarlas en una matriz).	UPPER
las letras de una cadena de caracteres ASCII en minúsculas (p. ej. convertir en minúsculas las entradas de un operador para poder buscarlas en una matriz).	LOWER

También puede utilizar las siguientes instrucciones para comparar o manipular caracteres ASCII:

Si desea:	Utilice esta instrucción:
Añadir caracteres al final de una cadena	CONCAT
Borrar caracteres de una cadena	DELETE
Determinar el carácter de inicio de una cadena secundaria	FIND
Insertar caracteres en una cadena	INSERT
Extraer caracteres de una cadena	MID
Reordenar los bytes de una etiqueta INT, DINT o REAL	SWPB
Comparar una cadena con otra	CMP
Determinar si unos caracteres son iguales que otros caracteres concretos	EQU
Determinar si unos caracteres son diferentes de otros caracteres concretos	NEQ
Determinar si unos caracteres son iguales o mayores que otros caracteres concretos	GEQ
Determinar si unos caracteres son mayores que otros caracteres concretos	GRT
Determinar si unos caracteres son iguales o menores que otros caracteres concretos	LEQ
Determinar si unos caracteres son menores que otros caracteres concretos	LES
Encontrar una cadena en una matriz de cadenas	FSC

#### Consulte también

[Códigos de error ASCII](#) en la página 868

[Tipos de cadena](#) en la página 867

## Instrucciones de conversión ASCII

#### Instrucciones disponibles

#### Diagrama de escalera y Texto estructurado

<a href="#">STOD</a>	<a href="#">STOR</a>	<a href="#">RTO S</a>	<a href="#">DTOS</a>	<a href="#">LOWER</a>	<a href="#">UPPE R</a>
----------------------	----------------------	---------------------------	----------------------	-----------------------	----------------------------

#### Bloque de funciones

No disponible

<b>Si desea convertir:</b>	<b>Utilice esta instrucción:</b>
las representaciones ASCII de valores enteros en valores SINT, INT, DINT o REAL (p. ej. convertir el valor de una báscula o de otro dispositivo ASCII en un número entero que pueda usar en su lógica).	STOD
las representaciones ASCII de un valor de punto flotante en un valor REAL (p. ej. convertir el valor de una báscula o de otro dispositivo ASCII en un número entero que pueda usar en su lógica).	STOR
valores SINT, INT, DINT o REAL en una cadena de caracteres ASCII (p. ej. convertir una variable en una cadena ASCII para enviarla a un terminal MessageView™).	DTOS
valores REAL en una cadena de caracteres ASCII (p. ej. convertir una variable en una cadena ASCII para enviarla a un terminal MessageView).	RTOS
las letras de una cadena de caracteres ASCII en mayúsculas (p. ej. convertir en mayúsculas las entradas de un operador para poder buscarlas en una matriz).	UPPER
las letras de una cadena de caracteres ASCII en minúsculas (p. ej. convertir minúsculas las entradas de un operador para poder buscarlas en una matriz).	LOWER

También puede utilizar las siguientes instrucciones para comparar o manipular caracteres ASCII:

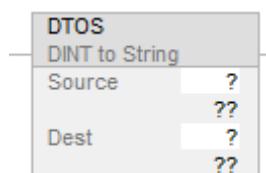
<b>Si desea:</b>	<b>Utilice esta instrucción:</b>
Añadir caracteres al final de una cadena	CONCAT
Borrar caracteres de una cadena	DELETE
Determinar el carácter de inicio de una cadena secundaria	FIND
Insertar caracteres en una cadena	INSERT
Extraer caracteres de una cadena	MID
Reordenar los bytes de una etiqueta INT, DINT o REAL	SWPB
Comparar una cadena con otra	CMP
Determinar si unos caracteres son iguales que otros caracteres concretos	EQU
Determinar si unos caracteres son diferentes de otros caracteres concretos	NEQ
Determinar si unos caracteres son iguales o mayores que otros caracteres concretos	GEQ
Determinar si unos caracteres son mayores que otros caracteres concretos	GRT

Determinar si unos caracteres son iguales o menores que otros caracteres concretos	LEQ
Determinar si unos caracteres son menores que otros caracteres concretos	LES
Encontrar una cadena en una matriz de cadenas	FSC

**Consulte también**[Códigos de error ASCII](#) en la página 868[Tipos de cadena](#) en la página 867**DINT en cadena  
(DTOS)**

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción DTOS produce la representación ASCII de un valor.

**Idiomas disponibles****Diagrama de escalera****Bloque de funciones**

Esta instrucción no está disponible en bloque de funciones.

**Texto estructurado**

```
DTOS(Source,Dest);
```

## Operandos

### Diagrama de escalera y Texto estructurado

Operando	Tipo	Formato	Descripción	Notas
Source	SINT INT DINT REAL	Etiqueta	La etiqueta que contiene el valor	Si el Source es un número REAL, la instrucción lo convierte en un valor DINT.
Destination	Tipo de cadena	Etiqueta	La etiqueta en la que se almacena el valor entero.	Los tipos de datos de cadena son: <ul style="list-style-type: none"> <li>• tipo de datos STRING predeterminado</li> <li>• cualquier tipo de cadena nuevo que usted cree</li> </ul>

### Descripción

La instrucción DTOS convierte el Source en una cadena de caracteres ASCII y coloca el resultado en el Destination.

### Afectar a las marcas de estado matemático

No

### Fallos mayores/menores

Tipo	Código	Causa	Método de recuperación
4	51	El valor LEN de la etiqueta de cadena es mayor que el tamaño de DATA de la etiqueta de cadena.	Comprueba que ninguna instrucción esté escribiendo en el miembro LEN de la etiqueta de cadena. En el valor LEN, introduzca el número de caracteres que contiene la cadena.
4	52	La cadena de salida es mayor que el destino.	Crear un nuevo tipo de cadena que sea lo bastante grande para la cadena de salida. Usar el nuevo tipo de cadena como tipo de datos para el destino.

Consulte los *Atributos comunes* para fallos relacionados con el operando.

## Ejecución

### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	La instrucción se ejecuta.
Post-escaneado	N/A

### Texto estructurado

Condición	Acción
Pre-escaneado	Consultar Pre-escaneado en la tabla Diagrama de escalera anterior
Ejecución normal	Consultar La condición de entrada de reglón es verdadera en la tabla Diagrama de escalera anterior
Post-escaneado	Consultar Post-escaneado en la tabla Diagrama de escalera anterior

## Ejemplo

Cuando temp\_high está establecido, la instrucción DTOS convierte el valor msg\_num en una cadena de caracteres ASCII y coloca el resultado en msg\_num\_ascii. Los siguientes reglones insertarán o concatenarán msg\_num\_ascii con otras cadenas para generar un mensaje completo que aparezca en un terminal.

### Diagrama de escalera



### Texto estructurado

```

IF temp_high THEN
    DT OS(msg_num,msg_num_ascii);
    temp_high := 0;
END_IF;

```

### Consulte también

[Atributos comunes](#) en la página 923

[Sintaxis de texto estructurado](#) en la página 955

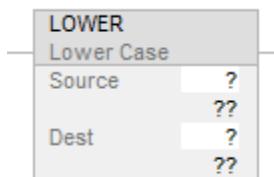
## Minúsculas (LOWER)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción LOWER convierte los caracteres alfabéticos en una cadena de caracteres en minúsculas.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

#### Texto estructurado

LOWER(Source,Dest);

#### Operandos

#### Diagrama de escalera y Texto estructurado

Operando	Tipo	Formato	Descripción
Source	Cadena	Etiqueta	Etiqueta que contiene los caracteres que desea convertir a minúsculas
Destinatio n	Caden a	Etiqueta	Etiqueta para almacenar los caracteres en minúscula

Consulte *Texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

### Descripción

La instrucción LOWER convierte a minúsculas todas las letras de Source y coloca el resultado en Destination.

- Los caracteres ASCII distinguen entre mayúsculas y minúsculas. La mayúscula A (\$41) no es igual a la minúscula a (\$61).
- Si los operadores introducen directamente caracteres ASCII, se convierten todos los caracteres a mayúsculas o minúsculas antes de compararlos.

Todos los caracteres de la cadena Source que no sean letras no se cambiarán.

### Afectar a las marcas de estado matemático

No

### Fallos mayores/menores

Tipo	Código	Causa	Método de recuperación
4	51	El valor LEN de la etiqueta de cadena es mayor que el tamaño de DATA de la etiqueta de cadena.	Comprueba que ninguna instrucción esté escribiendo en el miembro LEN de la etiqueta de cadena. En el valor LEN, introduzca el número de caracteres que contiene la cadena.
4	52	La cadena de salida es más mayor que el destino.	Crear un nuevo tipo de cadena que sea lo bastante grande para la cadena de salida. Usar el nuevo tipo de cadena como tipo de datos para el destino.

### Ejecución

#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	La instrucción se ejecuta.
Post-escaneado	N/A

### Texto estructurado

Condición	Acción
Pre-escaneado	Consultar Pre-escaneado en la tabla Diagrama de escalera anterior
Ejecución normal	Consultar La condición de entrada de región es verdadera en la tabla Diagrama de escalera anterior
Post-escaneado	Consultar Post-escaneado en la tabla Diagrama de escalera anterior

### Ejemplos

Para encontrar información sobre un elemento específico, un operador introduce el número de elemento en un terminal ASCII. Después de que el controlador haya leído la entrada desde el terminal (terminal\_read establecido), la instrucción LOWER convierte todos los caracteres de item\_number a mayúsculas y almacena el resultado en item\_number\_lower\_case. El siguiente reglón busca entonces una matriz de caracteres que coincide con item\_number\_lower\_case.

### Diagrama de escalera



### Texto estructurado

```

IF terminal_read THEN
    LOWER(item_number, item_number_lower_case);
    terminal_read := 0;
END_IF;

```

### Consulte también

[Atributos comunes](#) en la página 923

[Sintaxis de texto estructurado](#) en la página 955

## REAL en cadena (RTOS)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción RTOS produce la representación ASCII de un valor REAL.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

#### Texto estructurado

RTOS(Source,Dest);

#### Operandos

#### Diagrama de escalera y Texto estructurado

Operando	Tipo	Formato	Descripción	Notas
Source	REAL	Etiqueta	La etiqueta que contiene el valor REAL	
Destination	Tipo de cadena	Etiqueta	La etiqueta en la que se almacena el valor ASCII.	Los tipos de datos de cadena son: <ul style="list-style-type: none"><li>• Tipo de datos STRING predeterminado</li><li>• Cualquier tipo de cadena nuevo que usted cree</li></ul>

Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de expresiones.

#### Descripción

La instrucción RTOS convierte Source en una cadena de caracteres ASCII y coloca el resultado en Destination.

### Afectar a las marcas de estado matemático

No

#### Fallos mayores/menores

Tipo	Código	Causa	Método de recuperación
4	52	La cadena de salida es más mayor que el destino.	Crear un nuevo tipo de cadena que sea lo bastante grande para la cadena de salida. Usar el nuevo tipo de cadena como tipo de datos para el destino.

Consulte los *Atributos comunes* para fallos relacionados con el operando.

### Ejecución

#### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	La instrucción se ejecuta.
Post-escaneado	N/A

#### Texto estructurado

Condición	Acción
Pre-escaneado	Consultar Pre-escaneado en la tabla Diagrama de escalera anterior
Ejecución normal	Consultar La condición de entrada de reglón es verdadera en la tabla Diagrama de escalera anterior
Post-escaneado	Consultar Post-escaneado en la tabla Diagrama de escalera anterior

### Ejemplos

Cuando send\_data está habilitado, la instrucción RTOS convierte el valor data\_1 en una cadena de caracteres ASCII y coloca el resultado en data\_1\_ascii. Los siguientes reglones insertarán o concatenarán data\_1\_ascii con otras cadenas para generar un mensaje completo que aparezca en un terminal.

**Diagrama de escalera****Texto estructurado**

```
IF send_data THEN
    RTOS(data_1,data_1_ascii);
    send_data:= 0;
END_IF;
```

**Consulte también**

[Atributos comunes](#) en la página 923

[Sintaxis de texto estructurado](#) en la página 955

**Cadena en DINT  
(STOD)**

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción STOD convierte la representación ASCII de un entero a un valor entero o REAL.

**Idiomas disponibles****Diagrama de escalera**

## Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

## Texto estructurado

STOD(Source,Dest);

### Operандos

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción. Consulte *Conversión de datos*.

### Diagrama de escalera y Texto estructurado

Operando	Tipo	Formato	Descripción	Notas
Source	Tipo de cadena	Etiqueta	La etiqueta que contiene el valor ASCII	Los tipos de datos de cadena son: • Tipo de datos STRING predeterminado • Cualquier tipo de cadena nuevo que usted cree
Destination	SINT INT DINT	Etiqueta	La etiqueta en la que se almacena el valor entero.	Si el valor Source es un número de punto flotante, la instrucción convierte solo la parte no fraccionaria del número (independientemente del tipo de datos de destino).

Consulte *Sintaxis de texto estructurado* para obtener más información sobre la sintaxis de expresiones.

### Descripción

La instrucción STOD convierte el Source en un entero y coloca el resultado en Destination.

- La instrucción convierte números negativos y positivos.
- Si la cadena Source contiene caracteres no numéricos, STOD convierte el primer conjunto de números que esté contiguos:

La instrucción ignora cualquier control inicial o cualquier carácter no numérico, excepto el signo menos que está delante del número.

Si la cadena contiene muchos grupos de números separados por un delimitador (p. ej. /) la instrucción convierte solo el primer grupo de números.

### Afectar a las marcas de estado matemático

Solo en Diagramas de escalera. Consulte *Marcas de estado matemático*.

### Fallos mayores/menores

Tipo	Código	Causa	Método de recuperación
4	51	El valor LEN de la etiqueta de cadena es mayor que el tamaño de DATA de la etiqueta de cadena.	Comprueba que ninguna instrucción esté escribiendo en el miembro LEN de la etiqueta de cadena. En el valor LEN, introduzca el número de caracteres que contiene la cadena.
4	53	El número de salida sobrepasa los límites del tipo de datos de destino.	<ul style="list-style-type: none"> <li>• Reducir el tamaño del valor ASCII o</li> <li>• Usar un tipo de datos mayor para el destino</li> </ul>

Consulte los *Atributos comunes* para fallos relacionados con el operando.

### Ejecución

#### Diagrama de escalera

Condición	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	La instrucción se ejecuta. Destination se ha borrado La instrucción convierte el Source.
Post-escaneado	N/A

#### Texto estructurado

Condición	Acción
Pre-escaneado	Consultar Pre-escaneado en la tabla Diagrama de escalera anterior

Ejecución normal	Consultar La condición de entrada de reglón es verdadera en la tabla Diagrama de escalera anterior
Post-escaneado	Consultar Post-escaneado en la tabla Diagrama de escalera anterior

### Ejemplo

Cuando MV\_read.EM está establecido, la instrucción STOD convierte el primer grupo de caracteres numéricos de MV\_msg en un valor entero. Esta instrucción ignora el carácter de control inicial (\$06) y se detiene en el delimitador (\).

### Diagrama de escalera



### Texto estructurado

```

IF MV_read.EM THEN
    STOD(MV_msg,MV_msg_nmbr);
    MV_read.EM := 0;
END_IF;

```

### Consulte también

[Atributos comunes](#) en la página 923

[Sintaxis de texto estructurado](#) en la página 955

[Conversiones de datos](#) en la página 927

[Marcas de estado matemático](#) en la página 923

## Cadena en REAL (STOR)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción STOR convierte la representación ASCII de un valor de punto flotante en un valor REAL.

### Idiomas disponibles

#### Diagrama de escalera



#### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

#### Texto estructurado

STOR(Source,Dest);

#### Operандos

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción. Consulte *Conversión de datos*.

#### Diagrama de escalera y Texto estructurado

Operando	Tipo	Formato	Descripción	Notas
Source	Tipo de cadena	etiqueta	La etiqueta que contiene el valor ASCII	Los tipos de datos de cadena son: <ul style="list-style-type: none"> <li>• Tipo de datos STRING predeterminado</li> <li>• Cualquier tipo de cadena nuevo que usted cree</li> </ul>
Destination	REAL	etiqueta	La etiqueta en la que se almacena el valor REAL.	

Consulte Texto estructurado para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

## Descripción

La instrucción STOR convierte el Source en un número REAL y coloca el resultado en el Destination.

- La instrucción convierte números negativos y positivos.
- Si la cadena Source contiene caracteres no numéricos, STOR convierte el primer conjunto de números que esté contiguos incluido el punto decimal [.].

La instrucción ignora cualquier control inicial o cualquier carácter no numérico (excepto el signo menos que está delante del número).

Si la cadena contiene muchos grupos de números separados por un delimitador (p. ej. /) la instrucción convierte solo el primer grupo de números.

## Afectar a las marcas de estado matemático

Condicional, basado en el lenguaje de programación. Consulte *Marcas de estado matemático*.

## Fallos mayores/menores

Tipo	Código	Causa	Método de recuperación
4	51	El valor LEN de la etiqueta de cadena es mayor que el tamaño de DATA de la etiqueta de cadena.	Comprueba que ninguna instrucción esté escribiendo en el miembro LEN de la etiqueta de cadena. En el valor LEN, introduzca el número de caracteres que contiene la cadena.
4	53	El número de salida sobrepasa los límites del tipo de datos de destino.	<ul style="list-style-type: none"> <li>• Reducir el tamaño del valor ASCII o</li> <li>• Usar un tipo de datos mayor para el destino</li> </ul>

Consulte los *Atributos comunes* para fallos relacionados con el operando.

## Ejecución

### Diagrama de escalera

Condición	Acción del Diagrama de escalera
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A

La condición de entrada de reglón es verdadera	La instrucción se ejecuta.
Post-escaneado	N/A

### Texto estructurado

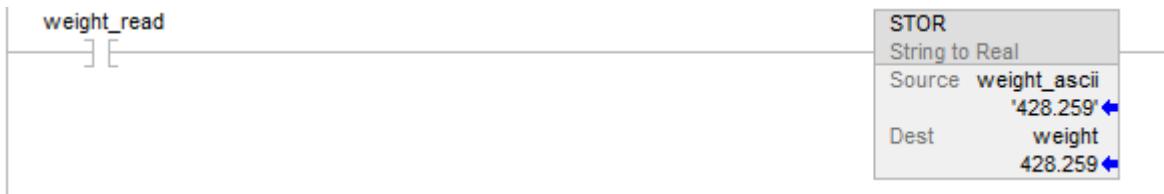
Condición	Acción
Pre-escaneado	Consultar Pre-escaneado en la tabla Diagrama de escalera anterior
Ejecución normal	Consultar La condición de entrada de reglón es verdadera en la tabla Diagrama de escalera anterior
Post-escaneado	Consultar Post-escaneado en la tabla Diagrama de escalera anterior

### Ejemplo

Después de leer el peso en una escala (weight\_read está establecido), la instrucción STOR convierte los caracteres numéricos de weight\_ascii en un valor REAL.

Es posible que perciba una pequeña diferencia entre las partes fraccionarias de Source y Destination.

### Diagrama de escalera



### Texto estructurado

IF weight\_read THEN

    STO R(weight\_ascii,weight);

END\_IF;

### Consulte también

[Atributos comunes](#) en la página 923

[Sintaxis de texto estructurado](#) en la página 955

[Conversiones de datos](#) en la página 927

[Marcas de estado matemático](#) en la página 923

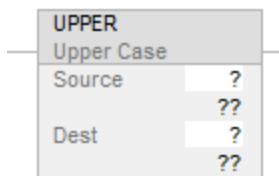
## Mayúsculas (UPPER)

Esta información es aplicable a los controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, GuardLogix 5570, Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580 y GuardLogix 5580.

La instrucción UPPER convierte los caracteres alfabéticos en una cadena de caracteres en mayúscula.

### Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

### Texto estructurado

UPPER(Source,Dest);

### Operandos

### Diagrama de escalera y Texto estructurado

Operando	Tipo	Formato	Descripción
Source	Cadena	etiqueta	Etiqueta que contiene los caracteres que desea convertir a mayúsculas
Destination	Cadena	etiqueta	Etiqueta para almacenar los caracteres en mayúscula

Consulte *Texto estructurado* para obtener más información sobre la sintaxis de las expresiones dentro de texto estructurado.

## Descripción

La instrucción UPPER convierte a mayúsculas todas las letras de Source y coloca el resultado en Destination.

- Los caracteres ASCII distinguen entre mayúsculas y minúsculas. La mayúscula A (\$41) no es igual a la minúscula a (\$61).
- Si los operadores introducen directamente caracteres ASCII, se convierten todos los caracteres a mayúsculas o minúsculas antes de compararlos.

Todos los caracteres de la cadena Source que no sean letras no se cambiarán.

## Afectar a las marcas de estado matemático

No

## Fallos mayores/menores

Tipo	Código	Causa	Método de recuperación
4	51	El valor LEN de la etiqueta de cadena es mayor que el tamaño de DATA de la etiqueta de cadena.	Comprueba que ninguna instrucción esté escribiendo en el miembro LEN de la etiqueta de cadena. En el valor LEN, introduzca el número de caracteres que contiene la cadena.
4	52	La cadena de salida es más mayor que el destino.	Crear un nuevo tipo de cadena que sea lo bastante grande para la cadena de salida. Usar el nuevo tipo de cadena como tipo de datos para el destino.

## Ejecución

### Diagrama de escalera

Condición/estado	Acción realizada
Pre-escaneado	N/A
La condición de entrada de reglón es falsa	N/A
La condición de entrada de reglón es verdadera	La instrucción se ejecuta.
Post-escaneado	N/A

## Texto estructurado

Condición	Acción
Pre-escaneado	Consultar Pre-escaneado en la tabla Diagrama de escalera anterior

Ejecución normal	Consultar La condición de entrada de reglón es verdadera en la tabla Diagrama de escalera anterior
Post-escaneado	Consultar Post-escaneado en la tabla Diagrama de escalera anterior

### Ejemplo

Para encontrar información sobre un elemento específico, un operador introduce el número de catálogo del elemento en un terminal ASCII. Después de que el controlador haya leído la entrada desde el terminal (terminal\_read se establece), la instrucción UPPER convierte todos los caracteres de catalog\_number a mayúsculas y almacena el resultado en catalog\_number\_upper\_case. El siguiente reglón busca entonces una matriz de caracteres que coincida con catalog\_number\_upper\_case.

### Diagrama de escalera



### Texto estructurado

```

IF terminal_read THEN
    UPPER(catalog_number,catalog_number_upper_case);
    terminal_read := 0;
END_IF;
  
```

### Consulte también

[Atributos comunes](#) en la página 923

[Sintaxis de texto estructurado](#) en la página 955



# **Instrucciones de depuración**

## **Instrucciones de depuración**

Estas instrucciones son compatibles con el software Studio 5000 Logix Emulate solo, que permiten emular un controlador LOGIX 5000 en una computadora personal.

Utilice las instrucciones de depuración para vigilar el estado de la lógica cuando está en las condiciones determinadas por usted.

### **Instrucciones disponibles**

<a href="#">BPT</a>	<a href="#">TPT</a>
---------------------	---------------------

### **Bloque de funciones**

No disponible

### **Texto estructurado**

No disponible

<b>Si desea:</b>	<b>Utilice esta instrucción:</b>
Detener la emulación de un programa cuando el reglón es verdadero.	BPT
Registrar los datos que usted seleccione cuando un reglón es verdadero.	TPT

### **Consulte también**

[Instrucciones de cálculo/matemáticas](#) en la página 425

[Comparar instrucciones](#) en la página 369

[Instrucciones de bit](#) en la página 143

[Instrucciones de cadena ASCII](#) en la página 871

[Instrucciones de conversión ASCII](#) en la página 892

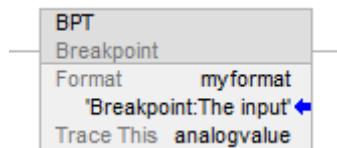
## Puntos de interrupción (BPT)

Esta instrucción es compatible con los controladores Studio 5000 Logix Emulate solo.

Utilice las instrucciones de depuración para vigilar el estado de la lógica cuando está en las condiciones determinadas por usted.

### Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

### Operandos

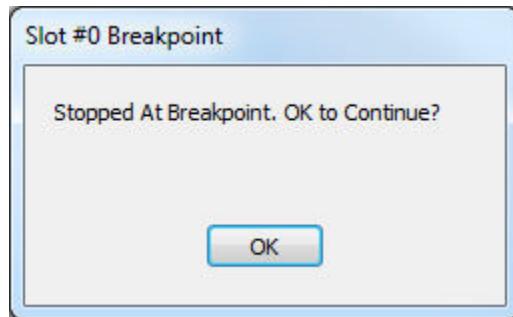
Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción. Consulte Conversión de datos.

### Diagrama de escalera

Operando	Tipo	Formato	Descripción
Format	Cadena	Etiqueta	Una cadena que establece el formato del texto que aparece en la ventana de rastreo del punto de interrupción.
Trace This	BOOL, SINT, INT, DINT, REAL	Etiqueta	La etiqueta que tiene el valor que usted desea que aparezca en la ventana de rastreo.

## Descripción

Los puntos de ruptura se programan con la instrucción de salida punto de ruptura (BPT). Cuando las entradas de un reglón que contiene instrucciones BPT son verdaderas, la instrucción BPT detiene la ejecución del programa. El programa muestra una ventana que indica que se produjo un punto de interrupción y el valor que lo desencadenó.



Cuando se desencadena un punto de interrupción el emulador muestra una ventana que le informa de que se ha producido un punto de interrupción. La barra de título de la ventana muestra la ranura que contiene el emulador que encontró el punto de interrupción.

Cuando usted hace clic en OK, el emulador reanuda la ejecución del programa. Si las condiciones que desencadenan el punto de interrupción persisten, volverá a desencadenarse el punto de interrupción.

Además, el emulador abre una ventana de rastreo del punto de interrupción. La ventana de rastreo muestra información sobre el punto de interrupción y los valores.

**Importante:** Cuando se desencadena el punto de interrupción, no podrá editar su proyecto hasta que permita que continúe la ejecución. Puede entrar en línea con el emulador para observar el estado de su proyecto, pero no podrá editarlo. Si intenta aceptar la edición de un reglón mientras el punto de interrupción está desencadenado, aparecerá un cuadro de diálogo indicando que el controlador no está en el modo correcto.

## Formato de cadena

Con la cadena de formato en las instrucciones de puntos de rastreo y de interrupción, podrá controlar cómo aparecen las etiquetas rastreadas en las ventanas de rastreo o de punto de interrupción. El formato de la cadena es:

- encabezado:(texto)%(%tipo)

donde el encabezado es una cadena de texto que identifica el punto de rastreo o el punto de interrupción, el texto es una cadena que describe la etiqueta (o cualquier otro texto que usted seleccione) y %(tipo) indica el formato de esa etiqueta.

Necesitará un indicador de tipo para cada etiqueta que esté rastreando con la instrucción de punto de rastreo o de punto de interrupción.

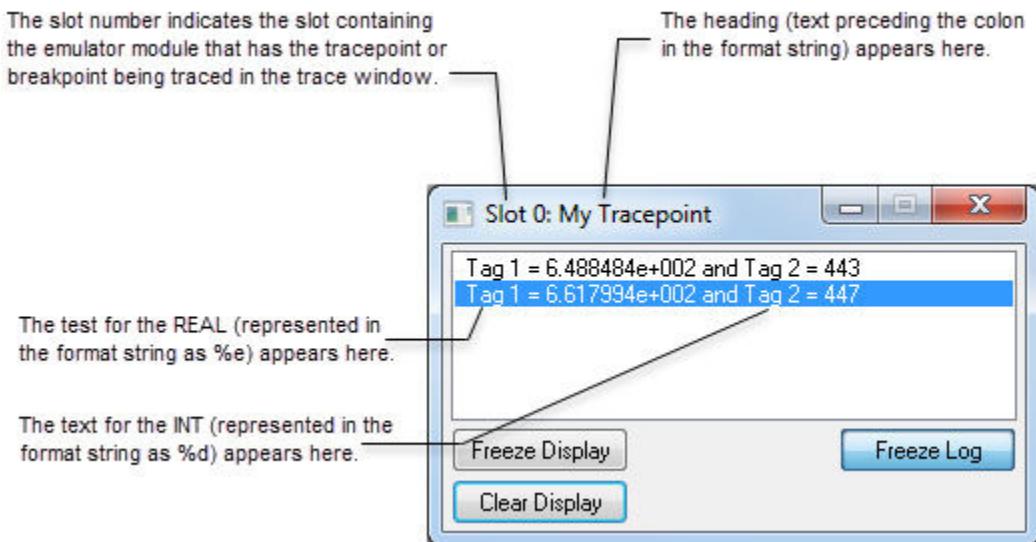
Por ejemplo, podría formatear la cadena de punto de rastreo como sigue.

- Mi punto de rastreo: Etiqueta 1 = %e y Etiqueta 2 = %d

El %e formatea la primera etiqueta rastreada como el punto flotante de doble precisión con un exponente y %d formatea la segunda etiqueta rastreada como entero decimal con signo.

En este caso tendría una instrucción de punto de rastreo que tiene dos operandos Trace This (uno para REAL y otro para INT, aunque el valor de cualquier etiqueta puede formatearse con cualquier marca).

La ventana de rastreo resultante que aparecería cuando se desencadena el punto de rastrea sería como la de este ejemplo.



### Afectar a las marcas de estado matemático

No

### Condiciones de fallo

No es específico para esta instrucción. Consulte los Atributos comunes para fallos relacionados con el operando.

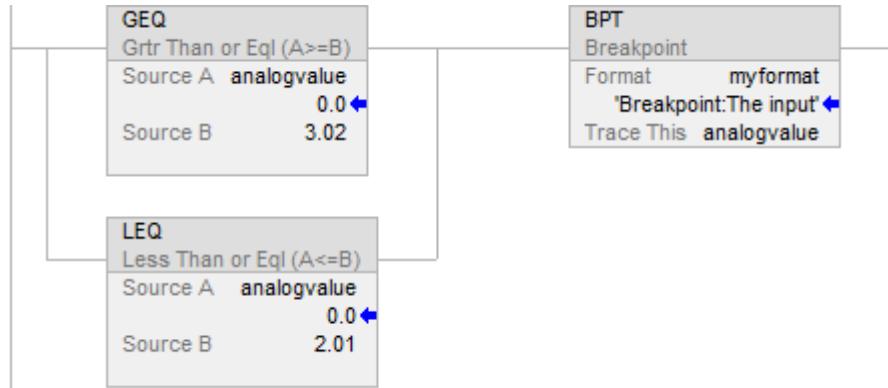
## Ejecución

Condición	Acción realizada
Pre-escaneado	El reglón pasa a falso.
La condición de entrada de reglón es falsa	El reglón pasa a falso.
La condición de entrada de reglón es verdadera	El reglón pasa a verdadero. La ejecución salta hasta el reglón que contiene la instrucción LBL con el nombre de etiqueta al que se hace referencia.
Post-escaneado	El reglón pasa a falso.

## Ejemplos

Con la instrucción BPT se pueden mostrar muchos valores de etiqueta. Sin embargo, la cadena de formato puede contener solo 82 caracteres. Puesto que la cadena de formato requiere de dos caracteres para cada una de las etiquetas que deseé en el punto de interrupción, no podrá rastrear más de 41 etiquetas con una sola instrucción BPT. Sin embargo, para separar los datos de las etiquetas en sus rastreos, necesitará incluir espacios y otro formato. De esta manera reducirá a mucho menos de 41 el número de valores de etiqueta que puede rastrear eficazmente con una instrucción BPT.

El reglón muestra un punto de interrupción que detiene la ejecución del programa cuando un valor analógico sea mayor de 3,02 o menor de 2,01.

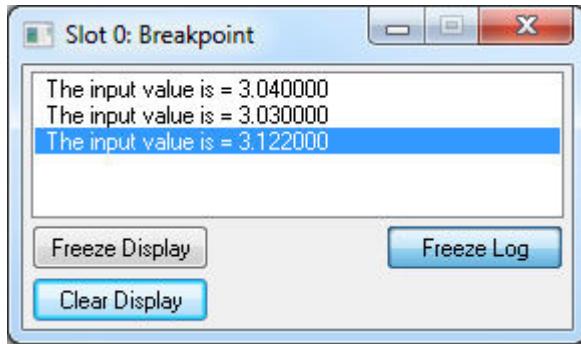


Muestra la información del punto de interrupción en la cadena de formato (miformato). En este caso, la cadena de formato contiene el siguiente texto:

- Punto de interrupción: El valor de entrada es %f

Cuando se activa el punto de interrupción, la ventana de rastreo del punto de interrupción muestra en su barra de título los caracteres que aparecen antes de los dos puntos ('Breakpoint'). Los otros caracteres son parte del rastreo. En este ejemplo, %f representa la primera (y, en este caso, la única) etiqueta que debe rastrearse ('analogvalue').

Los rastreos resultantes aparecerán como se muestra a continuación:



### Consulte también

[Atributos comunes](#) en la página 923

[Conversiones de datos](#) en la página 927

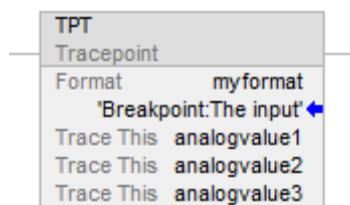
## Puntos de rastreo (TPT)

Esta instrucción es compatible con los controladores Studio 5000 Logix Emulate solo.

Datos de registro de puntos de rastreo que usted selecciona cuando un reglón es verdadero.

### Idiomas disponibles

### Diagrama de escalera



### Bloque de funciones

Esta instrucción no está disponible en bloque de funciones.

### Texto estructurado

Esta instrucción no está disponible en texto estructurado.

### Operandos

Existen reglas de conversión de datos para utilizar tipos de datos mixtos en una instrucción. Consulte Conversión de datos.

### Diagrama de escalera

Operando	Tipo	Formato	Descripción
Format	Cadena	Etiqueta	Una cadena que establece el formato de los informes de rastreo (tanto en pantalla como de los registrados en el disco).
Trace This	BOOL SINT INT DINT REAL	Etiqueta	La etiqueta que desea rastrear.

### Descripción

Los puntos de rastreo se programan con la instrucción de salida puntos de rastreo (TPT). Cuando las entradas de un reglón que contiene una instrucción TPT son verdaderas, la instrucción TPT escribe una entrada de rastreo en una pantalla de rastreo o en un archivo de registro.

Con la instrucción TPT se pueden rastrear muchas etiquetas. Sin embargo, la cadena de formato puede contener solo 82 caracteres. Puesto que la cadena de ganancia derivativa requiere dos caracteres para cada una de las etiquetas que deseé rastrear, no podrá rastrear más de 41 etiquetas con una sola instrucción TPT. Sin embargo, para separar los datos de las etiquetas en sus rastreos, necesitará incluir espacios y otro formato. De esta manera reducirá a mucho menos de 41 el número de etiquetas que puede rastrear eficazmente con una instrucción TPT.

### Formato de cadena

Con la cadena de formato en las instrucciones de puntos de rastreo y de interrupción, podrá controlar cómo aparecen las etiquetas rastreadas en las ventanas de rastreo o de punto de interrupción. El formato de la cadena se muestra a continuación:

- encabezado:(texto)% (tipo)

donde el encabezado es una cadena de texto que identifica el punto de rastreo o el punto de interrupción, el texto es una cadena que describe la etiqueta (o cualquier otro texto que usted seleccione) y %(tipo) indica el formato de esa etiqueta. Necesitará un indicador de tipo para cada etiqueta que esté rastreando con la instrucción de punto de rastreo o de punto de interrupción.

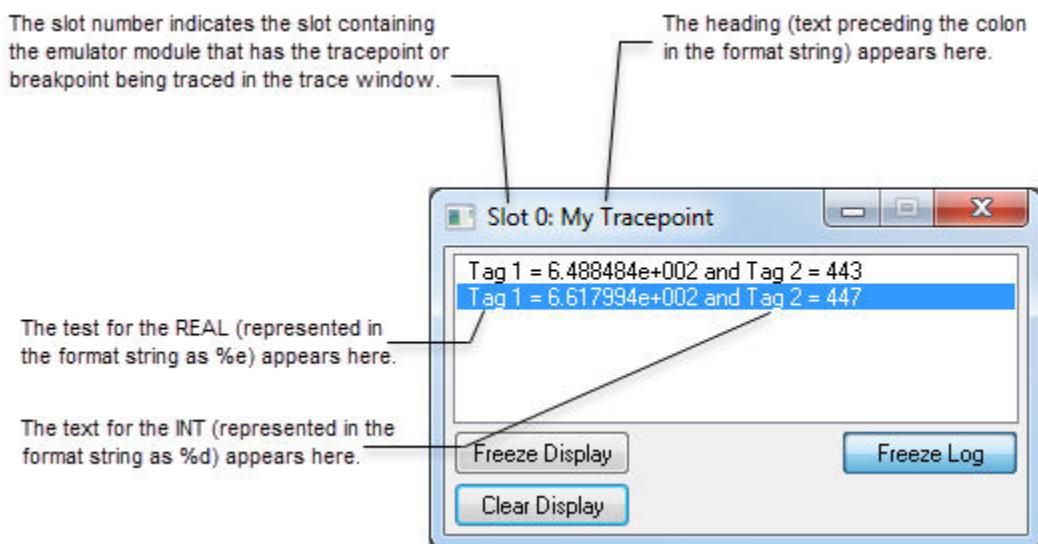
Por ejemplo, podría formatear la cadena de punto de rastreo como sigue:

- Mi punto de rastreo: Etiqueta 1 = %e y Etiqueta 2 = %d

El %e formatea la primera etiqueta rastreada como el punto flotante de doble precisión con un exponente y %d formatea la segunda etiqueta rastreada como entero decimal con signo.

En este caso tendría una instrucción de punto de rastreo que tiene dos operandos Trace This (uno para REAL y otro para INT, aunque el valor de cualquier etiqueta puede formatearse con cualquier marca).

La ventana de rastreo resultante que aparecería cuando se desencadena el punto de rastreo sería como la de este ejemplo.



### Afectar a las marcas de estado matemático

No

### Condiciones de fallo

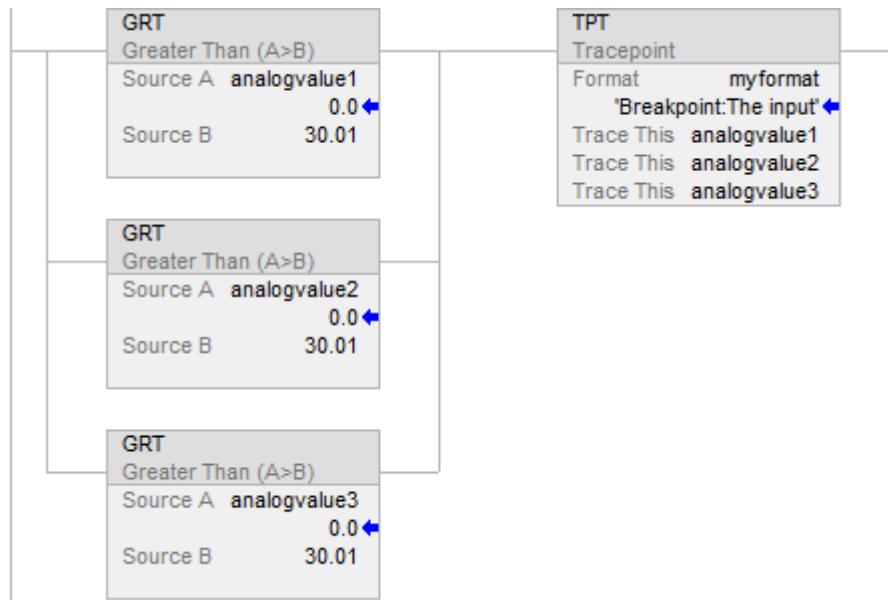
No es específico para esta instrucción. Consulte los Atributos comunes para fallos relacionados con el operando.

### Ejecución

Condición	Acción de la escalera de relés
Pre-escaneado	El reglón pasa a falso.
La condición de entrada de reglón es falsa	El reglón pasa a falso.
La condición de entrada de reglón es verdadera	El reglón pasa a verdadero. La ejecución salta hasta el reglón que contiene la instrucción LBL con el nombre de etiqueta al que se hace referencia.
Post-escaneado	El reglón pasa a falso.

## Ejemplo

El reglón desencadena un rastreo de tres valores analógicos cuando cualquiera de ellos excede un valor dado (30.01).



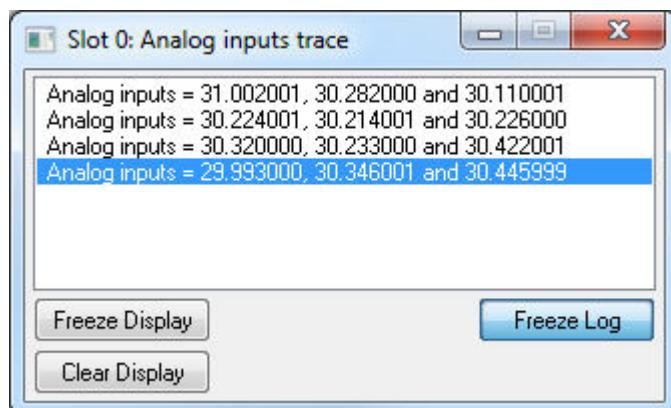
Muestra la información del punto de rastreo en la cadena de formato (miformato).

En este caso, la cadena de formato contiene el siguiente texto:

- Rastreo de entradas analógicas: Entradas analógicas = %f, %f, y %f

Cuando se desencadena el punto de rastreo, los caracteres que aparecen delante de los dos puntos ('Rastreo de entradas analógicas') aparecerán en la barra de títulos de la ventana de rastreo. Los otros caracteres son parte del rastreo. En este ejemplo, %f representa a las etiquetas que se van a rastrear ('analogvalue1', 'analogvalue2', and 'analogvalue3').

Los rastreos resultantes aparecerán como se muestra a continuación:



Cuando los rastreos se registran en el disco, los caracteres que aparecen antes de los dos puntos aparecen en el rastreo.

Esto indica qué punto de rastreo causó qué entrada de rastreo. Esto es un ejemplo de una entrada de rastreo. 'Rastreo de entradas analógicas:' es el encabezado de la cadena de formato del punto de rastreo.

Rastreo de entradas analógicas: Entradas analógicas = 31.00201, 30.282000 y 30.110001.

**Consulte también**

[Instrucciones de depuración](#) en la página 913

[Punto de interrupción \(BPT\)](#) en la página 914

[Atributos comunes](#) en la página 923

[Conversiones de datos](#) en la página 927

# **Atributos comunes para instrucciones generales**

Siga las pautas en este capítulo para los atributos comunes para las Instrucciones generales.

## **Atributos comunes**

Para más información sobre atributos comunes a las instrucciones de LOGIX 5000™, haga clic en cualquiera de los siguientes temas.

[Marcas de estado matemático](#) en la página 923

[Valores inmediatos](#) en la página 926

[Conversiones de datos](#) en la página 927

[Tipos de datos](#) en la página 931

[Tipos de datos LINT](#) en la página 933

[Valores de punto flotante](#) en la página 934

[Índice a través de matrices](#) en la página 936

[Direccionamiento de bit](#) en la página 936

## **Indicadores de estado matemáticos**

Siga las pautas proporcionadas en este tema para los Indicadores de estado matemáticos.

### **Descripción**

<b>Controladores</b>	<b>Descripción</b>
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Conjunto de Indicadores de estado matemáticos a los que puede acceder directamente con instrucciones. Estos indicadores solo se actualizan en rutinas de diagrama de escalera y no son etiquetas. Además, no son aplicables alias de indicadores.

Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Conjunto de Indicadores de estado matemáticos a los que puede acceder directamente con instrucciones. Estos indicadores se actualizan en todos los tipos de rutina, pero no son etiquetas. Además, no son aplicables alias de indicadores.
---	--

### Indicadores de estado

Indicador de estado	Descripción <b>(Para Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.)</b>	Descripción <b>(Para Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570)</b>
S:FS Indicador de primer escaneado	<p>El indicador de primer escaneado lo establece el controlador:</p> <ul style="list-style-type: none"> <li>• La primera vez que se escanea un programa después de que el controlador pase al modo Marcha.</li> <li>• La primera vez que se escanea un programa después de que se desinhiba el programa.</li> <li>• Cuando se llama una rutina desde una acción SFC y el paso que posee la Acción se escanea primero.</li> </ul> <p>Use el indicador de primer escaneado para inicializar datos para usarlos en posteriores escaneados. También se conoce como el bit de primer pase.</p>	<p>El indicador de primer escaneado lo establece el controlador:</p> <ul style="list-style-type: none"> <li>• La primera vez que se escanea un programa después de que el controlador pase al modo Marcha.</li> <li>• La primera vez que se escanea un programa después de que se desinhiba el programa.</li> <li>• Cuando se llama una rutina desde una acción SFC y el paso que posee la Acción se escanea primero.</li> </ul> <p>Use este indicador para inicializar datos para usarlos en posteriores escaneados. También se conoce como el bit de primer pase.</p>
S:N Indicador negativo	El controlador establece el indicador negativo cuando el resultado de una operación matemática o lógica es un valor negativo. Use este indicador como una prueba rápida de un valor negativo.	El controlador establece el indicador negativo cuando el resultado de una operación matemática o lógica es un valor negativo. Use este indicador como una prueba rápida de un valor negativo. Usar S:N es más eficiente que usar la instrucción CMP.
S:Z Indicador de cero	<p>El controlador establece el indicador de cero cuando el resultado de una operación matemática o lógica es cero. Use este indicador como una prueba rápida de un valor cero.</p> <p>El indicador de cero se borra al comienzo de la ejecución de una instrucción capaz de establecer este indicador.</p>	El controlador establece el indicador de cero cuando el resultado de una operación matemática o lógica es cero. Use este indicador como una prueba rápida de un valor cero.

S:V Indicador de desbordamiento	<p>El controlador establece el indicador de desbordamiento cuando:</p> <ul style="list-style-type: none"> <li>• El resultado de una operación matemática resulta en un desbordamiento. Por ejemplo, añadir 1 a un SINT genera un desbordamiento cuando el valor va de 127 a -128.</li> <li>• La etiqueta de destino es demasiado pequeña para retener el valor. Por ejemplo, si intenta almacenar el valor 123456 en una etiqueta SINT o INT.</li> </ul> <p>Use el indicador de desbordamiento para comprobar que el resultado de una operación siga estando dentro del rango. Si los datos almacenados son de tipo cadena, S:V se establece si la cadena es demasiado grande para que queda en la etiqueta de destino.</p> <p><b>Consejo:</b> Si corresponde, establezca S:V mediante una instrucción OTE u OTL.</p> <p>Haga clic en <b>Propiedades de controlador &gt; Pestaña Avanzada &gt; Informar de fallos de desbordamiento</b> para habilitar o deshabilitar la notificación de fallos de desbordamiento.</p> <p>Si se produce un desbordamiento mientras evalúa un subíndice de matriz, se generará un fallo menor y un fallo mayor que indica que el índice está fuera de rango.</p>	<p>El controlador establece el indicador de desbordamiento cuando:</p> <ul style="list-style-type: none"> <li>• El resultado de una operación matemática resulta en un desbordamiento. Por ejemplo, añadir 1 a un SINT genera un desbordamiento cuando el valor va de 127 a -128.</li> <li>• La etiqueta de destino es demasiado pequeña para retener el valor. Por ejemplo, si intenta almacenar el valor 123456 en una etiqueta SINT o INT.</li> </ul> <p>Use el indicador de desbordamiento para comprobar que el resultado de una operación siga estando dentro del rango.</p> <p>Siempre que se establezca un indicador de desbordamiento, se genera un fallo menor.</p> <p><b>Consejo:</b> Si corresponde, establezca S:V mediante una instrucción OTE u OTL.</p>
S:C Indicador de acarreo	<p>El controlador establece el indicador de acarreo cuando una operación matemática resultó en la generación de un acarreo del bit más significativo.</p> <p>Solo las instrucciones ADD y SUB, no los operadores + y -, con valores enteros afectan a este indicador.</p>	<p>El controlador establece el indicador de acarreo cuando una operación matemática resultó en la generación de un acarreo del bit más significativo.</p>
S:MINOR Indicador de fallo menor	<p>El controlador establece el indicador de fallo menor cuando hay por lo menos un fallo menor del programa.</p> <p>Use la etiqueta de fallo menor para comprobar si se ha producido un fallo menor. Este bit solo se activa mediante fallos de programación, como un desbordamiento. No se desencadena por un fallo de batería. El bit se borra al comienzo de cada escaneado.</p> <p><b>Consejo:</b> Si corresponde, establezca de forma explícita S:MINOR mediante una instrucción OTE u OTL.</p>	<p>El controlador establece el indicador de fallo menor cuando hay por lo menos un fallo menor del programa.</p> <p>Use el indicador de fallo menor para comprobar si se ha producido un fallo menor y tome las medidas pertinentes. Este bit solo se desencadena por fallos de programación, como un desbordamiento. No se desencadena por un fallo de batería. El bit se borra al comienzo de cada escaneado.</p> <p><b>Consejo:</b> Si corresponde, establezca de forma explícita S:MINOR mediante una instrucción OTE u OTL.</p>
<b>Importante:</b>	<p>Los indicadores de estado matemáticos se establecen en función del valor que se almacena. Las instrucciones que normalmente no afectan a los indicadores de estado matemáticos pueden parecer que lo hacen si la conversión de tipo se produce debido a tipos de datos mixtos para los parámetros de instrucción. El proceso de conversión de tipo establece los indicadores de estado matemáticos.</p>	

### Expresiones en subíndices de matriz

Controladores	Descripción
Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.	Las expresiones no establecen indicadores de estado basándose en los resultados de operaciones matemáticas. Si las expresiones se desbordan: <ul style="list-style-type: none"> <li>• Se genera un fallo menor si el controlador está configurado para hacerlo.</li> <li>• Se genera un fallo mayor (tipo 4, código 20) porque el valor resultante está fuera del rango.</li> </ul>
Controladores CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370 y GuardLogix 5570	Las expresiones establecen indicadores de estado basándose en los resultados de operaciones matemáticas. Si un subíndice de matriz es una expresión, tanto la expresión como la instrucción podrían generar fallos menores.

**Cons** Si un subíndice de matriz es demasiado grande (fuera ejo: de rango), se genera un fallo mayor (tipo 4, código 20).

## Valores inmediatos

Cuando introduce un valor inmediato (constante) en formato decimal (por ejemplo, -2, 3), el controlador almacena el valor usando 32 bits. Si introduce un valor en una base que no sea decimal, como binaria o hexadecimal, y no especifica todos los 32 bits, el controlador coloca un cero en los bits que no especifique (relleno de ceros).

---

**Importante:** Relleno de ceros de valores binarios inmediatos, octales o hexadecimales de menos de 32 bits.

---

Si introduce	El controlador almacena
-1	16#ffff ffff (-1)
16#ffff (-1)	16#0000 ffff (65535)
8#1234 (668)	16#0000 029c (668)
2#1010 (10)	16#0000 000a (10)

## Conversiones de datos

Las conversiones de datos se producen cuando mezcla tipos de datos en su programación.

Al programar:	Se pueden producir conversiones cuando:
Diagrama de escalera Texto estructurado	Mezcla tipos de datos para los parámetros dentro de una Instrucción
Bloque de funciones	Conecta dos parámetros que tienen diferentes tipos de datos

Las instrucciones se ejecutan más rápidamente y requieren menos memoria si todos los operandos de la instrucción utilizan:

- El mismo tipo de datos.
- Un tipo de datos óptimo:
  - Por lo general, los tipos de datos DINT y REAL son los óptimos.
  - Todas las instrucciones de bloque de funciones admiten un único tipo de datos para sus operandos.

Si mezcla tipos de datos o utiliza etiquetas que no son de un tipo de datos óptimo, el controlador convierte los datos conforme a estas reglas:

- Si el Destino es REAL, todos los operandos de fuente de enteros se convierten en REAL.
- Si el Destino es entero y uno o varios operandos de origen son REAL, todos los operandos de origen de enteros se convierten en REAL. El resultado REAL se convierte en el tipo de datos de Destino y se almacena a continuación.

Dado que la conversión de datos requiere tiempo y memoria adicionales, puede aumentar la eficacia de sus programas haciendo lo siguiente:

- Utilice el mismo tipo de datos en toda la instrucción.
- Minimice el uso del tipo de datos SINT o INT.
- En otras palabras, en sus instrucciones utilice únicamente etiquetas DINT o etiquetas REAL junto con los valores inmediatos.

### Convertir SINT o INT en DINT

Una etiqueta de origen de entrada SINT o INT son promocionadas a un valor DINT mediante una extensión de signo para la etiqueta de origen. Las instrucciones que convierten valores SINT o INT en valores DINT utilizan uno de los siguientes métodos de conversión.

Este método de conversión	Convierte datos colocando
Extensión de signo	El valor del bit situado más a la izquierda (el signo del valor) en cada posición de bit a la izquierda de existentes hasta que haya 32 bits.
Relleno de ceros	Añade ceros a la izquierda de los bits existentes hasta que haya 32 bits.

Las instrucciones lógicas (AND, OR, XOR, NOT, BTD, FRD, MVM, MEQ, SQI y SQO) utilizan el relleno de ceros. El resto de instrucciones utilizan la extensión de signo.

En el siguiente ejemplo se muestran los resultados de convertir un valor utilizando extensión de signo y relleno de ceros.

Este valor	2#1111_1111_1111_1111	(-1)
Se convierte a este valor mediante extensión de signo	2#1111_1111_1111_1111_1111_1111_1111_1111	(-1)
Se convierte a este valor mediante relleno de ceros	2#0000_0000_0000_1111_1111_1111_1111	(65535)

Si utiliza una etiqueta SINT o INT y un valor inmediato en una instrucción que convierte los datos mediante extensión de signo, utilice uno de estos métodos para manejar los valores inmediatos.

Especifique cualquier valor inmediato en la base decimal.

Si introduce el valor en una base que no sea la decimal, especifique todos los 32 bits del valor inmediato. Para hacerlo, introduzca el valor del bit situado más a la izquierda en cada posición de bit a su izquierda hasta que haya 32 bits.

Cree una etiqueta para cada operando y utilice el mismo tipo de datos en toda la instrucción. Para asignar un valor constante, realice una de las siguientes operaciones:

Introdúzcalo en una de las etiquetas.

Añada una instrucción MOV que mueva el valor en una de las etiquetas.

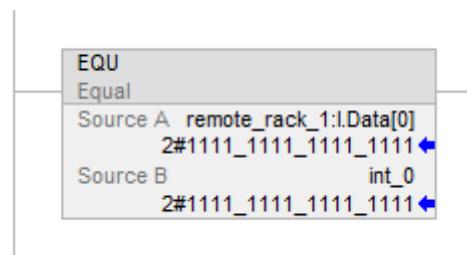
Utilice una instrucción MEQ para comprobar únicamente los bits requeridos.

Los siguientes ejemplos muestran dos maneras de mezclar un valor inmediato con una etiqueta INT. Ambos ejemplos comprueban los bits de un módulo E/S 1771 para determinar si todos los bits estén activados. Puesto que la palabra de datos de

entrada de módulo E/S 1771 es una etiqueta INT, es más fácil usar un valor constante de 16 bits.

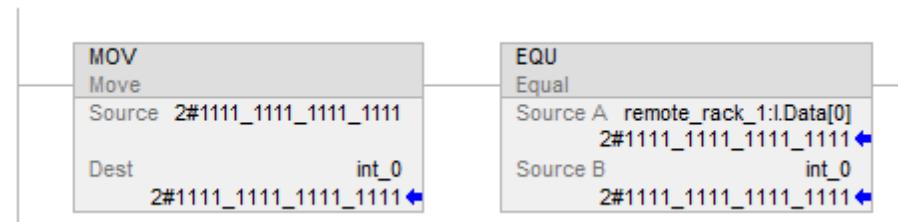
**Importante:** Mezclar una etiqueta INT con un valor inmediato

Puesto que `remote_rack_1:I.Data[0]` es una etiqueta INT, el valor con el que comprobarlo también se introduce como una etiqueta INT.



**Importante:** Mezclar una etiqueta INT con un valor inmediato

Puesto que `remote_rack_1:I.Data[0]` es una etiqueta INT, el valor con el que comprobarlo se mueve primero a `int_0`, que también es una etiqueta INT. A continuación, la instrucción EQU compara ambas etiquetas.



### Convertir Entero en REAL

El controlador almacena los valores REAL en un formato de número de punto flotante de precisión única IEEE. Utiliza un bit para el signo del valor, 23 bits para el valor de base y ocho bits para el exponente (en total son 32 bits). Si mezcla una etiqueta de entero (SINT, INT o DINT) con una etiqueta REAL como entrada de la misma instrucción, el controlador convierte el valor entero en un valor REAL antes de que se ejecute la instrucción.

- Un valor SINT o INT siempre se convierte en el mismo valor REAL.
- Un valor DINT puede que no se convierta en el mismo valor REAL:
- Un valor REAL usa hasta 24 bits para el valor de base (23 bits almacenados más uno ‘oculto’).

- Un valor DINT usa hasta 32 bits para el valor (uno para el signo y 31 para el valor).

Si el valor DINT requiere más de 24 bits significativos, puede que no se convierta en el mismo valor REAL. Si no lo hace, el controlador almacena los 24 bits superiores redondeados al valor par más cercano.

### Convertir DINT en SINT o INT

Para convertir un valor DINT en uno SINT o INT, el controlador trunca la porción superior del DINT y almacena los bits inferiores que quedan en el tipo de datos. Si el valor es demasiado grande, la conversión produce un desbordamiento.

Convertir un DINT en un INT y un SINT			
Este valor DINT	Se convierte en este valor menor		
16#0001_0081 (65,665)	INT:	16#0081 (129)	
	SINT:	16#81 (-127)	

### Convertir REAL en SINT, INT o DINT

Para convertir un valor REAL en un valor entero, el controlador redondea cualquier parte fraccionaria y almacena los bits que quedan en el tipo de datos del resultado. Si el valor es demasiado grande, la conversión produce un desbordamiento.

Los números se redondean igual que en los ejemplos mostrados a continuación.

Las fracciones inferiores a 0,5 se redondean al número entero por abajo más cercano.

Las fracciones superiores a 0,5 se redondean al número entero por arriba más cercano.

Las fracciones iguales a 0,5 se redondean al número par por arriba o por abajo más cercano.

IMPORTANTE Conversión de valores REAL en valores DINT	
Este valor REAL	Se convierte en este valor DINT
-2,5	-2
-3,5	-4
-1,6	-2
-1,5	-2
-1,4	-1
1,4	1
1,5	2
1,6	2

2,5	2
3,5	4

## Tipos de datos

El controlador admite los tipos de datos elementales definidos en los tipos de datos de la norma IEC 1131-3. Los tipos de datos atómicos predefinidos son:

Tipo de datos	Descripción	Rango
BOOL	Booleano de 1 bit	0 = borrado 1 = establecido
SINT	Entero de 1 byte	-128 a 127
INT	Entero de 2 bytes	-32.768 a 32.767
DINT	Entero de 4 bytes	-2.147.483.648 a 2.147.483.647
REAL	Número de punto flotante de 4 bytes	-3,402823E <sup>38</sup> a -1,1754944E <sup>38</sup> (valores negativos) y 0 y 1,1754944E <sup>38</sup> a 3,402823E <sup>38</sup> (valores positivos)

El controlador maneja todos los valores inmediatos como tipos de datos DINT

El tipo de datos REAL también almacena  $\pm$  in finito y  $\pm$  NAN, pero lo que el software muestra difiere según el formato de visualización.

Tipo de datos	Descripción
Estructura de AXIS_CONSUMED	estructura de control de un eje
Estructura de AXIS_VIRTUAL	
Estructura de AXIS_SERVO	
Estructura de AXIS_SERVO_DRIVE	
Estructura de CONTROL	estructura de control para instrucciones de matriz (archivo)
Estructura de COUNTER	estructura de control para instrucciones de contador
Estructura de MESSAGE	estructura de control para la instrucción MSG
Estructura de MOTION_GROUP	estructura de control para un grupo de movimiento
Estructura de MOTION_INSTRUCTION	estructura de control para instrucciones de movimiento
Estructura de PID	estructura de control para la instrucción PID
Estructura de TIMER	estructura de control para instrucciones de temporizador

### Conversiones de tipo de datos

Si se mezclan tipos de datos para operandos dentro de una instrucción, algunas instrucciones convierten automáticamente los datos en un tipo de datos óptimo

para esa instrucción. En algunos casos, el controlador convierte datos para adaptarse a un nuevo tipo de datos, pero a veces no los ajusta perfectamente.

<b>Conversión</b>	<b>Result</b>		
entero mayor a entero menor	El controlador trunca la porción superior del entero mayor y genera un desbordamiento. Por ejemplo:		
	<b>Decimal</b>		<b>Binario</b>
	DINT	65.665	0000_0000_0000_0001_0000_0000_1000_0001
	INT	129	0000_0000_1000_0001
	SINT	-127	1000_0001
SINT o INT a REAL	No se pierde precisión de los datos		
DINT a REAL	Se podría perder precisión de los datos. Ambos tipos de datos almacenan datos en 32 bits, pero el tipo REAL utiliza algunos de sus 32 bits para almacenar el valor del exponente. Si se pierde precisión, el controlador la toma de la porción menos significativa del DINT.		
REAL a entero	El controlador redondea la parte fraccionaria y trunca la porción superior de la parte no fraccionaria. Si se pierden datos, el controlador establece la marca de estado de desbordamiento. El redondeo es al número entero más cercano: menor de 0,5, redondeo hacia abajo; igual a 0,5, redondeo al número entero par más cercano; mayor de 0,5, redondeo hacia arriba Por ejemplo:		
	<b>REAL (origen)</b>		<b>DINT (resultado)</b>
	1,6	2	
	-1,6	-2	
	1,5	2	
	-1,5	-2	
	1,4	1	
	-1,4	-1	
	2,5	2	
	-2,5	-2	

No se pueden convertir datos a o desde el tipo de datos BOOL.

**Importante:** Las marcas de estado matemático se establecen en función del valor que se almacena. Las instrucciones que normalmente no afectan a las palabras clave de estado matemático pueden parecer que lo hacen si la conversión de tipo se produce debido a tipos de datos mixtos para los parámetros de instrucción. El proceso de conversión de tipo establece las palabras clave de estado matemático.

### Tipos de datos de seguridad

La aplicación Logix Designer impide la modificación de un tipo Definido por el usuario o Definido por instrucciones adicionales que podría provocar que a un tipo de datos no válido para los tipos definidos por el usuario o definido por instrucciones adicionales se le haga referencia directa o indirectamente por una etiqueta de Seguridad. (Eso incluye estructuras anidadas).

A continuación, se muestran los tipos de datos que no son válidos.

- REAL: Solo no válido en controladores GuardLogix(L6S) y Compact GuardLogix(L4S)
- LINT
- ALARMA ANALÓGICA
- ALARMA DIGITAL

### Consulte también

[Marcas de estado matemático](#) en la página 923

## Tipos de datos LINT

Cuando se usan tipos de datos LINT, hay muchas limitaciones aplicables. Un tipo de datos LINT no se puede usar en la mayoría de instrucciones. El tipo de datos LINT es:

- una palabra de 64 bits.
- No compatible con productos que usan palabras de 32 bits.
- No compatible en la mayoría de instrucciones.

**Cons** Los LINT solo se pueden utilizar con instrucciones para **ejo:** copiar (COP, CPS). Se utilizan con el atributo CST/WallClock Time, la sincronización de hora y las instrucciones adicionales. No puede sumar, restar, multiplicar ni dividir este tipo de etiqueta.

Si utiliza tipos de datos LINT, tenga en cuenta las siguientes descripciones cuando se produzcan estos problemas.

Cómo	Descripción
Mover/copiar dos valores DINT enteros dobles en uno LINT	Cree una matriz de doble entero de dos elementos, total de 64 bits (esto es, DINT[2]), que a continuación puede copiarse en un entero largo.

Corregir el error de visualización de fecha/hora	Cuando una etiqueta tiene un valor negativo, no puede mostrarse como Fecha/Hora. En el editor de etiquetas, compruebe si el valor es negativo cambiando el estilo de la etiqueta de Fecha/Hora a Binario. Cuando el bit más significativo (el situado más a la izquierda) es 1, el valor es negativo y por consiguiente no se puede mostrar como Fecha o Hora.
--	--

## Valores de punto flotante

Los controladores Logix manejan los valores de punto flotante conforme a la estándar IEEE 754 para aritmética de punto flotante. Este estándar define cómo se almacenan y calculan los números de punto flotante. El estándar IEEE 754 para matemáticas de punto flotante se diseñó para permitir y acelerar el tratamiento de números muy grandes en una cantidad razonable de espacio de almacenamiento.

Una etiqueta REAL almacena un número de punto flotante normalizado de precisión única.

Los números no normalizados y -0,0 se tratan como 0,0.

Si un cálculo resulta en un valor NAN, el bit de signo podría ser positivo o negativo. En esta situación el software muestra 1#NAN sin signo.

No todos los valores decimales se pueden representar con exactitud en este formato estándar, que da lugar a una pérdida de precisión. Por ejemplo, si a 10,1 le resta 10, espera que el resultado sea 0,1. En un controlador Logix, el resultado bien podría ser 0,10000038. En este ejemplo, la diferencia entre 0,1 y 0,10000038 es del 0,000038% o prácticamente cero. En la mayoría de las operaciones, esta pequeña imprecisión es insignificante. Para poner las cosas en perspectiva, si estuviese enviando un valor de punto flotante a un módulo de salida analógica, no habría diferencia entre el voltaje de salida de un valor enviado al módulo que difiriese un 0,000038%.

### Pautas para operaciones matemáticas de punto flotante

Siga estas pautas:

Al realizar determinadas operaciones matemáticas de punto flotante, podría producirse una pérdida de precisión debido a un error de redondeo. Los procesadores de punto flotante tienen su propia precisión interna que puede afectar a los valores resultantes.

No utilice matemáticas de punto flotante para valores monetarios o para funciones de totalizador. Use valores INT o DINT, escale el valor hacia arriba y realice un seguimiento de la posición decimal (o use un valor INT o DINT para dólares y un segundo valor INT o DINT para céntimos).

No compare números de punto flotante. En su lugar, compruebe los valores dentro de un rango. La instrucción LIM se proporciona específicamente para este fin.

### Ejemplos de totalizador

La precisión del tipo de datos REAL afecta a las aplicaciones de totalización, de tal modo que se producen errores al añadir números muy pequeños a números muy grandes.

Por ejemplo, añada 1 a un número durante un período de tiempo. En algún punto la adición dejará de afectar al resultado, porque la suma en ejecución es muy superior a 1 y no hay bits suficientes para almacenar todo el resultado. La adición almacena tantos bits superiores como sea posible y desecha los restantes bits inferiores.

Una solución alternativa a este problema es realizar los cálculos en los números pequeños hasta que los resultados sean grandes. A continuación, transfíralos a otra ubicación para realizar cálculos de números grandes adicionales. Por ejemplo:

- x es la variable incrementada pequeña.
- y es la variable incrementada grande.
- z es el conteo actual total que se puede usar en cualquier lugar.
- $x = x + 1;$
- si  $x = 100,000;$
- {
- $y = y + 100,000;$
- $x = 0;$
- }
- $z = y + x;$

Otro ejemplo sería:

- $x = x + \text{some\_tiny\_number};$
- si ( $x \geq 100$ )
- {
- $z = z + 100;$
- $x = x - 100; // \text{podría haber un resto diminuto}$
- }

## Índice a través de matrices

Para cambiar dinámicamente el elemento de matriz al que hace referencia su lógica, utilice una etiqueta o expresión como el subíndice para apuntar al elemento. Esto es parecido al direccionamiento indirecto de la lógica PLC-5.

Puede usar estos operadores en una expresión para especificar un subíndice de matriz:

Operador	Descripción
+	sumar
-	restar/negar
*	multiplicar
/	dividir
AND	Y
FRD	BCD a entero
NOT	complementario
OR	O
TOD	entero a BCD
SQR	raíz cuadrada
XOR	OR exclusivo

Por ejemplo:

Definiciones	Ejemplo	Descripción
my_list definido como DINT[10]	my_list[5]	Este ejemplo hace referencia al elemento 5 en la matriz. La referencia es estática porque el valor de subíndice permanece constante.
my_list definido como DINT[10] posición definida como DINT	MOV the value 5 into position my_list[position]	Este ejemplo hace referencia al elemento 5 en la matriz. La referencia es dinámica porque la lógica puede cambiar el subíndice cambiando el valor de posición.
my_list definido como DINT[10] posición definida como DINT desplazamiento definido como DINT	MOV the value 2 into position MOV the value 5 into offset my_list[position+offset]	Este ejemplo hace referencia al elemento 7 (2+5) de la matriz. La referencia es dinámica porque la lógica puede cambiar el subíndice cambiando el valor de posición o desplazamiento.

Asegúrese de que cualquier subíndice de matriz que introduzca esté dentro de los límites de la matriz especificada. Las instrucciones que ven las matrices como una colección de elementos generan un fallo mayor (tipo 4, código 20) si un subíndice supera su dimensión correspondiente.

## Direccionamiento de bit

El direccionamiento de bits se utiliza para acceder a un bit particular en un contenedor más grande. Los contenedores más grandes incluyen cualquier entero, estructura o matriz (de cualquier entero o BOOL). Por ejemplo:

Definición	Ejemplo	Descripción
variable1 definida como DINT tiene 32 bits	variable1.2	Este ejemplo da como referencia el bit 2 de la variable1.
variable2 definida como INT tiene 16 bits	variable2.15	Este ejemplo da como referencia el bit 15 de la variable2.
variable3 definida como SINT contiene 8 bits	variable3.[4]	Este ejemplo da como referencia el bit 4 de la variable3.
variable4 definida como estructura COUNTER tiene 5 bits de estado	variable4.DN	Este ejemplo da como referencia el bit DN de la variable4.
MyVariable definido como BOOL[100] MyIndex definido como SINT	MyVariable[(MyIndex AND NOT 7) / 8].[MyIndex AND 7]	Este ejemplo da como referencia un bit en una matriz BOOL.

### Consulte también

[Índice a través de matrices](#) en la página 936



# Atributos del bloque de funciones

Haga clic en un tema a continuación para obtener más información sobre los problemas que son exclusivos de la programación del bloque de funciones. Revise esta información para asegurarse de que entiende cómo funcionarán las rutinas del bloque de funciones.

### Consulte también

[Elegir los elementos del bloque de funciones](#) en la página 939

[Enclavamiento de datos](#) en la página 940

[Orden de ejecución](#) en la página 941

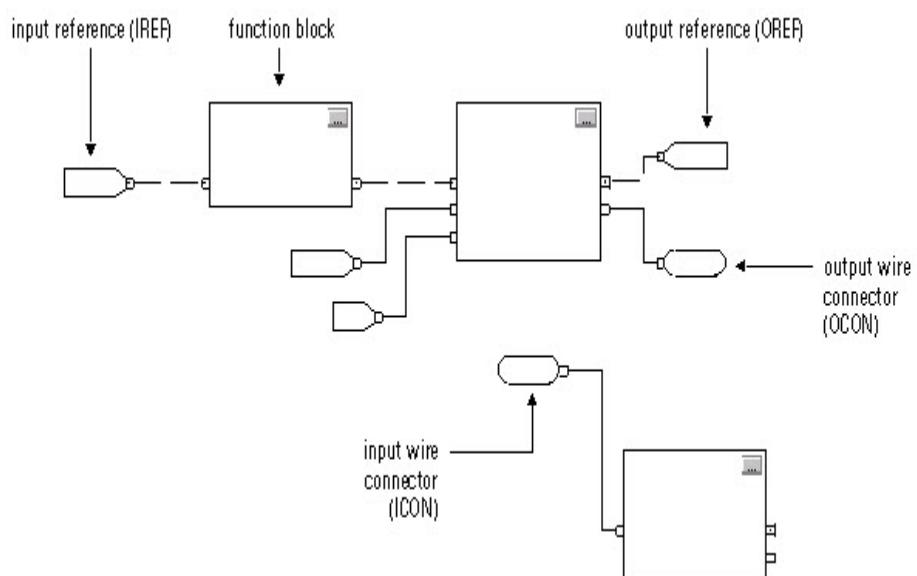
[Respuestas del bloque de funciones a las condiciones de desbordamiento](#) en la página 946

[Modos de temporización](#) en la página 946

[Control de programa/operador](#) en la página 950

## Elegir los elementos del bloque de funciones

Para controlar un dispositivo, utilice estos elementos:



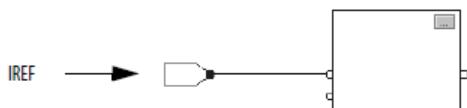
Utilice la tabla siguiente para seleccionar sus elementos del bloque de funciones:

Si desea aprovisionar un valor desde una etiqueta o dispositivo de entrada	Utilice una referencia de entrada (IREF)
Enviar un valor a una etiqueta o dispositivo de salida	Referencia de salida (OREF)
Realizar una operación en un valor o valores de entrada y generar un valor o valores de salida.	Bloque de funciones
Transferir datos entre bloques de funciones cuando: <ul style="list-style-type: none"> <li>• Están alejados en la misma hoja</li> <li>• Se encuentran en hojas diferentes en la misma rutina</li> </ul>	Conejero de cable de salida (OCON) y un conector de cable de entrada (ICON)
Dispersar datos a varios puntos de la rutina	Conejero de cable de una única salida (OCON) y conectores de cable de entrada múltiple (ICON)

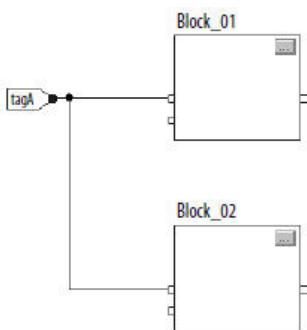
El bloque de funciones mueve las referencias de entrada a la estructura del bloque. En caso necesario, el bloque de funciones convierte estas referencias de entrada en valores REAL. El bloque de funciones ejecuta y mueve los resultados a las referencias de salida. De nuevo, en caso necesario, el bloque de funciones convierte estos valores de resultado de REAL a tipos de datos para las referencias de salida.

## Enclavamiento de datos

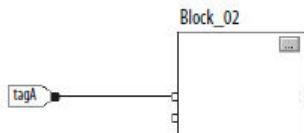
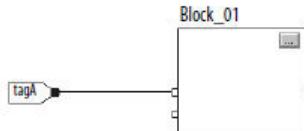
Si utiliza una IREF para especificar datos de entrada para una instrucción del bloque de funciones, los datos de esta IREF se enclavan para el escaneado de la rutina del bloque de funciones. La IREF bloquea datos de etiquetas en el alcance del programa y en el alcance del controlador. El controlador actualiza todos los datos de IREF al principio de cada escaneado.



En este ejemplo, el valor de tagA se almacena al principio de la ejecución de la rutina. El valor almacenado se utiliza cuando se ejecuta Block\_01. El mismo valor almacenado también se utiliza cuando se ejecuta Block\_02. Si el valor de tagA cambia durante la ejecución de la rutina, el valor almacenado de tagA en la IREF no cambia hasta la siguiente ejecución de la rutina.

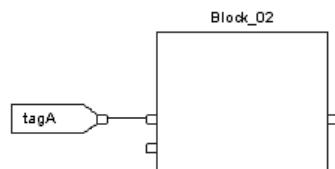
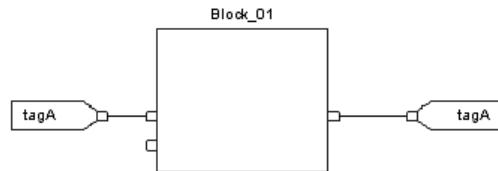


Este ejemplo es el mismo que el anterior. El valor de tagA se almacena solamente una vez al principio de la ejecución de la rutina. La rutina utiliza este valor almacenado en toda la rutina.



Puede utilizar la misma etiqueta en varias IREF y una OREF en la misma rutina. Debido a que los valores de las etiquetas de IREF se enclavan en todos los escaneados de la rutina, todos los IREF utilizarán el mismo valor, aunque OREF obtenga un valor de etiqueta diferente durante la ejecución de la rutina.

En este ejemplo, si tagA tiene un valor de 25,4 cuando la rutina comienza a ejecutar este escaneado y Block\_01 cambia el valor de tagA a 50,9, el segundo IREF conectado a Block\_02 seguirá utilizando un valor de 25,4 cuando Block\_02 ejecute este escaneado. En esta rutina, ninguna IREF utilizará el nuevo valor de tagA de 50,9 hasta el comienzo del siguiente escaneado.



## Orden de ejecución

La aplicación de programación Logix Designer determina automáticamente el orden de ejecución de los bloques de funciones en una rutina cuando:

- verifica una rutina del bloque de funciones
- verifica un proyecto que contiene una rutina del bloque de funciones

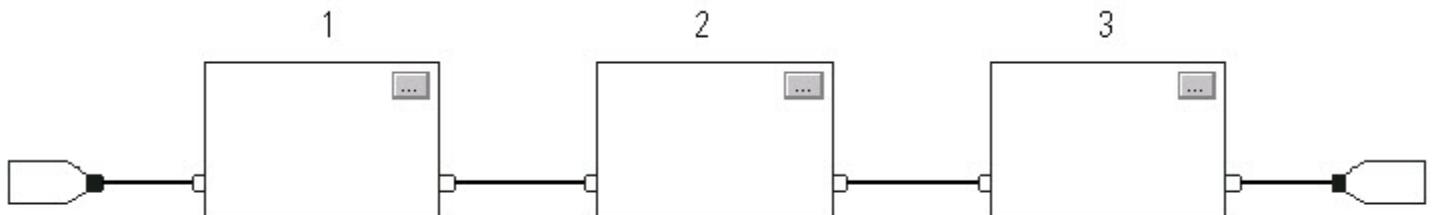
- descarga un proyecto que contiene una rutina del bloque de funciones

Se define el orden de ejecución conectando los bloques de funciones entre sí y se indica el flujo de datos de las líneas de retroalimentación, si es necesario.

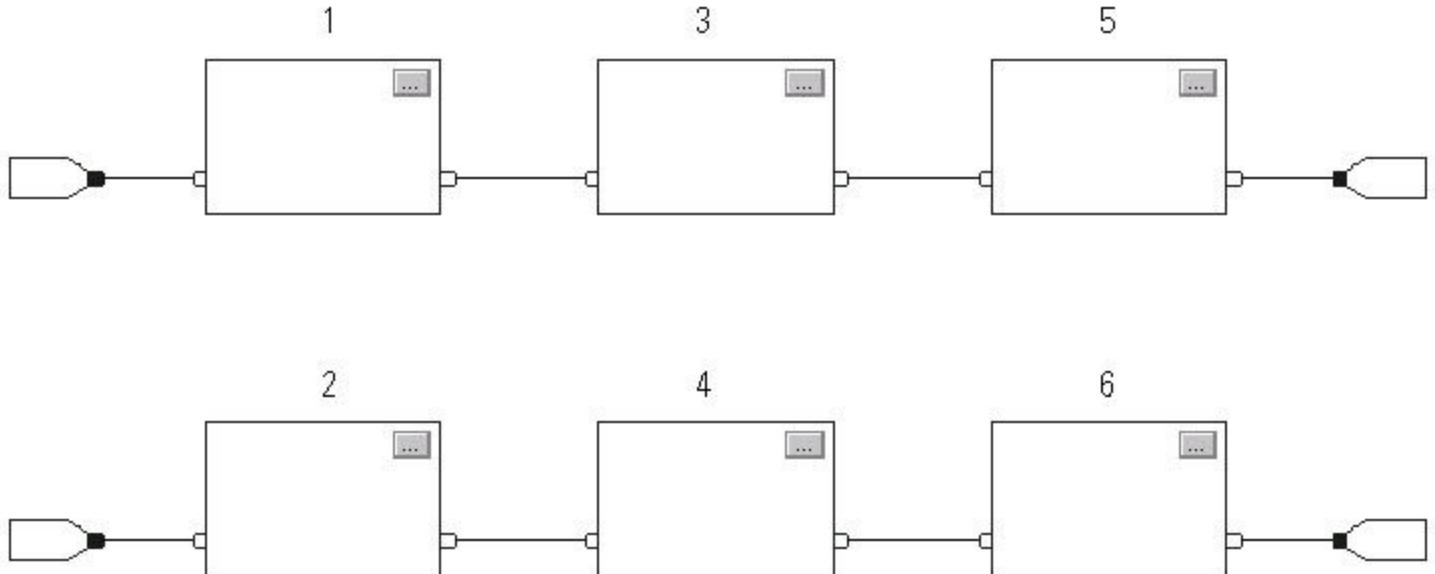
Si los bloques de funciones no están conectados entre sí, no importa cuál es el bloque de funciones que se ejecuta en primer lugar. No hay flujo de datos entre los bloques



Si conecta los bloques secuencialmente, el orden de ejecución se mueve de la entrada a la salida. Las entradas de un bloque requieren que los datos estén disponibles antes de que el controlador pueda ejecutar el bloque. Por ejemplo, el bloque 2 tiene que ejecutarse antes del bloque 3 porque las salidas del bloque 2 alimentan las entradas del bloque 3.

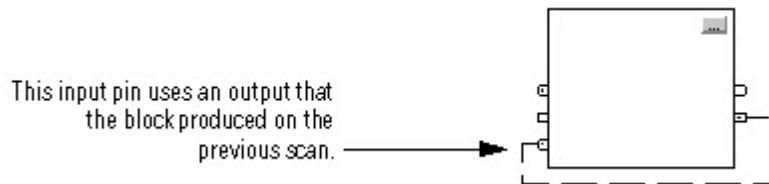


El orden de ejecución solo está relacionado con los bloques que están conectados entre sí. El ejemplo siguiente es correcto porque los dos grupos de bloques no están conectados entre sí. Los bloques de un grupo específico se ejecutan en el orden apropiado en relación con los bloques de ese grupo.

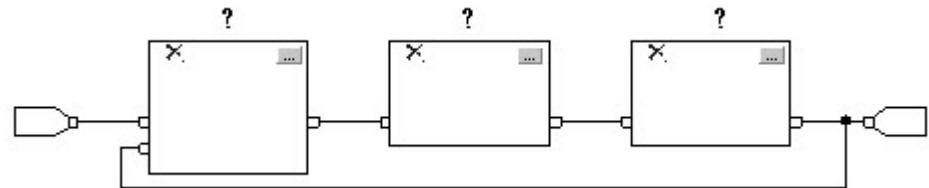


### Resolver un lazo

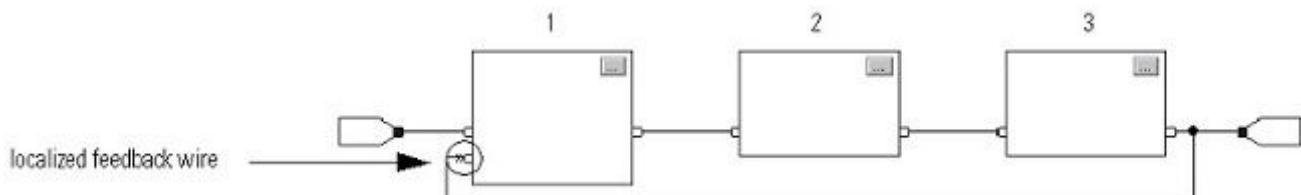
Para crear un lazo de retroalimentación alrededor de un bloque, conecte un pin de salida del bloque a un pin de entrada del mismo bloque. El ejemplo siguiente es correcto. El lazo solo contiene un único bloque, por lo que el orden de ejecución no importa.



Si un grupo de bloques está en un lazo, el controlador no puede determinar cuál es el bloque que debe ejecutarse primero. Dicho de otro modo, no puede resolver el lazo.



Para identificar cuál es el bloque que debe ejecutarse primero, marque el cable de entrada que crea el lazo (la línea de retroalimentación) con el indicador *Asumir datos disponibles*. En el ejemplo siguiente, el bloque 1 utiliza la salida del bloque 3 que se produjo en la ejecución anterior de la rutina.



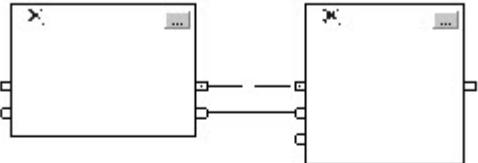
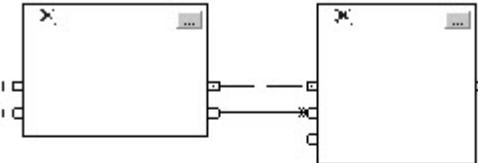
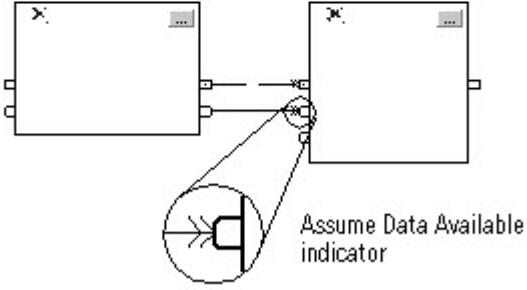
El indicador *Asumir datos disponibles* define el flujo de datos del lazo. La flecha indica que los datos sirven de entrada al primer bloque del lazo.

No marque todos los cables de un lazo con el indicador *Asumir datos disponibles*.

This is OK	This is NOT OK
<p>The <i>Assume Data Available</i> indicator defines the data flow within the loop.</p>	<p>The controller cannot resolve the loop because all the wires use the <i>Assume Data Available</i> indicator.</p>

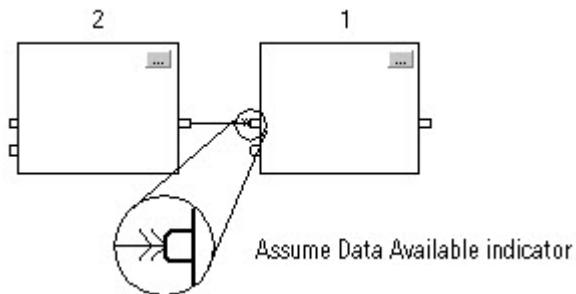
### Resolver el flujo de datos entre dos bloques

Si utiliza dos o más líneas para conectar dos bloques, use los mismos indicadores de flujo de datos para todos los cables que se encuentran entre los dos bloques.

This is OK	This is NOT OK
	
<p>Neither wire uses the <i>Assume Data Available</i> indicator.</p>  <p>Both wires use the <i>Assume Data Available</i> indicator.</p>	<p>One wire uses the <i>Assume Data Available</i> indicator while the other wire does not.</p>

### Crear un retardo de escaneado único

Para generar un retardo de escaneado único entre bloques, utilice el indicador Asumir datos disponibles. En el ejemplo siguiente, primero se ejecuta el bloque 1. Utiliza la salida del bloque 2 que se produjo en el escaneado anterior de la rutina.



### Resumen

En el resumen, una rutina del bloque de funciones realiza la ejecución en el orden siguiente:

1. El controlador enclava todos los valores de datos en IREF.
2. El controlador ejecuta los demás bloques de funciones en el orden determinado por la forma en que están conectados.
3. El controlador escribe salidas en OREF.

## Respuestas del bloque de funciones a las condiciones de desbordamiento

En general, las instrucciones del bloque de funciones que actualizan el historial no lo actualizan con valores  $\pm\text{NAN}$  o  $\pm\text{INF}$  si se produce un desbordamiento. Todas las instrucciones tienen una de estas respuestas a una condición de desbordamiento.

Respuesta	Instrucción
Respuesta 1 Los bloques ejecutan su algoritmo y comprueban el resultado de $\pm\text{NAN}$ o $\pm\text{INF}$ . Si $\pm\text{NAN}$ o $\pm\text{INF}$ , las salidas del bloque son: $\pm\text{NAN}$ o $\pm\text{INF}$ .	ALM NTCH DEDT PMUL DERV POSP ESEL RLIM FGEN RMPS HPF SCRV LDL2 SEL LDLG SNEG LPF SRTP MAVE SSUM MAXC TOT MINC UPDN MSTD MUX
Respuesta 2 Los bloques con limitación de salida ejecutan su algoritmo y comprueban el resultado de $\pm\text{NAN}$ o $\pm\text{INF}$ . Los parámetros de entrada HighLimit y LowLimit definen los límites de salida. Si $\pm\text{INF}$ , las salidas del bloque generan un resultado limitado. Si $\pm\text{NAN}$ , los límites de salida no se utilizan y las salidas del bloque son: $\pm\text{NAN}$ .	HLL, INTG, PI, PIDE, SCL, SOC
Respuesta 3 La condición de desbordamiento no se aplica. Normalmente, estas instrucciones tienen una salida booleana.	BAND, BNOT, BOR, BXOR, CUTD, D2SD, D3SD, DFF, JKFF, OSFI, OSRI, RESD, RTOR, SETD, TOFR, TONR

## Modos de temporización

Estas instrucciones de variadores y control de procesos admiten diferentes modos de temporización.

- DEDT
- DERV
- HPF
- INTG
- LDL2
- LDLG
- LPF
- NTCH
- PI
- RLIM
- SCRV
- SOC
- TOT
- PIDE

Hay tres distintos modos de temporización.

Modo de temporización	Descripción							
Periódico	<p>El modo Periódico es el modo predeterminado y es apto para la mayoría de aplicaciones de control. Le recomendamos que coloque las instrucciones que usen este modo en una rutina que se ejecute en una tarea periódica. El valor de tiempo de delta (DeltaT) para la instrucción se determina de la siguiente manera:</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 5px;"><b>Si la instrucción se ejecuta en una</b></td><td style="padding: 5px;"><b>Entonces el valor de DeltaT es igual a</b></td></tr> <tr> <td style="padding: 5px;">Tarea periódica</td><td style="padding: 5px;">Período de la tarea</td></tr> <tr> <td style="padding: 5px;">Evento o tarea continua</td><td style="padding: 5px;">Tiempo transcurrido desde la ejecución anterior El controlador trunca el tiempo transcurrido en milisegundos enteros (ms). Por ejemplo, si el tiempo transcurrido = 10,5 ms, el controlador establece DeltaT = 10 ms.</td></tr> </table> <p>La actualización de la entrada del proceso debe sincronizarse con la ejecución de la tarea o debe muestrearse entre 5 y 10 veces más rápido que el tiempo necesario para ejecutar la tarea a fin de reducir al mínimo el error de muestreo entre la entrada y la instrucción.</p>		<b>Si la instrucción se ejecuta en una</b>	<b>Entonces el valor de DeltaT es igual a</b>	Tarea periódica	Período de la tarea	Evento o tarea continua	Tiempo transcurrido desde la ejecución anterior El controlador trunca el tiempo transcurrido en milisegundos enteros (ms). Por ejemplo, si el tiempo transcurrido = 10,5 ms, el controlador establece DeltaT = 10 ms.
<b>Si la instrucción se ejecuta en una</b>	<b>Entonces el valor de DeltaT es igual a</b>							
Tarea periódica	Período de la tarea							
Evento o tarea continua	Tiempo transcurrido desde la ejecución anterior El controlador trunca el tiempo transcurrido en milisegundos enteros (ms). Por ejemplo, si el tiempo transcurrido = 10,5 ms, el controlador establece DeltaT = 10 ms.							
Sobremuestreo	<p>En el modo Sobremuestreo, el tiempo de delta (DeltaT) usado por la instrucción es el valor escrito en el parámetro OversampleDT de la instrucción. Si la entrada del proceso tiene un valor de marca de tiempo, use el modo de muestreo en tiempo real en su lugar.</p> <p>Añada lógica a su programa para controlar cuándo se ejecuta la instrucción. Por ejemplo, puede usar un temporizador establecido en el valor OversampleDeltaT para controlar la ejecución mediante el uso de la entrada EnableIn de la instrucción.</p> <p>La entrada del proceso debe muestrearse entre 5 y 10 veces más rápido que el tiempo necesario para ejecutar la instrucción a fin de reducir al mínimo el error de muestreo entre la entrada y la instrucción.</p>							
Muestreo en tiempo real	<p>En el modo Muestreo en tiempo real, el tiempo de delta (DeltaT) usado por la instrucción es la diferencia entre dos valores de marca de tiempo que corresponden a las actualizaciones de la entrada de proceso. Use este modo cuando la entrada de proceso tenga una marca de tiempo asociada a sus actualizaciones y necesite una coordinación precisa.</p> <p>El valor de marca de tiempo se lee desde el nombre de etiqueta introducido para el parámetro RTSTimeStamp de la instrucción. Por lo general, este nombre de etiqueta es un parámetro en el módulo de entrada asociado a la entrada de proceso.</p> <p>La instrucción compara el valor de RTSTime configurado (período de actualización previsto) con el valor de DeltaT calculado para determinar si la instrucción lee cada actualización de la entrada de proceso. Si la diferencia entre DeltaT y el tiempo de configuración no es menor o igual a 1 milisegundo, la instrucción establece el bit de estado de RTSMissed para indicar que hay un problema al leer actualizaciones de la entrada en el módulo.</p>							

Las instrucciones basadas en tiempo requieren un valor constante de DeltaT para que el algoritmo de control calcule correctamente la salida del proceso. Si DeltaT varía, se produce una discontinuidad en la salida del proceso. La gravedad de la discontinuidad depende de la instrucción y el rango de variación de DeltaT.

Se producirá una discontinuidad si sucede lo siguiente:

- La instrucción no se ejecuta durante un escaneo.
- La instrucción se ejecuta varias veces durante una tarea.
- La tarea se está ejecutando y cambia el régimen de escaneo o el tiempo de muestreo de la entrada del proceso.
- El usuario cambia el modo de base de tiempo mientras se está ejecutando la tarea.
- El parámetro Order se modifica en un bloque de filtro mientras se ejecuta la tarea.
- Al cambiar el parámetro Order, se selecciona otro algoritmo de control dentro de la instrucción.

#### **Parámetros de instrucción comunes para los modos de temporización**

Las instrucciones que admiten los modos de base de tiempo tienen estos parámetros de entrada y salida.

#### **Parámetros de entrada**

Parámetro de entrada	Tipo de datos	Descripción
TimingMode	DINT	<p>Selecciona el modo de ejecución de temporización. Value: Descripción: 0 Modo Periódico 1 Modo Sobremuestreo 2 Modo Muestreo en tiempo real Válido = de 0 a 2 Valor predeterminado = 0</p> <p>Cuando TimingMode = 0 y la tarea es periódica, se habilita el modo de temporización periódico y se establece DeltaT en el régimen de escaneo de la tarea. Cuando TimingMode = 0 y la tarea es un evento o es continua, se habilita el modo de temporización periódico y se establece DeltaT en el intervalo de tiempo transcurrido desde la última vez que se ejecutó la instrucción.</p> <p>Cuando TimingMode = 1, se habilita el modo de temporización de sobremuestreo y se establece DeltaT en el valor del parámetro OversampleDT. Cuando TimingMode = 2, se habilita el modo de muestreo en tiempo real y DeltaT se establece en la diferencia entre los valores de marca de tiempo actual y anterior leídos desde el módulo asociado a la entrada.</p> <p>Si el valor de TimingMode no es válido, la instrucción establece el bit correspondiente de Status.</p>

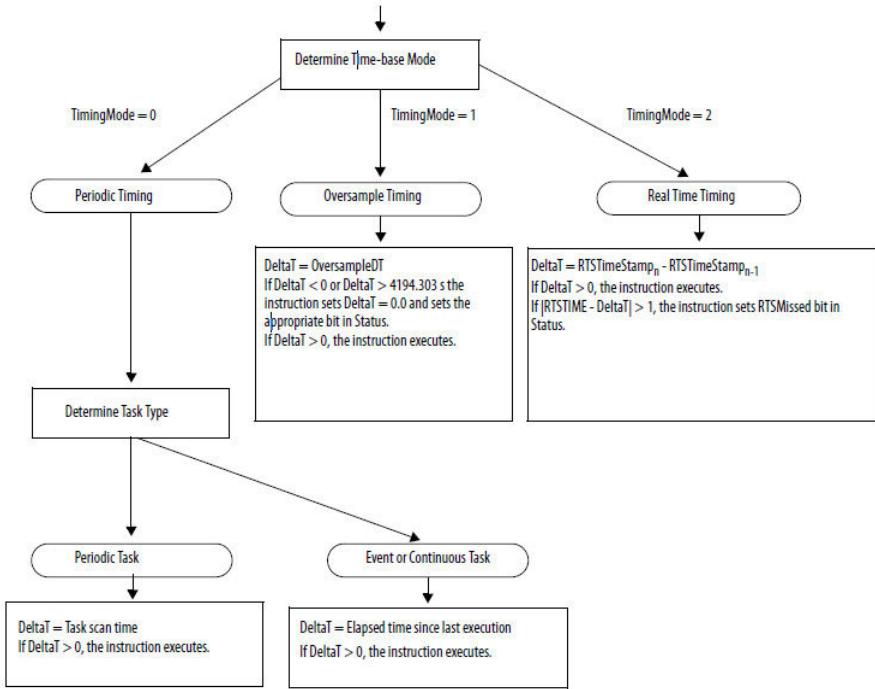
OversampleDT	REAL	Tiempo de ejecución para el modo de temporización de sobremuestreo. El valor usado para DeltaT se expresa en segundos. Si TimingMode = 1, OversampleDT = 0,0 deshabilita la ejecución del algoritmo de control. Si no es válido, la instrucción establece DeltaT = 0,0 y establece el bit correspondiente en Status. Válido = de 0 a 4194,303 segundos Valor predeterminado = 0,0
RTSTime	DINT	Período de actualización de módulo para el modo de temporización de muestreo en tiempo real. El período de actualización DeltaT esperado se expresa en milisegundos. El período de actualización es normalmente el valor que se utilizó para configurar el tiempo de actualización del módulo. Si su valor no es válido, la instrucción establece el bit correspondiente en Status y deshabilita la comprobación de RTSMissed. Válido = 1...32,767ms Valor predeterminado = 1
RTSTimeStamp	DINT	Valor de sello de tiempo de módulo para el modo de temporización de muestreo en tiempo real. El valor de sello de tiempo que corresponde a la última actualización de la señal de entrada. Este valor se utiliza para calcular DeltaT. Si su valor no es válido, la instrucción establece el bit correspondiente en Status, deshabilita la ejecución del algoritmo de control y deshabilita la comprobación de RTSMissed. Válido = 0...32,767ms (envuelve desde 32767 a 0) 1 conteo = 1 milisecondo Valor predeterminado = 0

### Parámetros de salida

Parámetro de salida	Tipo de datos	Descripción
DeltaT	REAL	Tiempo transcurrido entre actualizaciones. Este es el tiempo transcurrido en segundos utilizado por el algoritmo de control para calcular la salida del proceso. Periódico: DeltaT = régimen de escaneo de tarea si es una Tarea periódica, DeltaT = tiempo transcurrido desde la anterior ejecución de la instrucción si la tarea es una tarea de Evento o Continua Sobremuestreo: DeltaT = OversampleDT Muestreo en tiempo real: DeltaT = (RTSTimeStampn - RTSTimeStampn-1)
Status	DINT	Estado del bloque de funciones.
TimingModelInv (Status.27)	BOOL	Valor de TimingMode no válido.
RTSMissed (Status.28)	BOOL	Solo se utiliza en el modo de muestreo en tiempo real. Se establece cuando ABS   DeltaT - RTSTime   > 1 (0,001 segundos).
RTSTimeInv (Status.29)	BOOL	Valor de RTSTime no válido.
RTSTimeStampInv (Status.30)	BOOL	Valor de RTSTimeStamp no válido.
DeltaTInv (Status.31)	BOOL	Valor de DeltaT no válido.

### Descripción general de los modos de temporización

El siguiente diagrama muestra cómo una instrucción determina el modo de temporización adecuado.



## Control de programa/operador

Las siguientes instrucciones admiten el concepto de Control de programa/operador.

- Selección mejorada (ESEL)
- Totalizador (TOT)
- PID mejorado (PIDE)
- Rampa/estabilización (RMPS)
- Dispositivo de 2 estados discreto (D2SD)
- Dispositivo de 3 estados discreto (D3SD)

El Control de programa/operador le permite controlar estas instrucciones de forma simultánea tanto desde su programa de usuario como desde un dispositivo de interfaz de operador. Cuando está en el Control de programa, la instrucción es controlada las entradas del programa a la instrucción, cuando está en el Control de operador, la instrucción es controlada las entradas de operador a la instrucción.

El Control de programa o el Control de operador se determina usando estas entradas.

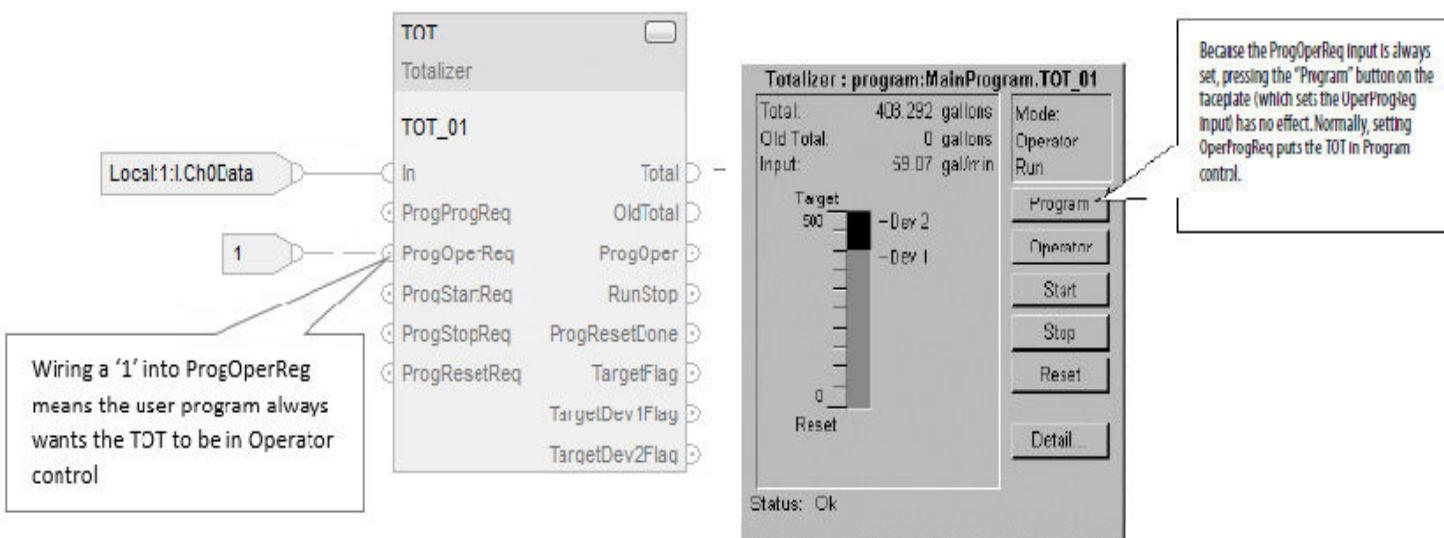
Entrada	Descripción
.ProgProgReq	Una solicitud de programa para activar el Control de programa.
.ProgOperReq	Una solicitud de programa para activar el Control de operador.
.OperProgReq	Una solicitud de operador para activar el Control de programa.
.OperOperReq	Una solicitud de operador para activar el Control de operador.

Para determinar si una instrucción está en el modo Control de programa o Control de operador, examine la salida ProgOper. Si se ha establecido ProgOper, la instrucción está en Control de programa; si se ha borrado el valor de ProgOper, la instrucción está en Control de operador.

El Control de operador tiene prioridad con respecto al Control de programa si se han establecido los dos bits de la solicitud de entrada. Por ejemplo, si se han establecido ProgProgReq y ProgOperReq, la instrucción pasa al modo Control de operador.

Las entradas de solicitud de programa tienen preferencia frente a las entradas de solicitud de operador. Esta proporciona la capacidad de usar las entradas ProgProgReq y ProgOperReq para bloquear una instrucción en un control deseado.

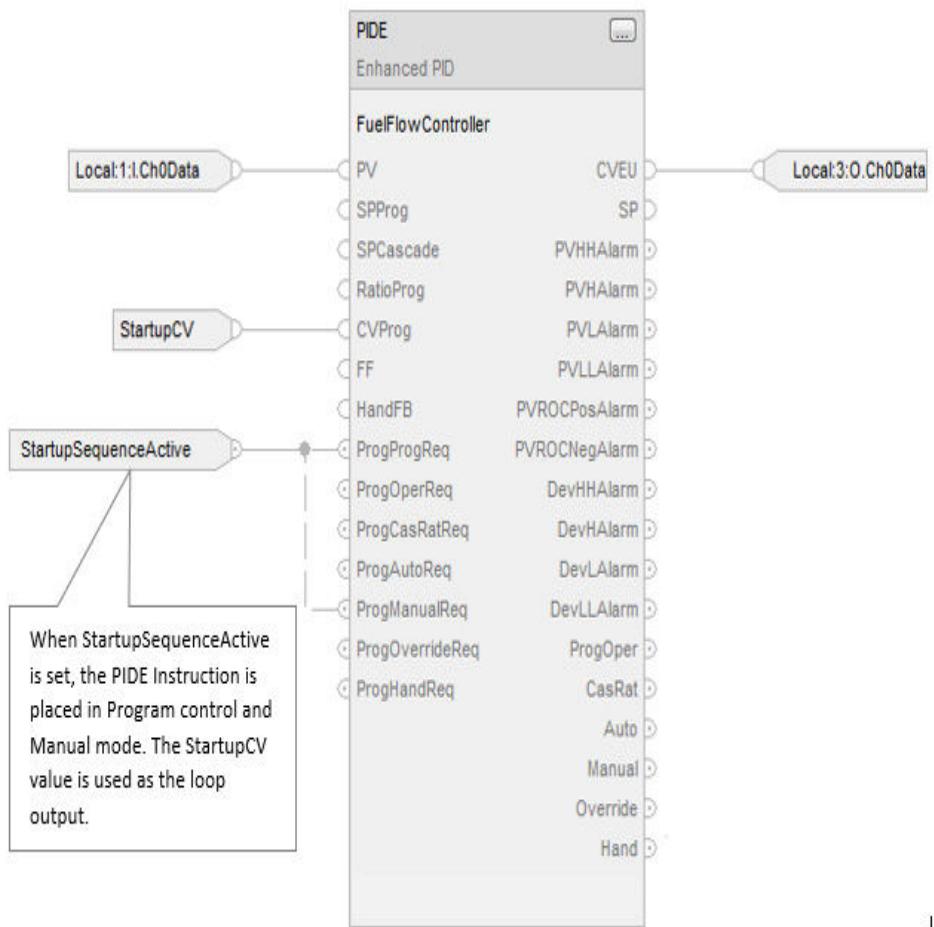
Por ejemplo, supongamos que una instrucción Totalizer siempre se utilizará en el modo Control de operador y su programa de usuario nunca controlará la ejecución ni la parada del Totalizador. En este caso, podría conectar un valor literal de 1 con el interior de ProgOperReq. Esto evitaría la posibilidad de que el operador colocase el Totalizador en el modo Control de programa estableciendo la entrada OperProgReq desde un dispositivo de interfaz de operador.



Del mismo modo, establecer constantemente la entrada ProgProgReq puede ‘bloquear’ la instrucción en el Control de programa. Esto es útil para secuencias de inicio automático cuando quiera que el programa controle la acción de la instrucción sin preocuparse por que un operador se haga con el control de la instrucción de manera no intencionada.

En este ejemplo, hace que el programa se establece la entrada ProgProgReq durante el inicio y, a continuación, se borra el valor de la entrada ProgProgReq una vez que se haya completado el inicio. Una vez que se haya borrado el valor de la entrada ProgProgReq, la instrucción permanece en el modo Control de programa hasta que reciba una solicitud de cambio. Por ejemplo, el operador podría establecer la entrada OperOperReq desde una presentación para hacerse con el control de la instrucción.

En el siguiente ejemplo se muestra cómo bloquear una instrucción en el modo de Control de programa.



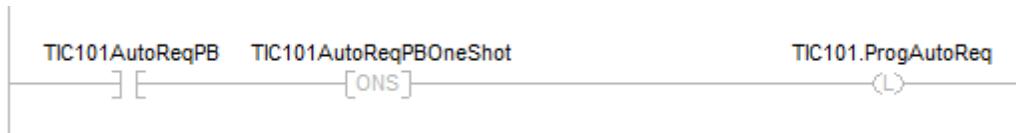
Las entradas de solicitud de operador a una instrucción son borradas siempre por la instrucción cuando se ejecuta. Esto permite a las interfaces de operador trabajar con estas instrucciones simplemente estableciendo el bit de solicitud de modo deseado. No tiene que programar la interfaz de operador para restablecer los bits

de solicitud. Por ejemplo, si una interfaz de operador establece la entrada OperAutoReq a una instrucción PIDE, cuando la instrucción PIDE se ejecuta, determina cuál debería ser la respuesta adecuada y borra el valor de OperAutoReq.

Por lo general, las entradas de solicitud de programa no son borradas por la instrucción porque normalmente se conecta como entradas a la instrucción. Si la instrucción borra estas entradas, la entrada conectada volvería a establecer la entrada. Puede que haya situaciones en las que le convenga usar otra lógica para establecer las solicitudes de programa de tal modo que quiera que la instrucción borre las solicitudes de programa. En este caso puede establecer la entrada ProgValueReset y la instrucción siempre borrará las entradas de solicitud del modo Programa cuando se ejecute.

En este ejemplo, un reglón de lógica de escalera de otra rutina se utiliza para bloquear con un impulso una entrada ProgAutoReq en una instrucción PIDE cuando se apriete un botón pulsador.

Cuando se pulsa el botón pulsador TIC101AutoReq, se produce un enclavamiento de un impulso de ProgAutoReq para la instrucción PIDE TIC101. TIC101 se ha configurado con la entrada ProgValueReset establecida. ProgAutoReq se restablece porque ProgValueReset siempre está establecida.





# **Programación de texto estructurado**

Estas son las cuestiones únicas de la programación de texto estructurado. Revise los siguientes temas para asegurarse de que comprende cómo se ejecuta la programación de texto estructurado.

[Sintaxis de texto estructurado](#) en la página 955

[Componentes de texto estructurado: Comentarios](#) en la página 957

[Componentes de texto estructurado: Asignaciones](#) en la página 958

[Componentes de texto estructurado: Expresiones](#) en la página 960

[Componentes de texto estructurado: Instrucciones](#) en la página 966

[Componentes de texto estructurado: Construcciones](#) en la página 967

[CASE...OF](#) en la página 970

[FOR...DO](#) en la página 972

[IF...THEN](#) en la página 975

[REPEAT\\_UNTIL](#) en la página 978

[WHILE\\_DO](#) en la página 981

## **Sintaxis de texto estructurado**

El texto estructurado es un lenguaje de programación textual que usa instrucciones para definir lo que se ejecuta.

- El texto estructurado no distingue entre mayúsculas y minúsculas.
- Use las tabulaciones y los retornos de carro (líneas aparte) para facilitar la legibilidad de su texto estructurado. No tienen ningún efecto en la ejecución del texto estructurado.

El texto estructurado no distingue entre mayúsculas y minúsculas. El texto estructurado puede contener estos componentes.

Término	Definición	Ejemplos
Asignación	Use una instrucción de asignación para asignar valores a etiquetas. El operador := es el operador de asignación. Termine la asignación con un punto y coma ;.	etiqueta := expresión;
Expresión	Una expresión es parte de una asignación completa o una instrucción de construcción. Una expresión evalúa hasta un número (expresión numérica), una cadena (expresión de cadena) o un estado verdadero o falso (expresión BOOL).	
Expresión de etiqueta	Un área con nombre de la memoria donde se almacenan los datos (BOOL, SINT, INT, DINT, REAL, Cadena).	value1
Expresión inmediata	Un valor constante	4
Expresión de operadores	Un símbolo o regla mnemotécnica que especifica una operación dentro de una expresión.	tag1 + tag2 tag1 >= value1
Expresión de función	Cuando se ejecuta, una función genera un valor. Use paréntesis para delimitar el operando de una función. Aunque su sintaxis es parecida, las funciones se distinguen de las instrucciones porque las funciones solo se pueden usar en expresiones. Las instrucciones no se pueden utilizar en expresiones.	función(tag1)
Instrucción	Una instrucción es independiente. Una instrucción utiliza los paréntesis para incluir sus operandos. Dependiendo de la instrucción, puede haber cero, uno o varios operandos. Cuando una instrucción se ejecuta, genera uno o varios valores que forman parte de una estructura de datos. Las instrucciones se terminan un punto y coma (;). Aunque su sintaxis es parecida, las instrucciones se distinguen de las funciones porque las instrucciones no se pueden usar en expresiones. Las funciones solo se pueden usar en expresiones.	instrucción();  instruction(operand);  instruction(operand1, operand2, operand3);
Construcción	Una declaración condicional usada para desencadenar el código de texto estructurado (es decir, otras instrucciones). Las construcciones se terminan un punto y coma (;).	IF...THEN CASE FOR...DO WHILE...DO REPEAT...UNTIL EXIT
Comentario	Texto que explica o aclara lo que hace una sección de texto estructurado. Use los comentarios para que sea más sencillo interpretar el texto estructurado. Los comentarios no afectan a la ejecución del texto estructurado. Los comentarios pueden aparecer en cualquier parte del texto estructurado.	//comment  (*start of comment . . . end of comment*)  /*start of comment . . . end of comment*/

### Consulte también

[Componentes de texto estructurado: Asignaciones](#) en la página 958

[Componentes de texto estructurado: Expresiones](#) en la página 960

[Componentes de texto estructurado: Instrucciones](#) en la página 966

[Componentes de texto estructurado: Construcciones](#) en la página 967

[Componentes de texto estructurado: Comentarios](#) en la página 957

## Componentes de texto estructurado: Comentarios

Para hacer que su texto estructurado sea más fácil de interpretar, añádale comentarios.

- Los comentarios le permiten usar un lenguaje llano para describir cómo funciona el texto estructurado.
- Los comentarios no afectan a la ejecución del texto estructurado.

**Para añadir comentarios a su texto estructurado:**

Para añadir un comentario	Use uno de estos formatos
en una única línea	//comment (*comment*)
al final de una línea de texto estructurado	/*comment*/
dentro de una línea de texto estructurado	(*comment*) /*comment*/
eso abarca más de una línea	(*start of comment...end of comment*) /*start of comment...end of comment*/

Por ejemplo:

Formato	Ejemplo
//comment	<b>Al comienzo de una línea</b> //Comprobar dirección de cinta transportadora IF conveyor_direction THEN... <b>Al final de una línea</b> ELSE //If conveyor isn't moving, set alarm light light := 1; END_IF;
(*comment*)	Sugar.Inlet[:]=1;(*open the inlet*) IF Sugar.Low (*low level LS*)& Sugar.High (*high level LS*)THEN... (*Controls the speed of the recirculation pump. The speed depends on the temperature in the tank.*) IF tank.temp > 200 THEN...
/*comment*/	Sugar.Inlet:=0; /*close the inlet*/ IF bar_code=65 /*A*/ THEN... /*Gets the number of elements in the Inventory array and stores the value in the Inventory_Items tag*/ SIZE(Inventory,0,Inventory_Items);

## Componentes de texto estructurado: Asignaciones

Utilice una asignación para cambiar el valor almacenado dentro de una etiqueta. Una asignación tiene esta sintaxis:

*etiqueta := expresión;*

donde:

Componente	Descripción	
Etiqueta	Representa la etiqueta que está obteniendo el nuevo valor; la etiqueta debe ser de tipo BOOL, SINT, INT, DINT, STRING o REAL. <b>Consejo:</b> La etiqueta STRING solo es aplicable a Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580..	
<code>:=</code>	Es el símbolo de asignación.	
Expresión	Representa el nuevo valor que se asigna a la etiqueta <b>Si la etiqueta es de este tipo de datos</b> <b>Use este tipo de expresión</b>	
<code>BOOL</code>	<code>BOOL</code>	
<code>SINT</code> <code>INT</code> <code>DINT</code> <code>REAL</code>	Numérico	
<code>STRING</code> (solo para Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.).	Tipo de cadena, incluyendo la etiqueta de cadena y el literal de cadena (solo Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.).	
<code>;</code>	Finaliza la asignación	

La etiqueta retiene el valor asignado hasta que otra asignación cambie su valor.

La expresión puede ser sencilla, como un valor inmediato u otro nombre de etiqueta, o compleja e incluir varios operadores y funciones o ambos. Consulte la sección “Expresiones” para obtener más información.

- Cons** Los datos del módulo E/S se actualizan de manera asíncrona con respecto a la ejecución de la lógica. Si hace referencia a una entrada varias veces en su lógica, la entrada podría cambiar de estado entre una referencia y otra. Si necesita que la entrada conserve el mismo estado para cada referencia, almacene en búfer el valor de la entrada y haga referencia a la etiqueta de ese búfer. Para obtener más información, consulte LOGIX 5000 Controllers Common Procedures, publication 1756-PM001.
- También puede usar los parámetros de programa Input y Output, que almacenan los datos en el búfer de manera automática durante la ejecución de la lógica. Consulte LOGIX 5000 Controllers Program Parameters Programming Manual, publication 1756-PM021.

### Especificar una asignación no retentiva

La asignación no retentiva es distinta a la asignación regular descrita anteriormente en cuanto a que la etiqueta de una asignación no retentiva se restablece a cero cada vez que el controlador:

- Pasa al modo Marcha
- Deja el paso de un SFC si configura el SFC para el Restablecimiento automático. Esto solo se aplica si incrusta la asignación en la acción del paso o usa la acción para llamar una rutina de texto estructurado mediante el uso de una instrucción JSR.

Una asignación no retentiva tiene esta sintaxis:

*etiqueta [:=] expresión ;*

donde:

Componente	Descripción	
<i>etiqueta</i>	Representa la etiqueta que está obteniendo el nuevo valor; la etiqueta debe ser de tipo BOOL, SINT, INT, DINT, STRING o REAL. <b>Consejo:</b> La etiqueta STRING solo es aplicable a Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580..	
<i>[:=]</i>	Es el símbolo de la asignación no retentiva.	
<i>expresión</i>	Representa el nuevo valor que se asigna a la etiqueta.	
	<b>Si la etiqueta es de este tipo de datos</b>	<b>Use este tipo de expresión</b>
BOOL	BOOL	
SINT		Numérico
INT		
DINT		
REAL		

	STRING (solo para Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.).	Tipo de cadena, incluyendo la etiqueta de cadena y el literal de cadena (solo Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.)
;	finaliza la asignación	

### Asignar un carácter ASCII a un miembro de datos de cadena

Use el operador de asignación para asignar un carácter ASCII a un elemento del miembro DATA de una etiqueta de cadena. Para asignar un carácter, especifique el valor del carácter o el nombre de la etiqueta, el miembro DATA y el elemento del carácter. Por ejemplo:

Esto está bien	Esto no está bien
string1.DATA[0] := 65;	string1.DATA[0] := A;
string1.DATA[0]:= string2.DATA[0];	string1 := string2; <b>Consejo:</b> Esto asigna todo el contenido de string2 a string1 en lugar de un solo carácter.

Para añadir o insertar una cadena de caracteres a una etiqueta de cadena, use una de las dos instrucciones de cadena ASCII indicadas a continuación:

Para	Usar esta instrucción
Añadir caracteres al final de una cadena	CONCAT
Insertar caracteres en una cadena	INSERT

### Consulte también

[Componentes de texto estructurado: Expresiones](#) en la página 960

[Literales de cadena de caracteres](#) en la página 968

## Componentes de texto estructurado: Expresiones

Una expresión es un nombre de etiqueta, ecuación o comparación. Para escribir una expresión, utilice cualquiera de los siguientes elementos:

- Nombre de etiqueta que almacena el valor (variable)
- Número que introduce directamente en la expresión (valor inmediato)
- Literal de cadena que introduce directamente en la expresión (solo Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580.)
- Funciones, tales como: ABS, TRUNC

- Operadores, como: +, -, <, >, And, Or

Cuando escriba expresiones, cumpla estas reglas generales:

- Utilice cualquier combinación de letras en mayúsculas y minúsculas. Por ejemplo, estas tres variaciones de "AND" son válidas: AND, And y and.
- Para requisitos más complejos, use paréntesis para agrupar expresiones dentro de expresiones. Esto hace que sea más fácil de leer toda la expresión y garantiza que la expresión se ejecute en el orden deseado. Consulte la sección "Orden de ejecución" para obtener más información..

En el texto estructurado se usan estos tipos de expresiones:

**Expresión BOOL:** Una expresión que produce un valor BOOL de 1 (verdadero) o 0 (falso).

- Una expresión booleana usa etiquetas bool, operadores relacionales y operadores lógicos para comparar valores o comprobar si el valor de las condiciones es verdadero o falso. Por ejemplo, tag1>65.
- Una expresión bool sencilla puede ser una etiqueta BOOL única.
- Por lo general, las expresiones booleanas se utilizan para condicionar la ejecución de otra lógica.

**Expresión numérica:** Una expresión que calcula un valor entero o de punto flotante.

- Una expresión numérica usa operadores aritméticos, funciones aritméticas y operadores a nivel de bits. Por ejemplo, tag1+5.
- A menudo, las expresiones numéricas se anidan dentro de expresiones BOOL. Por ejemplo, (tag1+5)>65.

**Expresión de cadena:** Una expresión que representa una cadena.

- Una expresión sencilla puede ser un literal de cadena o una etiqueta de cadena.

Use esta tabla para elegir operadores para sus expresiones.

Si desea	Usar
Calcular un valor aritmético	Operadores y funciones aritméticos
Comparar dos valores o cadenas	Operadores relacionales
Comprobar si el valor de determinadas condiciones es verdadero o falso	Operadores lógicos
Comparar los bits dentro de valores	Operadores a nivel de bits

**Consulte también**[Usar operadores y funciones aritméticos](#) en la página 962[Usar operadores relacionales](#) en la página 964[Usar operadores lógicos](#) en la página 963[Usar operadores a nivel de bits](#) en la página 963**Usar operadores y funciones aritméticos**

Puede combinar varios operadores y funciones en expresiones aritméticas.

Los operadores calculan nuevos valores.

Para	Use este operador	Tipo de datos óptimo
Sumar	+	DINT, REAL
Restar/negar	-	DINT, REAL
Multiplicar	*	DINT, REAL
Exponente (x a la fuente de alimentación de y)	**	DINT, REAL
Dividir	/	DINT, REAL
División módulo	MOD	DINT, REAL

Las funciones realizan operaciones matemáticas. Especifique una constante, una etiqueta no booleana o una expresión para la función.

Para	Use esta función	Tipo de datos óptimo
Valor absoluto	ABS (numeric_expression)	DINT, REAL
Arco coseno	ACOS (numeric_expression)	REAL
Arco seno	ASIN (numeric_expression)	REAL
Arco tangente	ATAN (numeric_expression)	REAL
Coseno	COS (numeric_expression)	REAL
Radianes en grados	DEG (numeric_expression)	DINT, REAL
Logaritmo natural	LN (numeric_expression)	REAL
Logaritmo de base 10	LOG (numeric_expression)	REAL
Grados en radianes	RAD (numeric_expression)	DINT, REAL
Seno	SIN (numeric_expression)	REAL
Raíz cuadrada	SQRT (numeric_expression)	DINT, REAL
Tangente	TAN (numeric_expression)	REAL
Truncar	TRUNC (numeric_expression)	DINT, REAL

En la siguiente tabla se muestran algunos ejemplos.

Use este formato	Ejemplo	
	Para esta situación	Escribir
valor1 operador valor2	Si gain_4 y gain_4_adj son etiquetas DINT y su especificación dice: ‘Añadir 15 a gain_4 y almacenar el resultado en gain_4_adj’	gain_4_adj := gain_4+15;
valor1 de operador	Si alarm y high_alarm son etiquetas DINT y su especificación dice: ‘Negar high_alarm y almacenar el resultado en alarm’	alarm:= -high_alarm;
función(numeric_expression)	Si overtravel y overtravel_POS son etiquetas DINT y su especificación dice: ‘Calcular el valor absoluto de overtravel y almacenar el resultado en overtravel_POS.’	overtravel_POS := ABS(overtravel);
valor1 operador (función((valor2+valor3)/2))	Si adjustment y position son etiquetas DINT y sensor1 y sensor2 son etiquetas de tipo REAL y su especificación dice: ‘Averiguar el valor absoluto de la media de sensor1 y sensor2, añadir el ajuste y almacenar el resultado en position.’	position := adjustment + ABS((sensor1 + sensor2)/2);

### Consulte también

[Componentes de texto estructurado: Expresiones](#) en la página 960

## Usar operadores a nivel de bits

Los operadores a nivel de bits manipulan los bits dentro de un valor basándose en dos valores.

Para	Use este operador	Tipo de datos óptimo
Y a nivel de bits	&, AND	DINT
O a nivel de bits	OR	DINT
O exclusivo a nivel de bits	XOR	DINT
complemento a nivel de bits	NOT	DINT

Este es un ejemplo:

Use este formato	Ejemplo	
	Para esta situación	Escribiría
valor1 operador valor2	Si input1, input2 y result1 son etiquetas DINT y su especificación dice: “Calcular el resultado a nivel de bits de input1 e input2. Almacenar el resultado en result1.”	result1 := input1 AND input2;

### Consulte también

[Componentes de texto estructurado: Expresiones](#) en la página 960

## Usar operadores lógicos

Los operadores lógicos le permiten comprobar si varias condiciones son

verdaderos o falsos. El resultado de una operación lógica es un valor BOOL.

Si la comparación es	El resultado es
verdadera	1
falsa	0

Use estos operadores lógicos.

Para esta comparación	Use este operador	Tipo de datos óptimo
AND lógico	&, AND	BOOL
OR lógico	OR	BOOL
OR exclusivo lógico	XOR	BOOL
complemento lógico	NOT	BOOL

En la siguiente tabla se muestran algunos ejemplos.

Use este formato	Ejemplo	Usar
	Para esta situación	
BOOLtag	Si photoeye es una etiqueta BOOL y su especificación dice: "Si photoeye_1 está activada,...."	IF photoeye THEN...
NOT BOOLtag	Si photoeye es una etiqueta BOOL y su especificación dice: "Si photoeye está desactivada,..."	IF NOT photoeye THEN...
expresión1 & expresión2	Si photoeye es una etiqueta BOOL, temp es una etiqueta DINT y su especificación dice: "Si photoeye está activada y temp es inferior a 100,...."	IF photoeye & (temp<100) THEN...
expresión1 OR expresión2	Si photoeye es una etiqueta BOOL, temp es una etiqueta DINT y su especificación dice: "Si photoeye está activada o temp es inferior a 100,....".	IF photoeye OR (temp<100) THEN...
expresión1 XOR expresión2	Si photoeye1 y photoeye2 son etiquetas BOOL y su especificación dice: "Si: photoeye1 está activada cuando photoeye2 está desactivada o photoeye1 está desactivada cuando photoeye2 está activada, Entonces..."	IF photoeye1 XOR photoeye2 THEN...
BOOLtag := expresión1 & expresión2	Si photoeye1, photoeye2 y open son etiquetas BOOL y su especificación dice: "Si photoeye1 y photoeye2 están activadas, establecer open en verdadero"	open := photoeye1 & photoeye2;

#### Consulte también

[Componentes de texto estructurado: Expresiones](#) en la página 960

## Usar operadores relacionales

Los operadores relacionales comparan dos valores o cadenas para proporcionar un resultado verdadero o falso. El resultado de una operación relacional es un valor BOOL.

Si la comparación es	El resultado es
Verdadera	1
Falsa	0

Use estos operadores relacionales.

Para esta comparación	Use este operador	Tipo de datos óptimo
Igual	=	DINT, REAL, tipo de cadena
Inferior a	<	DINT, REAL, tipo de cadena
Inferior a o igual que	<=	DINT, REAL, tipo de cadena
Superior a	>	DINT, REAL, tipo de cadena
Superior a o igual que	>=	DINT, REAL, tipo de cadena
No igual	<>	DINT, REAL, tipo de cadena

En la siguiente tabla se muestran algunos ejemplos.

Use este formato	Ejemplo	
	Para esta situación	Escribir
valor1 operador valor2	Si temp es una etiqueta DINT y su especificación dice: 'Si temp es inferior a 100,...'	IF temp<100 THEN...
stringtag1 operador stringtag2	Si bar_code y dest son etiquetas de cadena y su especificación dice: 'Si bar_code es igual a dest,...'	IF bar_code=dest THEN...
'literal de cadena de caracteres' de operador stringtag1	Si bar_code es una etiqueta de cadena y su especificación dice: 'Si bar_code es igual a 'Test PASSED', ...'	IF bar_code='Test PASSED' THEN...
carácter1 operador carácter2 Para introducir un carácter ASCII directamente en la expresión, introduzca el valor decimal del carácter.	Si bar_code es una etiqueta de cadena y su especificación dice: 'Si bar_code.DATA[0] es igual a "A",...'	IF bar_code.DATA[0]=65 THEN...
etiqueta_booleana := expresiones_booleanas	Si count y length son etiquetas DINT, done es una etiqueta BOOL y su especificación dice: 'Si count es superior o igual a length, terminó de contar.'	Done := (count >= length);

### Cómo se evalúan las cadenas

Los valores hexadecimales de los caracteres ASCII determinan si una cadena es inferior o superior a otra cadena.

- Cuando las dos cadenas se ordenan como en un directorio telefónico, el orden de las cadenas determina cuál es superior.

ASCII Characters	Hex Codes
1ab	\$31\$61\$62
1b	\$31\$62
A	\$41
AB	\$41\$42
B	\$42
a	\$61
ab	\$61\$62

- Las cadenas son iguales si sus caracteres coinciden.
- Los caracteres distinguen entre mayúsculas y minúsculas. "A" en mayúsculas (\$41) no es lo mismo que "a" en minúsculas (\$61).

#### Consulte también

[Componentes de texto estructurado: Expresiones](#) en la página 960

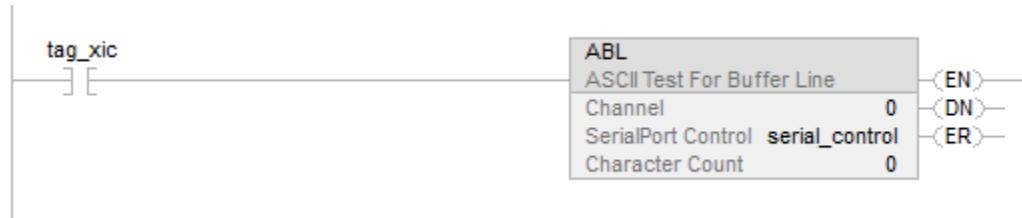
## Componentes de texto estructurado: Instrucciones

Las instrucciones de texto estructurado también pueden ser instrucciones. Una instrucción de texto estructurado se ejecuta cada vez que se escanea. Una instrucción de texto estructurado dentro de una construcción se ejecuta cada vez que las condiciones de la construcción son verdaderas. Si las condiciones de la construcción son falsas, no se escanean las instrucciones dentro de la construcción. No hay condición de reglón ni transición de estado que desencadene la ejecución.

Esto es distinto a las instrucciones del bloque de funciones que usan EnableIn para desencadenar la ejecución. Las instrucciones de texto estructurado se ejecutan como si EnableIn siempre estuviese establecido.

Esto también es distinto a las instrucciones de escalera de relé que usan condición de entrada de reglón para desencadenar la ejecución. Algunas instrucciones de escalera de relé solo se ejecutan cuando el valor de la condición de entrada de reglón cambia de falso a verdadero. Estas son las instrucciones de escalera de relé de transición. En el texto estructurado, las instrucciones se ejecutarán cada vez que se escaneen salvo que precondicione la ejecución del texto estructurado.

Por ejemplo, la instrucción ABL es una instrucción de transición en escalera de relé. En este ejemplo, la instrucción ABL solo se ejecuta en un escaneado cuando tag\_xic pasa de tener su valor borrado a establecido. La instrucción ABL no se ejecuta cuando el valor de tag\_xic se mantiene establecido o se borra.



En el texto estructurado, si escribe este ejemplo como:

```
IF tag_xic THEN ABL(0,serial_control);
END_IF;
```

La instrucción ABL se ejecutará cada escaneado en el que tag\_xic esté establecido, no solo cuando pase de tener su valor borrado a establecido.

Si quiere que la instrucción ABL solo se ejecute cuando tag\_xic pase de tener su valor borrado a establecido, debe condicionar la instrucciones de texto estructurado. Utilice un impulso para desencadenar la ejecución.

```
osri_1.InputBit := tag_xic;
OSRI(osri_1);
```

```
IF (osri_1.OutputBit) THEN
  ABL(0,serial_control);
END_IF;
```

## Componentes de texto estructurado: Construcciones

Las construcciones pueden programarse solos o anidarse dentro de otros construcciones.

### Si desea

Hacer algo si o cuando se produzcan unas determinadas condiciones  
Seleccionar qué hacer en función de un valor numérico

### Use esta construcción

IF... THEN  
CASE... OF

Hacer algo un número de veces determinado antes de hacer cualquier otra cosa	FOR...DO
Seguir haciendo algo mientras el valor de ciertas condiciones sea verdadero	WHILE...DO
Seguir haciendo algo hasta que una condición sea verdadera	REPEAT...UNTIL

### Algunas palabras clave están reservadas

Estas construcciones no están disponibles:

- GOTO
- REPEAT

La aplicación Logix Designer no le dejará usarlos como nombres de etiqueta o construcciones.

### Consulte también

[IF THEN](#) en la página 975

[CASE OF](#) en la página 970

[FOR DO](#) en la página 972

[WHILE DO](#) en la página 981

[REPEAT UNTIL](#) en la página 978

## Literales de cadena de caracteres

Los literales de cadenas de caracteres incluyen caracteres codificados de uno o dos bytes. Un literal de cadena de un byte es una secuencia de cero o más caracteres que van precedidos y seguidos por el carácter de comilla simple (''). En las cadenas de caracteres de un byte, la combinación de tres caracteres del símbolo del dólar (\$) seguida por dos dígitos hexadecimales se interpreta como la representación hexadecimal del código de caracteres de 8 bits, tal como se muestra en la siguiente tabla.

- Cons ejos:**
- Los literales de cadena de caracteres solo se aplican a Controladores CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380 y GuardLogix 5580..
  - Studio 5000 solo admite caracteres de un byte.

### Literales de cadena de caracteres

No.	Descripción	Ejemplo
1a	Cadena vacía (longitud cero)	"
1b	Cadena de longitud uno o carácter CHAR que contiene un solo carácter	'A'

1c	Cadena de longitud uno o carácter CHAR que contiene el carácter “space”.	''
1d	Cadena de longitud uno o carácter CHAR que contiene el carácter “single quote”	'\$"
1e	Cadena de longitud uno o carácter CHAR que contiene el carácter “double quote”	""
1f	Admite combinaciones de dos caracteres	'\$R\$L'
1g	Admite la representación de caracteres mediante «\$» y dos caracteres hexadecimales.	'\$0A'

### Combinaciones de dos caracteres en cadenas de caracteres

No.	Descripción	Ejemplo
1	Símbolo del dólar	\$\$
2	Comilla sencilla	\$'
3	Salto de línea	\$L o \$I
4	Nueva línea	\$N o \$n
5	Avance de página (página)	\$P o \$p
6	Retorno de carro	\$R o \$r
7	Tabulador	\$T o \$t

- Cons ejos:**
- El carácter de nueva línea proporciona un medio independiente de la implementación de definir el final de una línea de datos para E/S física y de archivo; para imprimir, el efecto es el de terminar una línea de datos y reanudar la impresión al comienzo de la siguiente línea.
  - La combinación \$' solo es válida dentro de literales de cadena de comilla sencilla.

### Consulte también

[Componentes de texto estructurado: Asignaciones](#) en la página 958

[Tipos de cadena](#) en la página 867

## Tipos de cadena

Almacene caracteres ASCII en etiquetas que usan datos de tipo cadena para:

- Usar el tipo de datos STRING predeterminado, que puede almacenar hasta 82 caracteres.
- Crear un nuevo tipo de cadena que almacene menos o más caracteres.

Para crear un nuevo tipo de cadena, consulte LOGIX 5000 Controllers ASCII Strings Programming Manual 1756-PM013.

Cada tipo de cadena contiene los siguientes miembros:

Nombre	Tipo de datos	Descripción	Notas
LEN	DINT	número de caracteres de la cadena	<p>El miembro LEN se actualiza automáticamente al nuevo conteo de caracteres cada vez que lo usa:</p> <ul style="list-style-type: none"> <li>• El Explorador de cadenas para introducir caracteres.</li> <li>• Instrucciones que lean, conviertan o manipulen una cadena.</li> </ul> <p>El miembro LEN muestra la longitud de la cadena actual. El miembro DATA puede contener caracteres adicionales y antiguos que no estén incluidos en el conteo de LEN.</p>
DATA	Matriz SINT	Caracteres ASCII de la cadena	<p>Para acceder a los caracteres de la cadena, dirige al nombre de la etiqueta. Por ejemplo, para acceder a los caracteres de la etiqueta string_1, introduzca string_1.</p> <p>Cada uno de los elementos de la matriz DATA contiene un carácter.</p> <p>Crear nuevos tipos de cadena que almacenen menos o más caracteres..</p>

### Consulte también

[Literales de cadena de caracteres](#) en la página 968

## CASE\_OF

Use CASE\_OF para seleccionar qué hacer con base en un valor numérico.

### Operandos

CASE numeric\_expression OF

selector1: statement;

selectorN: statement; ELSE

### Texto estructurado

Operando	Tipo	Formato	Introducir
Numeric_expression	SINT INT DINT REAL	Etiqueta Expresión	Etiqueta o expresión que evalúa hasta un número (expresión numérica)
Selector	SINT INT DINT REAL	Inmediato	Mismo tipo que numeric_expression

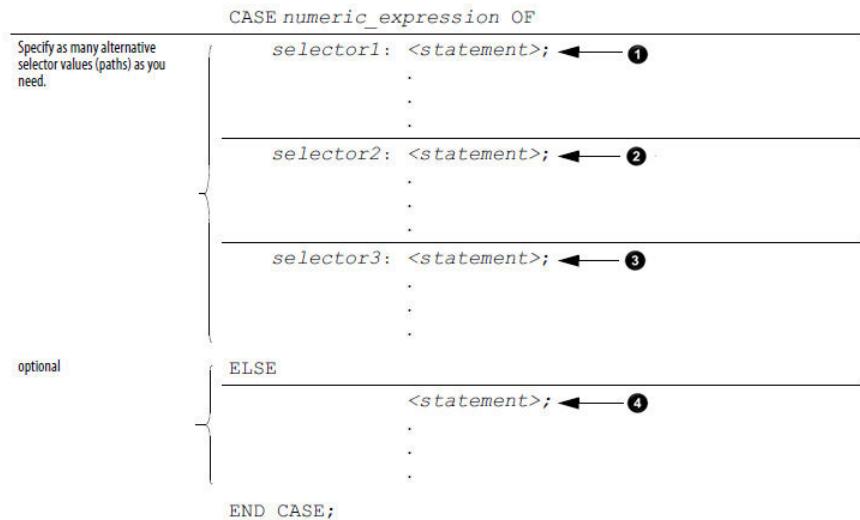
---

**Importante:** Si usa valores de tipo REAL, use un rango de valores para un selector porque un valor REAL tiene mayor probabilidad de estar dentro de un rango de valores que una coincidencia exacta de un valor específico.

---

## Descripción

La sintaxis se describe en la tabla.



Estas son las sintaxis para introducir los valores de selector.

Cuando el selector es	Introducir
Un valor	value: statement
Varios valores distintos	value1, value2, valueN : <statement> Use una coma (,) para separar cada valor.
Un rango de valores	value1..valueN : <statement> Use dos puntos(..) para identificar el rango.
Valores diferentes más un rango de valores	valuea, valueb, value1..valueN : <statement>

La construcción CASE es similar a una instrucción de interruptor en los lenguajes de programación C o C++. Sin embargo, con la construcción CASE, el controlador solo ejecuta las instrucciones asociadas al primer valor de selector coincidente. La ejecución siempre se interrumpe después de las instrucciones de ese selector y va a la instrucción END\_CASE.

**Afecta a las marcas de estado matemático**

No

**Fallos mayores/menores**

Ninguno

**Ejemplo**

<b>Si quiere esto</b>	<b>Introduzca este texto estructurado</b>
Si número de receta = 1, entonces Salida 1 de Ingrediente A = abierta (1) Salida 4 de Ingrediente B = abierta (1)	CASE recipe_number OF 1: Ingredient_A.Outlet_1 :=1; Ingredient_B.Outlet_4 :=1;
Si número de receta = 2 o 3, entonces  Salida 4 de Ingrediente A = abierta (1) Salida 2 de Ingrediente B = abierta (1)	2,3: Ingredient_A.Outlet_4 :=1; Ingredient_B.Outlet_2 :=1;
Si número de receta = 4, 5, 6 o 7, entonces Salida 4 de Ingrediente A = abierta (1) Salida 2 de Ingrediente B = abierta (1)	De 4 a 7: Ingredient_A.Outlet_4 :=1; Ingredient_B.Outlet_2 :=1;
Si número de receta = 8, 11, 12 o 13, entonces Salida 1 de Ingrediente A = abierta (1) Salida 4 de Ingrediente B = abierta (1)	8,11...13 Ingredient_A.Outlet_1 :=1; Ingredient_B.Outlet_4 :=1;
De lo contrario, todas las salidas = cerradas (0)	ELSE  Ingredient_A.Outlet_1 :=0; Ingredient_A.Outlet_4 :=0; Ingredient_B.Outlet_2 :=0; Ingredient_B.Outlet_4 :=0;  END_CASE;

[:=] también indica al controlador que borre las etiquetas de salida cuando el controlador hace lo siguiente:

Pasa al modo Marcha.

Deja el paso de un SFC si configura el SFC para el Restablecimiento automático.  
(Esto solo se aplica si incrusta la asignación en la acción del paso o usa la acción para llamar una rutina de texto estructurado mediante una instrucción JSR).

**FOR\_DO**

Use el lazo FOR\_DO para realizar algo un número de veces específico antes de hacer cualquier otra cosa.

**Operandos**

FOR count:= initial\_value TO

final\_value BY increment DO

<statement>;

END\_FOR;

Operando	Tipo	Formato	Descripción
count	SINT INT DINT	Etiqueta	Etiqueta para almacenar la posición de conteo a medida que se ejecuta el lazo FOR_DO.
initial_value	SINT INT DINT	Etiqueta Expresión Inmediato	Debe evaluar hasta un número Especifica el valor inicial de conteo
final_value	SINT INT DINT	Etiqueta Expresión Inmediato	Especifica el valor final de conteo, que determina cuándo salir del lazo
increment	SINT INT DINT	Etiqueta Expresión Inmediato	(Opcional) cantidad en la que se debe incrementar el conteo cada vez que se pasa el lazo Si no especifica un valor de incremento, el conteo se aumentará de 1 en 1.

**Importante:** Asegúrese de que no itere dentro del lazo demasiadas veces en un mismo escaneado.

El controlador no ejecuta otras instrucciones de la rutina hasta que complete el lazo.

Si el tiempo que tarda en completarse el lazo es superior al temporizador de vigilancia para la tarea, se produce un fallo mayor.

Consideré la posibilidad de usar otro construcción, como IF\_THEN.

## Descripción

La sintaxis se describe en la tabla.

```

FOR count := initial_value
    TO final_value
    optional [ BY increment ]                         If you don't specify an increment, the loop
                                                       increments by 1.
                                                      
    DO
        <statement>;
    optional { IF bool_expression THEN
        EXIT;
        END_IF; }                                     If there are conditions when you want to
                                                       exit the loop early, use other statements,
                                                       such as an IF...THEN construct, to
                                                       condition an EXIT statement.
                                                      
    END_FOR;
  
```

Los siguientes diagramas muestran cómo se ejecuta un lazo FOR\_DO y cómo sale del lazo de forma prematura una instrucción EXIT.

<pre> graph TD     S1[Statement 1] --&gt; S2[Statement 2]     S2 --&gt; S3[Statement 3]     S3 --&gt; S4[Statement 4]     S4[...] --&gt; Done{Done x Number Of Times?}     Done -- No --&gt; S2     Done -- Yes --&gt; Rest[Rest Of The Routine]   </pre>	<pre> graph TD     S1[Statement 1] --&gt; S2[Statement 2]     S2 --&gt; S3[Statement 3]     S3 --&gt; S4[Statement 4]     S4[...] --&gt; Done{Done x Number Of Times?}     Done -- No --&gt; S2     Done -- Yes --&gt; Rest[Rest Of The Routine]   </pre>
El lazo FOR_DO se ejecuta un número de veces específico.	Para detener el lazo antes de que el conteo alcance el último valor, use una instrucción EXIT.

### Afecta a las marcas de estado matemático

No

### Fallos mayores/menores

Se producirá un fallo mayor si	Tipo de fallo	Código de fallo
El lazo de la construcción es demasiado largo.	6	1

### Ejemplo 1

Si quiere esto	Introduzca este texto estructurado
Borrar el valor de bits 0...31 en una matriz de valores de tipo BOOL: Inicializar la etiqueta de subíndice a 0. Borrar i. Por ejemplo, cuando subscript = 5, borrar array[5]. Añadir 1 a subscript. Si subscript es ≤ 31, repetir 2 y 3. De lo contrario, detener.	Para subscript:=0 a 31 por 1, hacer array[subscript] := 0; End_for;

### Ejemplo 2

Si quiere esto	Introduzca este texto estructurado
Un tipo de datos definido por el usuario (estructura) almacena la siguiente información sobre un elemento en su inventario: <ul style="list-style-type: none"><li>• ID de códigos de barra del elemento (tipo de datos de cadena)</li><li>• Cantidad en depósito del elemento (tipo de datos DINT)</li></ul>	SIZE(Inventory,0,Inventory_Items); For position:=0 to Inventory_Items - 1 do If Barcode = Inventory[position].ID then Quantity := Inventory[position].Qty; Exit; End_if;

<p>Una matriz de la estructura anterior contiene un elemento para cada elemento diferente de su inventario. Debe buscar la matriz de un producto específico (use su código de barras) y determine la cantidad que hay en depósito.</p> <ol style="list-style-type: none"> <li>1. Obtenga el tamaño (número de elementos) de la matriz Inventory y almacene el resultado en etiqueta DINT).</li> <li>2. Inventory_Items (etiqueta DINT).</li> </ol> <p>Inicialice la etiqueta position a 0.</p> <ol style="list-style-type: none"> <li>3. Si Barcode coincide con el ID de un elemento de la matriz, entonces:</li> </ol> <p>Establecer la etiqueta Quantity = Inventory[position].Qty. Esto produce la cantidad que hay en depósito del elemento.</p> <p>Detener.</p> <p>Barcode es una etiqueta de cadena que almacena el código de barras del elemento que está buscando. Por ejemplo , cuando position = 5, comparar Barcode con Inventory[5].ID.</p> <ol style="list-style-type: none"> <li>4. Añadir 1 a position.</li> <li>5. Si position es <math>\leq</math> (Inventory_Items -1), repetir 3 y 4. Dado que los números de elemento comienzan en 0, el último elemento es menor que el número de elementos de la matriz en 1.</li> </ol> <p>De lo contrario, detener.</p>	End_for;
--	----------

## IF\_THEN

Use IF\_THEN para hacer algo si o cuando se produzcan unas determinadas condiciones.

### Operandos

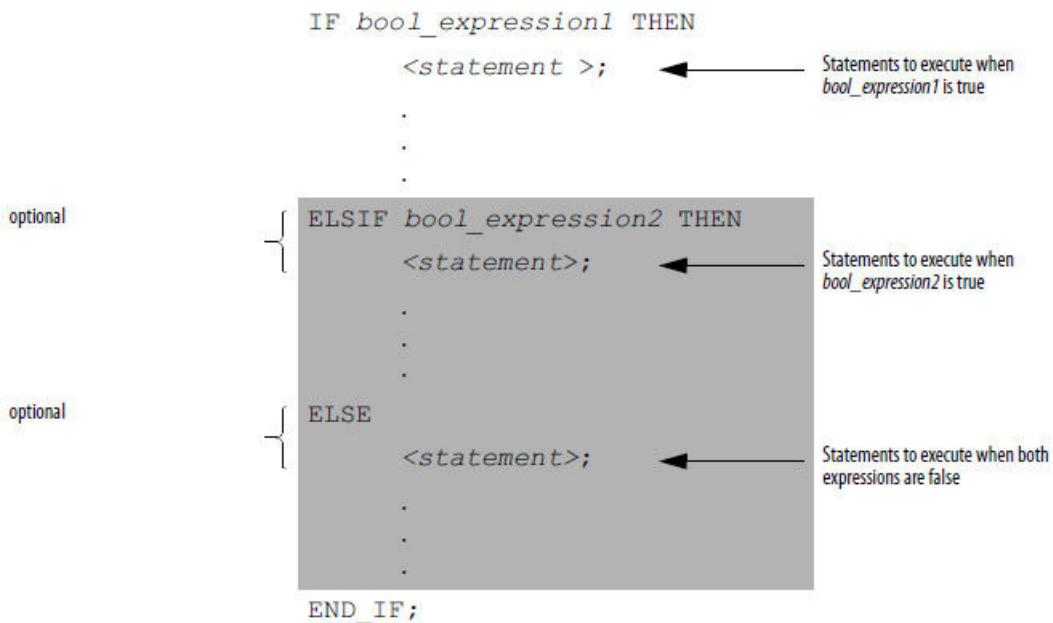
IF bool\_expression THEN

<statement>;

Operando	Tipo	Formato	Introducir
Bool_expression	BOOL	Etiqueta Expresión	Etiqueta o expresión BOOL que evalúa hasta un valor BOOL (expresión BOOL)

## Descripción

La sintaxis se describe en la tabla.



Para usar ELSIF o ELSE, siga estas pautas.

Para seleccionar de entre varios grupos de instrucciones posibles, añada una o varias instrucciones ELSIF.

Cada ELSIF representa una ruta alternativa.

Especifique tantas rutas ELSIF como necesite.

El controlador ejecuta el primer IF o ELSIF que sea verdadero e ignora el resto de las ELSIF y la ELSE.

Para realizar algo cuando todas las condiciones IF o ELSIF sean falsas, añada una instrucción ELSE.

La tabla resume diferentes combinaciones de IF, THEN, ELSIF y ELSE.

Si desea	Y	Use esta construcción
Hacer algo si o cuando las condiciones sean verdaderas	No hacer nada si las condiciones son falsas	IF_THEN
	Hacer otra cosa si las condiciones son falsas	IF_THEN_ELSE
Elegir de entre instrucciones alternativas (o grupos de	No hacer nada si las condiciones son falsas	IF_THEN_ELSIF

instrucciones) basándose en las condiciones de la entrada	Asignar instrucciones predeterminadas si todas las condiciones son falsas	IF_THEN_ELSIF_ELSE
---	---	--------------------

**Afecta a las marcas de estado matemático**

No

**Condiciones de fallo**

Ninguno.

**Ejemplos****Ejemplo 1**

IF...THEN

Si quiere esto	Introduzca este texto estructurado
Si el número de rechazos > 3, entonces	IF rejects > 3 THEN
transportador = apagado (0)	conveyor := 0;
alarma = activada (1)	alarm := 1;
	END_IF;

**Ejemplo 2**

IF\_THEN\_ELSE

Si quiere esto	Introduzca este texto estructurado
Si el contacto de dirección del transportador = avance (1), entonces	IF conveyor_direction THEN
luz = apagada	light := 0;
De lo contrario, luz = encendida	ELSE
	light [:=] 1;
	END_IF;

[:=] indica al controlador que borre la luz si el controlador hace lo siguiente:

Pasa al modo Marcha.

Deja el paso de un SFC si configura el SFC para el Restablecimiento automático.  
 (Esto solo se aplica si incrusta la asignación en la acción del paso o usa la acción para llamar una rutina de texto estructurado mediante una instrucción JSR).

**Ejemplo 3**

IF...THEN...ELSIF

<b>Si quiere esto</b>	<b>Introduzca este texto estructurado</b>
Si el interruptor limitador de nivel de azúcar bajo = bajo (activado) y el interruptor limitador de nivel de azúcar alto = no alto (activado), entonces	IF Sugar.Low & Sugar.High THEN
válvula de entrada = abierta (activado)	Sugar.Inlet [:=] 1;
Hasta que el interruptor limitador de nivel de azúcar alto = alto (desactivado)	ELSIF NOT(Sugar.High) THEN
	Sugar.Inlet := 0;
	END_IF;

[:=] indica al controlador que borre Sugar.Inlet si el controlador hace lo siguiente:

Pasa al modo Marcha.

Deja el paso de un SFC si configura el SFC para el Restablecimiento automático.  
 (Esto solo se aplica si incrusta la asignación en la acción del paso o usa la acción para llamar una rutina de texto estructurado mediante una instrucción JSR).

**Ejemplo 4**

IF...THEN...ELSIF...ELSE

<b>Si quiere esto</b>	<b>Introduzca este texto estructurado</b>
Si la temperatura del depósito > 100	IF tank.temp > 200 THEN
entonces la bomba = lenta	pump.fast :=1; pump.slow :=0; pump.off :=0;
Si la temperatura del depósito > 200	ELSIF tank.temp > 100 THEN
entonces la bomba = rápida	pump.fast :=0; pump.slow :=1; pump.off :=0;
De lo contrario la bomba = apagada	ELSE
	pump.fast :=0; pump.slow :=0; pump.off :=1;
	END_IF;

**REPEAT\_UNTIL**

Use el lazo REPEAT\_UNTIL para seguir haciendo algo hasta que el valor de ciertas condiciones sea verdadero.

**Operandos**

REPEAT

&lt;statement&gt;;

## Texto estructurado

Operando	Tipo	Formato	Introducir
bool_expression	BOOL	Etiqueta Expresión	Etiqueta o expresión BOOL que evalúa hasta un valor BOOL (expresión BOOL)

**Importante:** Asegúrese de que no itere dentro del lazo demasiadas veces en un mismo escaneado. El controlador no ejecuta otras instrucciones de la rutina hasta que complete el lazo. Si el tiempo que tarda en completarse el lazo es superior al temporizador de vigilancia para la tarea, se produce un fallo mayor. Considere la posibilidad de usar otro construcción, como IF\_THEN.

## Descripción

La sintaxis es:

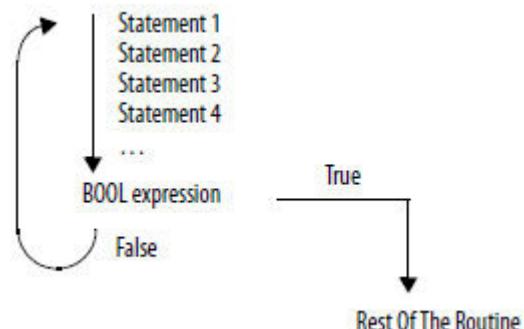
```

REPEAT
    <statement>;           ← statements to execute while
                           bool_expression1 is false
    optional {             ← If there are conditions when you want to
                           exit the loop early, use other statements,
                           such as an IF...THEN construct, to
                           condition an EXIT statement.
        IF bool_expression2 THEN
            EXIT;
        END_IF;
    }
    UNTIL bool_expression1
END_REPEAT;

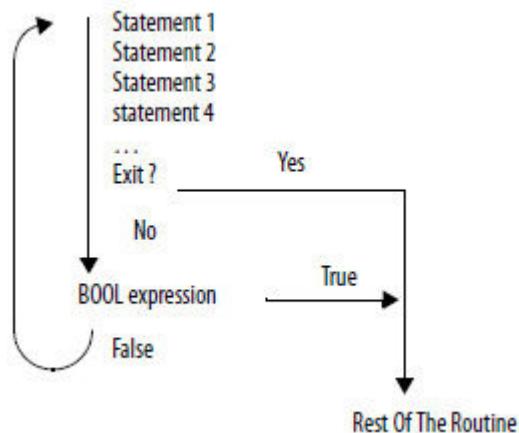
```

Los siguientes diagramas muestran cómo se ejecuta un lazo REPEAT\_UNTIL y cómo sale del lazo de forma prematura una instrucción EXIT.

Mientras el valor de bool\_expression sea falso, el controlador ejecutará únicamente las instrucciones que se encuentren dentro del lazo REPEAT\_UNTIL.



Para detener el lazo antes de que las condiciones sean falso, use una instrucción EXIT.



#### Afecta a las marcas de estado matemático

No

#### Condiciones de fallo

Se producirá un fallo mayor si	Tipo de fallo	Código de fallo
El lazo de la construcción es demasiado largo	6	1

#### Ejemplo 1

Si quiere esto	Introduzca este texto estructurado
<p>El lazo REPEAT_UNTIL ejecuta las declaraciones de la construcción y después determina si el valor de las condiciones es verdadero antes de volver a ejecutar las instrucciones. Esto se distingue del lazo WHILE_DO porque este evalúa sus condiciones primero.</p> <p>Si el valor de las condiciones es verdadero, el controlador ejecuta las instrucciones que se encuentren dentro del lazo. Las instrucciones de un lazo REPEAT_UNTIL siempre se ejecutan al menos una vez. Puede que las instrucciones de un lazo WHILE_DO nunca lleguen a ejecutarse.</p>	<pre> pos := -1; REPEAT   pos := pos + 2; UNTIL ((pos = 101) OR (structarray[pos].value = targetvalue)) end_repeat;   </pre>

#### Ejemplo 2

Si quiere esto	Introduzca este texto estructurado
Mover caracteres ASCII de una matriz SINT a una etiqueta de cadena. (En una matriz SINT cada elemento	<pre> element_number := 0; SIZE(SINT_array, 0, SINT_array_size);   </pre>

<p>tiene un carácter). Detener al alcanzar el retorno de carro.</p> <p>Inicializar Element_number a 0.</p> <p>Cuento el número de elementos en SINT_array (matriz que contiene los caracteres ASCII) y almacene el resultado en SINT_array_size (etiqueta DINT).</p> <p>Establecer String_tag[element_number] = el carácter en SINT_array[element_number].</p> <p>Añadir 1 al valor de element_number. Esto deja que el controlador compruebe el siguiente carácter de SINT_array.</p> <p>Establecer el miembro Length de String_tag = element_number. (Esto registra el número de caracteres que hay en String_tag hasta el momento).</p> <p>Si element_number = SINT_array_size, a continuación se detiene. (Se encuentra al final de la matriz y no contiene un retorno de carro).</p> <p>Si el carácter en SINT_array[element_number] = 13 (valor decimal del retorno de carro), parar.</p>	<pre>Repeat String_tag.DATA[element_number] := SINT_array[element_number]; element_number := element_number + 1; String_tag.LEN := element_number; If element_number = SINT_array_size then exit; end_if; Until SINT_array[element_number] = 13 end_repeat;</pre>
---	---

## WHILE\_DO

Use el lazo WHILE\_DO para seguir haciendo algo mientras el valor de ciertas condiciones sea verdadero.

### Operandos

WHILE *bool\_expression* DO

<statement>;

### Texto estructurado

Operando	Tipo	Formato	Descripción
<i>bool_expression</i>	BOOL	etiqueta expresión	Etiqueta o expresión BOOL que evalúa hasta un valor BOOL

**Importante:** Asegúrese de que no itere dentro del lazo demasiadas veces en un mismo escaneado. El controlador no ejecuta otras instrucciones de la rutina hasta que complete el lazo. Si el tiempo que tarda en completarse el lazo es superior al temporizador de vigilancia para la tarea, se produce un fallo mayor. Considere la posibilidad de usar otro construcción, como IF\_THEN.

### Descripción

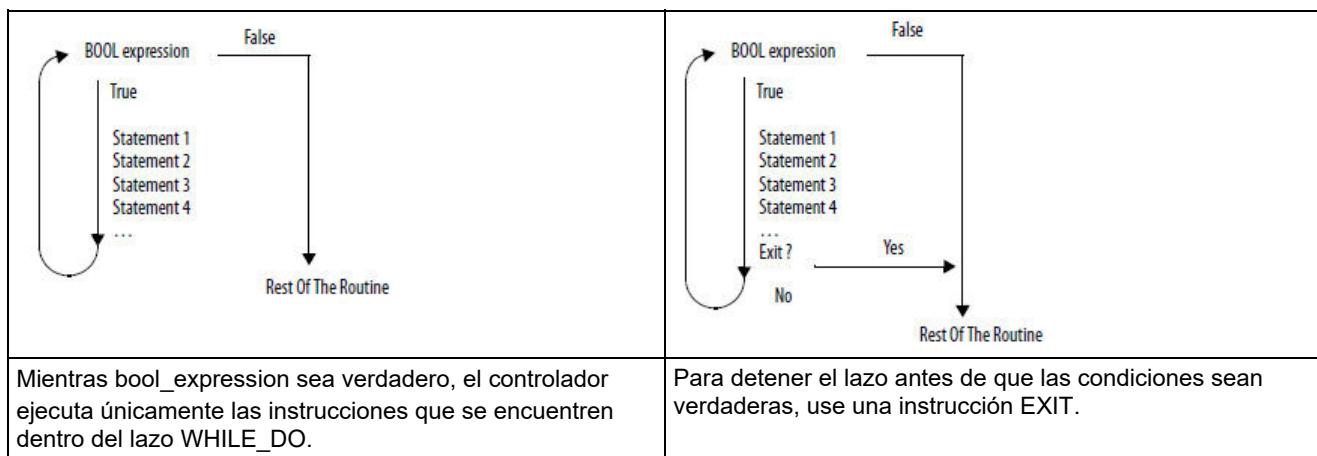
La sintaxis es:

```

WHILE bool_expression1 DO
    <statement>;           ← statements to execute while
                            bool_expression1 is true
    optional {             ←
        IF bool_expression2 THEN
            EXIT;           ← If there are conditions when you want to
                            exit the loop early, use other statements,
                            such as an IF...THEN construct, to
                            condition an EXIT statement.
        END_IF;
    }
END WHILE;

```

Los siguientes diagramas muestran cómo se ejecuta un lazo WHILE\_DO y cómo sale del lazo de forma prematura una instrucción EXIT.



### Afecta a las marcas de estado matemático

No

### Condiciones de fallo

Se producirá un fallo mayor si	Tipo de fallo	Código de fallo
el lazo de la construcción es demasiado largo	6	1

### Ejemplo 1

Si quiere esto	Introduzca este texto estructurado
<p>El lazo WHILE_DO evalúa primero sus condiciones. Si el valor de las condiciones es verdadero, el controlador ejecuta las instrucciones que se encuentren dentro del lazo.</p> <p>Esto es distinto al lazo REPEAT_UNTIL en cuanto a que este ejecuta las declaraciones del construcción y después determina si el valor de las condiciones es verdadero antes de volver a ejecutar las instrucciones. Las instrucciones de un lazo REPEAT_UNTIL siempre se ejecutan al menos una vez. Puede que las instrucciones de un lazo WHILE_DO nunca lleguen a ejecutarse.</p>	<pre>pos := 0; While ((pos &lt;= 100) &amp; structarray[pos].value &lt;&gt; targetvalue) do     pos := pos + 2;     String_tag.DATA[pos] := SINT_array[pos]; end_while;</pre>

### Ejemplo 2

Si quiere esto	Introduzca este texto estructurado
<p>Mover caracteres ASCII de una matriz SINT a una etiqueta de cadena. (En una matriz SINT cada elemento tiene un carácter). Detener al alcanzar el retorno de carro.</p> <p>Inicializar Element_number a 0.</p> <p>Cuento el número de elementos en SINT_array (matriz que contiene los caracteres ASCII) y almacene el resultado en SINT_array_size (etiqueta DINT).</p> <p>Si el carácter en SINT_array[element_number] = 13 (valor decimal del retorno de carro), parar.</p> <p>Establecer String_tag[element_number] = el carácter en SINT_array[element_number].</p> <p>Añadir 1 al valor de element_number. Esto deja que el controlador compruebe el siguiente carácter de SINT_array.</p> <p>Establecer el miembro Length de String_tag = element_number. (Esto registra el número de caracteres que hay en String_tag hasta el momento).</p> <p>Si element_number = SINT_array_size, a continuacion se detiene. (Se encuentra al final de la matriz y no contiene un retorno de carro).</p>	<pre>element_number := 0; SIZE(SINT_array, 0, SINT_array_size); While SINT_array[element_number] &lt;&gt; 13 do     String_tag.DATA[element_number] :=         SINT_array[element_number];     element_number := element_number + 1;     String_tag.LEN := element_number;     If element_number = SINT_array_size then         exit;     end_if; end_while;</pre>



## A

ABL 796  
ABS 390  
ACB 775  
ACL 778  
ACS 715  
activación de salida (OTE) 147  
ADD 393  
adición (ADD) 393  
AFI 614  
AHL 781  
alarma analógica 32, 53, 75  
alarma analógica ALMA  
  lógica de escalera 53  
  texto estructurado 75  
alarma digital 95, 106, 116  
alarma digital ALMD  
  bloque de funciones 95  
  lógica de escalera 106  
  texto estructurado 116  
AND 436  
aritmética y lógica de archivo (FAL) 493  
ASCII 773, 813, 832  
  Instrucciones de cadena ASCII 813, 832  
  Instrucciones de conversión ASCII 832  
  Instrucciones de puerto serial ASCII 773  
ASN 718  
AVE 512  
AWA 805  
AWT 799

## B

BAND 453  
BNOT 459  
Booleano 453, 456, 459, 461  
  NO booleano (BNOT) 459  
  O booleano (BOR) 461  
  O exclusivo booleano (BXOR) 456  
  Y booleano (BAND) 453  
BOR 461  
Borrar (CLR) 464  
BTD 428  
BTDT 432  
búsqueda y comparación de archivos (FSC) 519  
BXOR 456

## C

cadena central (MID) 820  
Carga LIFO (LFL) 581  
case...of 905  
CLR 464  
CMP 336  
códigos de error 224, 225, 229, 231, 811  
  ASCII 811  
  mensaje 224  
comparación de bits de archivo (FBC) 683  
comparar instrucciones 335  
conteo ascendente (CTU) 161  
conteo ascendente/descendente (CTUD) 166  
conteo descendente (CTD) 156  
COP 484  
copiar archivo (COP)\_ copiar archivo sincrónico (CPS) 484  
copiar archivo sincrónico - CPS 484

## D

DDT 675  
  detección de diagnóstico (DDT) 675  
derivada integral proporcional - PID 691  
desenclavamiento de salida (OTU) 151  
desplazamiento de bit a la izquierda (BSL) 558  
DINT en cadena (DTOS) 834  
distribuir campo de bits (BTD) 428  
distribuir campo de bits con receptor (BTDT) 432  
DIV 402  
dividir (DIV) 402

## E

enclavamiento de datos 876  
enclavamiento de salida (OTL) 149  
encontrar cadena (FIND) 814  
entrada de secuenciador (SQI) 598  
EQU 340  
etiqueta (LBL) 621  
EVENT 645  
examinar si abierto (XIO) 130  
examinar si cerrado (XIC) 128

## F

FAL 493  
    Diagrama de flujo de FAL (verdadero) 493  
    Diagrama de flujo FAL (falso) 493  
FBC 683  
    comparación de bits de archivo (FBC) 683  
FFL 567  
    Diagrama de flujo de FFL (falso) 567  
    Diagrama de flujo de FFL (pre-escaneado) 567  
    Diagrama de flujo de FFL (verdadero) 567  
FFU 574  
    Diagrama de flujo de FFU (falso) 574  
    Diagrama de flujo de FFU (pre-escaneado) 574  
    Diagrama de flujo de FFU (verdadero) 574  
FIFO 567, 574  
    Carga FIFO (FFL) 567  
    Descarga FIFO (FFU) 574  
fin temporal (TND) 643  
FLL 516  
FOR 657  
for...do 907

## G

GEQ 352  
grados (DEG) 761  
GSV 242  
GSV/SSV 257, 261, 307  
    ejemplo de programación 257  
    objetos 261  
    objetos de seguridad 307

## I

if...then 910  
igual a (EQU) 340  
instrucciones de alarma 31  
    alarma analógica 32, 53, 75  
    alarma digital 95, 106, 116  
instrucciones de bit 127  
Instrucciones de cadena ASCII 813, 814, 817, 820, 823, 828  
    cadena central (MID) 820  
    concatenar cadenas (CONCAT) 823  
    eliminar cadena (DELETE) 828  
    encontrar cadena (FIND) 814  
    insertar cadena (INSERT) 817

instrucciones de cálculo/matemáticas 389  
Instrucciones de conversión ASCII 832  
    cadena en DINT (STOD) 841  
    cadena en REAL (STOR) 844  
    DINT en cadena (DTOS) 834  
    Mayúsculas (UPPER) 847  
    minúsculas (LOWER) 836  
    REAL en cadena (RTOS) 839  
instrucciones de Logix 861  
    atributos comunes 861  
instrucciones de movimiento/lógicas 427  
instrucciones de puerto serial ASCII 773, 810, 811  
    Borrar ASCII búfer (ACL) 778  
    Caracteres ASCII en el búfer (ACB) 775  
    códigos de error 811  
    Escritura ASCII (AWT) 799  
    Escritura ASCII con anexo (AWA) 805  
    Instrucciones de puerto serial ASCII 773  
    Lectura ASCII (ARD) 786  
    Lectura ASCII de línea (ARL) 790  
    Líneas de handshake ASCII (AHL) 781  
    Prueba ASCII para línea de búfer (ABL) 796  
    tipos de cadena 810  
    tipos de datos 810  
instrucciones especiales 671  
instrucciones para/dividir 655  
intercambiar byte - SWPB 477

## J

JMP 621  
JSR 624  
JXR 617

## L

LBL 621  
LEQ 364  
LES 358  
LFL 581  
    Diagrama de flujo de LFL (falso) 581  
    Diagrama de flujo de LFL (pre-escaneado) 581  
    Diagrama de flujo de LFL (verdadero) 581  
LFU 588  
    Diagrama de flujo de LFU - verdadero 588  
    Diagrama de flujo de LFU (falso) 588  
    Diagrama de flujo de LFU (pre-escaneado) 588  
LIM 370

limitación de salida (PID) 710  
 llenar archivo (FLL) 516  
 LOG 740  
 logaritmo de base 10 (LOG) 740  
 logaritmo natural (LN) 743

**M**

máscara comparada igual a (MEQ) 375  
 mayor que (GRT) 346  
 MCR 631  
 menor o igual que (LEQ) 364  
 menor que (LES) 358  
 mensaje 224  
   códigos de error 224  
   códigos de error (.ERR) 225  
 MEQ 375  
 MID 820  
 minúsculas - LOWER 836  
 MOD 405  
 modo incremental 553, 555  
   diagrama de flujo del modo incremental (FSC) 555  
 modo numérico 551  
 modos de temporización 882  
 MOV 474  
 mover (MOV) 474  
 MSG 206, 215  
   ejemplos de configuración 215  
 MUL 410  
 multiplicar (MUL) 410  
 MVM 466  
 MVMT 469

**N**

NEG 414  
 negación (NEG) 414  
 NEQ 381  
 no igual a (NEQ) 381  
 NOP 636  
 NOT 445

**O**

o a nivel de bits (OR) 449  
 o exclusivo a nivel de bits (XOR) 441  
 obtener valor del sistema (GSV) 242

ONS 132  
 OR 449  
 orden de ejecución 877  
 OSF 134  
 OSFI 137  
 OSRI 144

**P**

pausa SFC - SFP 637  
 PID 691, 697, 702, 703, 704, 705, 709, 710  
   anti-windup 701  
   control de relación 704  
   derivada integral proporcional (PID) 691  
   establecer la banda muerta 709  
   lazos en cascada 703  
   prealimentación o sesgo de salida 705  
   reinicio sin perturbaciones 702  
   temporización de la instrucción 705  
   transferencia sin perturbaciones de manual a automático 701  
   usar instrucciones PID 697  
   utilizar el límite de salida 710  
 prueba del límite (LIM) 370

**R**

RAD 764  
 radianes (RAD) 764  
 raíz cuadrada (SQR) 417  
 REAL en cadena (RTOS) 839  
 repeat\_until 913  
 RES 171  
 restar (SUB) 421  
 retorno (RET) 624, 662  
 RTO 174  
 RTOR 179  
 RTOS 839

**S**

salida de secuenciador(SQO) 606  
 salida inmediata (IOT) 246  
 saltar a etiqueta (JMP) 621  
 saltar a rutina externa - JXR 617  
 SBR 624  
 seno (SIN) 730  
 SIN 730

sin instrucción de operación (NOP) 636

SQI 598

SQL 602

SQO 606

SQR 417

SQRT 417

SRT 535

SUB 421

subrutina (SBR) 624

## T

tamaño en elementos (SIZE) 544

TAN 734

tangente (TAN) 734

temporizador retentivo activado (RTO) 174

temporizador retentivo activado con restablecimiento

(RTOR) 179

texto estructurado 891, 892, 893, 896, 901, 902

  asignaciones 893

  comentarios 892

  construcciones 902

  expresiones 896

  Instrucciones 901

  sintaxis de programación 891

  sintaxis de texto estructurado 891

TND 643

TOD 754

TOF 184

TOFR 189

TON 194

TONR 199

## U

UID 650

UIE 650

un impulso (ONS) 132

un impulso en flanco ascendente (OSR) 140

un impulso en flanco ascendente con entrada

(OSRI) 144

un impulso en flanco descendente (OSF) 134

un impulso en flanco descendente con entrada

(OSFI) 137

## V

Valor absoluto (ABS) 390

valores inmediatos 863

## W

while\_do 916

## X

X a la potencia de Y (XPY) 747

XIC 128

XIO 130

XPY 747

# Soporte de Rockwell Automation

Rockwell Automation proporciona información técnica en la web para ayudarle a usar sus productos. En <http://www.rockwellautomation.com/support> podrá encontrar notas técnicas y de aplicación, código de muestra y vínculos a paquetes de servicios de software. También puede visitar nuestro centro de soporte en <https://rockwellautomation.custhelp.com> para obtener actualizaciones de software, chats y foros de soporte, información técnica, preguntas frecuentes y para registrarse para obtener actualizaciones de notificaciones de productos.

Además, ofrecemos múltiples programas de soporte para la instalación, configuración y solución de problemas. Para obtener más información, póngase en contacto con su distribuidor o agente de Rockwell Automation más cercano, o visite <http://www.rockwellautomation.com/services/online-phone>.

## Ayuda para la instalación

Si experimenta algún problema en las primeras 24 horas tras la instalación, revise la información contenida en este manual. Puede contactar con el servicio de soporte de clientes para obtener ayuda inicial para empezar a usar su producto.

Estados Unidos y Canadá	1.440.646.3434
Fuera de los Estados Unidos y Canadá	Utilice el localizador global disponible en <a href="http://www.rockwellautomation.com/locations">http://www.rockwellautomation.com/locations</a> o póngase en contacto con su agente de Rockwell Automation más cercano.

## Devolución de producto nuevo no satisfactorio

Rockwell Automation prueba todos sus productos para garantizar que funcionan correctamente al salir de fábrica. Sin embargo, si su producto no funciona y debe devolverlo, siga estos procedimientos.

Estados Unidos	Póngase en contacto con su distribuidor. Para completar el proceso de devolución, deberá proporcionar a su distribuidor un número de caso de soporte de cliente (llame al número de teléfono anterior para obtener uno).
Fuera de los Estados Unidos	Póngase en contacto con su agente de Rockwell Automation más cercano para obtener información sobre el procedimiento de devolución.

## Comentarios sobre la documentación

Sus comentarios sobre la documentación nos ayudarán a satisfacer mejor sus necesidades. Si tiene cualquier sugerencia sobre cómo mejorar este documento, cumplimente el formulario de comentarios, publicación [RA-DU002](#) [http://literature.rockwellautomation.com/idc/groups/literature/documents/du/ra-du002\\_-en-e.pdf](http://literature.rockwellautomation.com/idc/groups/literature/documents/du/ra-du002_-en-e.pdf).

Rockwell Otomasyon Ticaret A.Ş., Kar Plaza İş Merkezi E Blok Kat:6 34752 İçerenköy, İstanbul, Tel: +90 (216) 5698400

**[www.rockwellautomation.com](http://www.rockwellautomation.com)**

---

### Power, Control and Information Solutions Headquarters

Americas: Rockwell Automation, 1201 South Second Street, Milwaukee, WI 53204-2496 USA, Tel: (1) 414.382.2000, Fax: (1) 414.382.4444  
Europe/Middle East/Africa: Rockwell Automation NV, Pegasus Park, De Kleetlaan 12a, 1831 Diegem, Belgium, Tel: (32) 2 663 0600, Fax: (32) 2 663 0640  
Asia Pacific: Rockwell Automation, Level 14, Core F, Cyberport 3, 100 Cyberport Road, Hong Kong, Tel: (852) 2887 4788, Fax: (852) 2508 1846